A column-generation matheuristic approach for optimizing first-mile ridesharing services with publicly- and privately-owned autonomous vehicles

He, Ping; Jin, Jian Gang; Trépanier, Martin; Schulte, Frederik

DOI
10.1016/j.trc.2024.104516

Publication date
2024

Document Version
Final published version

Published in
Transportation Research Part C: Emerging Technologies

Transportation Research Part C 160 (2024) 104516

Contents lists available at ScienceDirect

Transportation Research Part C

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

A column-generation matheuristic approach for optimizing first-mile ridesharing services with publicly- and privately-owned autonomous vehicles

Ping He [a,e], Jian Gang Jin [a,b,*], Martin Trépanier [c,d], Frederik Schulte [e]

[a] *School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China*
[b] *Key Laboratory of Marine Intelligent Equipment and System of Ministry of Education, Shanghai Jiao Tong University, Shanghai, China*
[c] *Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Montreal, Canada*
[d] *Department of Mathematics and Industrial Engineering, Polytechnique Montreal, Montreal, Canada*
[e] *Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, Delft, The Netherlands*

ARTICLE INFO

Keywords:
Public transportation
Autonomous vehicles
First-mile ridesharing
Branch-and-price algorithm
Column-generation matheuristic algorithm

ABSTRACT

The burden of first-mile connection to public transit stations is a key barrier that discourages riders from taking public transportation. Public transit agencies typically operate a modest fleet of vehicles to provide first-mile services due to the high operating costs, thus failing to adequately meet the first-mile travel demands, especially during peak hours. At the same time, private cars are underutilized and have a lot of idle time. With the emergence of self-driving vehicles, new opportunities for addressing the current dilemma arise, such as integrating idle private self-driving vehicles to provide first-mile services, which is beneficial for public transportation agencies to provide high-quality services at low costs. This study investigates the first-mile ridesharing problem in which public transit agencies utilize idle privately-owned autonomous vehicles to dynamically inflate their fleet. This problem is more challenging in decision-making than conventional first-mile problems, as it involves decisions on heterogeneous fleet scheduling, vehicle routing, and time scheduling, all while taking into account the service quality for riders. To address this problem, an arc-based mixed-integer linear programming (MILP) model and a trip-based set-partitioning model are developed, both aiming to minimize total operational costs. To identify promising trips, we propose a tailored labeling algorithm with a novel dominance rule, along with a time window shift algorithm to determine the best schedule. To yield high-quality solutions in a short computation time, a tailored column-generation matheuristic algorithm is introduced. A branch-and-price exact algorithm and an adaptive large neighborhood search algorithm are developed to assess the matheuristic algorithm. Numerical experiments are conducted to demonstrate the effectiveness and applicability of the proposed models and algorithms. Experiments also show that this kind of ridesharing service can provide low-cost and high-quality services for the first-mile problem.

1. Introduction

Public transportation, such as metros and buses, plays a vital role in the urban transport system (Lin et al., 2021). However, in suburban areas, the population density is typically low, and thus the service radius of rail stations is often large, such as five kilometers. This causes inefficient first- and last-mile connectivity with public transit stations and is a key barrier that precludes

[*] Corresponding author at: School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China.
E-mail address: jiangang.jin@sjtu.edu.cn (J.G. Jin).

https://doi.org/10.1016/j.trc.2024.104516
Received 28 September 2023; Received in revised form 2 February 2024; Accepted 3 February 2024
Available online 10 February 2024

riders from using public transit systems (Park et al., 2021), as long walking access/egress distances (for instance, more than 500 m) would prevent them from easily reaching public transportation stations. At present, public transportation agencies tend to employ buses or mini-buses to provide demand-responsive connections to public transportation stations (Charisis et al., 2018). However, most demand-responsive connections are unable to offer door-to-door services (Vansteenwegen et al., 2022), and the service quality varies a lot due to the limited bus availability during rush hours, which may result in long waiting times for riders. Some riders take cabs or private cars for their first-mile or last-mile trips, but it is uneconomical and extremely difficult to hail a taxi during peak hours. As a result, it is critical to provide riders with efficient, convenient, and cost-effective first-mile and last-mile services.

With the increasing maturity of autonomous driving technology and its benefits in reducing transportation costs and improving the utilization of parking lots (Liang et al., 2020; Tang et al., 2021), autonomous vehicles (AVs) have been piloted to provide shuttle services in some city centers, universities, hospitals, parks, and more (RoboticResearch, 2022), as well as first/last-mile services (EasyMile, 2022). AVs do not require human drivers and are not subject to the constraint of drivers' workload (Chen et al., 2020), so using AVs to provide first-mile services may enable higher vehicle utilization and help public transit agencies in reducing operating costs. In addition, as ridesharing may further reduce operating costs, some studies (Chen et al., 2020; Shan et al., 2021) investigated using AVs to provide first-mile ridesharing services.

As of now, private vehicles are often underutilized and parked in parking lots for at least 90% of the day on average (Schmöller et al., 2015; Shaheen and Chan, 2016). To improve the utilization of private cars, carsharing has been promoted in recent years (Shaheen and Chan, 2016; Jochem et al., 2020). However, private car sharing still presents some inconveniences, such as requiring users to pick up and return vehicles at specific parking lots (Xu et al., 2018; Huang et al., 2020). Self-driving technologies offer a potential solution to these problems and could provide additional momentum for carsharing (Nazari et al., 2018). As a result, privately-owned autonomous vehicles have a greater opportunity to be leased to other private users, transportation network companies like Uber, Lyft, and Didi, and public transportation agencies during their idle times, and the owners can earn revenue. It is worth noting that the idle time of private vehicles may consist of multiple time slots. For instance, a vehicle may not be used during 8:30–11:00, 12:00–17:00, and 19:00–21:00 (Beirigo et al., 2022). Meanwhile, the time required for a complete first-mile round trip is usually less than 1 h (Huang et al., 2022) due to the small service area of first-mile problems, such as a 5 km service radius (Abe, 2021). Therefore, the first-mile problem is a promising application scenario for private autonomous vehicles with idle time slots. For public transit agencies, if they can leverage private vehicles to render on-demand first-mile services, they will be able to better meet travelers' demands, especially during rush hours (Grahn et al., 2022).

Compared to the classical first-mile problem illustrated in Fig. 1(a), the first-mile ridesharing problem with private AVs poses greater challenges for decision-making. First, this problem involves mixed fleet scheduling, which is more complicated than scheduling a homogeneous fleet (Ho et al., 2008; Lu et al., 2022). It requires determining which public vehicles and private vehicles are scheduled to provide first-mile services, taking into account the available capacity, earliest available time (also known as "ready time"), and ready location for each vehicle. In addition, it is necessary to ensure that the scheduled private vehicles can reach the designated parking lot before the end of their corresponding idle time slot. As shown in Fig. 1(b), once private vehicle 2 completes its current first-mile services, it must head to the parking lot, which could be located at the owner's residential area, workplace, or other locations.

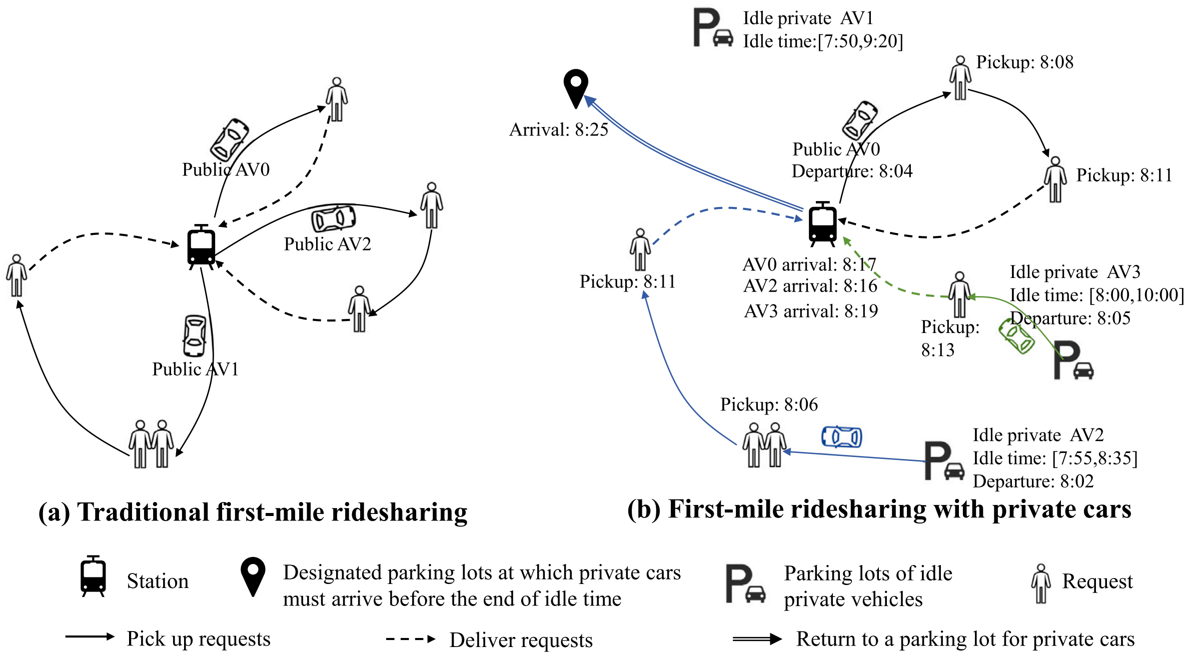
Second, to ensure that each rider's requirements for the latest arrival time at the hub station, the ride time, and the number of co-riders are well met, it is essential to determine the optimal ridesharing routes and identify the best schedule for each route. This schedule will specify the best departure time for each vehicle and the best pickup time at each node to reduce the waiting time for picking up requests and thus the ride time for on-board riders. The schedule decision increases the decision challenges for the first-mile ridesharing problem with private autonomous vehicles, which is often overlooked in existing studies since they assume that riders can be picked up immediately upon the arrival of vehicles. Fig. 2 illustrates the significance of schedule decisions in reducing ride times. Without schedule decisions, vehicles typically depart as early as possible and wait for requests if they arrive at the node before the corresponding earliest pickup time. However, this strategy may lead to longer ride times for on-board riders. As shown in Fig. 2(a), requests 1 and 2 have longer actual ride times than their maximum ride time requirements due to waiting for requests 3 and 4. By optimizing the departure and pickup times, the ride time requirements of requests 1 and 2 can be met on the same route, as demonstrated in Fig. 2(b).

Motivated by the aforementioned analysis, this paper investigates the first-mile ridesharing problem with private AVs available to schedule for public transit agencies. Our contributions consist of:

(1) investigating a first-mile ridesharing problem with publicly and privately owned autonomous vehicles to design high-quality and cost-effective first-mile services for passengers, developing an arc-based mixed integer programming model for this problem, and then reformulating it as a trip-based set partitioning model;

(2) presenting a novel column-generation math-heuristic algorithm to address the first-mile ridesharing problem with a mixed autonomous fleet. Two tailored strategies are incorporated in this algorithm to expedite the solving process of pricing sub-problems and enhance the diversity of the column pool, namely, reusing columns generated for other vehicles and the incremental column generation. Numerical experiments demonstrate that this algorithm can identify near-optimal solutions for instances with fewer than 50 requests in less than 35 s, outperforming the ALNS algorithm in terms of both computation time and solution quality. This math-heuristic algorithm framework is adaptable to other practical routing problems with mixed and heterogeneous fleets;

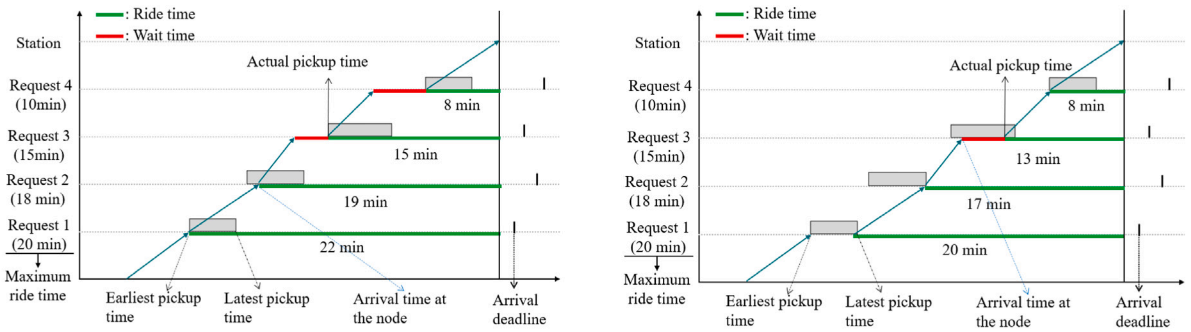
(3) proposing a time-window shift algorithm to determine the best schedule for a given ridesharing route and designing a customized labeling algorithm with a novel dominance rule to determine optimal ridesharing schemes for the pricing sub-problem of the column generation algorithm. This novel dominance rule is derived from the observation that the objective value of the sub-problem presents a piecewise linear convex function during the process of labeling extension. This labeling algorithm can be applied



(a) Traditional first-mile ridesharing

(b) First-mile ridesharing with private cars

Fig. 1. An illustrative example of the difference in vehicle scheduling between the traditional first-mile ridesharing problem and the first-mile ridesharing problem with private AVs. In (a), five requests are served by three public vehicles. In (b), two requests are served by one public vehicle, and three requests are served by two private vehicles.



(a) Without schedule decisions

(b) with schedule decisions

Fig. 2. Time schedule decisions for a route.

to other routing problems with travel time and latest arrival time constraints, such as the transportation of fresh and perishable goods with strict in-vehicle time requirements (Deng et al., 2022);

(4) employing the branch-and-price exact algorithm and the ALNS metaheuristic algorithm to assess the solution time and quality of the math-heuristic algorithm. The branch-and-price algorithm is capable of solving instances with up to 30 heterogeneous cars and 50 requests to optimality, and the ALNS algorithm can find high-quality solutions within 190 s for these instances; and

(5) demonstrating through case studies that the proposed approaches effectively address the first-mile travel problem with mixed fleets and passenger requirements on quality-of-service. Additionally, the integration of privately owned autonomous vehicles into ridesharing services can provide passengers with high-quality and cost-effective services. Moreover, ridesharing has the potential to substantially reduce operating costs for public transit agencies when the operating costs of privately owned autonomous vehicles are equivalent to those of publicly owned autonomous vehicles.

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature and positions our work. Section 3 describes the studied first-mile ridesharing problem in detail, including the problem description and two mathematical formulations.

Table 1
Overview of literature in first-mile ridesharing.

Reference	Main consideration						Solution method
	Fleet	Private cars	Public cars	IVT	LAT	MCR	
Shen et al. (2018)	Homo		✓	✓			Agent based simulation
Stiglic et al. (2018)	Homo	✓		✓	✓		Enumeration
Bian and Liu (2019a,b)	Homo	✓		✓	✓		Solution pooling heuristic
Bian et al. (2020)	Homo	✓		✓	✓		Solution pooling heuristic
Chen et al. (2020)	Homo		✓	✓			Cluster-based algorithm
Jiang et al. (2020)	Homo		✓	✓			Ant-colony optimization
Kumar and Khani (2021)	Homo	✓			✓		A matching algorithm
Ning et al. (2021)	Hete		✓	✓			Ant-colony optimization
Bian et al. (2022)	Hete	✓		✓	✓	✓	Solution pooling heuristic
Huang et al. (2022)	Homo		✓		✓		Dynamic pooling algorithm
This study	Hete	✓	✓	✓	✓	✓	CGM + BP

*IVT: rider's in-vehicle travel time; LAT: rider's latest arrival time; MCR: rider's maximum number of co-riders; Homo: Homogeneous fleet; Hete: Heterogeneous fleet; CGM: Column-generation matheuristic algorithm; BP: Branch-and-price algorithm.

A branch-and-price algorithm, an adaptive large neighborhood search algorithm, and a column-generation matheuristic algorithm are presented in Section 4. Numerical experiments are conducted in Section 5. Conclusions and some future work are summarized in Section 6.

2. Literature review

We review three streams of literature that are closely related to our research: (1) demand-responsive transit and first-mile ridesharing services; (2) vehicle sharing and fleet elasticity with autonomous vehicles; and (3) mixed fleets for first-mile and last-mile services.

2.1. Demand-responsive transit and first-mile ridesharing services

In practice, public transportation agencies often utilize buses or mini-buses to provide flexible demand-responsive transit (DRT) services to improve the connection of public transit stations (Aldaihani et al., 2004; Quadrifoglio and Li, 2009; Alshalalfah and Shalaby, 2012; Chandra and Quadrifoglio, 2013; Yu et al., 2015; Charisis et al., 2018; Lin et al., 2021; Yang et al., 2021). For instance, Aldaihani et al. (2004) developed an analytical model to design a hybrid grid network that integrates demand-responsive connectors with fixed-route major transit services. Quadrifoglio and Li (2009) utilized the continuous approximation method to solve the demand-responsive circulator service network design problem and compared the resulting demand-responsive and fixed-route solutions. Alshalalfah and Shalaby (2012) investigated the feasibility and benefits of leveraging flex-route feeder transit service connecting with the regional rail station. Yu et al. (2015) proposed a DRT system for commuting passengers departing their rail stations to their final work destinations in a seamless way. Charisis et al. (2018) presented a mathematical model and a genetic algorithm for designing a DRT feeder service for the first-mile and last-mile problem and applied their methods to the case of a residential area in Athens, Greece. Yang et al. (2021) designed DRT services by considering elastic demand and utilized the Augmented Lagrange Multiplier Method to solve this nonlinear optimization problem. However, passengers may have to wait a long time for the first-mile service due to the limited service capacity of DRTs during rush hours, and most DRTs do not provide door-to-door service (Vansteenwegen et al., 2022), which remains inconvenient for the elderly and disabled.

In this context, ridesharing via passenger cars has been proposed as a solution to the first-mile problem since it can provide more flexible and convenient door-to-door services (Shen et al., 2018; Stiglic et al., 2018; Ma et al., 2019; Bian and Liu, 2019a,b; Chen et al., 2020; Kumar and Khani, 2021). For instance, Stiglic et al. (2018) investigated how the seamless integration of ridesharing and public transit station can effectively solve the first-mile and last-mile problems as well as increase the use of public transport. Bian and Liu (2019a,b) proposed a novel mechanism for the first-mile ridesharing services and a heuristic algorithm to solve the problem. Ma et al. (2019) proposed a ridesharing strategy for private mobility service operators to render first- and last-mile services for riders. Chen et al. (2020) explored leveraging AVs to handle the first-mile ridesharing problem and designed a cluster-based solution method for it. Kumar and Khani (2021) integrated peer-to-peer ride-sharing and route-based transport to enhance the first-mile and last-mile connection. However, as shown in Table 1, most of the current studies, except Bian et al. (2022), have not given sufficient consideration for service quality: the latest arrival time, the maximum ride time, and the maximum number of co-riders. In addition, current studies typically assume riders can be picked up at any time after entering the system, without taking into account that some riders may have a preferred earliest pickup time. In this study, we consider this pickup time and allow vehicles to wait if they arrive earlier than the corresponding earliest pickup time. However, this waiting time will increase the ride time for on-board riders. Thus, a pickup schedule decision is necessary to minimize waiting time and better meet the ride time requirements for riders.

Table 2
Key distinctions between autonomous vehicles and human-driven vehicles for ridesharing services.

Vehicle category	Driving mode	Operation cost	Use experience	Vehicle availability
Human-driven vehicle	Manual driving	Labor and vehicle operation costs	Influenced by drivers	Dependent on drivers
Autonomous vehicle	Autonomous driving	Vehicle operation costs	Smoother and more comfortable	Available at all times when not in use

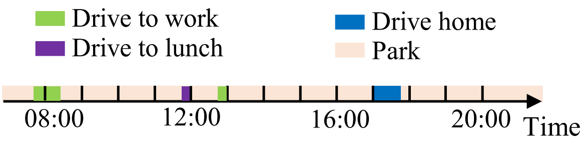


Fig. 3. An illustrative example demonstrating the trips of a driver and the parking time slots of the corresponding vehicle throughout the day.

2.2. *Vehicle sharing and fleet elasticity with autonomous vehicles*

In recent years, vehicle sharing has garnered significant attention as a promising complement to public transportation systems, since it increases the utilization of idle vehicles and provides more flexibility to meet travel demands (Schmöller et al., 2015; Shaheen and Chan, 2016). However, vehicle sharing still faces challenges, such as the inconvenience of picking up and returning vehicles at fixed parking spots (Jochem et al., 2020). With the emergence of self-driving vehicles, these issues can be well addressed (Nazari et al., 2018). Moreover, as presented in Table 2, self-driving vehicles have the following advantages over conventional human-driven vehicles: higher vehicle availability, lower operating costs, and enhanced user experiences (Bagloee et al., 2016; Hussain and Sherali, 2018). The integration of shared autonomous vehicles (SAVs) into transportation services can effectively reduce the total cost of ownership for transportation service providers, better align with the quality-of-service requirements for passengers, and diminish the idle rate for privately owned vehicles (Beirigo et al., 2022).

From an operational perspective, there are some differences between human-driven vehicles and self-driving vehicles, particularly concerning supply elasticity or vehicle availability. First, the elasticity of drivers is heavily influenced by factors such as wages and weather conditions (Farber, 2015; Sun et al., 2019; Angrist et al., 2021). For instance, higher wages typically attract more drivers, leading to an increase in the overall fleet supply. However, these factors have a limited impact on self-driving vehicles, ensuring a greater availability of vehicles. This increase in vehicle availability enhances the matching rate between passenger demand and vehicle supply, thus enhancing passenger service satisfaction. Second, human-driven vehicles are subject to the constraints of drivers' schedules and preferences. In contrast, self-driving cars can operate continuously during their idle periods, unrestricted by drivers' shifts or working hours.

Therefore, more and more studies focus on using self-driving cars to provide first-mile feeder services (Shen et al., 2018; Levin et al., 2019; Gurumurthy et al., 2020; Abe, 2021; Shan et al., 2021; Huang et al., 2022). For instance, Shen et al. (2018) simulated a first-mile transport system during morning rush hours by integrating SAVs. Levin et al. (2019) investigated a linear programming model for the optimal integration of SAVs with transit stations and proposed a rolling horizon method and a first-come-first-served strategy to obtain the suboptimal solution. Gurumurthy et al. (2020) carried out an agent-based simulation to identify the impact of using SAVs to provide first-mile and last-mile feeder services for the city of Austin, Texas. Shan et al. (2021) proposed an integrated optimization framework for railway transit systems and first-mile services using SAVs and developed a fixed-point algorithm for the problem. Huang et al. (2022) proposed a new service mode using SAVs to provide first-mile connections and introduced a novel pooling algorithm to match SAVs with riders.

To the best of our knowledge, current studies primarily rely on SAVs that remain idle throughout the day to offer first-mile services. However, the potential of using privately-owned autonomous vehicles with multiple idle time slots (for instance, 8:30-11:00, 12:00-17:00, and 19:00-21:00 (Pasaoglu et al., 2012; Beirigo et al., 2022)) for first-mile services has not yet been explored. Pasaoglu et al. (2012) presented compelling statistics indicating that privately owned vehicles are in motion for approximately 2 h per day, with the remaining time spent parked in areas like homes and offices. It is noteworthy that the parked time is not entirely consecutive, with roughly 6 h of parking time between all trips of their drivers (Pasaoglu et al., 2012). Moreover, private vehicle owners typically undertake an average of 2.5 trips per day (Pasaoglu et al., 2012), encompassing commuting to the office, returning from the office to home, and traveling from home or office to the mall, as illustrated in Fig. 3. As a result, the parking time between two consecutive trips is about 2.4 h, creating multiple idle time slots between these trips.

During these idle time slots, owners of autonomous private vehicles have the opportunity to lease their vehicles to generate some revenue. These idle and hireable autonomous vehicles could be temporarily employed to expand the fleets managed by public transit agencies or transportation network companies (TNCs). Fleet operators would gain benefits from this short-term fleet size elasticity by hiring privately-owned autonomous vehicles (Beirigo et al., 2022). For instance, this fleet elasticity enables fleet operators to better meet transportation demand with lower operating costs, especially during rush hours. At the same time, first-mile transportation services have the characteristic of relatively short trips, that is, relatively short transit times, which allows public transit agencies to schedule available private AVs to provide first-mile feeder services.

However, integrating privately owned self-driving vehicles to provide first- and last-mile ridesharing services also presents a couple of operational challenges. First, privately owned self-driving vehicles may have different seat numbers, resulting in heterogeneity in a previously homogeneous fleet and making fleet composition more complex. Second, privately owned vehicles may be dispersed throughout multiple parking lots, complicating vehicle scheduling and passenger matching decisions. Third, it is critical to ensure that the owner's trip chain remains uninterrupted when employing privately owned self-driving vehicles. Specifically, the ridesharing scheme provided by a privately owned vehicle must ensure that the vehicle can return to its designated parking lot before the owner's next trip.

2.3. Mixed fleets for offering first-mile and last-mile services

Several studies have explored the use of mixed fleets to address first- and last-mile transportation challenges. For instance, Grahn et al. (2022) found that the hybrid fleet with shuttle bus and ride-hailing vehicles to provide first-mile and last-mile services can reduce operating costs and tend to be more robust to variations in demand; Lu et al. (2022) investigated a last-mile problem with a mixed fleet of shared taxis to transport passengers and parcels; Grahn et al. (2023) explored the environmental impacts of a hybrid fleet consisting of ride-hailing vehicles and shared autonomous vehicles for the first-mile and last-mile travel services; Sipetas et al. (2023) studied a mixed fleet of both automated and human-driven vehicles to provide the feeder service in low-density areas. These studies concluded that mixed fleets are helpful for reducing emissions or operational costs for public transit agencies and improving service levels for passengers, especially during rush hours.

Motivated by the aforementioned analysis, we focus on the first-mile problem where public transit agencies integrate publicly-owned and privately-owned AVs to provide ridesharing services with the following considerations: (1) with the objective of minimizing the total operating costs; (2) riders' requirements for the latest arrival time, the maximum ride time, and the maximum number of co-riders; (3) heterogeneous fleet scheduling. To cope with this new challenging problem, we propose an arc-based and trip-based mathematical model, as well as develop a branch-and-price algorithm, an adaptive large neighborhood search algorithm, and a column-generation matheuristic algorithm.

3. First-mile ridesharing with private autonomous vehicles

This section will first elaborate on the problem of first-mile ridesharing by integrating private AVs. Then, an arc-based MILP model is presented. In light of the unique structure of this model, we reformulate it using Dantzig–Wolfe decomposition technology and develop a set-partitioning formulation.

3.1. Problem description

Consider a public transit agency that operates a fixed fleet of public AVs and can hire private AVs to dynamically expand the fleet to offer first-mile ridesharing services. For each hired private AV, the public transit agency may need to pay fees to its owner. Each request consists of a pickup location, the number of riders (p_i), the maximum number of co-riders (CO_i), the earliest service (pickup) time (ET_i), the maximum in-vehicle travel time (IVT_i), and the latest arrival time (LAT_i). The maximum number of co-riders refers to the number of travelers that the request is willing to share the trip with. The maximum in-vehicle travel time is employed to limit detours for each request, and it can be requested by riders according to their detour tolerance. Note that some riders may request a strict ride time or a strict number of co-riders, but these requirements may not always be met due to insufficient fleet availability or high service costs. If a request's detour or co-riding requirements cannot be met, a penalty cost is imposed on the service provider. Additionally, some passengers may need to reach their destination punctually, such as their offices, which necessitates arriving at the station on time. As a result, they will have the latest arrival time requirement. For instance, if a request needs to catch a train departing at 8:20, he or she may want to arrive at the station by 8:15.

The vehicles available for scheduling include en-route public and private vehicles, as well as public and private vehicles parked at the station and in garages. En-route vehicles are offering first-mile services and may already have riders on board ($\mathcal{O}_k$). If they still have available seats (q_k), they can be rescheduled to pick up other requests. Each vehicle has the following characteristics: a ready time (r_k), a ready location (g_k), and available seats at the ready location. The ready location denotes where the vehicle is available to provide first-mile services, while the ready time indicates the earliest available time to offer services. For each private vehicle scheduled, it is crucial to ensure that it can return to the designated parking lot ($\overline{g_k}$) before the end of the corresponding idle time slot ($\overline{r_k}$) to avoid disrupting the owner's travel plan.

To better respond to dynamic requests and reduce decision challenges, the rolling horizon method has been widely adopted for solving the dynamic vehicle routing problem and its variants (Pillac et al., 2013; Psaraftis et al., 2016; Ritzinger et al., 2016). This problem is also a variant of the dynamic vehicle routing problem with the arrival deadline, pickup time, ride time, and co-riding constraints. Therefore, we can also apply the rolling horizon method to solve this problem, similar to recent studies by Bian et al. (2020), Chen et al. (2020), and Bian et al. (2022). As illustrated in Fig. 4, the total planning horizon can be divided into a set of smaller horizons, with a fixed interval between each horizon, such as 10 min. This allows us to update the ridesharing schemes in response to incoming requests and available vehicles' states. At the beginning of each time horizon, the agency makes decisions on vehicles' scheduling and riders' matching for those requests whose earliest pickup time falls within the time horizon. Riders are then notified of their first-mile service information. Fig. 4 shows ridesharing schemes for two horizons, where one private car and one public car are scheduled to serve requests 1, 2, 3, and 4 in the first horizon. Before the re-optimization for the second horizon,

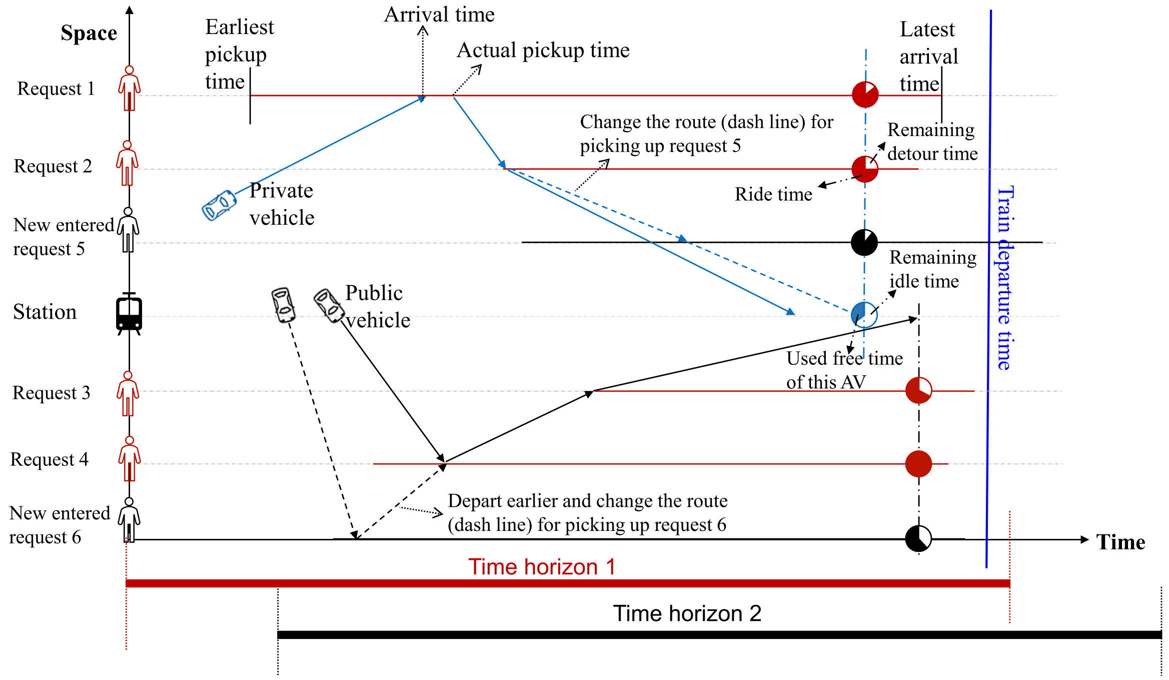


Fig. 4. An illustrative example of employing the rolling horizon method to dynamically solve the first-mile ridesharing problem.

requests 5 and 6 enter the system. Accordingly, the ridesharing schemes are updated at the beginning of the second horizon, with requests 5 and 6 assigned to private and public cars, respectively. To lower transportation costs, requests that have already been assigned previously but not yet picked up may be reassigned to other vehicles. To ensure service quality, the newly assigned pickup time must equal the previously assigned pickup time (LT_i), which means that the newly matched vehicle must arrive at this request before the LT_i.

The goal of public transit agencies is to determine the optimal dispatching scheme for vehicles and the high-quality ridesharing scheme for passengers with the objective of minimizing total costs while considering the following constraints: (1) If vehicles parked in the station and en-route vehicles are scheduled to provide first-mile services, they will depart from their ready locations, then pick up riders, and finally head to the station; (2) If private vehicles parked in garages are scheduled to provide services, they will depart from their garages, pick up riders, and take them to the station. Then they park in the station for the next service or return to the assigned garage of its owner; (3) The number of matched passengers cannot exceed the carrying capacity of each vehicle; (4) Each request must be served; (5) The pickup time window of each request must be met; (6) Vehicles should arrive at the station before the latest arrival time of requests it is serving; and (7) Penalty costs will incur if a rider's ride time tolerance or its willingness to share the trip cannot be met. Note that this study focuses on the problem setting of a single station for first-mile services. If there are multiple stations in the nearby area, they will be handled separately.

3.2. Arc-based MILP model

3.2.1. Notations

We represent the problem on a directed graph $G = (\mathscr{N}, \mathscr{A})$, where $\mathscr{N}$ denotes the set of all nodes, including requesting nodes $\mathscr{C} = 1, 2, 3 \ldots, n$, parking lots for private vehicles $\mathscr{G}$, and the depot at the hub station, which is denoted as 0. $\mathscr{A} = \{(i,j)|i,j \in \mathscr{N}, i \neq j\}$ denotes the set of arcs between all nodes. Other notations are defined in Table 3.

3.2.2. Mathematical model

Then, an arc-flow MILP model is formulated as follows:

$$min \sum_{k\in\mathscr{K}} \sum_{i\in\mathscr{N}} \sum_{j\in\mathscr{N}} c_k d_{i,j} x_{i,j}^k + \sum_{k\in\mathscr{K}_G} f y_k + \sum_{i\in\mathscr{C}\cup\mathscr{O}} \beta\zeta_i \tag{1}$$

subject to:

$$\sum_{k\in\mathscr{K}} \sum_{j\in\mathscr{N}\setminus\{\mathscr{G}\cup\{i\}\}} x_{i,j}^k = 1, \forall i \in \mathscr{C} \tag{2}$$

$$\sum_{j\in\mathscr{N}\setminus\{\mathscr{G}\cup\{i\}\}} x_{i,j}^k = \sum_{j\in\mathscr{N}\setminus\{i\}} x_{j,i}^k, \forall i \in \mathscr{C}, \forall k \in \mathscr{K} \tag{3}$$

Table 3
Parameters and decision variables.

Notations	Descriptions
Sets	
$\mathscr{K}_P^A/\mathscr{K}_P^B$	Set of public vehicles parked at the station/Set of en-route public vehicles
$\mathscr{K}_S^A/\mathscr{K}_S^B$	Set of previously scheduled private vehicles parked at the station (or parking lots)/en-route
$\mathscr{K}_P$	Set of total public vehicles, $\mathscr{K}_P = \mathscr{K}_P^A \cup \mathscr{K}_P^B$
$\mathscr{K}_S$	Set of total previously scheduled private vehicles, $\mathscr{K}_S = \mathscr{K}_S^A \cup \mathscr{K}_S^B$
$\mathscr{K}_G$	Set of private idle vehicles that have not been scheduled before
$\mathscr{K}$	Set of total vehicles, $\mathscr{K} = \mathscr{K}_P \cup \mathscr{K}_S \cup \mathscr{K}_G$
$\mathscr{C}$	Set of requesting nodes
$\mathscr{C}^+$	Set of requests have been matched previously with a vehicle but not yet picked up, $\mathscr{C}^+ \subset \mathscr{C}$
$\mathscr{O}$	Set of on-board requests for en-route vehicles, $\mathscr{O} = \{\mathscr{O}_1, \mathscr{O}_2, \dots \mathscr{O}_{\lvert\mathscr{K}_P^B \cup \mathscr{K}_S^B\rvert}\}$
$\mathscr{G}$	Set of garages for parking private vehicles
$\mathscr{N}$	Set of all nodes, $\mathscr{N} = \mathscr{C} \cup \{0\} \cup \mathscr{G}$
Parameters	
g_k	The garage, station, or parking lot where vehicle k will depart from it to serve requests, $g_k \in \mathscr{G} \cup \{0\}$
$\overline{g_k}$	The garage or parking lot where private car k must return by the end of this idle time slot, $\overline{g_k} \in \mathscr{G}$
q_k	Available seat number of vehicle k, $k \in \mathscr{K}$
r_k	Ready time for vehicle k, $k \in \mathscr{K}$
$\overline{r_k}$	Latest return time for private vehicle $k, k \in \mathscr{K}_S \cup \mathscr{K}_G$
a_k	Latest arrival time for vehicle k, $k \in \mathscr{K}_P^B \cup \mathscr{K}_S^B$
c_k	Transportation cost ($) per kilometer for vehicle k, $k \in \mathscr{K}$
p_i	Number of riders for request i, $i \in \mathscr{C} \cup \mathscr{O}$
CO_i	Maximum number of co-riders for request i, $i \in \mathscr{C} \cup \mathscr{O}$
LAT_i	Latest arrival time of request i, $i \in \mathscr{C} \cup \mathscr{O}$
IVT_i	Maximum (remaining) in-vehicle travel time for request i, $i \in \mathscr{C} \cup \mathscr{O}$
ET_i	Earliest service (pickup) start time for request i, $i \in \mathscr{C} \setminus \mathscr{C}^+$
LT_i	Latest service (pickup) start time for matched but not yet picked up request i, $i \in \mathscr{C}^+$ ($ET_i = LT_i$)
s_i	Service duration for request i, $i \in \mathscr{C}$
$d_{i,j}$	Distances (km) between nodes i and j, $i, j \in \mathscr{N}$
$t_{i,j}$	Travel time between nodes i and j, $i, j \in \mathscr{N}$
M	A large positive constant
f	Fixed cost ($) of hiring a private vehicle that has not been scheduled previously
β	Penalty cost coefficient ($) if the service quality for request i cannot be met, $i \in \mathscr{C} \cup \mathscr{O}$
Decision variables	
$x_{i,j}^k$	$\in \{0,1\}$. 1 if vehicle k traverses arc (i,j); otherwise, 0
t_i^k	≥ 0. Service start time for request i by vehicle k
ζ_i	$\in \{0,1\}$. 1 if the service quality requirements of request i cannot be met
ξ_i	≥ 0. In-vehicle ride time for request i
ψ_i	$\in \mathbb{Z}^+ \cup \{0\}$. Number of co-riders for request i
y_k	$\in \{0,1\}$. 1 if vehicle k is scheduled; otherwise, 0
t_k	≥ 0. The departure time of vehicle k

$$y_k = \sum_{j \in \mathscr{C}} x_{g_k,j}^k = \sum_{j \in \mathscr{C}} x_{j,0}^k \leq 1, \forall k \in \mathscr{K} \tag{4}$$

$$\sum_{i \in \mathscr{C}} \sum_{j \in \mathscr{N} \setminus \{i\}} p_i \; x_{i,j}^k + |\mathscr{O}_k| \leq q_k, \forall k \in \mathscr{K} \tag{5}$$

$$t_k \geq r_k, \forall k \in \mathscr{K} \tag{6}$$

$$t_0^k + s_0 + t_{0,\overline{g_k}} - M\left(1 - y_k\right) \leq \overline{r}_k, \forall k \in \mathscr{K} \setminus \mathscr{K}_P \tag{7}$$

$$t_j^k \geq t_i^k + t_{i,j} + s_i - M\left(1 - x_{i,j}^k\right), \forall i, j \in \mathscr{N} \setminus \{g_k\}, \forall k \in \mathscr{K} \tag{8}$$

$$t_j^k \geq t_k + t_{g_k,j} - M\left(1 - x_{g_k,j}^k\right), \forall j \in \mathscr{N} \setminus \{g_k\}, \forall k \in \mathscr{K} \tag{9}$$

$$t_i^k = LT_i, \forall i \in \mathscr{C}^+, \forall k \in \mathscr{K} \tag{10}$$

$$ET_i \leq t_i^k \leq LAT_i - t_{i,0} - s_i, \forall i \in \mathscr{C} \setminus \mathscr{C}^+, \forall k \in \mathscr{K} \tag{11}$$

$$t_0^k \leq LAT_i + M(1 - \sum_{j \in \mathscr{C} \cup \{0\}} x_{i,j}^k), \forall i \in \mathscr{C}, \forall k \in \mathscr{K} \tag{12}$$

$$t_0^k \le a_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{13}$$

$$t_0^k - t_i^k - M(1 - \sum_{j \in \mathcal{C} \cup \{0\}} x_{i,j}^k) \le \xi_i, \forall i \in \mathcal{C}, \forall k \in \mathcal{K} \tag{14}$$

$$t_0^k - r_k \le \xi_i, \forall i \in \mathcal{O}_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{15}$$

$$\psi_i \ge \sum_{m \in \mathcal{N}} \sum_{n \in \mathcal{N}} p_m x_{mn}^k + |\mathcal{O}_k| - p_i, i \in \mathcal{O}_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{16}$$

$$\psi_i \ge \sum_{m \in \mathcal{N} \setminus \{i\}} \sum_{n \in \mathcal{N}} p_m x_{mn}^k + |\mathcal{O}_k| - M(1 - \sum_{j \in \mathcal{N}} x_{i,j}^k), \forall i \in \mathcal{C}, \forall k \in \mathcal{K} \tag{17}$$

$$M\zeta_i \ge max\{\xi_i - IVT_i, \psi_i - CO_i\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{18}$$

$$x_{i,j}^k \in \{0, 1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{N}, \forall k \in \mathcal{K} \tag{19}$$

$$t_i^k \ge 0, \forall i \in \mathcal{N}, \forall k \in \mathcal{K} \tag{20}$$

$$t_k \ge 0, \forall k \in \mathcal{K} \tag{21}$$

$$\zeta_i \in \{0, 1\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{22}$$

$$\xi_i \ge 0, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{23}$$

$$\psi_i \in \mathbb{Z}^+ \cup \{0\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{24}$$

$$y_k \in \{0, 1\}, \forall k \in \mathcal{K} \tag{25}$$

The objective function (1) minimizes total operating costs, including transportation costs, fixed costs associated with hiring private autonomous vehicles that have not been scheduled previously, and penalty costs incurred when the actual ride time or the number of co-riders exceeds a request's requirements for detour tolerance or co-riding willingness. Constraints (2) guarantee that each request must be served. Constraints (3) indicate that each request is served by the same vehicle and the service flow is conserved. Constraints (4) denote whether the vehicle is scheduled to render services. Constraints (5) ensure that the number of served passengers, including both on-board and newly matched requests, does not exceed the capacity of each vehicle. Constraints (6) represent that the vehicle departure time is greater than or equal to its ready time. Constraints (7) ensure that the private vehicle must return to the designated parking lot before its latest return time. Constraints (8)–(9) represent the pickup time of each request. For those requests that have already been matched but not yet picked up, their ridesharing schemes may be changed, but the newly assigned pickup time should equal the previously assigned time, which is ensured by constraints (10). Constraints (11) ensure that the pickup time for each new incoming request meets its service (pickup) time window, in which the latest pickup time is equal to the corresponding latest arrival time minus service time and the direct travel time to the station. Constraints (12) and (13) guarantee that the arrival time at the station must meet the latest arrival time requirements of on-board and newly picked-up passengers. Constraints (14) and (15) determine the actual ride time for newly picked-up riders and on-board riders, respectively. Constraints (16) and (17) identify the number of co-riders for newly picked-up riders and on-board riders, respectively. Constraints (18) check whether the detour tolerance and the co-riding willingness for each request are met. Constraints (19)–(25) define the domain of decision variables.

It is worth noting that the private fleet in this model is not restricted to self-driving cars and can also be human-driven private vehicles or free-floating vehicles. With respect to human-driven vehicles, some drivers are willing to make a detour to pick up first-mile passengers and take them to the station before heading to their destinations (Stiglic et al., 2018). Accordingly, drivers may also have requirements for detours, the number of co-riders, and the latest arrival time at the station to ensure that they can reach their destinations on time. These requirements have already been considered in this model, making these human-driven vehicles a viable option for private fleets. Free-floating vehicles, similar to those operated by TNCs, typically do not have restrictions on the latest arrival time or ride time, so they can be directly incorporated into our model.

While this model primarily targets the first-mile ridesharing scenario, it can be extended to address the last-mile ridesharing problem by simplifying some parameters and constraints. This extension is feasible because passengers requiring last-mile services typically do not have strict pickup and arrival time requirements compared to first-mile cases (Chen and Wang, 2018; Agussurja et al., 2019; Wang, 2019).

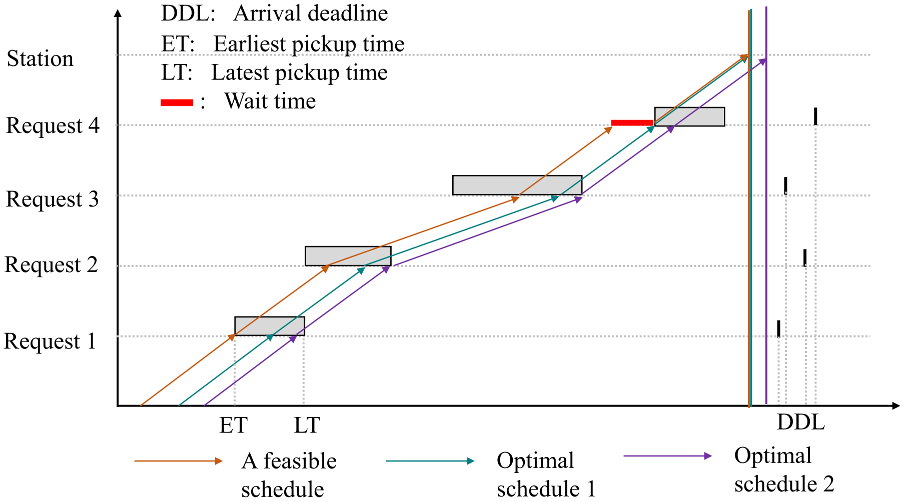


Fig. 5. Two optimal schedules for a route.

3.3. A set partitioning formulation

Before introducing the set partitioning model, we give the following definitions:

Definition 1 (*Schedule of a Route*)**.** For a given route with node sequence $\mathbb{S} = \{i_0 = g_k, i_1, \dots, i_n\}$, let $\mathbb{S}_t = \{t_0, t_1, \dots, t_n\}$ be a feasible schedule that meets the pickup time and latest arrival time requirements for each request visited. Here, t_0 refers to the departure time at the parking lot or the ready location of vehicle k, and t_n is the pickup time at the node n.

Definition 2 (*Optimal Schedule of a Route* $\mathbb{S}$)**.** Let $C(\mathbb{S}, \mathbb{S}_t)$ be the total cost of route $\mathbb{S}$ associated with schedule $\mathbb{S}_t$, and let $\Omega(\mathbb{S})$ be the set of feasible schedules for route $\mathbb{S}$. The schedule $\overline{\mathbb{S}_t}$ is the optimal schedule for route $\mathbb{S}$ if and only if $C(\mathbb{S}, \overline{\mathbb{S}_t}) \leq C(\mathbb{S}, \mathbb{S}_t)$, $\forall\ \mathbb{S}_t \in \Omega(\mathbb{S})$.

Definition 2 indicates that the optimal schedule for route $\mathbb{S}$ aims to minimize the penalty cost incurred by exceeding the ride time tolerances for each served request. It is worth noting that multiple optimal schedules for route $\mathbb{S}$ may exist, as illustrated in Fig. 5, which depicts two optimal schedules along with one feasible schedule. Additionally, there are many other potential optimal schedules between the two depicted in the figure. In such cases, we select the schedule with the earliest arrival time at node n as the best one, as it enables the vehicle to arrive at the destination earlier than other schedules and provide subsequent first-mile services sooner.

Definition 3 (*Trip*)**.** Let route $\mathbb{S}$ and its associated $\mathbb{S}_t$ denote a trip, which must meet all the constraints of the served requests and the employed vehicle.

Based on these definitions, the MILP model can be seen as finding the best trip for each car, which motivates us to reformulate it as a set-partitioning model using the Dantzig–Wolfe decomposition. The original model can be decomposed into a series of sub-problems to determine a promising trip for each car and a master problem to select the best combination of trips. We use $\mathcal{S}$ to denote the set of trips for vehicles. A binary decision variable ϑ_s is defined to determine whether the trip $s \in \mathcal{S}$ is employed. Additional parameters are as follows:

- $\alpha_i^s \in \{0, 1\}$: 1 if request i is served by trip s; 0 otherwise.
- $\mu_k^s \in \{0, 1\}$: 1 if vehicle k is employed by trip s; 0 otherwise.
- c_s: total costs of trip s, including total transportation costs, penalty costs if some requests' requirements on co-riding or detour are unmet by this trip, and the cost of employing a private vehicle that has not been scheduled before.

With these notations, the MILP model can be rewritten as the following set-partitioning model.

$$[\mathbf{MP}]\ min \sum_{s \in \mathcal{S}} c_s \vartheta_s \tag{26}$$

subject to:

$$\sum_{s \in \mathcal{S}} \alpha_i^s \vartheta_s = 1, \forall i \in \mathscr{C} \tag{27}$$

$$\sum_{s \in \mathcal{S}} \mu_k^s \vartheta_s \leq 1, \forall k \in \mathscr{K} \tag{28}$$

$$\vartheta_s \in \{0,1\}, \forall s \in \mathcal{S} \tag{29}$$

The objective function (26) minimizes the total costs of each trip. Constraints (27) ensure that each request is served by a trip of vehicles. Constraints (28) guarantee that each vehicle can serve at most one trip. Constraints (29) define the domain of decision variables.

4. Solution methodology

In this section, we develop a branch-and-price algorithm to determine the exact solution, a column-generation matheuristic (CGM) algorithm to obtain a near-optimal solution in less computation time than the branch-and-price algorithm, and an adaptive large neighborhood search (ALNS) algorithm to further evaluate the performance of the CGM algorithm.

4.1. Restricted master problem (RMP)

The number of complete columns in MP is exponential. Thus, enumerating all columns is computationally intractable for most cases. The common practice is repeatedly solving the restricted master problem (RMP) by focusing on a subset $\overline{\mathcal{S}} \subset \mathcal{S}$ and relaxing the decision variables ϑ_s to be continuous variables. To ensure RMP is always feasible, we directly employ a set of auxiliary decision variables $v_i, i \in \mathcal{C}$ to represent the initial column set $\mathcal{S}_0$. If the value of $v_i, i \in \mathcal{C}$ is positive, it will incur a penalty cost in the objective function.

$$[\textbf{RMP}]\ min \sum_{s\in\overline{\mathcal{S}}} c_s \vartheta_s + \sum_{i\in\mathcal{C}} M v_i \tag{30}$$

subject to:

$$\sum_{s\in\overline{\mathcal{S}}} \alpha_i^s \vartheta_s + v_i \geq 1, \forall i \in \mathcal{C} \tag{31}$$

$$\sum_{s\in\overline{\mathcal{S}}} \mu_k^s \vartheta_s \leq 1, \forall k \in \mathcal{K} \tag{32}$$

$$\vartheta_s \geq 0, \forall s \in \overline{\mathcal{S}} \tag{33}$$

$$v_i \geq 0, \forall i \in \mathcal{C} \tag{34}$$

4.2. Pricing subproblem (PSP)

Let π_i, ω_k denote dual variables corresponding to constraints (31) and (32) of RMP. Then, the reduced cost of trip s associated with vehicle k is:

$$\bar{c}_s^k = c_s - \sum_{i\in\mathcal{C}} \alpha_i^s \pi_i - \omega_k \tag{35}$$

Let binary variable $x_{i,j}$ denote whether arc(i,j) is traversed. t_i is the visiting time of request i. τ is the departure time of the vehicle. f_k is the fixed cost of scheduling a private vehicle that has not been scheduled previously; otherwise, f_k equals 0.

$$c_s = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} c_k d_{i,j} x_{i,j} + f_k + \sum_{i\in\mathcal{C}\cup\mathcal{O}} \beta \zeta_i \tag{36}$$

We need to determine the best trip for each vehicle due to the heterogeneous locations, ready times, and capacities of vehicles. Therefore, the subproblem for vehicle k can be formulated as:

$$[\textbf{PSP}]\ min\ \bar{c}_s^k = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} c_k d_{i,j} x_{i,j} + f_k + \beta \sum_{i\in\mathcal{C}\cup\mathcal{O}} \zeta_i - \sum_{i\in\mathcal{C}} \alpha_i^s \pi_i - \omega_k \tag{37}$$

subject to:

$$\sum_{j\in\mathcal{N}\setminus\{\mathcal{G}\cup\{i\}\}} x_{i,j} = \sum_{j\in\mathcal{N}\setminus\{i\}} x_{j,i}, \forall i \in \mathcal{C} \tag{38}$$

$$\sum_{j\in\mathcal{N}} x_{g_k,j} = \sum_{j\in\mathcal{C}} x_{j,0} = 1 \tag{39}$$

$$\sum_{i\in\mathcal{C}} \sum_{j\in\mathcal{N}\setminus\{i\}} p_i x_{i,j} + |\mathcal{O}_k| \leq q_k \tag{40}$$

$$\tau \geq r_k \tag{41}$$

$$t_0 + s_0 + t_{0,\overline{g_k}} \le \bar{r}_k, \forall k \in \mathcal{K}\backslash\mathcal{K}_P \tag{42}$$

$$t_j \ge t_i + t_{i,j} + s_i - M\left(1 - x_{i,j}\right), \forall i, j \in \mathcal{N}\backslash\{g_k\} \tag{43}$$

$$t_j \ge \tau + t_{g_k,j} - M\left(1 - x_{g_k,j}\right), \forall j \in \mathcal{N}\backslash\{g_k\} \tag{44}$$

$$t_i = LT_i, \forall i \in \mathcal{C}^+ \tag{45}$$

$$ET_i \le t_i \le LAT_i - t_{i,0} - s_i, \forall i \in \mathcal{C}\backslash\mathcal{C}^+ \tag{46}$$

$$t_0 \le LAT_i + M(1 - \sum_{j\in\mathcal{C}\cup\{g_k\}} x_{i,j}), \forall i \in \mathcal{C} \tag{47}$$

$$t_0 \le a_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{48}$$

$$t_0 - t_i - M(1 - \sum_{j\in\mathcal{C}\cup\{0\}} x_{i,j}) \le \xi_i, \forall i \in \mathcal{C} \tag{49}$$

$$t_0 - r_k \le \xi_i, \forall i \in \mathcal{O}_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{50}$$

$$\psi_i \ge \sum_{m\in\mathcal{N}} \sum_{n\in\mathcal{N}} p_m x_{mn} + |\mathcal{O}_k| - p_i, \forall i \in \mathcal{O}_k, \forall k \in \mathcal{K}_P^B \cup \mathcal{K}_S^B \tag{51}$$

$$\psi_i \ge \sum_{m\in\mathcal{N}\backslash\{i\}} \sum_{n\in\mathcal{N}} p_m x_{mn} + |\mathcal{O}_k| - M(1 - \sum_{j\in\mathcal{N}} x_{i,j}), \forall i \in \mathcal{C} \tag{52}$$

$$M\zeta_i \ge max\{\xi_i - IVT_i, \psi_i^k - CO_i\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{53}$$

$$x_{i,j} \in \{0, 1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{N} \tag{54}$$

$$t_i \ge 0, \forall i \in \mathcal{N} \tag{55}$$

$$\tau \ge 0 \tag{56}$$

$$\zeta_i \in \{0, 1\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{57}$$

$$\xi_i \ge 0, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{58}$$

$$\psi_i \in \mathbb{Z}^+ \cup \{0\}, \forall i \in \mathcal{C} \cup \mathcal{O} \tag{59}$$

The objective function (37) finds the trip with the minimum reduced cost for vehicle k. Other constraints are similar to the constraints corresponding to the original MILP model.

4.3. Solution techniques for the subproblem

The subproblem is a variant of the elementary shortest path problem with resource constraints (ESPPRC). The labeling algorithm (dynamic programming) is a commonly used approach to solve ESPPRC and its variants to optimality. This subproblem is more challenging than the ESPPRC since it involves not only routing decisions but also schedule decisions, which need to consider the ride time and latest arrival time of each request. To solve this more challenging problem, a time window shift algorithm is designed to determine the best schedule for a route, and then a tailored labeling algorithm is introduced. The computational efficiency of the subproblem is the critical part of the total computation time for the branch-and-price algorithm and the column-generation matheuristic algorithm. To speed up the computational efficiency, we combine the exact and heuristic labeling algorithms, along with heuristic dominance rules, to identify the vehicle trip with the minimum reduced cost.

Algorithm 1: Time window shift algorithm

```
1  Input: A route sequence $\mathbb{S} = \{i, i+1, \dots, n\}$ associated with a vehicle $k$
2  Output: A best schedule for the route
   //identify the earliest pickup time at node $n$;
3  for each node $i$ on this route do
4      $es_i \leftarrow es_{i-1}+t_{i-1,i}+s_{i-1}$; $ls_i \leftarrow ls_{i-1}+t_{i-1,i}+s_{i-1}$ ;
5      if $es_i > LT_i$ then
6          return false                    //violate the corresponding latest pickup time;
7      else if $ls_i < ET_i$ then
8          $es_i \leftarrow ET_i$; $ls_i \leftarrow ET_i$;
9      else
10         $es_i \leftarrow max\{es_i, ET_i\}$; $ls_i \leftarrow min\{ls_i, LT_i\}$;
   //check the feasibility of the latest arrival times;
11 if $es_n + s_n + t_{n,0} > a_k$ then
12     return false;
13 for each node $i$ on this route do
14     if $es_n + s_n + t_{n,0} > LAT_i$ then
15         return false;
   //determine the best schedule;
16 for each node $j$ on this route in a reverse order do
17     $es_j \leftarrow min\{es_{j+1} - s_j - t_{j,j+1}, LT_j\}$; $ls_j \leftarrow min\{ls_{j+1} - s_j - t_{j,j+1}, LT_j\}$;
18     update the ride time for request $j$ $(es_n - es_j)$;
19 return the best schedule $\{es_i, \forall i \in \mathbb{S}\}$
```

4.3.1. *A time window shift algorithm for identifying the best schedule of a route*

For a given route, the feasibility of ride times can be trivially checked with the eight-step time slack algorithm (Cordeau and Laporte, 2003), which has been well applied in dial-a-ride problems with ride time constraints. However, in this problem, it is necessary to identify the optimal schedule for each route, including the best departure and pickup times, rather than just checking the feasibility of ride times, which makes the eight-step feasibility check procedure inapplicable. As a result, we propose an algorithm with the computation complexity of $O(n)$ to determine the best schedule. The basic idea is that we first determine the earliest pickup time at the last node in the route to check whether the latest arrival time of each request can be met, and then identify the best pickup time for requests and departure time for the vehicle. The detailed procedure is presented in Algorithm 1, in which es_n and ls_n denote the earliest and latest feasible pickup times at node n, respectively. Note that we can use es_n of each node to construct a feasible schedule.

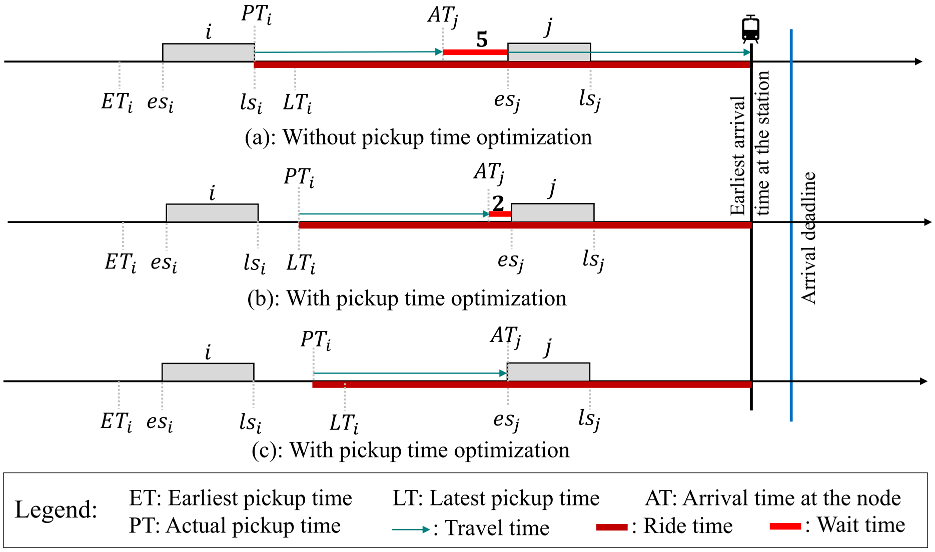
Lines 3 to 10 in Algorithm 1 are used to determine the earliest and latest feasible pickup time for each request, using a forward time window shift method (see Lim et al. (2017) for more details). Based on these calculations, we can check whether the latest arrival times can be met with lines 11 to 15. If these requirements are met, the algorithm proceeds to identify the best schedule. During this process, some wait times may be able to be reduced by postponing the pickup or departure time from the predecessor node (line 17). Taking Fig. 6 as an example, we assume that the pickup time equals the departure time, without considering the service duration for picking up a request. If the vehicle picks up request i at ls_i, it will wait for 5 min before picking up the request j, as shown in Fig. 6(a). However, if it postpones the pickup time to LT_i at request i, the wait time can be reduced to 2 min, and the total ride time of request i can be shortened by 3 min, as shown in Fig. 6(b). In addition, if the latest pickup time at request i is larger than the current value LT_i, the wait time may be able to be eliminated, resulting in a shorter ride time for request i, as shown in Fig. 6(c). By doing this process reversely from the last node of the route, we can finally get the earliest and latest pickup times with the minimum ride time for each request, and select these earliest pickup times as the best schedule.

4.3.2. *Customized labeling algorithm*

(1) Label definition

Define a label $L_k^i = \{C_i, V_i, U_i, P_i, LA_i, \{LIVT_n, LCO_n, DR_n, EVT_n, LVT_n, \forall n \in P_i \backslash \{g_k\}\}\}$ to denote the partial path and the corresponding information from the station or garage to a requesting node i for vehicle k, where:

- C_i: the reduced cost of the partial path;
- V_i: the consumed capacity of this label;
- U_i: a vector of visited requests and the inextensible requests for this label;
- P_i: the set of path nodes of label L_k^i;
- LA_i: the latest arrival time for all requests of the label;
- $LIVT_n$: the remaining in-vehicle travel time for request n without exceeding its detour tolerance;



PT_i
AT_j
i
5
j
ET_i es_i ls_i LT_i es_j ls_j
Earliest arrival time at the station
PT_i
AT_j
i
2
j
ET_i es_i ls_i LT_i es_j ls_j
Arrival deadline
PT_i
AT_j
i
j
ET_i es_i ls_i LT_i es_j ls_j
Legend:
ET: Earliest pickup time
LT: Latest pickup time
AT: Arrival time at the node
PT: Actual pickup time
→: Travel time
▬: Ride time
▬: Wait time

(a): Without pickup time optimization

(b): With pickup time optimization

(c): With pickup time optimization

Fig. 6. Shorten the ride time of a request with the backward pickup time shift method.

- LCO_n: the remaining number of co-riders without exceeding the maximum number of co-riders for request n;
- DR_n: the minimum ride time from node n to node i;
- EVT_n: the earliest visiting time at node n considering LA_i and $LIVT_n$; and
- LVT_n: the latest visiting time at node n considering LA_i and $LIVT_n$;

(2) Label extension and feasibility check

The label with the minimum reduced cost is selected and extended to all the reachable nodes. After the label extension procedure, we need to update several pieces of information, such as the latest arrival time of the route, the remaining ride time as well as the number of co-riders of requests, and the capacity of vehicles, which is more complicated than the conventional label extension procedure. Accordingly, we design a specialized label extension algorithm for this entire extension process, incorporating Algorithm 1 to determine the schedule. The label extension algorithm works as follows:

Step 1: Partial information update after extending on node j

$$P_j = P_i \cup \{j\} \tag{60}$$

$$V_j = V_i + p_j \tag{61}$$

$$EVT_j = max\left\{EVT_i + t_{i,j} + s_i, ET_j\right\} \tag{62}$$

$$LVT_j = \begin{cases} LT_j & if\, LVT_i > EVT_i,\ and\ j \in \mathscr{C}^+ \\ min\{max\{ET_j, LVT_i + t_{i,j} + s_i\}, LAT_j - t_{j,0} - s_j\} & if\, LVT_i > EVT_i,\ and\ j \in \mathscr{C}\backslash\mathscr{C}^+ \\ EVT_j & \text{if } LVT_i = EVT_i \end{cases} \tag{63}$$

Eqs. (60)–(61) respectively update the visited requests and the consumed capacity for the new label. Eq. (62) renews the earliest pickup time at node j, it should be the maximum value between the earliest service start time of request j and the earliest arriving time from requesting node i. Eq. (63) updates the latest pickup time at node j, which consists of three cases: (a) if the earliest pickup time is less than the latest pickup time at node i and node j is a request that has already been matched previously but not yet picked up, LVT_j will be updated to latest pickup time previously assigned to j; (b) if the earliest pickup time is less than the latest pickup time at node i and node j is a new incoming request, LVT_j will be updated to the minimum value between the latest arrival time from request i and the latest service start time for request j; and (c) if the earliest pickup time equals the latest pickup time at node i, this relationship exists at node j;

Step 2: Check whether the feasibility of the pickup time at request j is satisfied, and the extension is infeasible if $EVT_j > LAT_j - t_{j,0} - s_j$ or $EVT_j > LVT_j$.

Step 3: Starting from the last node, update the pickup time window for each preceding request step-by-step backward:

For each request n, the earliest and latest pickup times of its preceding node $n-1$ are updated via Eqs. (64) and (65):

$$EVT_{n-1} = \begin{cases} LT_{n-1} & \forall n-1 \in \mathscr{C}^+ \\ min\left\{EVT_n - s_{n-1} - t_{n-1,n}, LAT_{n-1} - s_{n-1} - t_{n-1,0}\right\} & \forall n-1 \in \mathscr{C}\backslash\mathscr{C}^+ \end{cases} \tag{64}$$

$$LVT_{n-1} = \begin{cases} LT_{n-1} & \forall n-1 \in \mathscr{C}^+ \\ min\left\{LVT_n - s_{n-1} - t_{n-1,n}, LAT_{n-1} - s_{n-1} - t_{n-1,0}\right\} & \forall n-1 \in \mathscr{C} \backslash \mathscr{C}^+ \end{cases} \tag{65}$$

Step 4: Update the ride time for each request served by the new label L_k^j

$$DR_n = \begin{cases} EVT_j - EVT_n & \forall n \in P_j \\ EVT_j - r_k & \forall n \in \mathscr{O}_k \end{cases} \tag{66}$$

Step 5: Update the latest arrival time for L_k^j

The latest arrival time of the new label can be updated by Eq. (67), and the extension is infeasible if $t_{j,0} + s_j + EVT_j > min\{LA_i, LAT_j\}$.

$$LA_j = min\{LA_i, LAT_j\} \tag{67}$$

Step 6: Renew the remaining in-vehicle travel time and the remaining number of co-riders without penalty costs for each served request

$$LIVT_n = max\left\{0, IVT_n - DR_n, \forall n \in P_j \cup \mathscr{O}_k\right\} \tag{68}$$

$$LCO_n = max\left\{0, CO_n - (V_j - p_n), \forall n \in P_j \cup \mathscr{O}_k\right\} \tag{69}$$

Step 7: Renew the inextensible nodes for the new label

Let $U_j \leftarrow P_j \cup U_i$, then the above steps can be utilized to check whether it is feasible to extend the label L_k^j to node $n \in \mathscr{C} \backslash U_i$. If the extension to node n is infeasible, the inextensible nodes will be : $U_j \leftarrow U_j \cup \{n\}$.

Step 8: Update the reduced cost for the new label L_k^j

$$C_j = f_k + c_k \sum_{n \in P_j \backslash \{g_k\}} d_{n-1,n} + \sum_{n \in P_j \cup \mathscr{O}_k} \beta \; min\{1, max\{0, \lceil DR_n - IVT_n \rceil, V_j - p_n - CO_n\}\} - \sum_{n \in P_j} \alpha_n^s \pi_n - \omega_k \tag{70}$$

(3) Dominance Rule

Consider two different labels ending at the same request i : $L_k^{i1} = \{C_i^1, V_i^1, U_i^1, P_i^1, LA_i^1, \{LIVT_n^1, LCO_n^1, DR_n^1, EVT_n^1, LVT_n^1, \forall n \in P_i \backslash \{g_k\}\}\}$ and $L_k^{i2} = \{C_i^2, V_i^2, U_i^2, P_i^2, LA_i^2, \{LIVT_n^2, LCO_n^2, DR_n^2, EVT_n^2, LVT_n^2, \forall n \in P_i \backslash \{g_k\}\}\}$. Let $\mathbb{Q}(L_k^{i1})$ be the set of feasible extensions of label L_k^{i1} to the station. For each $\ell \in \mathbb{Q}(L_k^{i1})$, we use $L_k^{i1} \oplus \ell$ denote a feasible complete path by concatenating ℓ to label L_k^{i1}, and $C(L_k^{i1} \oplus \ell)$ is the resulting reduced cost. Then, the standard dominance rule concluded by Dabia et al. (2013) can be introduced, namely Proposition 1.

Proposition 1. *For labels L_k^{i1} and L_k^{i2} ending at the same node i, L_k^{i1} dominates L_k^{i2} if conditions (a) and (b) hold.*

$$\mathbb{Q}(L_k^{i2}) \subseteq \mathbb{Q}(L_k^{i1}) \tag{a}$$

$$C(L_k^{i1} \oplus \ell) \le C(L_k^{i2} \oplus \ell), \forall \ell \in \mathbb{Q}(L_k^{i2}) \tag{b}$$

See Dabia et al. (2013) for proofing of Proposition 1. Condition (a) means that L_k^{i1} can concatenate with more feasible extensions to the station to obtain a complete path than L_k^{i2}. To ensure that condition (a) always holds in this PSP problem, we must ensure that L_k^{i1} has more seats available, fewer inextensible nodes, a later latest arrival time, an earlier earliest pickup time, and a later latest pickup time at request i. In the conventional shortest path problem to minimize total transportation costs, condition (b) is met if L_k^{i1} generates a smaller reduced cost than L_k^{i2}. However, a lower reduced cost does not always guarantee that condition (b) holds for this PSP, as the same extension may result in different penalty cost increases for L_k^{i1} and L_k^{i2}. Fig. 7 provides an illustrative example. Before extending to request m, we assume that C_i^1 is lower than C_i^2. However, after extending to request m, label L_k^{i1} incurs more penalties than label L_k^{i2}, which may result in $C_m^1 > C_m^2$. As a result, we must take additional measures to ensure that condition (b) always holds. The most straightforward approach is to check that each request (served by L_k^{i1} and L_k^{i2}) in L_k^{i1} has a longer remaining ride time and a larger number of co-riders than L_k^{i2}. Consequently, the following conditions need to be checked to identify whether L_k^{i1} dominates L_k^{i2}.

$$\text{Lower reduced cost: } C_i^1 \le C_i^2 \tag{I}$$

$$LCO_n^1 \ge LCO_n^2, and \; LIVT_n^1 \ge LIVT_n^2, \forall n \in P_i^1 \cup P_i^2 \cup \mathscr{O}_k \tag{II}$$

$$\text{Fewer occupied seats: } V_i^1 \le V_i^2 \tag{III}$$

$$\text{Fewer inextensible nodes: } U_i^1 \le U_i^2 \tag{IV}$$

$$\text{Longer latest arrival time: } LA_i^1 \ge LA_i^2 \tag{V}$$

$$\text{Smaller earliest pickup time at request } i\text{: } EVT_i^1 \le EVT_i^2 \tag{VI}$$

$$\text{Larger latest pickup time at request } j\text{: } LVT_i^1 \ge LVT_i^2 \tag{VII}$$

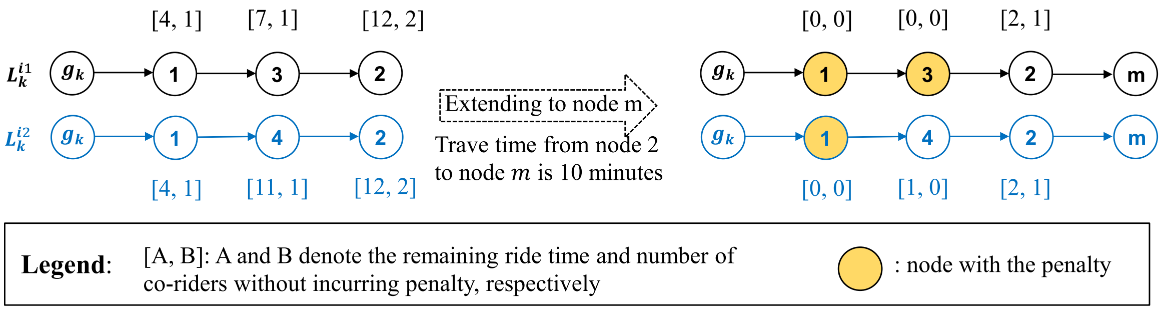


Fig. 7. Two different labels, ending with the same node 2, extend to the same node m but result in different penalties.

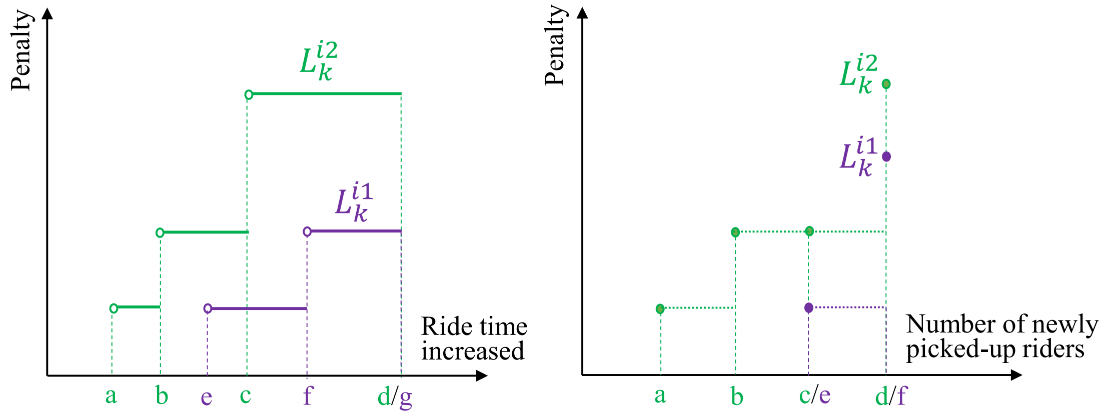


Fig. 8. Visualizations of penalty functions for the label extension.

Definition 4 (*Dominance Rule 1*)**.** If inequalities (I)–(VI) hold and at least one is strict, then L_k^{i1} dominates L_k^{i2}.

However, applying this dominance rule cannot eliminate too many unpromising labels since condition (II) is weak. As a result, we propose a stronger dominance rule. As shown in Fig. 8, the total penalty costs incurred by detour (or co-riding) for served requests of L_k^i are in the form of a non-decreasing step function (or a non-decreasing discrete-point function) as the ride time (or the number of co-riders) increases. Let $\Phi_{L_k^i}(u)$ and $\Phi_{L_k^i}(v)$ denote the total penalty costs for served requests of L_k^i when u riders and v ride time are added to L_k^i, respectively. Then, we can ensure that condition (b) in Proposition 1 always holds via inequality (VIII).

$$\Phi_{L_k^{i1}}(u) \quad \le \Phi_{L_k^{i2}}(u), \Psi_{L_k^{i1}}(v) \le \Psi_{L_k^{i2}}(v), \forall u \in \mathbb{Z}^+, u \le min\{q_k - V_i^1, q_k - V_i^2\}, \forall v > 0, v \le min\{LA_i^1 - EVT_j(L_k^{i1}), LA_i^2 - EVT_j(L_k^{i2})\} \tag{VIII}$$

Given the characteristics of the non-decreasing step/discrete-point function, inequality (VIII) can be met if the penalty of L_k^{i1} is always lower than that of L_k^{i2} at each inflection point (such as a, b, c, d, e, f, and g in Fig. 8(a)) of the penalty functions for L_k^{i1} and L_k^{i2}. The inflection points are all attributed to the increased ride time exceeding the detour tolerance of some requests. Consequently, we just need to ensure that the penalty of L_k^{i1} is always lower than that of L_k^{i2} when the increased ride time equals the remaining ride time without penalty for each request in the labels L_k^{i1} and L_k^{i2}, as well as when the increased ride time equals the minimum difference between the latest arrival time and the earliest pickup time for L_k^{i1} and L_k^{i2}. Likewise, the penalty corresponding to the co-riding can be treated in this way. This simplifies the check for inequality (VIII).

Definition 5 (*Dominance Rule**)**.** If inequalities (VIII), (I), and (III)–(VII) hold and at least one is strict, then L_k^{i1} dominates L_k^{i2}.

Based on these preparations, the standard framework of the labeling algorithm (see, for example, Feillet et al. (2004) and Irnich and Desaulniers (2005)) can be employed for this problem.

4.3.3. Heuristic labeling algorithm

The dominance rule mentioned above is still not very strong. To accelerate the solving process for the pricing problem, we propose two heuristic dominance rules based on the aforementioned exact dominance rule, which enables more labels to be dominated. The computation time can be significantly reduced for the entire solving process via the heuristic labeling algorithm, but the optimality of the subproblem is sacrificed. Therefore, when there are no negative columns that can be identified by the heuristic labeling algorithm, the exact labeling algorithm becomes necessary as a supplement.

Heuristic dominance rule 1. If inequalities (I) and (III)–(VII) hold, L_k^{i1} dominates L_k^{i2}.

As inequalities (II) and (IV) are too strict in dominance rule *, we can remove them to derive this heuristic dominance rule, which helps eliminate more labels and therefore effectively improves the computational efficiency of the labeling algorithm, but may also eliminate some promising labels that constitute the optimal route.

Prior to presenting heuristic dominance rule 2, we provide the following two definitions.

- $LCO(L_k^i)$: remaining number of co-riders without co-riding penalty costs for label L_k^i.
- $LIVT(L_k^i)$: remaining ride time without exceeding detour tolerances for requests of label L_k^i.

Heuristic dominance rule 2. If inequalities (I), (III)–(VII) and (IX) hold, then L_k^{i1} dominates L_k^{i2}.

$$LCO(L_k^{i1}) \geq LCO(L_k^{i2}), LIVT(L_k^{i1}) \geq LIVT(L_k^{i2}) \tag{IX}$$

The inequality (IX) relaxes (II) since (IX) focuses only on the minimum remaining ride time and the number of co-riders for all requests on the label, which also helps to find high-quality routes and facilitates a reduction in computation time for the labeling algorithm.

4.4. Branch-and-price algorithm

The column generation procedure can be used to solve the linear relaxation of the restricted master problem. However, the solution obtained may not be an integer solution. In such cases, the branch-and-bound tree is employed to search for an integer solution. The tree is explored using the best-first search strategy, whereby the node with the lowest lower bound will be explored first.

Once the fractional solution is found at a tree node, we will branch on the arc flow variables. First, we need to determine the values of each arc flow variable $x_{i,j} = \sum_{s \in S} \alpha_{i,j}^s \vartheta_s$ Then, we select the $arc\,(i,j) \leftarrow argmin|x_{i,j} - 0.5|$ and generate two child nodes with imposed constraints $x_{i,j} = 0$ and $x_{i,j} = 1$, respectively. Accordingly, the directed graph $G(\mathcal{N}, \mathcal{A})$ and the columns of the restricted master problem will be modified. Details are as follows:

- Impose the constraint $x_{i,j} = 0$ to one child node.

At this node, the $arc\,(i,j)$ is forbidden in the subproblem and master problem. Therefore, the distance of the $arc\,(i,j)$ can be set to positive infinity, and columns containing this arc are removed from the master problem.

- Impose the constraint $x_{i,j} = 1$ to another child node.

At this node, the successor of node i must be node j. Accordingly, the distances of $arcs\,\{(i,k)\,|k! = j\}$ and $arcs\,\{(m,j)\,|m! = i\}$ can be set to positive infinity, and columns that contain node i without the successor node j and contain node j without the predecessor node i are deleted from the master problem.

4.5. Column-generation matheuristic algorithm

The process of closing the optimality gap of the branch-and-bound tree may be slow, even if each node already provides a tight linear relaxation gap thanks to the property of the set-partitioning model. To obtain high-quality solutions more efficiently, we design a column-generation matheuristic algorithm focusing on the root node of the branch-and-bound tree. The column-generation matheuristic algorithm incorporates the following two techniques:

- **Technique1: reusing columns of other vehicles**

As mentioned before, we solve the subproblem to identify a trip for each vehicle. However, some vehicles with similar properties, such as the ready time, the capacity, and the latest return time, can use the same trip. As a result, before applying the labeling algorithm to determine the best trip for the vehicle, we can check if the trips generated for other vehicles in the current iteration of the column-generation matheuristic can be employed directly for this vehicle. By doing so, the overall computation time can be reduced, and the diversity of the column pool can be enhanced.

- **Technique2: incremental column generation**

First, we incrementally set the number of visited requests (N) from 1 to L (Jin et al., 2021) for the pricing problem. With the given number of visited requests for each route, we employ column generation with technique 1 to generate the column set $\mathcal{S}_2$. Without the restriction on the number of visited requests for the pricing problem, the column generation procedure tends to generate columns (trips) visiting requests as many as possible. However, the optimal integer solution often consists of routes with a mixed number of visited requests (Jin et al., 2021). The incremental column generation procedure can promote the diversity of columns in terms of route length.

The procedure of the column-generation matheuristic algorithm is as follows, with a flowchart in Fig. 9.

Step 1: Initializing the column set $\mathcal{S}_0$.

Step 2: Solving the RMP and PSP with techniques 1 and 2 to get the column set $\mathcal{S}_1$.

Step 3: Solving the RMP and PSP with technique 1 to get the column set $\mathcal{S}_2$.

Step 4: Converting variables ϑ_s into binary variables and determining the integer solution based on the union of $\mathcal{S}_1$ and $\mathcal{S}_2$.

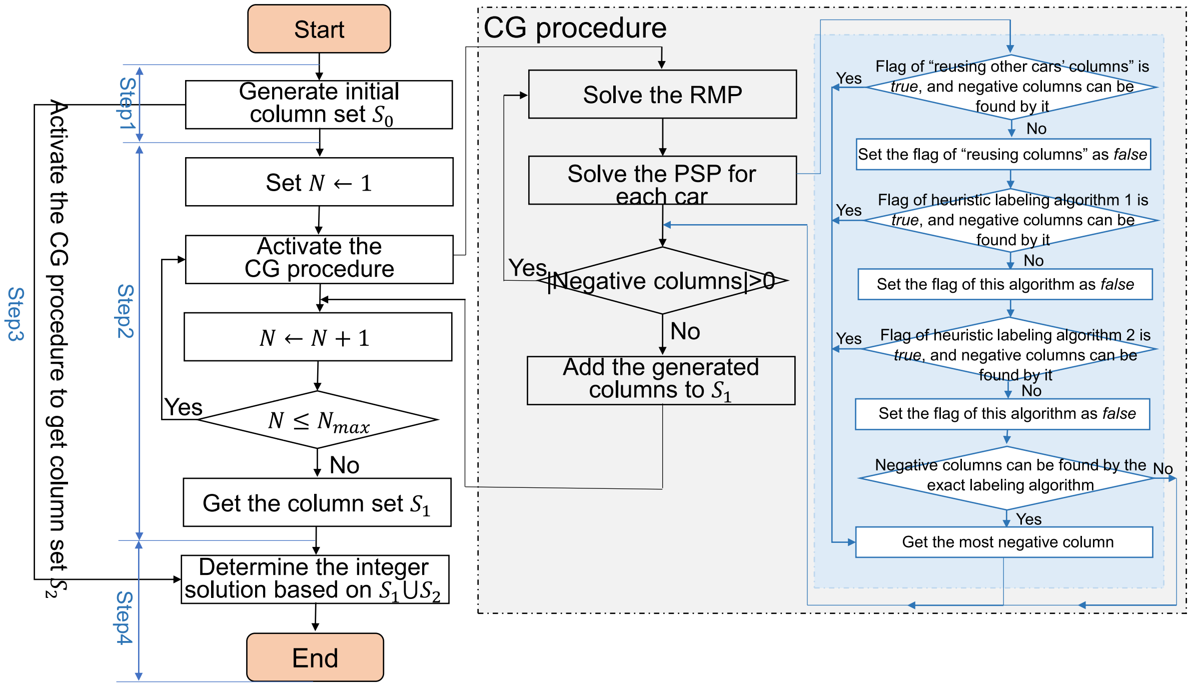

Start
Generate initial column set S_0
Set $N \leftarrow 1$
Activate the CG procedure
$N \leftarrow N+1$
Yes
$N \leq N_{max}$
No
Get the column set S_1
Determine the integer solution based on $S_1 \cup S_2$
End
Step1
Step2
Step3
Step4
Activate the CG procedure to get column set S_2
CG procedure
Solve the RMP
Solve the PSP for each car
Yes
|Negative columns|>0
No
Add the generated columns to S_1
Flag of "reusing other cars' columns" is true, and negative columns can be found by it
Yes
No
Set the flag of "reusing columns" as false
Flag of heuristic labeling algorithm 1 is true, and negative columns can be found by it
Yes
No
Set the flag of this algorithm as false
Flag of heuristic labeling algorithm 2 is true, and negative columns can be found by it
Yes
No
Set the flag of this algorithm as false
Negative columns can be found by the exact labeling algorithm
No
Yes
Get the most negative column

Fig. 9. Flowchart of the column-generation matheuristic algorithm.

4.6. An adaptive large neighborhood search algorithm

We leverage the adaptive large neighborhood search algorithm (ALNS) to evaluate the performance of the column generation algorithm. The main steps of the ALNS algorithm are outlined in Pisinger and Ropke (2007). First, an initial solution is constructed by randomly inserting each request into a feasible position of a car's route. Then, four removal operators and four repair operators are utilized to search for neighborhood solutions, with the Metropolis rule accepting the new solution. Note that Algorithm 1 can be directly employed in this ALNS algorithm to determine the best schedule for a route.

4.6.1. Removal operators

Random Removal. Requests can be randomly chosen and removed from the routes.

Worst Removal. First, we calculate the cost reduction for each request after it is removed. Then, the request with the highest cost reduction is removed from the current solution. Third, repeat these two steps to remove the remaining requests that need to be removed.

Route Removal. A route is chosen randomly and removed.

Shaw Removal. We define three shaw removal operators based on the closeness ($CL(i,j)$) of two requests i and j on travel time ($CL(i,j) = t_{i,j}$), latest arrival time and maximum ride time ($CL(i,j) = |IVT_i - IVT_j| + |LAT_i - LAT_j|$), and reachability ($CL(i,j) = max\{|ET_i - LT_j - t_{i,j} - s_i|, |ET_j - LT_i - t_{j,i} - s_j|\}$), respectively.

4.6.2. Repair operators

Random Repair. Requests are randomly inserted into a feasible insertion position.

Best Repair. Requests will be inserted into the lowest-cost position if feasible insertion positions exist in the available routes.

Greedy Repair. We first compute the insertion costs for each request at every available position. Then, we select the lowest-cost request-position pair as the request and position for insertion. If the lowest insertion cost equals M (an extremely large number used to indicate the infeasibility of a route), then there is no feasible position to insert this request.

Regret Repair. The regret repair operator (Pisinger and Ropke, 2007), a variant of the greedy repair operator, uses the look-ahead information to select the request and position for insertion. In this study, 2-regret and 3-regret repair operators are employed and are randomly selected in each iteration.

Note that during the repair process, some requests may not have any feasible positions for insertion. In such cases, these requests are not inserted in the current iteration and are instead assigned a penalty value of M.

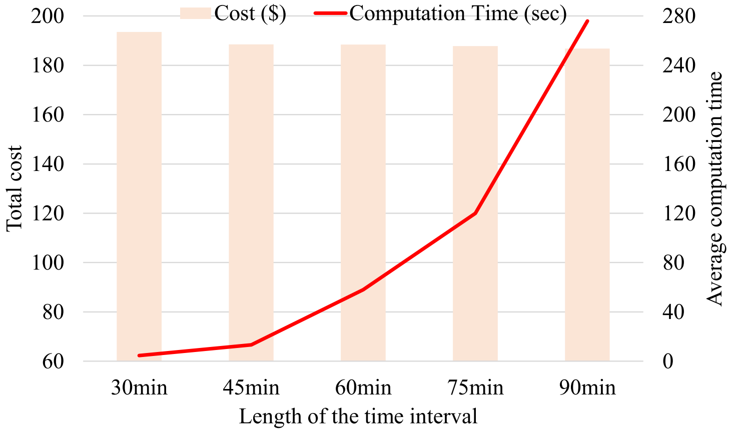


Fig. 10. Total cost and average computation time under different lengths of time horizon.

5. Case study

To evaluate the performance of the proposed branch-and-price method and the column-generation matheuristic approach, we conducted computational experiments based on real-world case studies and randomly generated test instances. The algorithms are coded in C++ and run on a 64-bit computer with a 2.3 GHz Intel Core i5-10500T processor and 8 GB RAM. The MILP is solved by the CPLEX 12.8. The maximum running time of the CPLEX and the branch-and-price algorithm is 7200 s. The maximum number of iterations of the ALNS algorithm is set to 20000. Other parameters of the ALNS algorithm are the same as He et al. (2023).

5.1. Instance generation and parameters settings

Since there is no available benchmark for this problem, we conducted experiments on randomly generated instances based on the Xinzhuang metro station in Shanghai. Each instance consists of requests and vehicles, with their locations randomly generated from the road network. The set of vehicles includes public and private vehicles, and it is assumed that there are enough vehicles to serve all requests. The number of riders for each request is set to 1 or 2, with the probability of 75% and 25%, respectively. The maximum ride time is calculated as the direct travel time plus the tolerable detour time (5 min + 15 × rand, Bian et al. (2022)). The latest arrival times are randomly generated. The earliest pickup times are also randomly generated within the current time horizon, for instance [8:00,9:00]. The latest pickup times for new incoming requests in this time horizon are set to the latest arrival time minus the direct travel time and service time. Each request's maximum number of co-riders is randomly generated from 2, 3, or 4. Other parameters are as follows:

- 4-seat cars are available;
- The fixed cost of hiring a private vehicle that has not yet been scheduled previously is $10;
- The travel costs of public and private vehicles are 0.5 and 0.8 $/mile, respectively;
- The service duration for each request is 1 min;
- The penalty cost coefficient (β) is 3 $/person;
- The vehicle's travel speed is 25 km/h.

To determine the best length for each time horizon, we conduct experiments with five different time intervals, using 180 requests that are randomly distributed over three hours. The results, shown in Fig. 10, indicated that the total operating cost will decrease with the length of intervals increasing, with a cost reduction of 4% when the length increased from 30 min to 90 min. However, the computation time varies significantly with different intervals, with the average computation time increasing from 5 s to 276 s when the length increases from 30 min to 90 min. As a result, we rule out the 90-minute interval and select 60 min as the best interval based on both computation time and operating cost. Compared to the 90-minute interval, using the 60-minute interval results in only a 1% increase in operating costs, while the computation time is very efficient at only 58 s. In addition, the time interval we chose is consistent with Bian's work (Bian et al., 2022).

5.2. Computational performance

To assess the computational performance of the column-generation matheuristic algorithm, we compare the results with the CPLEX solver, the branch-and-price algorithm, and the ALNS algorithm based on small- and large-scale instances.

Table 4
Computational performance of small-scale instances.

Instances		CPLEX solver			Branch-and-Price		ALNS		CG matheuristic		Gap* (%)		
		Obj	Time	Gap(%)	Obj^{BP}	Time	Obj^{ALNS}	Time	Obj^{CGM}	Time	Gap1	Gap2	Gap3
(1)		(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)
Set1	6_5_1	21.46	0.2	0.00	21.46	0.1	21.46	1.9	21.46	0.2	0.00	0.00	0.00
	7_5_1	22.66	0.6	0.00	22.66	0.1	22.66	2.0	22.66	0.2	0.00	0.00	0.00
	8_5_2	24.17	1.2	0.00	24.17	0.1	24.17	2.4	24.17	0.3	0.00	0.00	0.00
	9_5_2	24.39	1.3	0.00	24.39	0.1	24.39	3.1	24.39	0.4	0.00	0.00	0.00
	10_5_3	27.35	5.1	0.00	27.35	0.1	27.35	4.6	27.35	0.5	0.00	0.00	0.00
	11_5_3	32.52	9.3	0.00	32.52	0.1	32.52	4.1	32.52	0.4	0.00	0.00	0.00
	12_5_4	36.05	9.1	0.00	36.05	0.2	36.05	5.3	36.05	0.6	0.00	0.00	0.00
	13_5_4	37.61	144.8	0.00	37.61	0.2	37.61	5.8	37.61	0.7	0.00	0.00	0.00
	14_5_5	43.42	1928.6	0.00	43.42	1.8	43.42	7.3	43.42	0.7	0.00	0.00	0.00
	15_5_5	51.86	7206.3	6.02	51.86	3.4	51.86	8.4	51.86	0.8	0.00	0.00	0.00
	16_5_6	60.87	7201.2	19.70	60.87	0.7	60.87	9.7	60.87	1.1	0.00	0.00	0.00
	17_5_6	65.05	7200.1	13.20	65.05	1.2	65.05	11.7	65.05	1.2	0.00	0.00	0.00
	18_5_7	60.13	7217.7	24.60	60.13	0.6	60.13	13.9	60.13	1.4	0.00	0.00	0.00
	19_5_7	64.38	7206.3	34.90	64.38	6.4	64.38	13.1	64.38	1.5	0.00	0.00	0.00
	20_5_8	71.04	7224.6	53.10	71.04	0.9	71.04	15.5	71.04	1.8	0.00	0.00	0.00
Set2	6_6_0	18.81	0.1	0.00	18.81	0.1	18.81	1.9	18.81	0.2	0.00	0.00	0.00
	7_6_0	20.02	0.3	0.00	20.02	0.1	20.02	2.1	20.02	0.2	0.00	0.00	0.00
	8_7_0	20.26	1.4	0.00	20.26	0.1	20.26	3.7	20.26	0.2	0.00	0.00	0.00
	9_7_0	20.48	2.5	0.00	20.48	0.1	20.48	3.3	20.48	0.2	0.00	0.00	0.00
	10_8_0	23.54	3.2	0.00	23.54	2.3	23.54	4.2	23.54	0.3	0.00	0.00	0.00
	11_8_0	27.04	8.2	0.00	27.04	0.1	27.04	4.4	27.04	0.3	0.00	0.00	0.00
	12_9_0	29.02	48.2	0.00	29.02	0.1	29.02	5.4	29.02	0.4	0.00	0.00	0.00
	13_9_0	29.95	195.9	0.00	29.95	0.2	29.95	5.6	29.95	0.6	0.00	0.00	0.00
	14_10_0	29.57	274.8	0.00	29.57	0.2	29.57	7.7	29.57	0.5	0.00	0.00	0.00
	15_10_0	33.57	5490.5	0.00	33.57	0.2	33.57	8.0	33.57	0.7	0.00	0.00	0.00
	16_11_0	37.19	7215.4	27.60	37.19	0.2	37.19	8.4	37.19	0.8	0.00	0.00	0.00
	17_11_0	35.14	7261.4	35.60	35.14	0.4	35.14	10.5	35.14	1.0	0.00	0.00	0.00
	18_12_0	33.94	7232.9	36.30	33.94	0.4	33.94	12.8	33.94	1.2	0.00	0.00	0.00
	19_12_0	34.13	7226.8	41.30	34.13	0.6	34.13	13.0	34.13	1.3	0.00	0.00	0.00
	20_13_0	38.23	7230.2	46.00	38.23	0.5	38.23	15.8	38.23	1.5	0.00	0.00	0.00
Average		**35.80**	**2918.27**	**11.96**	**35.80**	**0.7**	**35.80**	**7.2**	**35.80**	**0.7**	**0.00**	**0.00**	**0.00**

Note: Obj, Obj^{BP}, Obj^{ALNS}, and, Obj^{CGM} are the best objective value identified by the CPLEX solver, the Branch-and-price algorithm, the ALNS algorithm, and the CG math-heuristic algorithm, respectively. Column of 'Time' is the computation time of the corresponding algorithm. Units of 'Obj' and 'Time' are $ and seconds, respectively. Gap1 = [(5) - (2)]/(2)×100%, Gap2 = [(7) - (2)]/(2)×100%, and Gap3 = [(9) - (2)]/(2)×100%.

5.2.1. *Small-scale instances*

The experiments are first conducted on two sets of small-scale instances, each set consisting of 15 instances with the number of requests ranging from 6 to 20. In set 1, vehicles include en-route publicly- and privately-owned vehicles, publicly- and privately-owned vehicles parked in the station, and privately-owned vehicles parked in garages. In set 2, all vehicles are public vehicles with the state of parking at the station or en route. The detailed computational results are included in Table 4. The first column "Instance" is named in the format of "A_B_C", with "A" denoting the number of requests, "B" indicating the number of public vehicles, and "C" representing the number of private vehicles. The columns "Obj", "Obj^{BP}", "Obj^{ALNS}", and "Obj^{CGM}" are the objective values obtained by the CPLEX solver, the branch-and-price algorithm, the ALNS algorithm, and the CGM algorithm, respectively. It is worth noting that we run the ALNS algorithm five times for each instance, and we find that the objective values for each instance are identical. This means that the average and best objective values obtained by the ALNS algorithm for small-scale instances are equivalent.

As shown in Table 4, CPLEX can find the optimal solutions for all small-scale instances, but it can only prove optimality for some instances within a time limit of 7200 s. In terms of solution quality, all the algorithms can find optimal solutions for these instances. However, regarding the computation time, both the branch-and-price algorithm and the CGM algorithm perform better than the ALNS algorithm.

5.2.2. *Large-scale instances*

To evaluate the computational performance of the CGM algorithm on large-scale instances, we compare its results with those of the branch-and-price algorithm and the ALNS algorithm on 16 instances, each with 20 to 50 requests and 15 public vehicles. Table 5 summarizes the detailed results. The meaning of columns "Obj^{BP}" and "Obj^{CGM}" is similar to that in Table 4. The columns "Obj^{Best}" and "Obj^{Ave}" are the best objective values and the average objective values identified by running the ALNS algorithm five times, respectively. In terms of solution quality, the CGM algorithm performs remarkably well in obtaining high-quality solutions, with a maximum gap of only 3.75 compared to the exact solution. Moreover, the vast majority of its solutions are better than or equal to those found by the ALNS algorithm. In terms of computational efficiency, the CGM algorithm outperforms the other two algorithms, being capable of solving all instances in less than 35 s.

Table 5
Computational performance of large-scale instances.

Instances	Branch-and-Price		ALNS			CG matheuristic		Gap	
	Obj^{BP} ($)	Time (s)	Obj^{Best} ($)	Obj^{Ave} ($)	Time (s)	Obj^{CGM} ($)	Time (s)	Gap1 (%)	Gap2 (%)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
20_15_0	46.10	0.5	46.10	46.10	25.4	46.10	1.5	0.00	0.00
22_15_1	46.58	0.7	46.58	46.58	27.5	46.58	2.0	0.00	0.00
24_15_2	56.96	3.9	56.96	56.96	27.4	56.96	3.1	0.00	0.00
26_15_3	67.05	55.40	67.05	67.05	34.7	67.05	4.3	0.00	0.00
28_15_4	72.71	125.1	72.71	73.11	38.0	72.71	4.3	0.00	0.00
30_15_5	79.90	8.2	79.90	80.59	54.8	79.90	5.1	0.00	0.00
32_15_6	88.59	156.2	88.59	89.31	63.5	88.59	6.6	0.00	0.00
34_15_7	91.70	101.4	91.70	92.83	80.7	92.89	8.4	0.00	1.30
36_15_8	97.20	91.1	97.20	98.79	86.3	97.20	8.3	0.00	0.00
38_15_9	106.34	151.2	114.25	116.30	90.6	107.55	11.3	7.44	1.14
40_15_10	111.45	64.2	120.61	123.48	114.9	111.45	13.6	8.22	0.00
42_15_11	124.57	20.2	133.19	136.86	124.0	129.24	15.6	6.92	3.75
44_15_12	115.77	314.8	123.49	126.48	148.4	118.31	25.3	6.67	2.19
46_15_13	124.40	220.5	129.12	133.85	169.2	124.40	33.8	3.79	0.00
48_15_14	130.26	160.5	136.53	140.93	187.5	130.39	33.0	4.81	0.10
50_15_15	135.64	194.5	142.16	145.97	220.3	137.43	33.6	4.81	1.32
Average	**93.45**	**104.3**	**96.63**	**98.45**	**93.3**	**94.22**	**13.1**	**2.67**	**0.61**

*Gap1 = [(4) - (2)]/(2)×100%; Gap2 = [(7) - (2)]/(2)×100%.

Table 6
Computational performance of large-scale instances without incorporating private vehicles.

Instances	Branch and Price		ALNS			CG Heuristic		Gap	
	Obj^{BP} ($)	Time (s)	Obj^{Best} ($)	Obj^{Ave} ($)	Time (s)	Obj^{CGM} ($)	Time (s)	Gap1 (%)	Gap2 (%)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
20_15	36.87	0.7	36.87	36.87	18.3	36.87	0.8	0.00	0.00
22_16	39.51	1.1	39.51	39.51	20.8	39.51	1.2	0.00	0.00
24_17	41.72	1.3	41.72	41.72	30.1	41.72	1.4	0.00	0.00
26_18	47.44	2.5	47.44	47.44	31.7	47.44	1.6	0.00	0.00
28_19	50.98	3.2	50.98	50.98	38.8	50.98	2.1	0.00	0.00
30_20	51.60	4.7	51.60	51.60	46.6	51.60	2.1	0.00	0.00
32_21	53.58	3.7	53.58	53.58	53.6	53.58	3.1	0.00	0.00
34_22	57.08	4.3	57.08	57.77	61.3	57.08	5.5	0.00	0.00
36_23	60.15	10.4	60.15	60.15	74.5	60.97	6.4	0.00	1.36
38_24	60.62	16.4	60.62	60.82	80.4	60.62	6.2	0.00	0.00
40_25	62.36	13.6	62.36	62.40	88.4	64.43	7.6	0.00	0.00
42_26	74.10	14.2	74.10	74.60	107.7	74.10	11.2	0.00	0.00
44_27	82.03	166.9	82.03	83.50	147.1	82.44	11.1	0.00	0.50
46_28	70.04	48.1	70.23	70.75	146.3	70.23	15.2	0.27	0.27
48_29	70.78	38.6	70.78	70.83	171.8	70.78	20.2	0.00	0.00
50_30	74.73	458.8	74.93	75.97	189.3	75.84	26.1	0.27	1.49
Average	**58.36**	**49.3**	**58.37**	**58.66**	**74.5**	**58.51**	**7.7**	**0.03**	**0.23**

*Gap1 = [(4) - (2)]/(2)×100%; Gap2 = [(7) - (2)]/(2)×100%.

To demonstrate that incorporating private vehicles would increase the complexity of decision-making, we conduct additional experiments on the same instances with all vehicles being homogeneous publicly-owned vehicles, and the results are reported in Table 6. In terms of solution quality, both the ALNS and the CGM algorithms outperform the scenario considering private cars, as their average gap decreased from 2.67% and 0.61% to 0.03% and 0.23%, respectively. This implies that the presence of private vehicles increases the decision-making challenge, as the heterogeneous fleet may enlarge the solution space. It should be noted that in this scenario, the ALNS algorithm outperforms the CGM algorithm in terms of solution quality. Combining the results in Table 5, we can conclude that the ALNS algorithm is more suitable for scenarios without private cars, while the CGM algorithm is better suited for scenarios where private cars are included in the fleet. With regard to computation time, all algorithms exhibit decreased computation time, further proving the increased complexity of decision-making with the inclusion of a private fleet.

5.3. Benefits and drawbacks of ridesharing

To identify the benefits and drawbacks associated with first-mile ridesharing, we compared the total travel distances, a measure of total operating costs, and in-vehicle travel time variation with the non-ridesharing mode where one car serves only one request per trip. All the large-scale instances in Section 5.2.2 are used to conduct the experiment, and the results are reported in Fig. 11, which displays the distribution of distance-savings and in-vehicle ride time increases of ridesharing for these instances. We can conclude that ridesharing is effective in reducing travel distances while only moderately increasing riders' in-vehicle travel time.

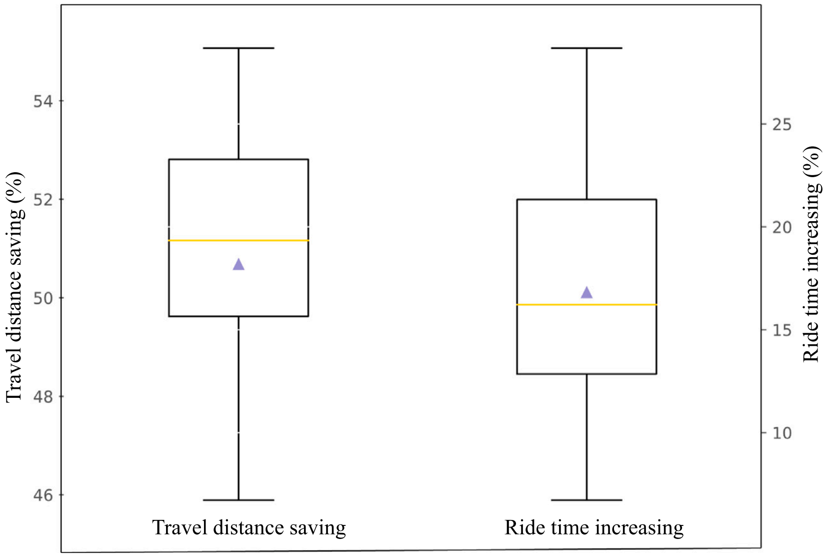


Fig. 11. Benefits and drawbacks of ridesharing compared to the non-ridesharing model.

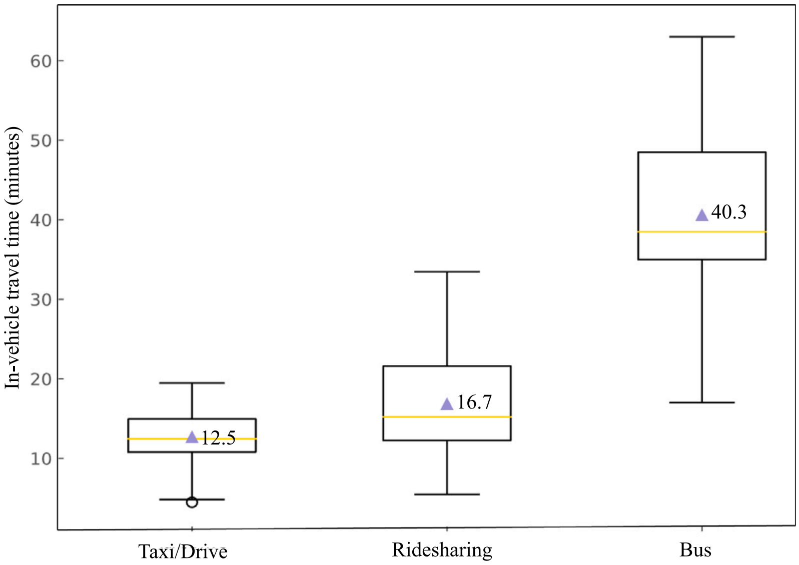


Fig. 12. Travel times under different first-mile travel modes at Xinzhuang station.

On average, the percentage of distance saved is about three times the percentage increase in ride time. Specifically, the average distance-saving percentage is 50.67%, while the percentage increase in ride time is 16.81%. Additionally, we also compared riders' travel time across different travel modes, as shown in Fig. 12. Although ridesharing increases riders' in-vehicle travel time when compared to non-ridesharing modes such as travel by taxi, the travel time remains highly attractive when compared to bus travel. For instance, the average bus journey takes 40.3 min, whereas the average travel time for ridesharing is less than 16.7 min.

5.4. Impact of operating costs of private vehicles on ridesharing schemes

To analyze how the difference in operating costs between private and public vehicles affects vehicle dispatching, total operating costs, and passengers' ride times. Two groups of experiments, different and identical operating costs for privately-owned and publicly-owned vehicles, are conducted for each large-scale instance in Section 5.2.2, and the results are shown in Table 7. The "Used car" column denotes the constitution of scheduled cars, defined in the form of "Q/W/E". Here, "Q" denotes the number of scheduled public vehicles, "W" indicates the total number of scheduled private vehicles, and "E" represents the number of newly hired private vehicles that are parked in garages and have not been scheduled before. Two conclusions are drawn:

Table 7
Results comparing under the same/different operating costs for private and public vehicles.

Instance (1)	Different operating cost			Same operating cost			Results comparing	
	Operator's cost($) (2)	Used cars (3)	In-vehicle time (m) (4)	Operator's cost($) (5)	Used cars (6)	In-vehicle time (m) (7)	Cost saving(%) (8)	Time increasing(%) (9)
20_15_0	46.10	10/0/0	16.13	46.10	10/0/0	16.13	0	0
22_15_1	46.58	10/0/0	16.48	43.71	9/1/1	16.52	6.16	−0.24
24_15_2	56.96	11/0/0	16.89	51.61	10/2/2	17.69	9.39	−4.52
26_15_3	67.05	12/0/0	16.69	55.40	10/2/2	17.96	17.38	−7.07
28_15_4	72.71	12/1/0	16.67	58.59	10/3/2	17.49	19.42	−4.69
30_15_5	79.90	13/1/0	17.62	65.05	11/3/2	16.93	18.59	4.08
32_15_6	88.59	13/2/0	17.53	70.15	10/5/2	17.22	20.81	1.80
34_15_7	92.89	13/3/0	16.28	68.58	11/5/3	16.54	26.17	−1.57
36_15_8	97.20	13/4/0	16.56	72.25	10/7/4	16.65	25.67	−0.54
38_15_9	107.55	13/4/0	17.46	74.11	11/8/5	16.06	31.09	8.72
40_15_10	111.45	12/5/1	17.09	74.09	11/8/6	16.06	33.52	6.41
42_15_11	129.24	13/5/1	16.87	79.04	11/8/6	16.41	38.84	2.80
44_15_12	118.31	14/6/1	16.76	76.08	12/9/6	16.06	35.69	4.36
46_15_13	124.40	14/6/1	16.75	79.04	11/10/7	16.95	36.46	−1.18
48_15_14	130.39	15/6/1	16.83	77.41	12/10/8	16.60	40.63	1.39
50_15_15	137.43	15/7/2	16.70	75.26	11/12/9	15.74	45.24	6.10
Average							25.32	0.99

*(8) = [(5) - (2)]/(2)×100%; (9) = [(4) - (7)]/(7)×100%.

(1) When the operating costs of private and public vehicles are identical, the more private vehicles scheduled, the lower the operating costs. Since privately-owned vehicles are typically parked in different garages, scheduling them to serve nearby demands can effectively reduce transport distances and thus operating costs. In addition, we conducted numerical experiments with differences in transportation costs per mile between private and public vehicles of $0.05, $0.1, $0.2, and $0.3, respectively. We found that when the transportation cost of private vehicles is higher, even if it is only by $0.05 per mile, priority should be given to publicly owned vehicles.

(2) The impact of scheduling private vehicles to provide feeder services on riders' in-vehicle travel times is not significant. For instance, the average in-vehicle travel time in both modes varied by only 0.99%.

5.5. Sensitivity analysis on penalty cost coefficients

To analyze the effect of penalty cost factors on the satisfaction rate of passenger preferences for co-riding and detouring, as well as the impact on the operating costs for the public transit agency, we conduct the experiment using the instances in Section 5.2.2 The results, showing the variation in total operating costs and the total number of unmet requests, are presented in Fig. 13. It can be observed that even a small penalty factor, such as 0.5, immediately makes the passenger preferences well satisfied, with unmet requests decreasing from 79.63% to 23.33%. As the penalty factor increases, the satisfaction rate also increases. When the penalty factor exceeds 2, all passengers' preferences are satisfied, with the operating cost increased by almost 1/7. The increase in cost is primarily due to some passengers' reluctance to share the trip with too many other people or to take longer routes. Therefore, taking measures to encourage passengers to share trips and tolerate detours as much as possible can help public transit agencies further reduce operating costs.

5.6. Simulation for dynamic scenarios

To verify the effectiveness of the models and algorithms in dynamic scenarios, we carry out a series of experiments to examine how ridesharing schemes respond to variations in demands and travel times. Initially, we establish a base scenario with ten requests and seven vehicles, comprising five private cars parked in garages and two public cars parked at the station. Next, as illustrated in Fig. 14(a), we employ the CG math-heuristic algorithm to generate a ridesharing scheme for this scenario, scheduling five vehicles to serve these requests. Subsequently, we assume that some road segments experience severe congestion during the execution of the ridesharing scheme, resulting in increased travel times for some requests. To ensure time-related requirements, such as travel time, are adequately met following this variation, we update the ridesharing scheme based on the new travel time information. Although the routes of vehicles 6 and 7 are modified, the new ridesharing scheme, as shown in Fig. 14(b), remains unchanged.

Second, we assume that during the execution of the ridesharing scheme in the base scenario, five new requests (requests 11, 12, 13, 14, and 15) entered, and request 8 canceled its order. In addition, three hireable vehicles entered the system, namely vehicles 8, 9, and 10. Based on the demand and vehicle information, we obtain a new ridesharing scheme as shown in Fig. 14(c). In this scheme, vehicle 5, after picking up request 6, instead of fulfilling the canceled order of request 8, picks up the new request 15. The newly available vehicles 8 and 10 are assigned to serve requests 11, 12, and 13. Meanwhile, new request 14 is picked up by vehicle 1 with a detour. The ridesharing scheme for other passengers and vehicles remains unchanged.

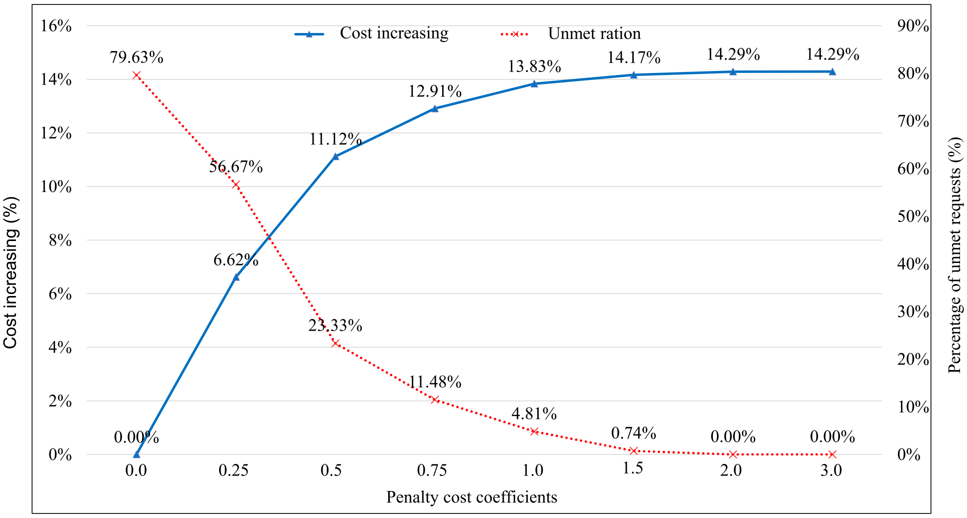


Fig. 13. Sensitivity analysis on the coefficient of penalty costs.

Third, we assume that there are simultaneous changes in demand and travel time in the two dynamic scenarios mentioned above. Similarly, we obtain a new ridesharing scheme for vehicles and passengers, as shown in Fig. 14(d). The ridesharing scheme in this scenario is consistent with the last scenario, but the travel routes of vehicles 6 and 7 have been altered. From these dynamic scenarios, we observe that our algorithm can adapt the ridesharing scheme to dynamic changes in both demand and travel time.

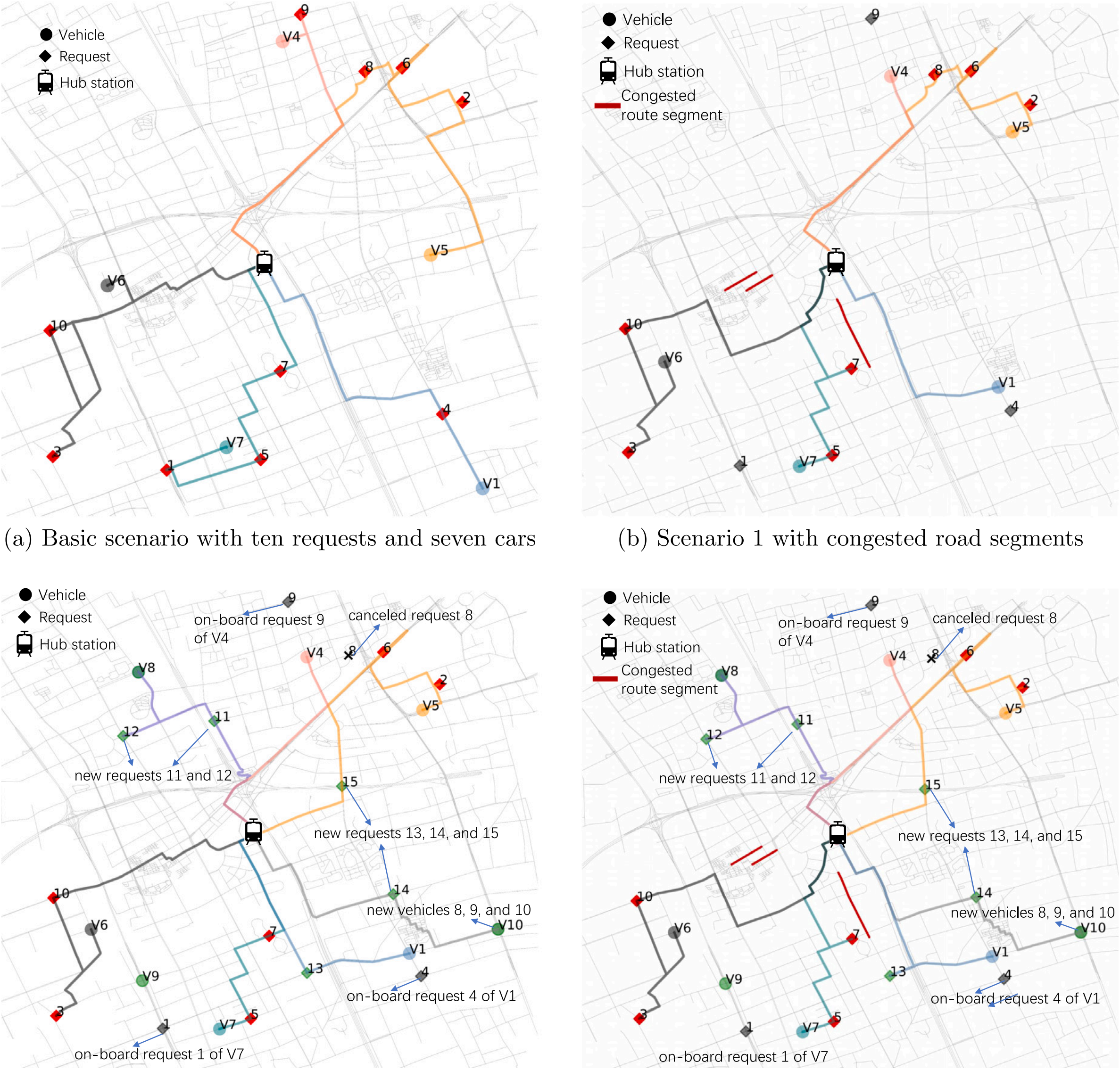
Based on these case studies, we could infer that: (1) Changes in passenger demand have a more pronounced impact on the ridesharing scheme compared to travel time uncertainty. This is because ridesharing schemes can still be executed by altering vehicle paths when there is congestion on specific road segments. However, significant changes in passenger demand require not only path modifications but also scaling the employed fleet to align with the demand variations. (2) Detours resulting from traffic congestion may increase passenger ride times and vehicle arrival times, while changes in passenger demand may lead to reductions in these values. For instance, in case 2, the average passenger ride time increased from 15.1 min to 16.9 min compared to case 1, as some paths were detoured. Conversely, average ride times decreased in cases 3 (13.5 min) and 4 (15.3 min) compared to cases 1 or 2.

6. Conclusion

In this paper, we investigate a first-mile issue where public transit agencies can integrate privately-owned autonomous vehicles during their idle time to provide high-quality first-mile ridesharing services, which involves both routing and time scheduling decisions. To address this issue, an arc-based MILP model is developed to minimize the total operating cost while considering the available time and location of vehicles, the return time of private vehicles, ride times, the number of co-riders, and latest arrival times. Then, we reformulate the MILP model as a trip-based set-partitioning model using the Dantzig–Wolfe decomposition and develop an exact branch-and-price algorithm. To identify promising trips, we propose a tailored labeling algorithm with a novel dominance rule, along with a time window shift algorithm to determine the best schedule. To yield near-optimal solutions in a shorter amount of computation time, we devise a customized column-generation matheuristic procedure that includes two strategies for generating more high-quality columns: (1) reusing columns of other vehicles; and (2) incremental column generation. Additionally, we develop an adaptive large neighborhood search algorithm to evaluate the performance of the column-generation algorithm.

Real-world case studies are conducted based on the road network in the vicinity of the Xinzhuang metro station. Results show that (i) the branch-and-price algorithm can obtain the optimal solution for each given instance within 500 s. (ii) The column-generation matheuristic algorithm can find high-quality solutions for instances in less than 35 s, outperforming the ALNS algorithm in terms of solution quality and computation time. (iii) Compared to the non-ridesharing mode, ridesharing can reduce travel distances by about 50% while maintaining service quality for passengers.

The current work still has some limitations and there are several promising research directions for future research. First, this study updates the ridesharing scheme using the rolling horizon method, without considering real-time request-vehicle matching and routing. As a result, developing a real-time algorithm to assign real-time requests to en-route vehicles would be a promising extension. Second, this work does not investigate how to determine the optimal publicly-owned fleet size for the first-mile transportation service provider. Although they can inflate the fleet by dynamically hiring privately-owned autonomous vehicles to meet travel demand, especially during rush hours, the number of publicly-owned vehicles that need to be deployed remains a key decision problem. Third, it would be worthwhile to investigate the mechanism design problem for boosting the willingness of private owners to participate in vehicle sharing.



(a) Basic scenario with ten requests and seven cars

(b) Scenario 1 with congested road segments

(c) Scenario 2 with demand changed

(d) Scenario 3 with both travel time and demand changed

Fig. 14. Ridesharing schemes for one base scenario and three dynamic scenarios.

CRediT authorship contribution statement

Ping He: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Conceptualization. **Jian Gang Jin:** Writing – review & editing, Writing – original draft, Methodology, Supervision, Funding acquisition, Conceptualization. **Martin Trépanier:** Writing – review & editing, Conceptualization. **Frederik Schulte:** Writing – review & editing, Conceptualization.

Acknowledgments

This work is supported by the National Natural Science Foundation of China [Grant 72122014, 72061127003], and National Key Research and Development Program of China [Grant 2020AAA0107600]. The first author appreciates the support of the China Scholarship Council [Grant 202106230253].

References

Abe, R., 2021. Preferences of urban rail users for first- and last-mile autonomous vehicles: Price and service elasticities of demand in a multimodal environment. Transp. Res. C 126, 103105.

Agussurja, L., Cheng, S.F., Lau, H.C., 2019. A state aggregation approach for stochastic multiperiod last-mile ride-sharing problems. Transp. Sci. 53 (1), 148–166.

Aldaihani, M.M., Quadrifoglio, L., Dessouky, M.M., Hall, R., 2004. Network design for a grid hybrid transit service. Transp. Res. Part A: Policy Pract. 38 (7), 511–530.

Alshalalfah, B., Shalaby, A., 2012. Feasibility of flex-route as a feeder transit service to rail stations in the suburbs: Case study in toronto. J. Urban Plan. Dev. 138 (1), 90–100.

Angrist, J.D., Caldwell, S., Hall, J.V., 2021. Uber versus taxi: A driver's eye view. Am. Econ. J.: Appl. Econ. 13 (3), 272–308.

Bagloee, S.A., Tavana, M., Asadi, M., Oliver, T., 2016. Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. J. Mod. Transp. 24, 284–303.

Beirigo, B.A., Negenborn, R.R., Alonso-Mora, J., Schulte, F., 2022. A business class for autonomous mobility-on-demand: Modeling service quality contracts in dynamic ridesharing systems. Transp. Res. C 136, 103520.

Bian, Z., Bai, Y., Liu, X., Wang, B., 2022. An online hybrid mechanism for dynamic first-mile ridesharing service. Transp. Res. C 138, 103585.

Bian, Z., Liu, X., 2019a. Mechanism design for first-mile ridesharing based on personalized requirements Part I: Theoretical analysis in generalized scenarios. Transp. Res. B 120, 147–171.

Bian, Z., Liu, X., 2019b. Mechanism design for first-mile ridesharing based on personalized requirements Part II: Solution algorithm for large-scale problems. Transp. Res. B 120, 172–192.

Bian, Z., Liu, X., Bai, Y., 2020. Mechanism design for on-demand first-mile ridesharing. Transp. Res. B 138, 77–117.

Chandra, S., Quadrifoglio, L., 2013. A model for estimating the optimal cycle length of demand responsive feeder transit services. Transp. Res. B 51, 1–16.

Charisis, A., Iliopoulou, C., Kepaptsoglou, K., 2018. DRT route design for the first/last mile problem: Model and application to Athens, Greece. Public Transp. 10 (3), 499–527.

Chen, Y., Wang, H., 2018. Pricing for a last-mile transportation system. Transp. Res. B 107, 57–69.

Chen, S., Wang, H., Meng, Q., 2020. Solving the first-mile ridesharing problem using autonomous vehicles. Comput.-Aided Civ. Infrastruct. Eng. 35 (1), 45–60.

Cordeau, J.F., Laporte, G., 2003. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. Transp. Res. B 37 (6), 579–594.

Dabia, S., Ropke, S., Van Woensel, T., De Kok, T., 2013. Branch and price for the time-dependent vehicle routing problem with time windows. Transp. Sci. 47 (3), 380–396.

Deng, F., Qin, H., Li, J., Cheng, C., 2022. The pickup and delivery problem with time windows and incompatibility constraints in cold chain transportation. Transp. Sci. 57 (2), 444–462.

EasyMile, 2022. EasyMile's driverless EZ10 passenger shuttles: innovative, safe and efficient mobility. https://easymile.com/vehicle-solutions/ez10-passenger-shuttle, 2022[EB/OL].

Farber, H.S., 2015. Why you can't find a taxi in the rain and other labor supply lessons from cab drivers. Q. J. Econ. 130 (4), 1975–2026.

Feillet, D., Dejax, P., Gendreau, M., Gueguen, C., 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. Networks 44 (3), 216–229.

Grahn, R., Qian, S., Hendrickson, C., 2022. Optimizing first-and last-mile public transit services leveraging transportation network companies (TNC). Transportation 1–28.

Grahn, R., Qian, S., Hendrickson, C., 2023. Environmental impacts of first-mile-last-mile systems with shared autonomous electric vehicles and ridehailing. Transp. Res. Part D: Transp. Environ. 118, 103677.

Gurumurthy, K.M., Kockelman, K.M., Zuniga-Garcia, N., 2020. First-mile-last-mile collector-distributor system using shared autonomous mobility. Transp. Res. Rec.: J. Transp. Res. Board 2674 (10), 638–647.

He, P., Jin, J., Schulte, F., Trépanier, M., 2023. Optimizing first-mile ridesharing services to intercity transit hubs. Transp. Res. C 150, 104082.

Ho, W., Ho, G.T., Ji, P., Lau, H.C., 2008. A hybrid genetic algorithm for the multi-depot vehicle routing problem. Eng. Appl. Artif. Intell. 21 (4), 548–557.

Huang, K., An, K., Rich, J., Ma, W., 2020. Vehicle relocation in one-way station-based electric carsharing systems: A comparative study of operator-based and user-based methods. Transp. Res. Part E: Logist. Transp. Rev. 142, 102081.

Huang, Y., Kockelman, K.M., Garikapati, V., 2022. Shared automated vehicle fleet operations for first-mile last-mile transit connections with dynamic pooling. Comput. Environ. Urban Syst. 92, 101730.

Hussain, R., Sherali, Z., 2018. Autonomous cars: Research results, issues, and future challenges. IEEE Commun. Surv. Tutor. 21 (2), 1275–1313.

Irnich, S., Desaulniers, G., 2005. Shortest path problems with resource constraints. In: Column Generation. Springer, New York, pp. 33–65.

Jiang, G., Lam, S.K., Ning, F., He, P., Xie, J., 2020. Peak-hour vehicle routing for first-mile transportation: Problem formulation and Algorithms. IEEE Trans. Intell. Transp. Syst. 21 (8), 3308–3321.

Jin, J.G., Meng, Q., Wang, H., 2021. Feeder vessel routing and transshipment coordination at a congested hub port. Transp. Res. B 151, 1–21.

Jochem, P., Frankenhauser, D., Ewald, L., Ensslen, A., Fromm, H., 2020. Does free-floating carsharing reduce private vehicle ownership? The case of SHARE NOW in European cities. Transp. Res. Part A: Policy Pract. 141, 373–395.

Kumar, P., Khani, A., 2021. An algorithm for integrating peer-to-peer ridesharing and schedule-based transit system for first mile/last mile access. Transp. Res. C 122, 102891.

Levin, M.W., Odell, M., Samarasena, S., Schwartz, A., 2019. A linear program for optimal integration of shared autonomous vehicles with public transit. Transp. Res. C 109, 267–288.

Liang, X., Correia, G.H.d.A., An, K., Arem, B.v., 2020. Automated taxis' dial-a-ride problem with ride-sharing considering congestion-based dynamic travel times. Transp. Res. C 112, 260–281.

Lim, A., Zhang, Z., Qin, H., 2017. Pickup and delivery service with manpower planning in Hong Kong public hospitals. Transp. Sci. 51 (2), 688–705.

Lin, D., Nelson, J.D., Cui, J., 2021. Exploring influencing factors on metro development in China from urban and economic perspectives. Tunn. Underg. Space Technol. 112 (4), 1–10.

Lu, C.C., Diabat, A., Li, Y.T., Yang, Y.M., 2022. Combined passenger and parcel transportation using a mixed fleet of electric and gasoline vehicles. Transp. Res. Part E: Logist. Transp. Rev. 157, 102546.

Ma, T.Y., Rasulkhani, S., Chow, J.Y.J., Klein, S., 2019. A dynamic ridesharing dispatch and idle vehicle repositioning strategy with integrated transit transfers. Transp. Res. Part E: Logist. Transp. Rev. 128, 417–442.

Nazari, F., Noruzoliaee, M., Mohammadian, A.K., 2018. Shared versus private mobility: Modeling public interest in autonomous vehicles accounting for latent attitudes. Transp. Res. C 97, 456–477.

Ning, F., Jiang, G., Lam, S.K., Ou, C., He, P., Sun, Y., 2021. Passenger-centric vehicle routing for first-mile transportation considering request uncertainty. Inform. Sci. 570, 241–261.

Park, K., Farb, A., Chen, S., 2021. First-/last-mile experience matters: The influence of the built environment on satisfaction and loyalty among public transit riders. Transp. Policy 112, 32–42.

Pasaoglu, G., Fiorello, D., Martino, A., Scarcella, G., Alemanno, A., Zubaryeva, A., Thiel, C., et al., 2012. Driving and Parking Patterns of European Car Drivers—A Mobility Survey. European Commission Joint Research Centre, Luxembourg.

Pillac, V., Gendreau, M., Guéret, C., Medaglia, A.L., 2013. A review of dynamic vehicle routing problems. European J. Oper. Res. 225 (1), 1–11.

Pisinger, D., Ropke, S., 2007. A general heuristic for vehicle routing problems. Comput. Oper. Res. 34 (8), 2403–2435.

Psaraftis, H.N., Wen, M., Kontovas, C.A., 2016. Dynamic vehicle routing problems: Three decades and counting. Networks 67 (1), 3–31.

Quadrifoglio, L., Li, X., 2009. A methodology to derive the critical demand density for designing and operating feeder transit services. Transp. Res. B 43 (10), 922–935.

Ritzinger, U., Puchinger, J., Hartl, R.F., 2016. A survey on dynamic and stochastic vehicle routing problems. Int. J. Prod. Res. 54 (1), 215–231.

RoboticResearch, 2022. First and Last-mile solution with autonomous electric vehicles. https://www.roboticresearch.com/first-last-mile/.org, 2022[EB/OL].

Schmöller, S., Weikl, S., Müller, J., Bogenberger, K., 2015. Empirical analysis of free-floating carsharing usage: The munich and berlin case. Transp. Res. C 56, 34–51.

Shaheen, S., Chan, N., 2016. Mobility and the sharing economy: Potential to facilitate the first- and last-mile public transit connections. Built Environ. 42 (4), 573–588.

Shan, A., Hoang, N.H., An, K., Vu, H.L., 2021. A framework for railway transit network design with first-mile shared autonomous vehicles. Transp. Res. C 130, 103223.

Shen, Y., Zhang, H., Zhao, J., 2018. Integrating shared autonomous vehicle in public transportation system: A supply-side simulation of the first-mile service in Singapore. Transp. Res. Part A: Policy Pract. 113, 125–136.

Sipetas, C., Roncoli, C., Mladenović, M., 2023. Mixed fleets of automated and human-driven vehicles in public transport systems: An evaluation of feeder line services. Transp. Res. Interdiscip. Perspect. 18, 100791.

Stiglic, M., Agatz, N., Savelsbergh, M., Gradisar, M., 2018. Enhancing urban mobility: Integrating ride-sharing and public transit. Comput. Oper. Res. 90, 12–21.

Sun, H., Wang, H., Wan, Z., 2019. Model and analysis of labor supply for ride-sharing platforms in the presence of sample self-selection and endogeneity. Transp. Res. B 125, 76–93.

Tang, Z.Y., Tian, L.J., Wang, D.Z., 2021. Multi-modal morning commute with endogenous shared autonomous vehicle penetration considering parking space constraint. Transp. Res. Part E: Logist. Transp. Rev. 151, 1–24.

Vansteenwegen, P., Melis, L., Aktas, D., David, B., Montenegro, G., Vieira, F.S., Sorensen, K., 2022. A survey on demand-responsive public bus systems. Transp. Res. C 137, 1–39.

Wang, H., 2019. Routing and scheduling for a last-mile transportation system. Transp. Sci. 53 (1), 131–147.

Xu, M., Meng, Q., Liu, Z., 2018. Electric vehicle fleet size and trip pricing for one-way carsharing services considering vehicle relocation and personnel assignment. Transp. Res. B 111, 60–82.

Yang, H., Zhang, Z., Fan, W., Xiao, F., 2021. Optimal design for demand responsive connector service considering elastic demand. IEEE Trans. Intell. Transp. Syst. 22 (4), 2476–2486.

Yu, Y., Machemehl, R.B., Xie, C., 2015. Demand-responsive transit circulator service network design. Transp. Res. Part E: Logist. Transp. Rev. 76, 160–175.