Transportation Research Part C 138 (2022) 103585

Contents lists available at ScienceDirect

Transportation Research Part C

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

An online hybrid mechanism for dynamic first-mile ridesharing service

Zheyong Bian [a,b], Yun Bai [b,*], Xiang Liu [b,c], Bijun Wang [d]

[a] Department of Construction Management, College of Technology, University of Houston, 4730 Calhoun Road, Houston, TX 77004, USA
[b] Center for Advanced Infrastructure and Transportation, Rutgers, The State University of New Jersey, 100 Brett Road, Piscataway, NJ 08854, USA
[c] Department of Civil and Environmental Engineering, Rutgers, The State University of New Jersey, 500 Bartholomew Rd, Piscataway, NJ 08854, USA
[d] School of Systems and Enterprises, Stevens Institute of Technology, Castle Point Terrace, Hoboken, NJ 07030, USA

ARTICLE INFO

Keywords:
Dynamic ridesharing
First mile
Mechanism design
Dynamic optimization
Dynamic pricing

ABSTRACT

This paper leverages the “Mechanism Design” theory to design the ridesharing-based transit feeder service with mixed scheduled and on-demand passenger requests. An online hybrid mechanism is proposed with four incentive objectives: promoting passengers to participate by satisfying their mobility preferences, inducing passengers to truthfully reveal their mobility preferences, incentivizing the service provider to be financially sustainable, and incentivizing more regular commuters to early schedule the service. We propose and prove four properties, “preference-based individual rationality”, “preference-based incentive compatibility”, “financial sustainability”, and “scheduling preferability” to achieve the four incentive objectives, respectively. This online hybrid mechanism is comprised of a dynamic re-optimization methodology for re-matching and re-routing and a hybrid real-time pricing mechanism discriminative against different passenger types. In order to obtain the large-scale solutions for the online hybrid mechanism, this paper improves the solution pooling approach (SPA), which was originally proposed in our previous work for a static offline mechanism, to adapt for the online hybrid mechanism. The improved SPA successfully sustains the “preference-based individual rationality”, “financial sustainability”, and “scheduling preferability” properties. The simulation results demonstrate the superiority of the proposed online hybrid mechanism over the static offline mechanism and the outperformance of the improved SPA over the original SPA.

1. Introduction

One critical issue that an industrialized nation faces is the need for a low-cost, reliable, and environmentally sustainable public transit system. Despite the huge investment in infrastructure and operations, American public transit system does not have significant growth in ridership. In fact, the transit ridership in the U.S. decreased by 2.7 percent from 2017 to 2018 (O’Toole, 2020). Among various issues, one well-recognized factor that hinders the use of public transit is the “first mile” accessibility gap, which significantly affects a passenger’s choice of public transit (Lesh, 2013; Perera et al., 2018). Enabled by recent advances in technology, ridesharing holds the promise of bridging the accessibility gap in public transit and reducing the on-road vehicles. As an innovative way to provide

* Corresponding author.
E-mail addresses: zbian2@central.uh.edu (Z. Bian), yunbai.cait@rutgers.edu (Y. Bai), xiang.liu@rutgers.edu (X. Liu), bwang27@stevens.edu (B. Wang).

https://doi.org/10.1016/j.trc.2022.103585
Received 22 January 2021; Received in revised form 15 September 2021; Accepted 24 January 2022

"transit feeder service," recent literature has discussed the possibility to increase national-wide public transit ridership by leveraging ridesharing as a first-mile service (Lesh, 2013; Stiglic et al., 2018).

Our research uses mechanism design theory to investigate how a multi-agent (vehicles and travelers) first-mile ridesharing (FMR) system can operate effectively in such a way that all participants collectively achieve the system goals, while satisfying their personal needs via customized incentives. This paper accounts for mixed scheduled and on-demand passenger requests because, in FMR service, a portion of passengers are commuters and can schedule the service in advance, while other spontaneous travelers may send requests up to the "last minute". Therefore, an online hybrid mechanism (ONHBM) is designed with four incentive objectives achieved:

1. Promoting passengers to participate by satisfying their mobility preferences. Many people are not so enthusiastic about sharing rides because the system is not designed in a manner that satisfies their mobility preferences. Thus, the designed mechanism should ensure that the mechanism can satisfy passengers' mobility preferences (e.g., the latest arrival time, the longest tolerable detour, the maximum tolerable number of shared riders, maximum price they are willing to pay, etc.).
2. Promoting to truthfully reveal users' mobility preferences. The mechanism allows passengers to directly report their personalized mobility preferences. However, rational passengers may intentionally misreport the preferences to maximize their utilities. For example, a passenger may report a very low price that he/she is willing to pay. This untruthful behavior may impair the system-wide optimization and other passengers' and the service provider's benefits. Therefore, the mechanism should be misreporting preventable.
3. Incentivizing the service provider to be financially sustainable. The designed mechanism incentivizes the service provider to continually provide the service without a financial deficit so that the service is self-sustainable without external investment.
4. Promoting passengers to early schedule the service as well as incentivizing regular commuters to use the FMR service. A large portion of the passengers scheduling the service are commuters, who may use the FMR service regularly and frequently (Kumar and Khani, 2021). They deserve a discount compared with single-use on-demand passengers. In addition, scheduling the service promotes to reduce the uncertainty of trip requests and provides priori information for the system-wide global optimization. Thus, the designed pricing scheme is particularly discriminatory against different types of passengers, providing scheduled passengers with more incentive. This objective of the mechanism is similar to the concept of group purchasing, which usually give customers some discount as incentive (Matsuo et al. 2005).

We propose four mechanism properties, namely "preference-based individual rationality", "preference-based incentive compatibility", "financial sustainability", and "scheduling preferability", to achieve the four incentive objectives, respectively, which will be detailed in Section 4.

The mechanism is modeled through a dynamic re-optimization approach to enable re-matching and re-routing and a real-time dual-layer incentive pricing scheme, which is implemented by a rolling horizon planning (RHP) approach. The system continuously receives and uses RHP to process passengers' requests. Emerging passengers' requests are accommodated either in existing routing plans, which need to be re-adjusted, or in new matching plans and routing sequences. The real-time, dual-layer pricing scheme consists of an intermediate pricing scheme and a final pricing scheme. The intermediate pricing scheme, determined by the passengers' request locations and times, is embedded as a baseline in the final pricing layer. The main objectives of the intermediate pricing layer are to differentiate prices of passenger requests with different urgencies (defined as time to train departure time) as well as to ensure financial sustainability of the service. The final pricing layer is determined by passengers' mobility preferences as well as the matching plan and the routing sequence. The final pricing layer prevents passengers' misreporting strategy.

The re-optimization problem of matching and routing is NP-hard and is difficult to be solved within a short time when the problem scale is sufficiently large, while the real-time FMR service requires a computationally efficient algorithm with near-optimal results. Traditional heuristic or approximate algorithms may not maintain crucial properties in solving large-scale problems (Mu'alem and Nisan, 2008; Nisan and Ronen, 2007; Bian and Liu 2019b). Our previous work (Bian and Liu, 2019b) developed an efficient algorithm, Solution Pooling Approach (SPA), to solve an offline mechanism design problem for scheduled FMR, which successfully holds "individual rationality" and "incentive compatibility". Nevertheless, it is not sufficiently fast for the real-time FMR service with on-demand passenger requests. In addition, the original SPA may not hold "financial sustainability" and "scheduling preferability" (the reason is given in Section 5). Thus, this paper improves the original SPA. The improved SPA can solve mechanism design problem within a short time and can hold "preference-based individual rationality", "financial sustainability", and "scheduling preferability".

The remainder of the paper is structed as follows. Section 2 reviews the state-of-the-art and summarizes the contributions of this research. Section 3 describes the FMR system and clarifies the mechanism design problem. Section 4 introduces the online hybrid mechanisms. The improved SPA algorithm is presented in Section 5. Experimental results are presented in Section 6. Section 7 summarizes conclusion remarks.

2. Literature review

Section 2.1 reviews the literature on FMR, mechanism types for ridesharing, user types in ridesharing, and solution algorithms for ridesharing mechanisms. Section 2.2 introduces our previous work on mechanism design for FMR service. Section 2.3 identifies the knowledge gaps and introduces the intended contributions of this paper.

2.1. Existing work

2.1.1. First-mile ridesharing

Research has been aware of the efficacy of ridesharing to address first-mile accessibility problem and proposed relevant research. Most of the research applied optimization and simulation approaches to design or improve FMR systems, but to our best knowledge, incentive problem, which is critical for FMR organization, is not well studied. Masoud et al. (2017a) studied the first- and last-mile problem in Los Angeles Metro Red Line and proposed a mechanism that selectively serves passengers with an incentive pricing scheme. Bian and Liu (2017) designed the optimal FMR service connecting to train schedules using the simulated annealing algorithm. Shen et al. (2018) conducted a simulation of the shared autonomous vehicle FMR service. Bian and Liu (2018) designed a detour-based discounting mechanism for the FMR service. Jiang et al. (2020) studied the FMR service problem with a limited vehicle fleet capacity, aiming at maximizing the number of served passengers. Chen et al. (2020) built a mixed integer programming model for autonomous vehicle FMR. Kumar and Khani (2021) studied a first- and last-mile ridesharing matching problem in a multimodal transportation network.

2.1.2. Mechanism design types for ridesharing

Recently, researchers have proposed various types of ridesharing mechanisms to achieve different incentive objectives. These mechanisms can be classified into two general categories, which are rule-based and auction-based mechanisms.

1. Rule-based mechanisms. Rule-based mechanisms determine the matching plan, routing sequence, and prices based on pre-determined rules. There are four basic rule-based mechanisms. The first is called the "flat rate" mechanism, which determines the price via a constant (flat) rate based on travel distance and/or travel time (Ungemah et al., 2006). The second is the supply–demand-based mechanism (e.g., surging pricing scheme), which uses dynamic pricing scheme to balance the supply and demand (Liu and Li, 2017; Wei et al., 2019; Ma et al., 2020; Ke et al., 2020). The third type is fair cost sharing mechanism that splits the fare among riders with a certain degree of fairness (Zhang et al., 2020; Hu et al., 2020; Chau et al., 2020, etc.). Lastly, the fourth mechanism simultaneously optimizes matching plan, routing sequence, and prices to achieve one of certain objectives (e.g., minimizing total vehicle mileage, maximizing profits, and minimizing transportation cost) (Santos and Xavier, 2015; Lei et al., 2019). In these "rule-based" mechanisms, passengers cannot report their mobility preferences and thus the service and pricing are difficult to be customized to account for users' needs.
2. Auction-based mechanisms. Auction-based mechanisms induce passengers to bid for the service. Passengers with higher values have higher priority to be served (Zhao et al., 2015; Nguyen, 2013; Cheng et al., 2014; Kleiner et al., 2011; Kamar and Horvitz, 2009; Asghari et al., 2016; Asghari and Shahabi, 2017; Shen et al., 2016; Ma et al., 2018; Zheng et al., 2019; Luo, 2019; Hsieh et al., 2019; Shi et al., 2020). The Vickrey-Clarke-Groves (VCG) mechanism is an example of this type (Vickrey, 1961; Clarke, 1971; Groves, 1973). However, the VCG mechanism is not budget-balanced and may not be financially sustainable without external investment (Parkes et al., 2001). Recently, researchers further modified the VCG mechanisms. For example, researchers have applied deficit control mechanisms to address the revenue shortage problem (Zhao et al., 2014; Lloret-Batlle et al., 2017; Zhang et al., 2018; etc.). Although passengers can customize the service through reporting their personalized valuations, their mobility preferences associated with the FMR (e.g., detour tolerance) have not been considered in the existing research.

2.1.3. User types in ridesharing

1. Scheduled passengers. Many passengers may schedule the ridesharing service because they know their schedules in advance, for example, the commuters and those who have scheduled business, events, or long-distance travels. In a system where all passenger requests are scheduled in advance, the matching plan can be pre-determined with less computational constraint. Mechanisms have been developed to handle such scheduled requests (Zhao et al., 2015; Nguyen, 2013; Cheng et al., 2014; Zheng et al., 2019; Hsieh et al., 2019). These mechanisms are offline and static and do not need to handle dynamically occurring passenger requests.
2. On-demand passengers. Other passengers may send on-demand requests because they do not report their travel plans until the "last minute" (Daganzo and Ouyang, 2019). In the literature, researchers primarily focused on the mechanism for dynamic ridesharing (Kleiner et al., 2011; Luo, 2019; Zhang et al., 2017; 2018; Shen et al., 2016; Shi et al., 2020), in which passengers send on-demand requests for immediate car usage. Compared with mechanisms for scheduled passengers, on-demand mechanisms are more difficult to design because: 1) The mechanism results must be dynamically obtained; 2) The obtainment of mechanism results should be quickly responsive, since on-demand passengers require prompt responses to their requests for service.

The prior work has focused on either scheduled or spontaneous passengers, but little work has accounted for both types simultaneously. For the FMR, it is possible that a scheduled rider may share a ride with a spontaneous passenger if both of their personalized mobility preferences are satisfied. When designing mechanisms for mixed types of passengers, the incentive for early scheduling needs to be incorporated. Those passengers (e.g., commuters) using the FMR service regularly and frequently can receive a discount, because the early scheduled trips are easier to accommodate and can be coordinated to reduce the system cost. In order to achieve system-wide optimization using passengers' prior information, the mechanism should be able to provide incentives for passengers who schedule the service earlier (aka. the "scheduling preferability" property in our proposed mechanism design).

2.1.4. Solution algorithms for ridesharing mechanisms

Certain mechanism design problems (e.g., VCG mechanism) for ridesharing are NP-hard and can be computationally challenging (Lloret-Batlle et al., 2017). Although several heuristic algorithms in the literature (Rückert et al., 2019; Hua and Qi, 2019; Haferkamp and Ehmke 2020) can optimally match vehicles and passengers and optimize the routing sequence, they barely maintain some desirable mechanism design properties (e.g., incentive compatibility) (Nisan and Ronen 2007). Much research either focused on small-scale mechanism design problems or simplified their mechanisms to reduce the computational complexity.

1. Small-scale mechanisms. Several auction-based mechanisms proposed by the researchers (Cheng et al., 2014; Kamar and Horvitz, 2009; Masoud et al., 2017b; Yan et al., 2021, etc.) are inherently computationally challenging. Some studies developed prevalent algorithms (e.g., branch and cut, local-search-based heuristics) (Masoud et al., 2017b; Nguyen, 2013). However, we are not aware of these studies investigating the performance of their algorithms in solving large-scale ridesharing problems.
2. Simplified mechanisms. Some researchers simplified their mechanisms to handle the computational challenge. For example, the parallel mechanism in Kleiner et al. (2011) is restricted to the simplified dual-ridesharing, i.e., one passenger and one driver sharing the ride. Kamar and Horvitz (2009) developed a local VCG-based pricing scheme which computes the VCG payment of agents only among the agents that share the same carpool. Masoud and Lloret-Batlle (2016) simplified the optimization of many-to-many ridesharing to one-to-many ridesharing cases. Zhang et al. (2020) developed two greedy algorithms to maximize liquidity and utility, respectively, in obtaining their mechanism. Yan et al. (2021) proposed a pricing scheme to ensure the ride-sharing solution to be stable, system-wide optimal, and financially sustainable, but did not prove the property of "incentive compatibility". Some other researchers (Asghari and Shahabi, 2017; Zhang et al., 2018) use greedy or approximate algorithms for matching and routing to reduce the computational complexity. However, the simplified mechanism may not fully capture the complex characteristics of real-world ridesharing services and thus may result in sub-optimal solutions.

2.2. Our previous work

This paper is built upon our previous research. One of our recent publications (Bian and Liu, 2019a) developed a new mechanism for the scheduled FMR service. After that, in Bian et al. (2020), we extended the mechanism for the application in the on-demand scenario with the aid of rolling horizon planning (RHP) approach. However, these two articles studied the mechanisms for the scheduled and on-demand trip request separately, without considering mixed scheduled and on-demand passengers. Consequently, the incentive objectives of promoting passengers to early schedule the service as well as incentivizing regular commuters to use the FMR service have not been considered. In addition, although the mechanism proposed in Bian et al. (2020) is able to handle on-demand passenger requests, the mechanism is offline and static, under which, once the matching plan and routing sequence is determined, it will never be changed to accommodate newly occurred passengers' requests. This is because the mechanism lacks a dynamic re-optimization approach to re-routing vehicles to accommodate newly occurred passenger requests. Therefore, the mechanism sustains a relatively high vehicle empty seat rate and transportation cost (see Section 6 for demonstration).

We also developed a new heuristic algorithm, solution pooling algorithm (SPA), to solve the mechanism design problem in our previous work (Bian and Liu, 2019b; Bian et al., 2020). The SPA was proven to satisfy individual rationality and incentive compatibility (Bian and Liu, 2019b). Nevertheless, two other incentive objectives (financial sustainability and scheduling preferability) were not addressed yet. Also, the SPA did not address dynamic re-optimization of matching and routing in the online hybrid mechanism considering mixed scheduled and on-demand passengers, which is more challenging than the offline static mechanism.

2.3. Knowledge gaps and intended research contributions

Based on the literature review, we identify the following knowledge gaps that this proposed research aims to address:

- Knowledge Gap 1: Mobility-Preference-Based Mechanism. Existing mechanisms (rule-based or auction-based mechanisms) did not fully account for passengers' personalized mobility preferences, such as passengers' maximum willing-to-pay prices, arrival deadlines, tolerance of inconvenience attributes (e.g., detour, tolerable number of shared passengers) which can significantly influence travelers' choice of ridesharing (BBC news, 2016; Li et al., 2020; Daganzo et al., 2020; Fielbaum and Alonso-Mora, 2020). The existing mechanisms may fail to incentivize passengers to share the ride because the provided incentives may not necessarily satisfy their mobility preferences.
- Knowledge Gap 2: User Types. Existing mechanisms are designed for either scheduled or on-demand passengers. None of the previous studies specifically accounted for **mixed types of passengers**. Also, the incentive objective of incentivizing more regular passengers (e.g., commuters) to use the service and promoting passengers to early schedule the service has not been fully studied in the literature.
- Knowledge Gap 3: Solutions Algorithms. Existing research developed algorithms to solve small-scale or simplified mechanism design to circumvent the computational complexity. To our best knowledge, no previous research has addressed such a complex dynamic ridesharing mechanism design problems with computationally efficient solution algorithms for large scale instances or validated the capability of heuristic algorithms to sustain the above-mentioned important mechanism design properties for ridesharing.

To address these knowledge gaps, this paper brings the following contributions:

- This paper designs an online hybrid mechanism for mixed types of passengers accounting for their personalized mobility preferences. The online hybrid mechanism simultaneously achieves four incentive objectives: 1) promoting passengers to participate by satisfying their mobility preferences, 2) promoting passengers to truthfully report their mobility preferences, 3) incentivizing the service provider to be financially sustainable, 4) promoting passengers to early schedule the service and incentivizing more regular commuters to use the FMR service.
- This paper develops a more advanced algorithm, SPACL, which improves the original SPA developed in Bian and Liu, 2019b. A closed loop is added to the original SPA to ensure the properties of "preference-based individual rationality", "scheduling preferability" and "financial sustainability" (detailed in Section 5). The simulation results show that the improved SPA, i.e., SPACL, outperforms the original SPA and the commercial solver CPLEX in term of both the solution quality and computational speed for large-scale problems.

3. Description of the system

3.1. Information revelation

For scheduled service, passengers can report the arrival deadline (θ_i^{AD}), the maximum tolerable detour time (θ_i^{EIVT}), and the maximum tolerable number of shared riders (θ_i^{NR}). We use $\theta_i = \{\theta_i^{AD}, \theta_i^{EIVT}, \theta_i^{NR}\}$ to denote scheduled passengers' mobility preference. For on-demand service, in addition to the scheduled passengers' mobility preferences, passengers can report another parameter, the maximum willing-to-pay (WTP) price, to bid for the service, because unlike the scheduled service, on-demand service requires vehicles to be in an "on-call state", ready for providing the service within a very short time. Sometimes, either the number of vehicles or time is insufficient to serve all passengers. The concept of the auction-based mechanisms has been proposed by many other studies (Kleiner et al., 2011; Zhang et al., 2017; Zhang et al., 2018; Asghari et al., 2016; Shi et al., 2020; etc.). We use $\alpha_i = \{\alpha_i^{AD}, \alpha_i^{EIVT}, \alpha_i^{NR}, \alpha_i^{P}\}$ to denote the on-demand passengers' mobility preferences, where α_i^{AD} is the arrival deadline, α_i^{EIVT} is the maximum tolerable detour time, α_i^{NR} is the tolerable number of shared passengers, and α_i^{P} is the maximum WTP price.

3.2. Events' sequence of the hybrid mechanism

There is a key difference between scheduled service and on-demand service. In scheduled service, passengers do not report the maximum WTP price but will be notified of a maximum possible price, upon which the passenger can accept the offer or not, while in on-demand service, passengers can bid for the service by reporting the maximum WTP price and the final price will never exceed the reported maximum WTP price. Based on this difference, the event sequences of scheduled service and on-demand service are also different, as clarified below.

For scheduled service, the event sequence is as follows.

(1) A passenger types in the train departure location (i.e., the transit hub) and time.
(2) The application interface will show a price that will never be exceeded by the final price. The passenger can accept the offer or not. This highest price is the intermediate price detailed in Section 4.
(3) If the passenger accepts the offer, he can input the mobility preferences. If passengers do not key in these mobility preferences, the system deems that they do not have the requirements on these attributes.
(4) The request is received. The service ensures that each passenger whose request is sent will be served.
(5) When the service is approaching (e.g., 60 min before the train departure time), the system consolidates and processes all passengers' requests.
(6) Passengers are notified of the pickup time and the final price. The drivers are notified of the pickup sequence and the latest arrival time.

Note that the mechanism for the scheduled service has a drawback. In the second event of the scheduled service, the system will determine a maximum possible price that will never be exceeded by the final price for a specific passenger, and given this price, the passenger can accept or reject the offer. This possibly leads to loss of demand for those passengers whose actual WTP prices are higher than the final prices but lower than the given maximum prices. The possible reduced amounts in price and discount are quantified based on simulation examples in Section 6 and Appendix D. To avoid such demand loss, we suggest estimating the reduced/discounted price based on historical prices in nearby locations or based on simulation if historical data is not available. The system could place this estimated price along with the maximum possible price in the application interface to help the passenger decide whether to accept the offer or not. With the estimated price shown to the passenger, the proposed mechanism could avoid loss of demand to the maximum extent.

For on-demand service, the event sequence is as follows.

(1) A passenger types in the request information, including train departure location (i.e., the transit hub) and time as well as his mobility preferences. When the mobility preferences are not reported by the passenger, the system can set default reasonable values.
(2) The request is sent.

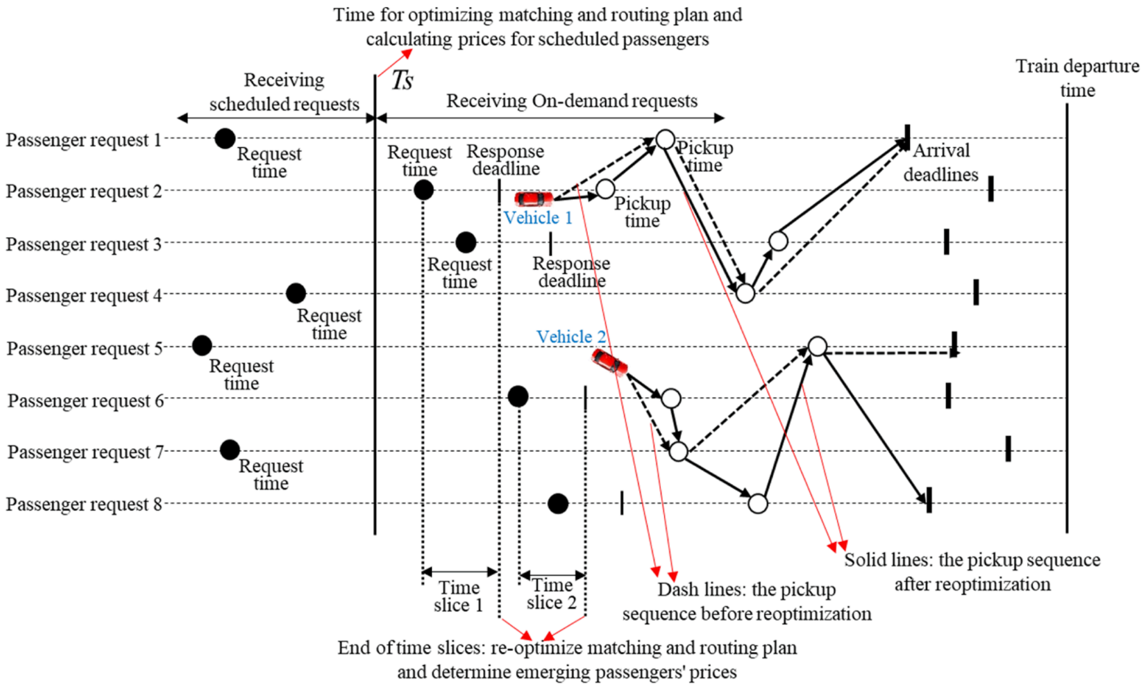

Fig. 1. Rolling horizon planning (RHP) approach to implementation of the mechanism.

(3) When a passenger's response deadline (e.g., 1 min after the request is sent) is reached, the system consolidates all passengers' requests sent within the time interval between the first passenger's requesting time and his response deadline, conducts routing optimization, and calculates the prices. Note that if the on-demand request's requirements are too strict to be satisfied (e.g., the maximum will-to-pay price is 0.1 dollar), the request will be rejected.

(4) Passengers are notified of the estimated pickup time and the final price. The drivers are notified of the updated pickup sequence.

3.3. Mechanism implementation by rolling horizon planning (RHP) approach

We use the RHP approach (Fig. 1), which is widely used to handle dynamic ridesharing problem (Agatz et al., 2010; Kleiner et al., 2011; Agatz et al., 2011), to implement the mechanism. Passenger requests 1, 4, 5, and 7 in Fig. 1 schedule the service in advance. The matching plan and routing sequence are optimized at time *Ts* and passengers will receive a response shortly. Two vehicles are dispatched to serve the four passengers. The optimal matching plan and routing sequence is "V1 → P1 → P4 → H" and "V2 → P7 → P5 → H" (the dash line; V: vehicle, P: passenger, H: the transit hub). After scheduled passengers' requests are processed, the system starts to receive passengers' on-demand requests. Each on-demand request has a response deadline because passengers are impatient to wait too long for the response of the system. If the system response time is too long, passengers may not have enough time to arrive at the transit hub. The rolling horizon approach cuts a succession of time slices for processing emerging passengers' requests. Each time slice starts when a passenger request occurs after the last time slice and ends at the response deadline of this passenger request regardless of other subsequent requests that may occur. For example, in Fig. 1, after the first time slice, Passenger 6's request is the first request, and thus the second time slice starts at Passenger 6's request time and ends at its response deadline. The system will simultaneously process all passengers' requests when each time slice ends. The matching plan and routing sequence are re-optimized: "V1 → P2 → P1 → P4 → P3 → H" and "V2 → P6 → P7 → P8 → P5 → H" (the solid line). The system also calculates the exact prices of the passenger requests that occur within the time slice.

There are three vehicle states. The first vehicle state is empty and without assigned passenger requests. This type of vehicles is able to provide the FMR service immediately. The second type of vehicles have already been dispatched and instructed to pick up passengers in a specific sequence. These vehicles with assigned passengers may still have empty seats for additional emerging passengers. The third type of vehicles are not available immediately since they may have other unfinished dropping off tasks but will finish within a short time. The times and the locations when and where these vehicles will be available can be estimated and are treated as known parameters. Thus, the inputs of the vehicle information include the available location, available time, number of passengers in the vehicle, passengers who are already assigned to this vehicle, and remaining seat capacity.

3.4. Assumptions

We clarify the assumptions of this paper as follows.

- We assume that the travel time between two locations is deterministic.
- Each passenger is rational with the goal of maximizing his utility when choosing their report strategies.
- It is assumed that passengers have no incentive to misreport the train departure location (i.e., the transit hub) and time.
- There are sufficient vehicles that can serve all scheduled passengers, and every scheduled passenger will be responded and all those who accept the offer will be served.
- This paper considers a single transit hub for the FMR matching, routing, and pricing. If multiple transit hubs exist in the nearby area, the mechanism is designed separately for different transit hubs.

4. The online hybrid mechanism

This section introduces the online hybrid mechanism. For the notation except specified in the text, please refer to Appendix A.

4.1. Intermediate pricing

Let us firstly introduce the intermediate pricing layer, which will be incorporated into the online hybrid mechanism. Note that the intermediate price is not the final price but an intermediate variable to determine the final price. This intermediate price plays the role of dividing line of the scheduled passengers' prices and the on-demand passengers' prices. A scheduled passenger's price will always be less than the intermediate price while an on-demand passenger's price will be always higher than the intermediate price. Therefore, the mechanism can promote passengers to schedule the service early and provide an incentive for regular commuters to use the service. Another function of this intermediate price is to avoid unreasonably low prices so that the mechanism can be financially sustainable. We suggest that intermediate pricing layer apply the traditional taxi pricing scheme (Taxi calculator, 2018), with an adjustment factor based on passengers' urgency as follows (Formula (1)).

$$ip_g(rt_g) = \begin{cases} cf + dr \times d_g, \text{if } rt_g < Ts \text{ (scheduled)} \\ cf + UG(\Delta t_g) \times dr \times d_g, \text{if } rt_g \geqslant Ts \text{ (on demand)} \end{cases} \tag{1}$$

where Ts is the deadline to receive scheduled passengers' requests, rt_g is the request time, cf is a constant initial fee, d_g is Passenger(s) g's travel distance of direct shipment to the transit hub without detouring, dr is the distance rate, and $UG(\Delta t_g) > 1$ is the urgency coefficient, a monotone decreasing function of Δt_g, which is the time difference between the train departure time DT_g and the request time rt_g ($\Delta t_g = DT_g - rt_g$). In order to differentiate scheduled and on-demand passengers' prices, we use $ip_g(rt_g^-)$ to represent scheduled passenger(s) g's intermediate price and use $ip_g(rt_g^+)$ to denote on-demand passenger(s) g's intermediate price, where rt_g^- is scheduled passenger(s) g's request time earlier than time Ts and rt_g^+ is on-demand passenger(s) g's request time later than time Ts. The intermediate price is a monotone increasing function of a passenger's request time.

4.2. Matching plan and routing sequence

This section introduces how to determine the matching plan and routing sequence considering mixed scheduled and on-demand passengers. First, we build a static optimization model of preliminary matching and routing for scheduled service. Then, as on-demand passengers' requests occur, a re-optimization model is developed for dynamic re-matching and re-routing.

4.2.1. Preliminary optimization for scheduled service

The preliminary optimization model is formulated by Formulas (2)–(14). The notation in the optimization model is presented in Appendix A (a. notation in the preliminary optimization).

Objective function:

$$max \sum_{i \in PS} VA_i - fs(X) \tag{2}$$

where VA_i is passenger i's value, and $fs(X)$ is the transportation cost. In Formula (2), $\sum_{i \in PS} VA_i$ is a constant. The reason is as follows. Under the mechanism for the scheduled service, before the completed request is sent, passengers must accept or reject the offer given the maximum possible price and an estimated final price in the application interface after they key in the origin and destination as well as their mobility preference. Thereafter, the matching and routing plan is optimized only for the scheduled passengers who accept the offers. Passengers who do not participate the service will never be involved in the optimization for matching and routing. The scheduled passengers who accepted the offers must be served (Formula (6)), and thus all passengers' valuations are independent of all decision variables and their summation ($\sum_{i \in PS} VA_i$) is a constant. This indicates that the objective function of Formula (2) is equivalent to Formula (3).

$$minfs(X) = \sum_{k \in VS} \sum_{i \in PS} \sum_{j \in PS} xs_{ijk} c_{ij} + \sum_{k \in VS} \sum_{i \in PS} ys_{ki} dc_{ki} + \sum_{k \in VS} \sum_{i \in PS} zs_{ki} ch_i \tag{3}$$

Subject to

$$ys_{kj} + \sum_{i \in PS} xs_{ijk} = ws_{kj} \text{ for all } k \in VS, j \in PS \tag{4}$$

$$zs_{ki} + \sum_{j \in PS} xs_{ijk} = ws_{ki} \text{ for all } k \in VS, i \in PS \tag{5}$$

$$\sum_{k \in VS} ws_{ki} = 1 \text{ for all } i \in PS \tag{6}$$

$$\sum_{i \in PS} ys_{ki} \leqslant 1 \text{ for all } k \in VS \tag{7}$$

$$\sum_{i \in PS} ws_{ki} np_i \leqslant Q_k \text{ for all } k \in VS \tag{8}$$

$$\sum_{i \in PS} \sum_{j \in PS} xs_{ijk} t_{ij} + \sum_{i \in PS} ys_{ki} t_{ki} + \sum_{i \in PS} zs_{ki} t_{i0} \leqslant w_{kg} \left(\theta_g^{AD} - Ts \right) + \left(1 - w_{kg} \right) M \text{ for all } g \in PS, k \in VS \tag{9}$$

$$IVT_i = \sum_{k \in VS} \sum_{j \in PS} xs_{ijk} \left(IVT_j + t_{ij} \right) + \sum_{k \in VS} zs_{ki} t_{i0} \text{ for all } i \in PS \tag{10}$$

$$IVT_i - t_{i0} \leqslant \theta_i^{EIVT} \text{ for all } i \in PS \tag{11}$$

$$\sum_{j \in PS} \sum_{k \in VS} ws_{jk} np_j \leqslant ws_{ki} \left(\theta_i^{NR} + np_i \right) + (1 - ws_{ki}) M \text{ for all } i \in PS, k \in VS \tag{12}$$

$$xs_{ijk}, ys_{ki}, zs_{ki}, ws_{ki} = \{0, 1\} \text{ for all } i, j \in PS, k \in VS \tag{13}$$

$$IVT_i \geqslant 0 \text{ for all } i \in PS \tag{14}$$

Formula (3) specifies the objective function that minimizes the transportation cost. Formula (4) and Formula (5) form each vehicle's routing sequence. Formula (6) represents that each passenger must be served. Formula (7) ensures that each vehicle cannot be dispatched more than once. Formula (8) is the vehicle capacity constraint, stipulating that the vehicle capacity should not be exceeded. Formula (9) avoids late arrival. Formula (10) formulates each passenger's total time spent at the vehicle after pickup. This formula can also prevent illegal sub-tours. Formula (11) ensures that each passenger's tolerable detour time is never exceeded. Formula (12) is the constraint that each passenger's tolerable number of shared passengers is not exceeded. Formula (13) means that the decision variables xs_{ijk}, ys_{ki}, zs_{ki}, ws_{ki} are all binary. Formula (14) ensures that each passenger's travel time in the vehicle is non-negative.

This model is a nonlinear programming, but can be reformulated as a linear mixed integer programming. In this model, only the constraint of Formula (10) is nonlinear. We use the following reformulation method.

We introduce a new variable us_{ijk} so that $us_{ijk} = xs_{ijk} \times IVT_j$ and thus Formula (10) can be rewritten as

$$IVT_i = \sum_{k \in VS} \sum_{j \in PS} \left(us_{ijk} + xs_{ijk} t_{ij} \right) + \sum_{k \in VS} zs_{ki} t_{i0} \text{ for all } i \in PS \tag{15}$$

In order to ensure $u_{ijk} = xs_{ijk} \times IVT_j$, we introduce two linear constraints, Formula (16) and Formula (17).

$$0 \leqslant us_{ijk} \leqslant IVT_j \text{ for all } i, j \in PS, k \in VS \tag{16}$$

$$IVT_j - \left(1 - xs_{ijk} \right) M \leqslant us_{ijk} \leqslant xs_{ijk} M, \text{ for all } i, j \in PS, k \in VS \tag{17}$$

where "M" is a positive number that is sufficiently large.

This reformulated model is denoted as

Model MS_0

Objective function: Formula (3)

Constraints: Formulas (4)–(9), (11)–(17).

Note that if there are not sufficient vehicles to serve the demand, it will lead the preliminary optimization model to be infeasible. We propose two strategies to avoid the possible infeasibility.

(1) When the passenger demand exceeds the vehicle supply for the scheduled service, the system may use a surging pricing scheme to increase the maximum possible price (i.e., the intermediate price) to reduce the demand. The surging pricing scheme has been studied by many researchers (Liu and Li, 2017; Wei et al., 2019; Ma et al., 2020; Ke et al., 2020) and is beyond the scope of this paper.
(2) In the scheduled service, the system will receive passengers' requests early enough before the train departure time at the transit hub, so the service provider has sufficient time to allocate more vehicle resource in advance. The service provider could reserve

vehicles for regular commuters for the first priority and single-time scheduled passengers for the second priority. If the vehicle fleet size is still not enough to serve the scheduled passengers' demand, the system could stop receiving scheduled passengers' requests when vehicles' total supply capacity is reached, and the passengers who send requests after the supply capacity is reached will be notified that no reserved vehicles are available in the current time and area, and they will be advised to schedule another time slot or send on-demand requests.

4.2.2. Re-optimization for on-demand service

Based on the RHP approach, the mechanism re-optimizes the matching plan and routing sequence at the end of each time slice. The re-optimization methodology makes no difference among time slices. We build a re-optimization model below, which is denoted as "Mts_0", for each time slice. The notation is presented in Appendix A (b. notation in the re-optimization model).

The objective of the model is to maximize passengers' cumulative values beyond the service provider's minimum acceptable prices minus the total transportation cost (Formula (18))

$$max\sum_{i\in P_{ts}}SV_i - TC(X) \tag{18}$$

where SV_i is a passenger's surplus value beyond the service provider's minimum acceptable price, which is defined as the intermediate price here. That is

$$SV_i = VA_i - ip_i\left(rt_i^+\right)\sum_{k\in V_{ts}}w_{ki} \tag{19}$$

where VA_i is passenger i's value:

$$VA_i = \alpha_i^p\sum_{k\in V_{ts}}w_{ki} \tag{20}$$

If the passenger is served ($\sum_{k\in V_{ts}}w_{ki}$=1), his value equals his maximum WTP price, α_i^p, and his surplus value is $VA_i - ip_i$. Otherwise, $\sum_{k\in V_{ts}}w_{ki}$= 0, his value equals "0" and his surplus value is "0" as well.

$TC(X)$ represents the transportation cost, which is formulated by Formula (21)

$$TC(X) = \sum_{k\in V_{ts}}\sum_{i\in P_{ts}}\sum_{j\in P_{ts}}x_{ijk}c_{ij} + \sum_{k\in V_{ts}}\sum_{i\in P_{ts}}y_{ki}dc_{ki} + \sum_{k\in V_{ts}}\sum_{i\in P_{ts}}z_{ki}ch_i + \sum_{k\in V_{ts}}cv_k\left(1 - \sum_{i\in P_{ts}}z_{ki}\right) \tag{21}$$

Subject to

$$y_{kj} + \sum_{i\in P_{ts}}x_{ijk} = w_{kj} \text{ for all } k\in V_{ts}, j\in P_{ts} \tag{22}$$

$$z_{ki} + \sum_{j\in P_{ts}}x_{ijk} = w_{ki} \text{ for all } k\in V_{ts}, i\in P_{ts} \tag{23}$$

$$\sum_{k\in V_{ts}}w_{ki}\leqslant 1 \text{ for all } i\in P_{ts} \tag{24}$$

$$\sum_{i\in P_{ts}}y_{ki}\leqslant 1 \text{ for all } k\in V_{ts} \tag{25}$$

$$\sum_{i\in P_{ts}}w_{ki}np_i\leqslant Q_k \text{ for all } k\in V_{ts} \tag{26}$$

$$IVT_i = \sum_{k\in V_{ts}}\sum_{j\in P_{ts}}x_{ijk}\left(IVT_j + t_{ij}\right) + \sum_{k\in V_{ts}}z_{ki}t_{i0} \text{ for all } i\in P_{ts} \tag{27}$$

$$IVT_i - t_{i0}\leqslant\alpha_i^{EIVT} \text{ for all } i\in P_{ts} \tag{28}$$

$$NPIV_k + \sum_{j\in P_{ts}}w_{kj}np_j\leqslant w_{ki}\left(\alpha_i^{NR} + np_i\right) + (1 - w_{ki})M \text{ for all } i\in P_{ts}, k\in V_{ts} \tag{29}$$

$$\sum_{i\in P_{ts}}w_{ki}np_i\leqslant RNR_k + \sum_{i\in P_{ts}}\delta_{ki}np_i \text{ for all } k\in V_{ts} \tag{30}$$

$$\sum_{i\in P_{ts}}\sum_{j\in P_{ts}}x_{ijk}t_{ij} + \sum_{i\in P_{ts}}y_{ki}t_{ki} + \sum_{i\in P_{ts}}z_{ki}t_{i0}\leqslant w_{kg}\left(\alpha_g^{AD} - AT_k\right) + \left(1 - w_{kg}\right)M \text{ for all } g\in P_{ts}, k\in V_{ts} \tag{31}$$

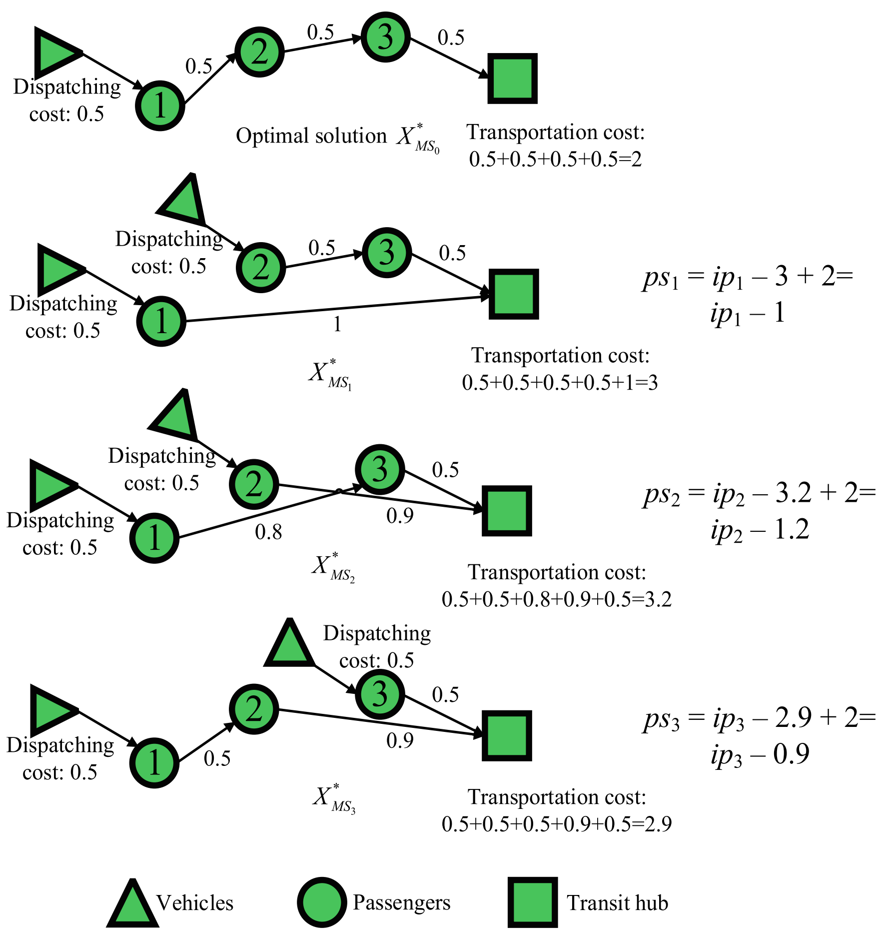

Fig. 2. An example to calculate scheduled passengers' prices.

$$\sum_{i\in P_{ts}}\sum_{j\in P_{ts}} x_{ijk}t_{ij} + \sum_{i\in P_{ts}} y_{ki}t_{ki} + \sum_{i\in P_{ts}} z_{ki}t_{i0} \leqslant min(RIVT_k, DLS_k - AT_k) \text{ for all } k \in V_{ts} \tag{32}$$

$$w_{ki} \geqslant \delta_{ki} \text{ for all } i \in P_{ts}, k \in V_{ts} \tag{33}$$

$$x_{ijk}, y_{ki}, z_{ki}, w_{ki} = \{0,1\} \text{ for all } i,j \in P_{ts}, k \in V_{ts} \tag{34}$$

$$IVT_i \geqslant 0 \text{ for all } i \in P_{ts} \tag{35}$$

Formulas (22), (23), (25), (27), (28), (34), and (35) in model Mts_0 achieve identical constraints with Formulas (4), (5), (7), (10), (11), (13), and (14) in model MS_0, respectively. Formula (24) indicates that each passenger request will either be served by at most one vehicle or be rejected. For on-demand service, unlike scheduled service, not necessarily all passengers can be served, because number of available vehicles or the remaining time to the arrival deadline may not be sufficient. Thus, we change the constraint from Formula (6) to Formula (24). Formula (26) ensures that the remaining seat capacity of each vehicle (Q_k) will not be exceeded. Formula (29) is to ensure that those passengers', who have not been picked up yet but will be served, maximum tolerable number of shared riders should not be exceeded. Formula (30) is to ensure that passengers', who are already in the vehicle, maximum tolerable number of shared riders should not be exceeded. Formula (31) avoids late arrival for those passengers who have not been picked up but will be served. Formula (32) avoids late arrival for those passengers who are already picked up by the vehicles and ensures that the detour time is within these in-vehicle passengers' tolerance. Formula (33) indicates that once a passenger request is assigned to a vehicle, this vehicle must pick up the passenger(s) and this passenger(s) will not be transferred to other vehicles.

Similar to model MS_0, model Mts_0 has a nonlinear constraint (Formula (27)), which can be linearized (Formulas (36)–(38)) in the same way.

$$IVT_i = \sum_{k\in V_{ts}}\sum_{j\in P_{ts}} \left(u_{ijk} + x_{ijk}t_{ij}\right) + \sum_{k\in V_{ts}} z_{ki}t_{i0} \text{ for all } i \in P_{ts} \tag{36}$$

$$0 \leqslant u_{ijk} \leqslant IVT_j \text{ for all } i,j \in P_{ts}, k \in V_{ts} \tag{37}$$

$$IVT_j - (1 - x_{ijk})M \leqslant u_{ijk} \leqslant x_{ijk}M, \text{ for all } i,j \in P_{ts}, k \in V_{ts} \tag{38}$$

Thus, this model is denoted as
Model Mts_0
Objective function: Formula (18)
Constraints: Formulas (22)–(26) and (28)–(38)

4.3. Hybrid pricing schemes

4.3.1. The pricing scheme for scheduled passengers

The pricing scheme is determined by a set of optimization models, MS_g, for all passenger requests $g \in PS$.
Model MS_g
Objective function: Formula (3)
Constraints: Formulas (4)–(9), (11)–(17), and (39)

$$\sum_{i \in P_{ts}} ws_{ki} np_i \leqslant ws_{kg} np_g + (1 - ws_{kg})M, \text{ for all } k \in VS \tag{39}$$

Compared with model MS_0, model MS_g has an additional constraint (Formula (39)) that passenger(s) g does not share the trip with other riders. Let $X^*_{MS_g}$ and $X^*_{MS_0}$ represent the optimal solutions of models MS_g and MS_0, respectively. Each passenger request's price can be formulated by Formula (40).

$$ps_g = ip_g(rt^-_g) - fs(X^*_{MS_g}) + fs(X^*_{MS_0}) \tag{40}$$

Fig. 2 gives a simple example to demonstrate how to calculate scheduled passengers' prices. Note that in Fig. 2, only $X^*_{MS_0}$ is the optimal matching plan and routing sequence that will be adopted. $X^*_{MS_1}$, $X^*_{MS_2}$, and $X^*_{MS_3}$ are used to calculate the three passengers' prices, respectively, which will never be used for matching and routing.

4.3.2. The pricing scheme for on-demand passengers

When the end of each time slice is approaching, the mechanism needs to determine the prices of passenger requests emerging within this time slice. We define a set "PO_{ts}" to denote the passenger requests sent within time slice ts. It is obvious that $PO_{ts} \in P_{ts}$. Before we give the pricing scheme for on-demand passengers, we reformulate the objective function (Formula (18)) of model Mts_0 to an equivalent form (Formula (41)) so as to formulate the pricing scheme. Thus, Formula (41) is the objective function of model Mts_0 as well.

$$\max \sum_{i \in P_{ts}} SV_i - TC(X) = \sum_{i \in P_{ts}} \left(VA_i - ip_i(rt^+_i) \sum_{k \in V_{ts}} w_{ki} \right) - TC(X)$$

$$\Leftrightarrow \max \sum_{i \in P_{ts}} \left(VA_i - ip_i(rt^+_i) \sum_{k \in V_{ts}} w_{ki} \right) - TC(X) + \sum_{i \in P_{ts}} ip_i(rt^+_i)$$

$$= \max fo(X) = \sum_{i \in P_{ts}} VA_i - TC(X) + \sum_{i \in P_{ts}} ip_i(rt^+_i) \left(1 - \sum_{k \in V_{ts}} w_{ki} \right) \tag{41}$$

Then, we construct a set of new optimization models, which is denoted as Mts_g, $g \in PO_{ts}$, to calculate the prices of all passengers' requests. Model Mts_g can be mathematically defined as
Model Mts_g
Objective function: Formula (41)
Constraints: Formulas (22)–(26), (28)–(38), and (42).

$$w_{kg} = 0, \text{ for all } k \in V_{ts} \tag{42}$$

We use "$X^*_{Mts_g}$" and "$X^*_{Mts_0}$" to represent the optimal solutions of model Mts_g ($g \in PO_{ts}$) and model Mts_0, respectively. Then the price of passenger request g is calculated by Formulas (43)

$$po_g = fo(X^*_{Mts_g}) - fo(X^*_{Mts_0}) + VA_g(X^*_{Mts_0}) \tag{43}$$

where "$fo()$" is the objective function of model Mts_0 (Formula (41)), and "$VA_g(X^*_{Mts_0})$" is Passenger(s) g's value given the optimal matching plan and routing sequence $X^*_{Mts_0}$. Specifically, $VA_g(X^*_{Mts_0}) = \alpha^p_i \sum_{k \in V_{ts}} w^*_{ki}$ based on Formula (20) (w_{ki}* is the value of decision

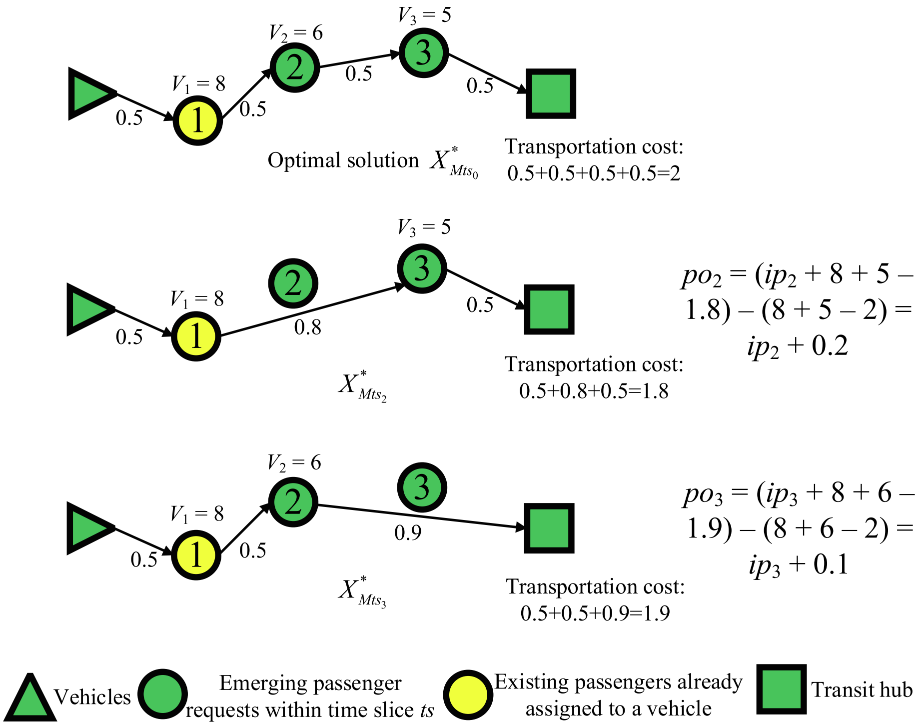

Fig. 3. An example to calculate on-demand passengers' prices.

Table 1
Incentive objectives and mechanism properties.

Incentive objectives	Mechanism properties	Related Propositions
Promoting passengers for participation by satisfying their mobility preferences	Preference-Based Individual Rationality (Definition 1)	Proposition 2
Promoting passengers to truthfully report their mobility preferences	Preference-Based Incentive Compatibility (Definition 2)	Proposition 3
Incentivizing the service provider to be financially sustainable	Financial Sustainability (Definition 4)	Proposition 4
Promoting passengers, especially regular commuters, to early schedule the service	Scheduling Preferability (Definition 5)	Proposition 8

variable w_{ki} in the optimal solution $X^*_{Mts_0}$).

Fig. 3 displays a simple example to calculate on-demand passengers' prices. In Fig. 3, Passenger 1 is an existing passenger who has already been assigned to the vehicle and whose price has already been given prior to this time slice. Passengers 2 and 3 send on-demand request within the current time slice *ts*. The mechanism only needs to calculate Passengers 2's and 3's prices. Note that in Fig. 3, only $X^*_{Mts_0}$ is the adopted matching plan and routing sequence. $X^*_{Mts_2}$ and $X^*_{Mts_3}$ are used to calculate the two passengers' prices, respectively, which will never be used for matching and routing.

4.4. Theoretical analysis for incentive objectives

This section uses theoretical analysis to demonstrate how the four incentive objectives can be achieved by mechanism properties under the proposed mechanism, as shown in Table 1. Proofs of some propositions are listed in Appendix B.

Proposition 1. *For scheduled service, the final price is always smaller than or equal to the intermediate price*:

$$ps_g \leqslant ip_g\left(rt_g^-\right)$$

The proof is presented in Appendix B.

Definition 1. (*Preference-based individual rationality*) If a mechanism is preference-based individual rational, for any passenger either scheduling the service or sending an on-demand request, all of the personalized requirements will always be satisfied (i.e. the following four conditions are satisifed):

Table 2
Constraints to satisfy passengers' mobility preferences.

Conditions in Definition 1	Scheduled passengers
(1)	Formulas (9), (31), and (32)
(2)	Formulas (11), (28), and (32)
(3)	Formulas (12), (29), and (30)

Condition (1) All passengers are able to arrive before their deadlines: $ART_g \leq \theta_g{}^{AD}$ or $\alpha_g{}^{AD}$ (ART_g is Passenger(s) g's arrival deadline).
Condition (2) All passengers' tolerable detour time will never be exceeded: $EIVT_g \leq \theta_g{}^{EIVT}$ or $\alpha_g{}^{EIVT}$ ($EIVT_g$ is Passenger(s) g's detour time).
Condition (3) All passengers' tolerable number of shared riders will never be exceeded: $NSR_g \leq \theta_g{}^{NR}$ or $\alpha_g{}^{NR}$ (NSR_g is the number of passengers sharing the ride with Passenger(s) g)
Condition (4) All passengers' prices will never exceed their maximum WTP prices: ps_g and $po_g \leq VA_g$

Proposition 2. *The hybrid mechanism is preference-based individual rational.*

Proof. The first three conditions are always satisfied because of the constraints of the optimization models MS_0 and Mts_0, as shown in Table 2.

No constraints in the optimization model are imposed to satisfy Condition (4) in Definition 1, and thus we need to use the following method to prove the validity of Condition (4) in Definition 1. We prove it for scheduled and on-demand passengers, separately.

For scheduled service, we discuss two cases based on a passengers' choice to accept the offer or not.

(1) If the passenger(s) does not accept the choice, then his value and the price are both zero:

$$VA_g = ps_g = 0$$

(2) If the passenger(s) accepts the offer, the passenger(s) will be served, and his value is deemed to be no less than the intermediate price (otherwise he will not accept the offer):

$$VA_g \geqslant ip_g\left(rt_g^-\right)$$

Based on Proposition 1, we have

$$ps_g \leqslant ip_g\left(rt_g^-\right)$$

Since $VA_g \geq ip_g(rt_g{}^-)$ in this case, we have

$$ps_g \leqslant ip_g\left(rt_g^-\right) \leqslant VA_g$$

For on-demand service, based on Formula (43), we have

$$VA_g\left(X^*_{Mts_0}\right) - po_g = VA_g\left(X^*_{Mts_0}\right) - \left(fo\left(X^*_{Mts_g}\right) - fo\left(X^*_{Mts_0}\right) + VA_g\left(X^*_{Mts_0}\right)\right)$$

$$= fo\left(X^*_{Mts_0}\right) - fo\left(X^*_{Mts_g}\right)$$

Compared with model Mts_0, each model Mts_g ($g \in PO_{ts}$) has an additional constraint (Formula (42)). This indicates that

$$fo\left(X^*_{Mts_0}\right) \geqslant fo\left(X^*_{Mts_g}\right)$$

Thus,

$$VA_g\left(X^*_{Mts_0}\right) - po_g = fo\left(X^*_{Mts_0}\right) - fo\left(X^*_{Mts_g}\right) \geqslant 0$$

We proved that both scheduled and on-demand passengers' mobility preferences are always satisfied, and thus the mechanism is preference-based individual rational. □

Definition 2. (*Preference-based incentive compatibility*) Truthful reporting of the mobility preference is a passenger's dominant strategy. Misreporting the mobility preference will cause at least one of the following consequences:

(1) Late arrival: $ART_g > \theta_g{}^{AD}$ or $\alpha_g{}^{AD}$.
(2) His maximum tolerable detour time will be exceeded: $EIVT_g > \theta_g{}^{EIVT}$ or $\alpha_g{}^{EIVT}$.
(3) His maximum tolerable number of shared riders will be exceeded: $NSR_g > \theta_g{}^{NR}$ or $\alpha_g{}^{NR}$.

(4) His utility, defined as the value minus the actual paid price ($Us_g = VA_g - ps_g$ and $Uo_g = VA_g - po_g$) will never increase.

A case demonstration of the property "preference-based incentive compatibility" is presented in Appendix E.

Proposition 3. *The hybrid mechanism is preference-based incentive compatible.*

Proof. It is equivalent to prove that under the circumstance in which any passenger misreports any mobility preferences, if the first three consequences do not happen, then the fourth consequence must happen. Thus, we only need to prove that if 1) the passenger does not arrive late, 2) his maximum tolerable detour time is not exceeded, and 3) his maximum tolerable number of shared riders is not exceeded, then the passenger's utility will never increase.

Firstly, we prove that the mechanism for scheduled service is incentive compatible.

Suppose that Passenger(s) g misreports any of the mobility preferences $\theta_g' = \{\theta_g'^{AD}, \theta_g'^{EIVT}, \theta_g'^{NR}\}$ instead of the truthful values $\theta_g = \{\theta_g^{AD}, \theta_g^{EIVT}, \theta_g^{NR}\}$. We denote the changed matching and routing plan as $X_{MS_0}^{'*}$ (the optimal solution to model MS_0' changed from model MS_0 due to the passenger's misreport) given that the three above-mentioned conditions are satisfied though the passenger misreports any mobility preferences. In the plan$X_{MS_0}^{'*}$, arrival deadline constraint (Formula (9)), maximum tolerable detour time constraint (Formula (11)), and maximum tolerable number of shared rides constraint (Formula (12)) are all satisfied. All other constraints of the model MS_0 are naturally satisfied given the plan $X_{MS_0}^{'*}$. Thus, $X_{MS_0}^{'*}$is still feasible to the model MS_0.

Model MS_g does not change regardless of Passenger(s) g's report, because in model MS_g, constrained by Formula (39), Passenger(s) g does not share the trip with others and his mobility preferences will be always satisfied regardless of his report. Thus, the optimal solution of model MS_g is constant ($X_{MS_0}^{'*}$).

Then, if Passenger(s) g misreports the mobility preferences, his price becomes to

$$ps_g' = ip_g\left(rt_g^-\right) - fs\left(X_{MS_g}^*\right) + fs\left(X_{MS_0}^{'*}\right)$$

Then, the scheduled passenger's utility (Us_g) becomes to

$$Us_g^{'} = VA_g - ps_g' = VA_g - ip_g\left(rt_g^-\right) + fs\left(X_{MS_g}^*\right) - fs\left(X_{MS_0}^{'*}\right)$$

As explained above,$X_{MS_0}^{'*}$ is feasible to model MS_0. $X_{MS_0}^*$ is optimal to model MS_0. Thus

$$fs\left(X_{MS_0}^{'*}\right) \geq fs\left(X_{MS_0}^*\right)$$

Then

$$Us_g' = VA_g - ip_g\left(rt_g^-\right) + fs\left(X_{MS_g}^*\right) - fs\left(X_{MS_0}^{'*}\right) \leq VA_g - ip_g\left(rt_g^-\right) + fs\left(X_{MS_g}^*\right) - fs\left(X_{MS_0}^*\right) = Us_g$$

where Us_g is Passenger(s) g's utility given that the mobility preferences are truthfully revealed.

Thus, if a passenger misreports mobility preferences for the scheduled service, the passenger's utility will never increase.

Secondly, we prove that the mechanism for on-demand service is preference-based incentive compatible. Suppose that Passenger(s) g misreports any of the mobility preferences $\alpha_i' = \{\alpha_i'^{AD}, \alpha_i'^{EIVT}, \alpha_i'^{NR}, \alpha_i'^{P}\}$ instead of the truthful values $\alpha_i = \{\alpha_i^{AD}, \alpha_i^{EIVT}, \alpha_i^{NR}, \alpha_i^{P}\}$. Then, the optimization model Mts_0 changes to Mts_0', and the optimal matching plan and routing sequence changes to $X_{Mts_0}^{'*}$ from $X_{Mts_0}^*$ (the optimal plan given that Passenger(s) g truthfully reports the mobility preferences).

Then, if Passenger(s) g misreports the mobility preferences, the system will mistake his actual value $VA_g(X)$ for $VA_g'(X)$, where $VA_g(X) = \alpha_i^P \sum_{k\in V_{ts}} w_{ki}$ and $VA_g'(X) = \alpha_i'^P \sum_{k\in V_{ts}} w_{ki}$ based on Formula (20), given the solution X. The objective function of Mts_0 becomes

$$fo^{'}(X) = \sum_{i\in P_{ts}, i\neq g} VA_i(X) + VA_g^{'}(X) - TC(X) \ + \ \sum_{i\in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k\in V_{ts}} w_{ki}\right)$$

However, model Mts_g does not change because in model Mts_g, constrained by Formula (42), Passenger(s) g is never served regardless of his report of the mobility preferences.

Thus, his price becomes to

$$po^{'}_g = fo\left(X_{Mts_g}^*\right) - fo^{'}\left(X_{Mts_0}^{'*}\right) + VA^{'}_g\left(X_{Mts_0}^{'*}\right)$$

Then the on-demand passenger's utility (Uo_g) becomes

$$Uo'_g = VA_g\left(X_{Mts_0}^{'*}\right) - po_g'$$

$$= VA_g\left(X_{Mts_0}^{'*}\right) - fo\left(X_{Mts_g}^*\right) + fo^{'}\left(X_{Mts_0}^{'*}\right) - VA^{'}_g\left(X_{Mts_0}^{'*}\right)$$

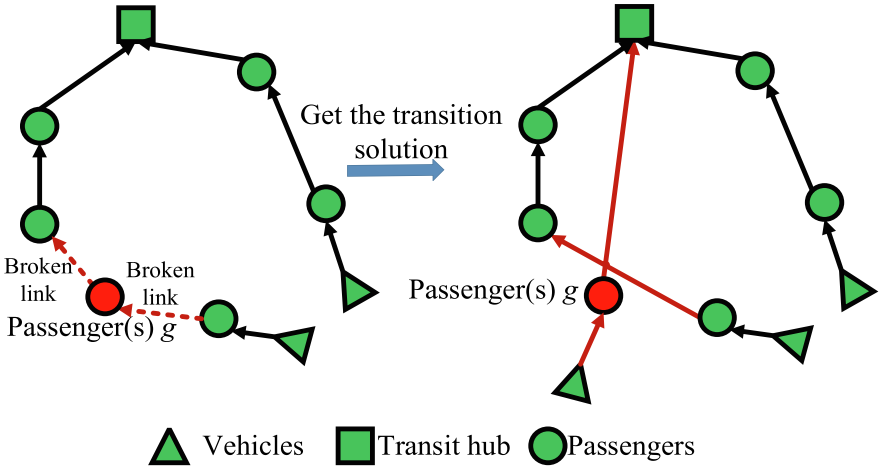

Fig. 4. Transition solution from model MS_0 to model MS_g.

$$= VA_g\left(X^{'*}_{Mts_0}\right) - fo\left(X^{*}_{Mts_g}\right) + \sum_{i\in P_{ts}, i\neq g} VA_i\left(X^{'*}_{Mts_0}\right) + VA'_g\left(X^{'*}_{Mts_0}\right) - TC\left(X^{'*}_{Mts_0}\right) + \sum_{i\in P_{ts}} ip_i\left(rt_i^{+}\right)\left(1 - \sum_{k\in V_{ts}} w^{'*}_{ki}\right) - VA'_g\left(X^{'*}_{Mts_0}\right)$$

$$= -fo\left(X^{*}_{Mts_g}\right) + \sum_{i\in P_{ts}} VA_i\left(X^{'*}_{Mts_0}\right) - TC\left(X^{'*}_{Mts_0}\right) + \sum_{i\in P_{ts}} ip_i\left(rt_i^{+}\right)\left(1 - \sum_{k\in V_{ts}} w^{'*}_{ki}\right)$$

$$= fo\left(X^{'*}_{Mts_0}\right) - fo\left(X^{*}_{Mts_g}\right)$$

where $w^{'*}_{ki}$ is the value of the decision variable of w_{ki} in the solution $X^{'*}_{Mts_0}$.

If Passenger(s) g's preferences are all satisfied in the plan $X^{'*}_{Mts_0}$, then $X^{'*}_{Mts_0}$ is feasible to model Mts_0. $X^{*}_{Mts_0}$ is optimal to Mts_0, and thus we have

$$fo\left(X^{'*}_{Mts_0}\right) \leq fo\left(X^{*}_{Mts_0}\right)$$

Therefore, we have

$$Uo_g' = fo\left(X^{'*}_{Mts_0}\right) - fo\left(X^{*}_{Mts_g}\right) \leq fo\left(X^{*}_{Mts_0}\right) - fo\left(X^{*}_{Mts_g}\right) = VA_g\left(X^{*}_{Mts_0}\right) - po_g = Uo_g$$

where "Uo_g" is Passenger(s) g's utility if he truthfully reveals the mobility preferences. This indicates that any on-demand passenger's utility will never increase due to misreporting the mobility preferences if their mobility preferences are all satisfied. □ The Appendix E gives a detailed example to demonstrate the property "preference-based incentive compatibility", showing that misreport will never help gain larger utility given that the mobility preferences are satisfied.

Definition 3. (*Transition solution from model MS_0 to model MS_g for scheduled service*) Transition solution for scheduled service, denoted as $Y_g = TS_g(X)$ for $g \in PS$, maps a feasible solution X of the model MS_0 to a feasible solution Y_g of the model MS_g. The pseudocode of this transition solution is presented in Algorithm 1 (see Appendix C).

Descriptively, if Passenger(s) g does not share the trip with others in X, then the transition solution remains the same as X: $Y_g = X$; otherwise, shown in Fig. 4, in the transition solution $TS_g(X)$, another vehicle with the lowest dispatching cost replaces the original vehicle to serve Passenger(s) g. The definition of transition solution is used to prove some properties (Propositions 5 and 10).

Proposition 4. *Suppose that X is any feasible solution of model MS_0, and $Y_g = TS_g(X)$. Then we have*

$$fs(X) \geqslant fs\left(Y_g\right) - \overline{dc}_g - ch_g$$

where $\overline{dc}_g$ is the maximum possible dispatching cost to pick up Passenger(s) g.
For the derailed proof, please refer to Appendix B.

Definition 4. (*Financial sustainability*) If a mechanism is financially sustainable, the collected prices are sufficient to cover the total transportation cost:

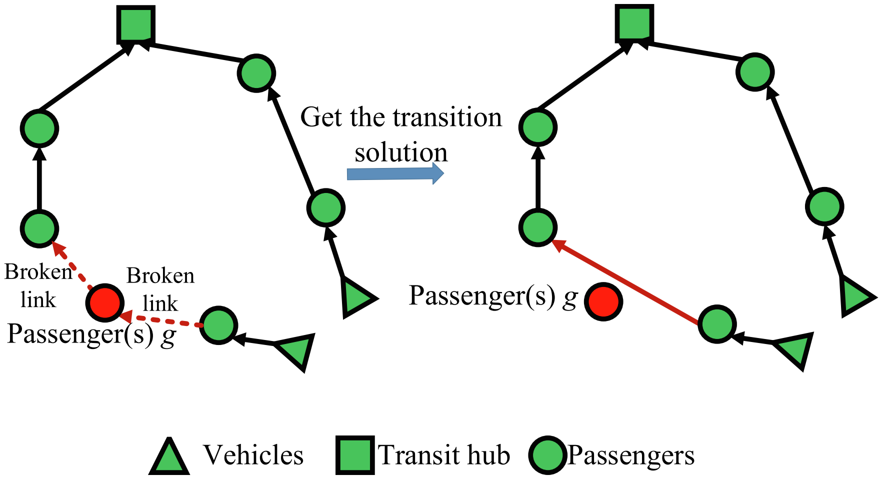

Fig. 5. Transition solution from model Mts_0 to model Mts_g.

$$\sum_g p_g \geqslant TC$$

Proposition 5. *If each passenger's intermediate price is no less than* $2(\overline{dc}_g + ch_g)$, *the scheduled service is financially sustainable. If* $ip_g(rt_g^-) \geq 2(\overline{dc}_g + ch_g)$, *then*

$$\sum_{g \in PS} ps_g \geqslant fs\left(X^*_{MS_0}\right)$$

where $fs\left(X^*_{MS_0}\right)$ *is the transportation cost given the solution* $X^*_{MS_0}$ *based on Formula (3).*

Proof. Let $Y_g^* = TS_g(X^*_{MS_0})$ (Definition 3). Y_g^* is feasible to model MS_g while $X^*_{MS_g}$ is optimal to model MS_g. Then

$$fs\left(Y^*_g\right) \geqslant fs\left(X^*_{MS_g}\right)$$

Based on Proposition 4, we have

$$fs\left(X^*_{MS_0}\right) \geqslant fs\left(Y^*_g\right) - \overline{dc}_g - ch_g$$

Therefore,

$$fs\left(X^*_{MS_0}\right) \geqslant fs\left(X^*_{MS_g}\right) - \overline{dc}_g - ch_g$$

Based on Formula (40), we have

$$ps_g = ip_g\left(rt^-_g\right) - fs\left(X^*_{MS_g}\right) + fs\left(X^*_{MS_0}\right)$$

$$\geqslant ip_g\left(rt^-_g\right) - \overline{dc}_g - ch_g$$

Given the condition $ip_g(rt_g^-) \geq 2(\overline{dc}_g + ch_g)$, we have

$$ps_g \geqslant ip_g\left(rt^-_g\right) - \overline{dc}_g - ch_g \geqslant \overline{dc}_g + ch_g$$

This indicates that a scheduled passenger's final price is no less than the minimum transportation cost for driving him to the transit hub.

Now consider a plan X_0, in which all passengers do not share the trip with others. Then, the transportation cost of X_0 is

$$fs(X_0) = \sum_{g \in PS} \sum_{k \in VS} ys_{ks} dc_{kg} + \sum_{g \in PS} ch_g$$

In addition, since $\overline{dc}_g$ is maximum possible dispatching cost to pick up Passenger(s) g, then

Table 3
Conditions to prove Proposition 8.

Utility of scheduling the service	Utility of sending an on-demand request	
(a) $Us_g \geq 0$	Rejected by the service	(b) $Uo_g = 0 \leq Us_g$
	Successfully assigned to a vehicle	(c) $Uo_g \leq Us_g$

$$\sum_{k\in VS} ys_{kg} dc_{kg} \leqslant \overline{dc}_g$$

Then,

$$fs(X_0) = \sum_{g\in PS}\sum_{k\in VS} ys_{ks} dc_{kg} + \sum_{g\in PS}(ch_g) \leqslant \sum_{g\in PS}\left(\overline{dc}_g + ch_g\right)$$

In addition, X_0 is feasible to model MS_0 and $X^*_{MS_0}$ is optimal to model MS_0. Then

$$fs\left(X^*_{MS_0}\right) \leqslant fs(X_0) \leqslant \sum_{g\in PS}\left(\overline{dc}_g + ch_g\right) \leqslant \sum_{g\in PS} ps_g$$

indicating that the scheduled service is financially sustainable if the condition $ip_g(rt_g^-) \geq 2(\overline{dc}_g + ch_g)$ is satisfied. □

We do not give detailed proof for the property of financial sustainability for the on-demand service here, but we draw the conclusion that as long as the intermediate is sufficiently high, the on-demand service will be financially sustainable, this is because on-demand passengers' prices are always no less than scheduled passengers' prices (please refer to Proposition 8).

Definition 5. (*Transition solution from model Mts_0 to model Mts_g in on-demand service*) The definition of this transition solution $Y_g = TO_g(X)$ for on-demand service can be found in Bian et al. (2020): in the transition solution Y_g, Passenger(s) g is simply removed from the service plan X. The pseudocode to obtain the transition solution is presented in Algorithm 2. The difference between the transition solutions for scheduled service and for on-demand service can be straightforwardly differentiated from the comparison of Figs. 4 and 5.

Proposition 6. *For any $g\in PO_{ts}$ when $Y_g = TO_g(X)$, we have*

$$TC\left(Y_g\right) \leqslant TC(X)$$

where the TC(X) is the transportation cost given the plan X.
The proof is listed in Appendix B.

Proposition 7. *For any $g\in PO_{ts}$, when $Y_g = TO_g(X)$, we have*

$$\sum_{i\in P_{ts}\backslash g} VA_i(X) = \sum_{i\in P_{ts}} VA_i\left(Y_g\right)$$

This proposition is proved in Appendix B.

Definition 7. (*Scheduling preferability*) If the mechanism is scheduling preferable, then any passenger's utility of scheduling the service is greater than or equal to that of sending an on-demand request. Mathematically, we have

$$Us_g = VA_g - ps_g \geqslant Uo_g = VA_g - po_g$$

Proposition 8. *The mechanism is scheduling preferable.*

Proof. If we can prove the validity of the following inequations in Table 3, Proposition 8 is proved.
Proof of inequation (a):
Proposition 2 proved that

$$Us_g = VA_g - ps_g \geqslant 0 \quad for \quad any\ g \in PS$$

Proof of inequation (b):
If the passenger sends an on-demand request, but he is rejected by the service, then $VA_g(X^*_{Mts_0}) = 0$, and $po_g = 0$ based on Formula (43) and thus $Uo_g = 0$. Thus, in this case we have

$$Us_g \geqslant Uo_g = 0 \quad for \quad any\ g \in PO_{ts}$$

Proof of inequation (c):
If the passenger sends an on-demand request, and he is successfully assigned to a vehicle, then, based on Formula (43), the price is

$$po_g = fo\left(X^*_{Mts_g}\right) - fo\left(X^*_{Mts_0}\right) + VA_g\left(X^*_{Mts_0}\right)$$

Let $Y_g^* = TO_g\left(X_{Mts_0}^*\right)$. Y_g* is feasible to model Mts_g and $X_{Mts_g}^*$ is optimal to model Mts_g. Thus,

$$fo\left(X_{Mts_g}^*\right) \geqslant fo\left(Y_g^*\right)$$

Then, we have

$$po_g = fo\left(X_{Mts_g}^*\right) - fo\left(X_{Mts_0}^*\right) + VA_g\left(X_{Mts_0}^*\right)$$

$$\geqslant fo\left(Y_g^*\right) - fo\left(X_{Mts_0}^*\right) + VA_g\left(X_{Mts_0}^*\right)$$

in which

$$fo\left(Y_g^*\right) = \sum_{i\in P_{ts}} VA_i\left(Y_g^*\right) - TC\left(Y_g^*\right) + \sum_{i\in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k\in V_{ts}} w_{ki}\right)$$

$$fo\left(X_{Mts_0}^*\right) = \sum_{i\in P_{ts}} VA_i\left(X_{Mts_0}^*\right) - TC\left(X_{Mts_0}^*\right) + \sum_{i\in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k\in V_{ts}} w_{ki}\right)$$

Based on the definition of the transition solution (Definition 5) in on-demand service, except the difference, $\sum_{k\in V_{ts}} w_{kg} = 0$ in Y_g* and $\sum_{k\in V_{ts}} w_{kg} = 1$ in $X_{Mts_0}^*$, all of other elements in $\sum_{i\in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k\in V_{ts}} w_{ki}\right)$ are identical. Then we have

$$po_g \geqslant fo\left(Y_g^*\right) - fo\left(X_{Mts_0}^*\right) + VA_g\left(X_{Mts_0}^*\right)$$

$$= ip_g\left(rt_g^+\right) + \left(\sum_{i\in P_{ts}} VA_i\left(Y_g^*\right) - \sum_{i\in P_{ts}\backslash g} VA_i\left(X_{Mts_0}^*\right)\right) + \left(TC\left(X_{Mts_0}^*\right) - TC\left(Y_g^*\right)\right)$$

Based on Propositions 6 and 7, respectively, we have

$$TC\left(X_{Mts_0}^*\right) - TC\left(Y_g^*\right) \geqslant 0$$

and

$$\sum_{i\in P_{ts}} VA_i\left(Y_g^*\right) - \sum_{i\in P_{ts}\backslash g} VA_i\left(X_{Mts_0}^*\right) = 0$$

Thus, we have

$$po_g \geqslant ip_g\left(rt_g^+\right)$$

In Proposition 1, we proved that
$ps_g \leqslant ip_g\left(rt_g^-\right)$.
Since ip_g is a monotone increasing function of a passenger's request time rt_g and $rt_g^+ > rt_g^-$, thus we have

$$ip_g\left(rt_g^+\right) \geqslant ip_g\left(rt_g^-\right)$$

Thus,

$$po_g \geqslant ip_g\left(rt_g^+\right) \geqslant ip_g\left(rt_g^-\right) \geqslant ps_g$$

Suppose that α_g^p is Passenger(s) g's WTP price if he is served. Then

$$VA_g\left(X_{Mts_0}^*\right) = VA_g\left(X_{MS_0}^*\right) = \alpha_g^p$$

because in this case the passenger(s) is served in both scheduled and on-demand service.
Thus, we have

$$Uo_g\left(X_{Mts_0}^*\right) = VA_g\left(X_{Mts_0}^*\right) - po_g \leqslant VA_g\left(X_{MS_0}^*\right) - ps_g = Us_g\left(X_{MS_0}^*\right) \quad \square$$

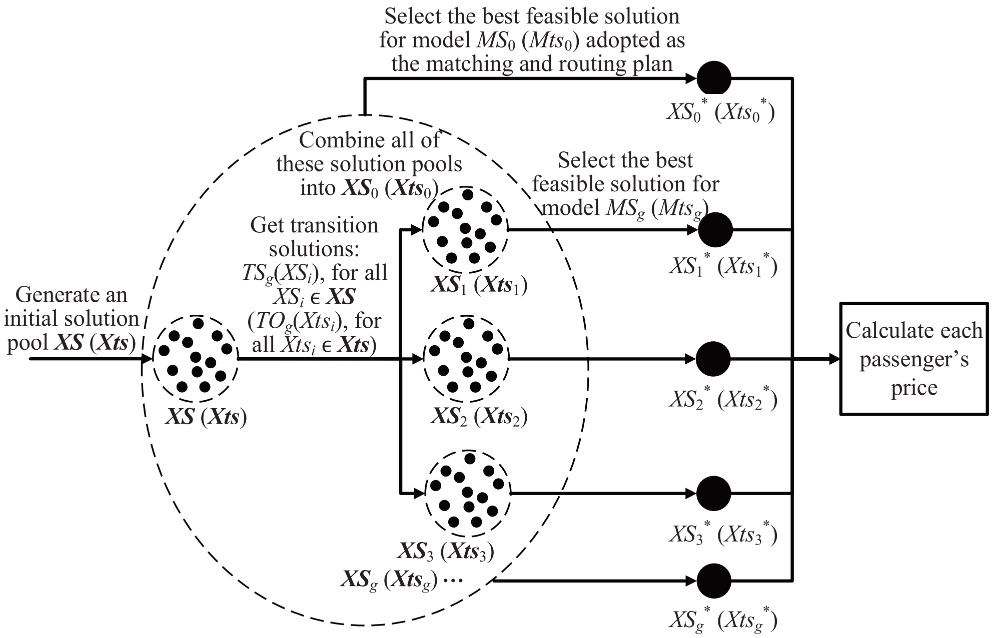

Fig. 6. Original Solution Pooling Approach (SPA).

5. Solution approach for large-scale problems

This section introduces an efficient heuristic algorithm for large-scale problems. For the notation except specified in the text, please refer to Appendix A.

5.1. Review of the original solution pooling approach (SPA)

Since the models for the FMR mechanism design problem are NP-hard Vehicle Routing problems (VRP), large-scale problems are difficult to be exactly solved. In our previous work, we developed a novel heuristic algorithm SPA (Bian and Liu, 2019b; Bian et al., 2020). SPA generates a large solution pool for each model. When the system requires the mechanism results, SPA selects the best feasible solution from each pool. SPA is still applicable for the dynamic FMR in this paper. The optimization models that need to be solved for scheduled ridesharing and on-demand ridesharing are MS_0 & MS_g (for all $g \in PS$) and Mts_0 & Mts_g (for all $g \in PO_{ts}$ for each time slice ts), respectively. Let $\boldsymbol{XS}_0$ and $\boldsymbol{Xts}_0$ represent the solution pools of model MS_0 and Mts_0, respectively, and let $\boldsymbol{XS}_g$ and $\boldsymbol{Xts}_g$ denote the solution pools of model MS_g and Mts_g, respectively. We let $XS_0{}^*$, $XS_g{}^*$, $Xts_0{}^*$, and $Xts_g{}^*$ to represent the best feasible solutions in solution pools $\boldsymbol{XS}_0$, $\boldsymbol{XS}_g$, $\boldsymbol{Xts}_0$, and $\boldsymbol{Xts}_g$, respectively. The solutions $XS_0{}^*$ and $Xts_0{}^*$ replace the real optimal solutions $X^*_{MS_0}$ and $X^*_{Mts_0}$ of the models MS_0 and Mts_0 to be adopted as the matching plan and routing sequence for scheduled service and on-demand service, respectively. The solutions $XS_g{}^*$ and $Xts_g{}^*$, instead of the real optimal solutions $X^*_{MS_0}$ and $X^*_{Mts_0}$ of the models MS_g and Mts_g, are used to determine the scheduled passengers' and on-demand passengers' prices, respectively, based on Formulas (40) and (43). Based on Bian and Liu (2019b), if the solution pool $\boldsymbol{XS}_g$ ($\boldsymbol{Xts}_g$) is included in $\boldsymbol{XS}_0$ ($\boldsymbol{Xts}_0$), then the mechanism is preference-based individual rational, and if the generation of solution pools $\boldsymbol{XS}_0$ ($\boldsymbol{Xts}_0$) and $\boldsymbol{XS}_g$ ($\boldsymbol{Xts}_g$) is independent of passengers' report of mobility preferences, then the mechanism is preference-based incentive compatible. The original SPA algorithm for the dynamic FMR problem in this paper can be described as follows.

- Step 1. Generate an initial solution pool, $\boldsymbol{XS}$ ($\boldsymbol{Xts}$), for model MS_0 (Mts_0).
- Step 2. Use Algorithm 1 (2) to get all transition solutions of each solution in the pool $\boldsymbol{XS}$ ($\boldsymbol{Xts}$): $XS_{gi} = TS_g(XS_i)$ ($Xts_{gi} = TO_g(Xts_i)$), where XS_i (Xts_i) is the ith solution in the pool $\boldsymbol{XS}$ ($\boldsymbol{Xts}$). The solution pool of model MS_g (Mts_g), $\boldsymbol{XS}_g$ ($\boldsymbol{Xts}_g$), consists of these transition solutions: $\boldsymbol{XS}_g = \{XS_{gi}$, for all $i\}$ ($\boldsymbol{Xts}_g = \{Xts_{gi}$, for all $i\}$).
- Step 3. Combine solution pools $\boldsymbol{XS}$ and $\boldsymbol{XS}_g$ (for all $g \in PS$) to get the final solution pool $\boldsymbol{XS}_0$: $\boldsymbol{XS}_0 = \{\boldsymbol{XS}, \boldsymbol{XS}_g$ (for all $g \in PS$)$\}$. (Combine solution pools $\boldsymbol{Xts}$ and $\boldsymbol{Xts}_g$ (for all $g \in PO_{ts}$) to get the final solution pool $\boldsymbol{Xts}_0$: $\boldsymbol{Xts}_0 = \{\boldsymbol{Xts}, \boldsymbol{Xts}_g$ (for all $g \in PO_{ts}$)$\}$).
- Step 4. Select the highest-quality solution from each pool, $XS_0{}^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS}_0\}$ ($Xts_0{}^* = \text{argmax}\{fo(X), X \in \boldsymbol{Xts}_0\}$) and $XS_g{}^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS}_g\}$ ($Xts_g{}^* = \text{argmax}\{fo(X), X \in \boldsymbol{Xts}_g\}$). Then, calculate all passengers' prices.

Fig. 6 gives the straightforward flow chart of the original algorithm. For the detailed introduction of the original SPA, please refer to Bian and Liu (2019b).

However, SPA has the following shortcomings in solving the mechanism design problem for the dynamic FMR in this paper:

(1) SPA sometimes obtains relatively low-quality solutions within a short computing time (e.g. a few seconds), as shown in the simulation results. This is because SPA does not use passengers' actual mobility preferences, but uses virtual values instead to generate solution pools, so that the generation of solutions pools ($\boldsymbol{XS}_0$, $\boldsymbol{XS}_g$, $\boldsymbol{Xts}_0$, and $\boldsymbol{Xts}_g$) is independent of passengers' report of their mobility preferences so as to sustain "incentive compatibility" (Bian and Liu, 2019b).
(2) SPA cannot necessarily sustain the properties of "financial sustainability" and "scheduling preferability". The reason is given below.

The mechanism is not necessarily financially sustainable. Let us go back to the proof of Proposition 5. In the proof, we have $fs\left(Y_g^*\right) \geqslant fs\left(X_{MS_g}^*\right)$, where $Y_g{}^*$ is the gth transition solution of $X_{MS_0}^*$ from model MS_0 to model MS_g, $Y_g^* = TS_g\left(X_{MS_0}^*\right)$. This is because $Y_g{}^*$ is a feasible solution of model MS_g and $X_{MS_g}^*$ is the optimal solution of model MS_g. In the SPA algorithm, $X_{MS_0}^*$ is replaced by $XS_0{}^*$, which is the best feasible solution selected from the pool $\boldsymbol{XS}_0$: $XS_0{}^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS}_0\}$. Also, $X_{MS_g}^*$ is replaced by $XS_g{}^*$, the best feasible solution in the pool $\boldsymbol{XS}_g$: $XS_g{}^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS}_g\}$. We let $YS_g{}^*$ be the gth transition solution of the solution $XS_0{}^*$: $YS_g{}^* = TS_g(XS_0{}^*)$. However, unlike $fs\left(Y_g^*\right) \geqslant fs\left(X_{MS_g}^*\right)$, we do not necessarily have $fs(YS_g{}^*) \geq fs(XS_g{}^*)$. This is because $YS_g{}^*$ is not necessarily in the solution pool $\boldsymbol{XS}_g$, and thus $YS_g{}^*$ is possible to be superior to the best solution in $\boldsymbol{XS}_g$ (i.e. $XS_g{}^*$). Thus, we cannot necessarily ensure the property of "financial sustainability".

The mechanism is not necessarily scheduling preferable. In the proof of inequation (c) in Proposition 8, we have $fo\left(X_{Mts_g}^*\right) \geqslant fo\left(Y_g^*\right)$, where $Y_g^* = TO_g\left(X_{Mts_0}^*\right)$, because $Y_g{}^*$ is feasible to model Mts_g while $X_{Mts_g}^*$ is optimal to model Mts_g. In the SPA algorithm, $X_{Mts_0}^*$ is replaced by $Xts_0{}^*$, which is the best solution in $\boldsymbol{Xts}_0$: $Xts_0{}^* = \text{argmax}\ \{fo(X), X \in \boldsymbol{Xts}_0\}$. Also, $X_{Mts_g}^*$ is replaced by $Xts_g{}^*$, the best feasible solution in the pool $\boldsymbol{Xts}_g$: $Xts_g{}^* = \text{argmax}\ \{fo(X), X \in \boldsymbol{Xts}_g\}$. We let $Yts_g{}^*$ be the gth transition solution of the solution $Xts_0{}^*$: $Yts_g{}^* = TO_g(Xts_0{}^*)$. However, unlike $fo\left(X_{Mts_g}^*\right) \geqslant fo\left(Y_g^*\right)$, we do not necessarily have $fo(Xts_g{}^*) \geq fo(Yts_g{}^*)$. This is because $Yts_g{}^*$ is not necessarily in the solution pool $\boldsymbol{Xts}_g$, and thus $Yts_g{}^*$ is possible to be superior to the best solution in $\boldsymbol{Xts}_g$ (i.e. $Xts_g{}^*$). Thus, we cannot necessarily ensure the property of "scheduling preferability".

5.2. Improved SPA

The improved SPA algorithm, namely Solution Pooling Approach with Closed Loop (SPACL), has the two following improvements compared with the original SPA.

(1) The improved SPA obtains higher quality matching plan and routing sequence than the original SPA does. The generation of solution pools ($\boldsymbol{XS}_0$, $\boldsymbol{XS}_g$, $\boldsymbol{Xts}_0$, and $\boldsymbol{Xts}_g$) in the improved SPA depends on passengers' reported mobility preferences, while the original SPA is "blind to" (independent of) passengers' reported mobility preferences when generating the solution pools. That is why the improved SPA may obtain higher quality matching plan and routing sequence than the original SPA does.
(2) The improved SPA overcomes the incapability of the original SPA to sustain the properties of "financial sustainability" and "scheduling preferability". In summary, the fundamental reason why original SPA cannot necessarily sustain the property of "financial sustainability" is that it is possible that the transition solution of $XS_0{}^*$, $YS_g{}^*$, is not in the solution pool $\boldsymbol{XS}_g$ ($YS_g{}^* \notin \boldsymbol{XS}_g$). Similarly, the fundamental reason why original SPA cannot necessarily sustain "scheduling preferability" is that it is possible that $Yts_g{}^*$ is not in the solution pool $\boldsymbol{Xts}_g$ ($Yts_g{}^* \notin \boldsymbol{Xts}_g$). Thus, the improved SPA will add a closed loop to ensure that the transition solutions of $XS_0{}^*$ and $Xts_0{}^*$ are in the solution pools $\boldsymbol{XS}_g$ and $\boldsymbol{Xts}_g$, respectively: $YS_g{}^* \in \boldsymbol{XS}_g$ and $Yts_g{}^* \in \boldsymbol{Xts}_g$, so that the properties "financial sustainability" and "scheduling preferability" are sustained.

However, due to the two improvement strategies, SPACL does not necessarily hold the property of "preference-based incentive compatibility". This is because the generation of solution pools depends on passengers' report of their mobility preferences given the two improvement strategies above, while "preference-based incentive compatibility" requires that solution pool generation must be independent of passengers' report of their mobility preferences (Bian and Liu, 2019b). We sacrifice "preference-based incentive compatibility" due to the following reasons.

In large-scale problems, it would be extremely challenging to simultaneously guarantee the "incentive compatibility" property, the solution quality, and computational efficiency. As demonstrated by the experimental results, the solution quality obtained by the original SPA is significantly lower than SPACL for large-scale examples although SPA is able to achieve "incentive compatibility". SPACL, though sacrifices the property of "incentive compatibility", ensures high-quality solutions, high computational speed, as well as the other three mechanism properties satisfied, i.e., preference-based individual rationality, financial sustainability, and scheduling preferability. We also consider proposing strategies in our future work to prevent passenger manipulating the system even without the property "incentive compatibility".

In Bian and Liu (2019b), it was proved that, in the SPA algorithm, when the solution pools for all models are completely independent of passengers' report of mobility preferences, then the mechanism can guarantee the property of "incentive compatibility". Thus, the basic idea of possible strategies for preventing manipulation is to fix the solution pools for scheduled service to the maximum

Table 4
Properties held by exact algorithms, SPA, and SPACL.

Algorithms	Mechanism design properties			
	Preference-based individual rationality	Preference-based incentive compatibility	Financial sustainability	Scheduling preferability
Exact algorithms	√ (Proposition 2)	√ (Proposition 3)	√ (Proposition 4)	√ (Proposition 8)
SPA	√ (Proved in Bian and Liu, 2019b)	√ (Proved in Bian and Liu, 2019b)	×	×
SPACL	√ (Proposition 9)	×	√ (Proposition 10)	√ (Proposition 11)

Note: "√" represents that the property is proved; "×" represents that the property cannot be proved but does not necessarily mean this property does not hold.

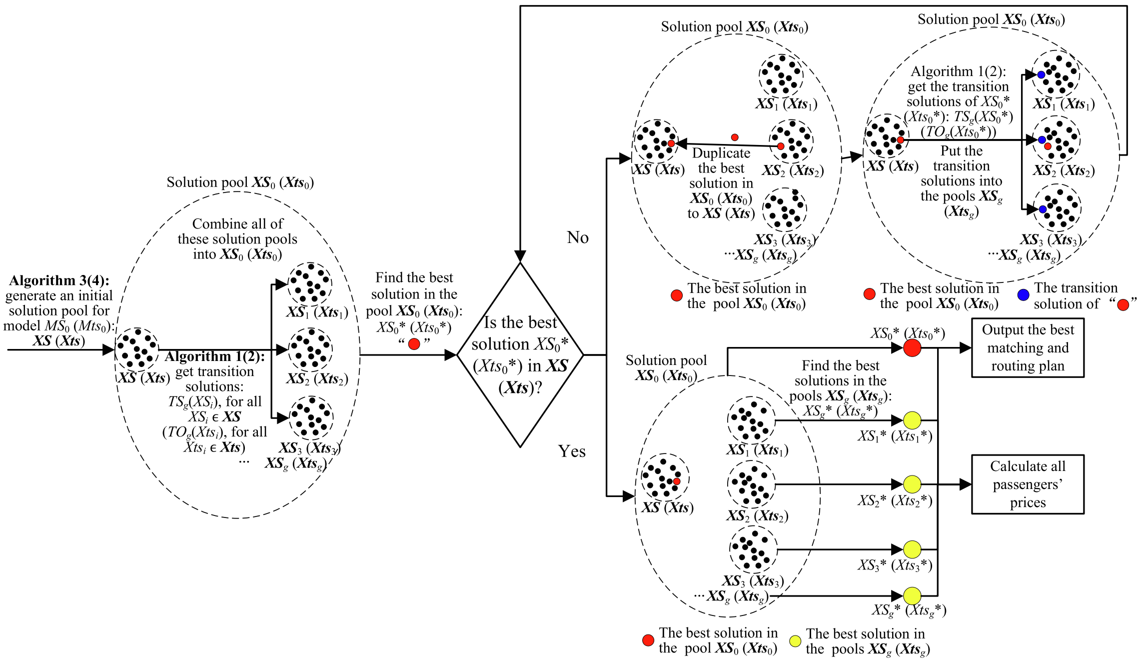

Fig. 7. Flow chart of SPACL.

extend and generate the solution pools for on-demand service as randomly as possible.

For scheduled service, a large portion of passengers could be commuters. After the service is implemented for a while, it is possible to estimate recurrent passenger requests, and thus we can keep a fixed solution pool corresponding to the optimization model for each time slot in the mechanism. Thus, we could maintain the property of "incentive compatibility" in most scenarios if the solution pools are mostly fixed. In the future work, we may develop a learning algorithm that preserves high-quality solutions for each solution pool based on historical data so that the mechanism could sustain the property "incentive compatibility" for large-scale problems.

For on-demand service, passengers' requests and mobility preferences are more difficult to predict. Thus, the strategy is to generate solution pools as randomly as possible so that passengers are difficult to anticipate the results of misreporting while truthfully reporting ensures their mobility preferences always satisfied given that the "preference-based individual rationality" always holds for SPACL (Proposition 9). In fact, SPACL has already incorporated this strategy because the solution pools in SPACL are generated by a Tabu Search algorithm that generates neighborhood solutions randomly in each iteration.

We develop a numerical example to demonstrate how SPACL could prevent passengers misreporting in Appendix E.

In summary, Table 4 presents the three solution approaches (exact algorithms, SPA, and SPACL) in sustaining the four important properties, preference-based individual rationality, preference-based incentive compatibility, financial sustainability, and scheduling preferability.

The steps of the SPACL are presented below. The flow chart of SPACL is shown in Fig. 7. The pseudocodes of the sub-algorithms used in SPACL are presented in "Algorithms 1, 2, 3, and 4" (See Appendix C). The pseudocodes of the main algorithm of SPACL to obtain the mechanism results for scheduled and on-demand service are presented in Algorithms 5 and 6 (See Appendix C), respectively.

- Step 1. Use Algorithm 3 (4) to generate an initial solution pool, ***XS*** (***Xts***), for model MS_0 (Mts_0). When generating the initial solution pool, the algorithm uses passengers' reported mobility preferences as input.

Table 5
Simulation parameter setting.

Simulation parameters	Values or methods for setting the values
Train departure time	9:00 am
Time to stop receiving scheduled requests (Ts)	8:00 am
Scheduled and on-demand passengers' arrival deadlines (θ_i^{AD} and α_i^{AD})	Randomly and uniformly generate from 8:40 am to 8:55 am
Number of scheduled requests	10
Number of on-demand requests	60
Number of passengers in each request (np_i)	1
Passenger origins	Uniformly distributed in an annular area (0.5-mile-radius inner circle and 5-mile-radius outer circle)
On-demand passengers' requesting time (rt_i)	Shown as Fig. 8
On-demand passengers' response deadline (RD_i) depending on the request urgency	$tnow + \max(0.5, 3 - (DT_i - rt_i)/30 \times 2.5)$ $tnow$: the current time DT_i: the train departure time rt_i: the request time
Number of emerging vehicles before re-optimization (em_{ts})	$em_{ts} = max(\lceil en_{ts}/2 \times (1 + rand) \rceil, 4)$ en_{ts}: number of emerging requests within time slice ts $rand$: a uniform distributed number between 0 and 1
Number of vehicles that are not immediately available	$\lceil rand \times 0.5 \times em_{ts} \rceil$ (available within 5 × $rand$ minutes)
Vehicle locations (x_i, y_i)	The coordinate (x_i, y_i) is uniformly distributed in the annular area where passenger origins are generated
Travel distance between two locations (d_{ij})	Using Euclidean distance
Transportation cost (c_{ij})	$c_{ij} = 0.5d_{ij}$
Travel time between two locations (t_{ij})	$t_{ij} = 2.5\sqrt{(xv_i - xv_j)^2 + (yv_i - yv_j)^2} + pt$ $xv_i = x_i + \varepsilon$ and $yv_i = y_i + \varepsilon$ $\varepsilon \sim N(\mu = 0, \sigma^2 = 0.1)$ $pt = 2$
Passengers' maximum WTP prices (α_i^P)	$\alpha_i^P = 3c_{i0} + \varepsilon_i$ c_{i0}: transportation cost to travel to the transit hub directly $\varepsilon_i \sim N(\mu = 4, \sigma^2 = 1)$
Tolerable number of shared riders (θ_i^{NR} and α_i^{NR})	$\lceil 5 \times rand \rceil$
Tolerable detour time (θ_i^{EIVT} and α_i^{EIVT})	$5 + 15 \times rand$
Parameters of the intermediate price	$cf = 1.5$, $dr = 1$, $UG(\Delta t_g) = 1.6 - \Delta t_g/100$

- Step 2. Use Algorithm 1 (2) to get all transition solutions of each solution in the pool $\boldsymbol{XS}$ ($\boldsymbol{Xts}$): $XS_{gi} = TS_g(XS_i)$ ($Xts_{gi} = TO_g(Xts_i)$), where XS_i (Xts_i) is the ith solution in the pool $\boldsymbol{XS}$ ($\boldsymbol{Xts}$). The solution pool of model MS_g (Mts_g), $\boldsymbol{XS_g}$ ($\boldsymbol{Xts_g}$), consists of these transition solutions: $\boldsymbol{XS_g} = \{XS_{gi}, \text{ for all } i\}$ ($\boldsymbol{Xts_g} = \{Xts_{gi}, \text{ for all } i\}$).
- Step 3. Combine solution pools $\boldsymbol{XS}$ and $\boldsymbol{XS_g}$ (for all $g \in PS$) to get the solution pool $\boldsymbol{XS_0}$: $\boldsymbol{XS_0} = \{\boldsymbol{XS}, \boldsymbol{XS_g} \text{ (for all } g \in PS)\}$. (Combine solution pools $\boldsymbol{Xts}$ and $\boldsymbol{Xts_g}$ (for all $g \in PO_{ts}$) to get the solution pool $\boldsymbol{Xts_0}$: $\boldsymbol{Xts_0} = \{\boldsymbol{Xts}, \boldsymbol{Xts_g} \text{ (for all } g \in PO_{ts})\}$).
- Step 4. Find the best solution from the solution pool $\boldsymbol{XS_0}$ ($\boldsymbol{Xts_0}$), $XS_0^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS_0}\}$ ($Xts_0^* = \text{argmax}\{fo(X), X \in \boldsymbol{Xts_0}\}$).
- Step 5. If $XS_0^* \notin \boldsymbol{XS}$ (if $Xts_0^* \notin \boldsymbol{Xts}$), duplicate XS_0^* into the pool $\boldsymbol{XS}$ (duplicate Xts_0^* into the pool $\boldsymbol{Xts}$) and then go to Step 6, otherwise go to Step 7.
- Step 6. Use Algorithm 1 (2) to get the transition solutions of XS_0^* (Xts_0^*): $YS_g^* = TS_g(XS_0^*)$, for all $g \in PS$ ($Yts_g^* = TO_g(Xts_0^*)$, for all $g \in PO_{ts}$). Then put YS_g^* (Yts_g^*) into the solution pools $\boldsymbol{XS_g}$ ($\boldsymbol{Xts_g}$). Return to Step 4.
- Step 7. Find the best solution from the pool $\boldsymbol{XS_g}$ ($\boldsymbol{Xts_g}$): $XS_g^* = \text{argmin}\ \{fs(X), X \in \boldsymbol{XS_g}\}$ ($Xts_g^* = \text{argmax}\{fo(X), X \in \boldsymbol{Xts_g}\}$).
- Step 8. Output XS_0^* (Xts_0^*) as the matching plan and routing sequence. Passengers' prices are determined by Formula (44) (45).

$$ps_g = ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs\left(XS_0^*\right) \tag{44}$$

$$po_g = fo\left(Xts_g^*\right) - fo\left(Xts_0^*\right) + VA_g\left(Xts_0^*\right) \tag{45}$$

In Step 1 in this SPACL, Algorithm 3 (4) uses the passengers' reported mobility preferences as input to generate the initial solution pool $\boldsymbol{XS}$ ($\boldsymbol{Xts}$) in order to improve the solution quality. Steps 5 and 6 can ensure $YS_g^* \in \boldsymbol{XS_g}$ and $Yts_g^* \in \boldsymbol{Xts_g}$ so that the mechanism can hold "financial sustainability" and "scheduling preferability".

5.3. Theoretical properties of the improved SPA

This section theoretically proves that SPACL sustains "preference-based individual rationality", "financial sustainability", and "scheduling preferability".

Proposition 9. *The hybrid mechanism obtained by SPACL is preference-based individual rational.*

The proof is presented in Appendix B.

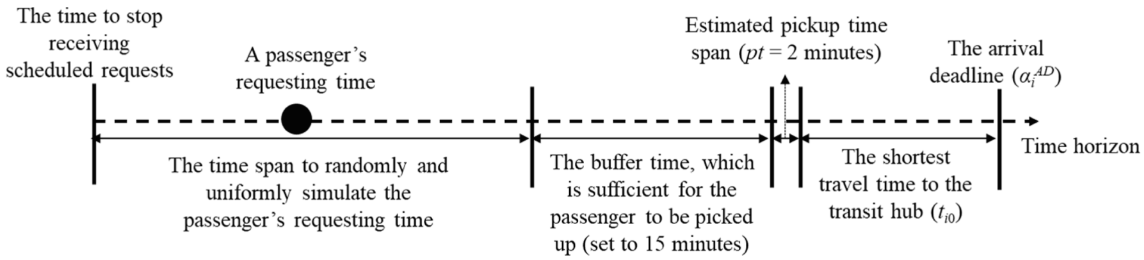

Fig. 8. Set of on-demand passengers' requests.

Proposition 10. *The scheduled service is financially sustainable if each passenger's intermediate price is no less than* $2(\overline{dc}_g+ch_g)$ *under the mechanism obtained by the SPACL algorithm. If* $ip_g(rt_g^-) \geq 2(\overline{dc}_g+ch_g)$, *then*

$$\sum_{g\in PS} ps_g \geqslant fs(XS_0^*)$$

where ps_g *is obtained by Formula (44):*$ps_g = ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs(XS_0^*)$.

The proof is presented in Appendix B.

Proposition 11. *The mechanism obtained by SPACL is scheduling preferable*

The proof is presented in Appendix B.

6. Simulation experiments

Simulation experiments are randomly generated by a computer. We generate a small-scale simulation experiment and 21 medium- and large-scale simulation examples. Note that the simulation examples achieve the purpose of test only and the developed methodologies can be adapted for practical application as well.

6.1. Small-scale simulation

6.1.1. Simulation setting

We assume that a train will depart from a certain train station at 9:00 am. Passengers can schedule the FMR service one hour before the train departure time (before 8:00 am). Passenger requests sent after 8:00 am will be treated as on-demand requests. We use the same setting method of simulation parameters for on-demand service as that in Bian et al. (2020) so that the designed mechanism and algorithm can be fairly compared with those in Bian et al. (2020). Table 5 presents the values or methods for setting the values of the simulation parameters. For the detailed setting of simulation examples, please refer to Bian et al. (2020).

6.1.2. Simulation results

Fig. 9 displays the simulation results for the scheduled service and partial on-demand service in the rolling horizon. We give the following clarifications for this figure. The mechanism results include matching plan and routing sequences and prices. The mechanism result for the scheduled service is present in Fig. 9(a). Vehicle 1, Vehicle 2, and Vehicle 3 cannot pick up any more additional passengers because some assigned passengers' maximum tolerable numbers of shared riders are reached. Thus, the routing plan of these three vehicles will never change in the future and the routes are removed from the planning horizon. Other vehicles (Vehicles 4 and 5) which can pick up additional passengers are remained in the planning horizon (see Fig. 9 (b)), rendering the possibility that scheduled passengers share the trip with subsequent on-demand passengers. After the scheduled service is determined, on-demand passenger requests occur. Fig. 9(b) shows the emerging passenger requests sent within the first time slice. Vehicle 4 has already arrived at a scheduled passenger's origin, and is serving or waiting for this passenger. The estimated remaining serving time is 1.34 min. The system deems that this vehicle will be available to be dispatched for picking up additional passengers after this "1.34 min". The end of the first time slice is 8:03:25 AM, which reaches one passenger's response deadline. The system conducts re-optimization and calculates the prices of emerging requests. The re-optimization results and the emerging passengers' prices are presented in Fig. 9(c). Then, the system continues to receive and process on-demand passengers' requests and obtain mechanism results.

Table 6 summarizes the results of scheduled service and all time slices of the on-demand service, including the start of time slice, the length of time slice, number of passenger requests, number of served passenger requests, collected price, transportation cost, and the total profit. We discuss the following findings from Table 6:

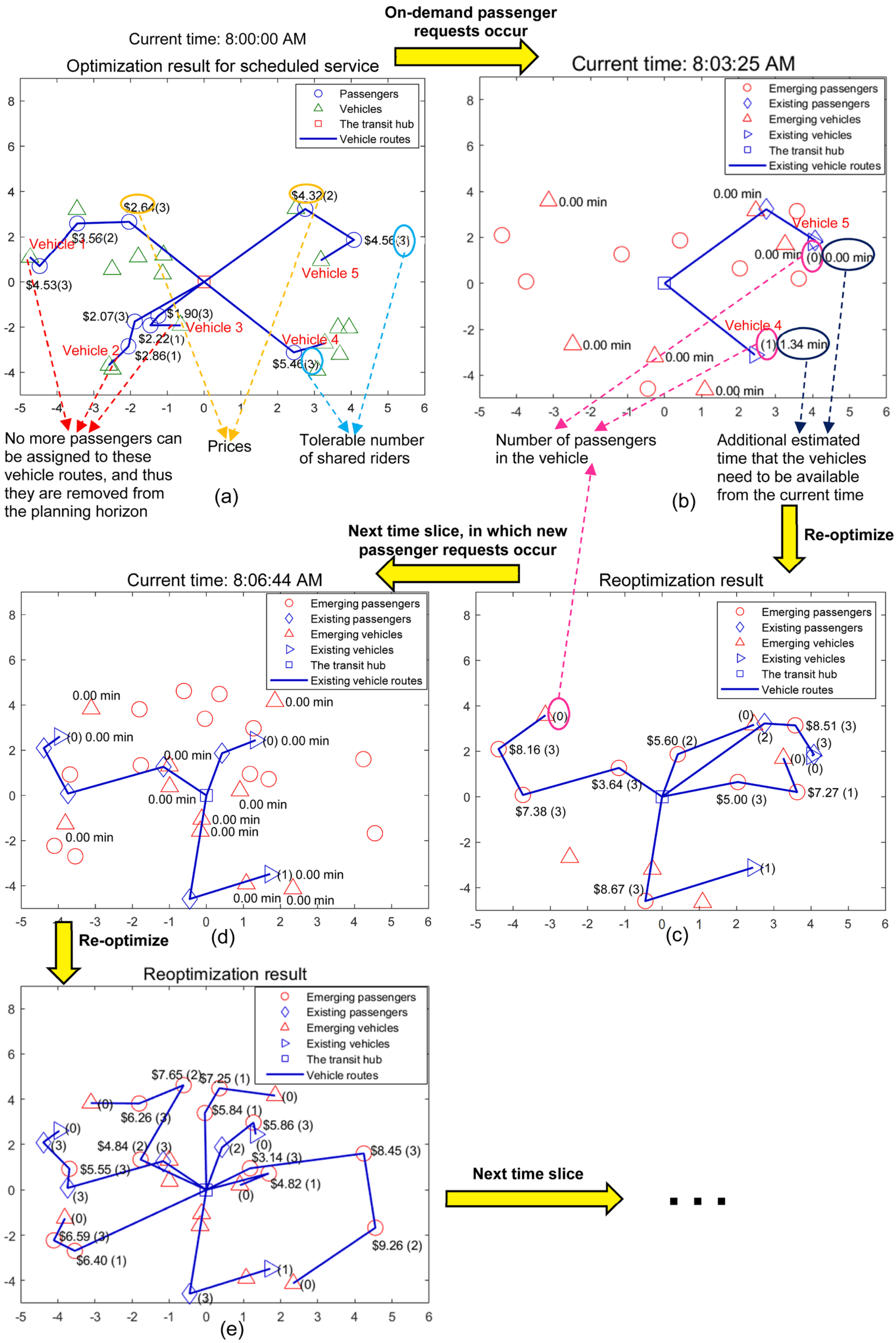

Fig. 9. The mechanism result presented on the rolling horizon.

Table 6
Summary of the mechanism results.

Service types		Start of time slice (AM.)	Time slice length (minutes)	Number of new passenger requests	Number of served passengers	Collected price ($)	Transportation cost ($)	Total profit ($)
Scheduled service		Before 8:00	Enough long	10	10	34.11	13.79	20.31
On-demand service: time slices 1–14	1	8:00:25	3.00	8	8	54.24	11.24	42.99
	2	8:04:04	2.66	13	13	81.91	17.84	64.06
	3	8:06:46	2.44	9	9	66.84	16.38	50.46
	4	8:10:11	2.15	4	4	23.81	4.95	18.86
	5	8:12:43	1.94	3	3	24.02	7.02	17.00
	6	8:15:21	1.72	5	4	26.16	0.73	25.43
	7	8:17:05	1.58	3	2	20.62	7.40	13.22
	8	8:19:19	1.39	4	4	23.10	2.90	20.20
	9	8:21:31	1.21	4	4	21.51	3.70	17.81
	10	8:23:24	1.05	1	0	0.00	0.00	0.00
	11	8:25:48	0.85	1	0	0.00	0.00	0.00
	12	8:26:56	0.76	1	1	2.81	0.05	2.76
	13	8:30:16	0.50	3	2	19.78	6.03	13.75
	14	8:32:11	0.50	1	0	0.00	0.00	0.00
			Total	70	64	398.90	92.04	306.86

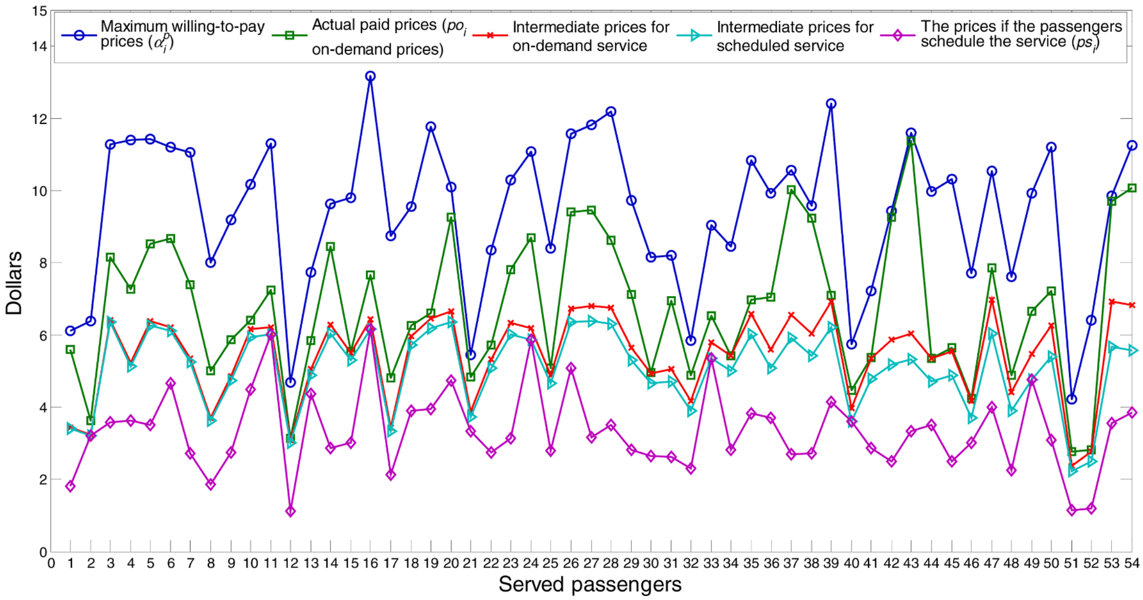

Fig. 10. On-demand passengers' price information.

(1) The length of time slice decreases as the time approaches to the train departure time. This is commensurate with the rule of determining the time slice length. When passengers are more urgent to catch the train at the transit hub, the response time should be close enough so that these passengers can have less waiting time, leading to a shorter time slice, and vice versa.
(2) Under this mechanism, totally 64 passengers are served among the 70 passengers requested. Six passengers are not served. There are two possible reasons: 1) the passenger requirements are so strict that they cannot be satisfied by the service; and 2) available vehicles are insufficient to serve all passengers.
(3) For each time slice, the collected prices can always cover the transportation cost. Totally 398.90 dollars are collected from the 64 passengers, which covers the 92.04-dollar transportation costs. The total profit in this simulation is 306.86 dollars, making the mechanism financially sustainable.

Fig. 10 presents the on-demand passengers' price information. There are five lines in this figure, from high to low representing passengers' maximum WTP prices, paid prices, the intermediate prices for on-demand service, the intermediate prices for scheduled service, and the prices if the passengers schedule the service in advance. From this figure, we can draw two conclusions.

Table 7
Comparison of algorithm performances.

Time slices	Problem scale		Objective function value (Formula (43))			Computing time (seconds)				
	Number of passengers	Number of available vehicles	CPLEX	SPA	SPACL	CPLEX			SPA	SPACL
						Optimization	Calculating prices	Total		
1	10	8	81.27	81.27	81.27	1.80	14.80	16.61	3.04	1.28
2	18	13	140.77	**139.73**	140.77	25.87	168.95	194.82	6.69	2.40
3	11	11	86.86	86.86	86.86	3.00	20.68	23.68	4.07	1.67
4	8	8	59.23	59.23	59.23	2.14	6.31	8.45	1.89	0.93
5	9	8	67.69	67.69	67.69	1.59	3.61	5.20	1.68	0.81
6	13	10	98.34	98.34	98.34	1.58	8.35	9.93	2.73	1.19
7	9	8	70.28	70.28	70.28	1.81	3.66	5.47	1.54	0.79
8	11	10	90.51	90.51	90.51	1.57	5.97	7.55	2.48	1.10
9	11	9	85.46	85.46	85.46	1.70	6.02	7.72	2.14	1.00
10	7	8	51.89	51.89	51.89	0.88	0.66	1.55	0.78	0.50
11	5	8	33.25	33.25	33.25	0.67	0.85	1.52	0.73	0.58
12	5	8	33.87	33.87	33.87	0.56	0.66	1.22	0.79	0.58
13	5	8	31.39	31.39	31.39	0.61	2.05	2.66	1.08	0.77
14	4	10	24.25	24.25	24.25	0.67	0.75	1.42	0.93	0.73

Table 8
Comparison between the online hybrid mechanism and the offline static mechanism (small-scale problem).

Time slices	Number of served passenger requests		Number of additional dispatched vehicles		Collection price ($)		Transportation cost ($)		Total Profit ($)	
	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM
Scheduled service	10	10	5	5	34.11	34.11	13.79	13.79	20.31	20.31
1	8	8	3	4	54.24	48.99	11.24	13.20	42.99	35.79
2	13	13	5	6	81.91	82.09	17.84	20.84	64.06	61.26
3	9	9	4	5	66.84	72.70	16.38	17.67	50.46	55.03
4	4	3	1	1	23.81	19.63	4.95	4.22	18.86	15.41
5	3	3	2	2	24.02	24.02	7.02	7.02	17.00	17.00
6	4	3	0	2	26.16	21.35	0.73	6.14	25.43	15.20
7	2	2	2	2	20.62	20.62	7.40	7.40	13.22	13.22
8	4	4	1	2	23.10	24.50	2.90	6.59	20.20	17.90
9	4	4	1	3	21.51	25.00	3.70	6.66	17.81	18.34
10	0	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00
11	0	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00
12	1	1	0	1	2.81	4.81	0.05	2.05	2.76	2.76
13	2	2	2	2	19.78	19.78	6.03	6.03	13.75	13.75
14	0	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00
Total	64	62	26	35	398.90	397.60	92.04	111.63	306.86	285.97

ONHBM: the online hybrid mechanism.
OFFSM: the offline static mechanism.

(1) Passengers' maximum WTP prices are no less than the actual paid price. This property can always be satisfied for any passenger in any scenario, because we proved the property "preference-based individual rationality", which ensures that passengers' mobility preferences can always be satisfied.
(2) If these on-demand passengers schedule the service early, they will be charged with a lower price. Thus, passengers have the incentive to schedule the service instead of sending on-demand requests. In Proposition 8, we have proved that on-demand price (the green line in Fig. 10) is always greater than or equal to the intermediate price for on-demand service (the red line in Fig. 10), the intermediate price for on-demand service is never less than the intermediate price for the scheduled service (the light blue line in Fig. 10), and the intermediate price for scheduled service is never less than the price of scheduling the service (the purple line in Fig. 10). This is commensurate with Fig. 10.

Table 7 displays the performance of the three algorithms, CPLEX solver, SPA, and SPACL, for this small-scale numerical example. We find that, except the second time slice, in which the SPA algorithm obtains a lower objective function value (the bold number) than the other two algorithms, the three algorithms can obtain the exactly identical objective function values for all of the other time slices. In this small-scale numerical example, the solver CPLEX obtains the exactly optimal solutions for all time slices. Thus, we can conclude that both SPA and SPACL can obtain very high-quality solutions for small-scale mechanism design problems. For the computational

Table 9
Comparison of algorithm performances for medium-scale examples.

Simulation examples	Number of time slices	Number of time slices in which the algorithm gets the best solutions among the three algorithms			Summation of objective function values in all time slices (Formula (43))			Average computing time of all time slices (seconds)		
		CPLEX	SPA	SPACL	CPLEX	SPA	SPACL	CPLEX	SPA	SPACL
S15_O90	16	**14**	10	**14**	**1263.92**	1257.32	1263.86	79.52	3.46	**1.46**
S20_O120	16	**16**	9	13	**1825.53**	1813.58	1825.05	145.40	4.21	**1.66**
S25_O150	19	16	7	**18**	2260.06	2266.94	**2279.16**	130.61	4.89	**1.90**
S30_O180	19	12	7	**19**	2592.99	2623.18	**2638.72**	242.02	7.01	**2.58**
S35_O210	20	16	7	**18**	2660.06	2777.68	**2805.57**	242.90	7.16	**2.60**
S40_O240	19	12	8	**18**	2918.82	3068.04	**3097.90**	347.39	10.17	**3.57**
S45_O270	20	13	5	**19**	2893.43	3152.47	**3195.27**	367.98	9.33	**3.28**
S50_O300	23	16	6	**23**	2 failures	3999.90	**4052.24**	356.32	9.76	**3.62**
S55_O330	20	10	6	**18**	2 failures	3557.25	**3613.09**	569.00	14.06	**4.81**
S60_O360	22	13	7	**21**	3 failures	4646.67	**4714.43**	562.66	14.08	**4.66**

speed, the solver CPLEX can obtain the mechanism results very quickly within a short time for all time slices except the second time slice with the largest scale, in which CPLEX spends 194.82 s solving the mechanism design problem. By comparison, SPA is faster than the CPLEX solver in obtaining the mechanism results for all time slices. The longest time spent by SPA is only 6.69 s. However, in contrast, the SPACL is even faster than CPLEX solver and SPA. The longest computing time is only 2.40 s among the 14 time slices. We also use the medium- and large-scale simulations to compare the three solution approaches in Section 6.2.

In order to demonstrate the superiority of the online hybrid mechanism (ONHBM), we compare it with the static offline mechanism (OFFSM) proposed in our previous work (Bian et al., 2020), in terms of the number of served passenger requests, number of dispatched vehicles, collected price, transportation cost, and profit. The comparison results are presented in Table 8. In the static offline mechanism, once the matching plan and routing sequence is determined, it will never be changed to accommodate newly occurred passengers' requests. In this small-scale simulation, 64 passengers are served under the ONHBM mechanism, two more than that under the OFFSM mechanism. Under the ONHBM mechanism, totally 26 vehicles are dispatched to serve the 64 passengers. In contrast, the OFFSM mechanism will dispatch 35 vehicles to serve the 62 passengers. It seems that ONHBM mechanism can ensure a higher vehicle occupancy rate than the OFFSM mechanism by dispatching less vehicles and serve more passengers. The two mechanisms do not make a significant difference in the total collected prices. However, the ONHBM mechanism induces 92.04-dollar transportation cost, saving nearly 20 dollars' transportation cost compared with the OFFSM mechanism. Thus, ONHBM mechanism makes more profit than the OFFSM mechanism (ONHBM $306.86 versus OFFSM $285.97). The possible reason why OFFSM is outperformed by ONHBM is that OFFSM lacks a dynamic re-optimization approach to re-matching and re-routing to accommodate newly occurred passenger requests, and therefore, OFFSM sustains a relatively high vehicle empty seat rate and transportation cost.

6.2. *Medium- and large-scale simulation examples*

We generate 10 medium-scale and 11 large-scale simulation examples to compare CPLEX, SPA, and SPACL. For the medium-scale examples, the numbers of scheduled passenger requests (*ns*) increase from 15 to 60 by the interval of 5. For the large-scale examples, the numbers of scheduled passenger requests (*ns*) increase from 100 to 200 by the interval of 10. The total numbers of on-demand requests are set to 6 times of scheduled passengers ($6 \times ns$) in both medium- and large-scale examples. The numerical example is denoted as "S*ns*_O*no*", where *ns* is the number of scheduled passengers and *no* is the number of on-demand passengers. For example, "S15_O90" represents the numerical example with 15 scheduled passengers and 90 on-demand passengers.

6.2.1. *Results of medium-scale simulation examples*

The setup of the data for the medium-scale simulation examples is identical with that of the small-scale simulation in Section 6.1. The medium-scale simulation examples are run at a Dell computer with processor Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz and 8 GB RAM. We set the maximum computing time of CPLEX, SPA, and SPACL in solving the optimization model Mts_0 to 30 s (excluding the pre-processing time), which is deemed as the maximum allowable response time of the algorithm in the on-demand ridesharing scenario.

Table 9 displays the performance of CPLEX, SPA, and SPACL (the bold numbers indicate the best performance). We record the number of time slices in which each solution approach gets the best solutions among the three algorithms. When the scale of the simulation example is relatively small (e.g., S15_O90 and S20_O120), CPLEX can obtain the best solutions with the highest frequency compared with the other two algorithms, SPA and SPACL. However, with the scale increasing, SPACL has the highest frequency in

Table 10
Performances of solution approaches in obtaining mechanism results in the time slice with the largest-scale problem (medium-scale examples).

Simulation examples	The largest problem scale among all time slices		Computing time (seconds)			Objective function value (Formula (43))		
	Number of passengers	Number of vehicles	CPLEX (for optimization only)	SPA	SPACL	CPLEX	SPA	SPACL
S15_O90	21	16	10004.61	8.85	**3.12**	**166.89**	162.69	166.86
S20_O120	27	21	64.34	10.39	**3.97**	**215.57**	213.02	215.35
S25_O150	25	21	10019.33	13.90	**4.57**	**189.78**	187.99	189.50
S30_O180	31	27	10009.29	17.61	**5.85**	**251.04**	248.00	248.08
S35_O210	34	27	10014.28	23.56	**7.01**	270.67	265.15	**271.76**
S40_O240	41	38	10010.03	33.86	**12.43**	**326.95**	319.49	326.12
S45_O270	45	37	10154.67	38.38	**13.57**	355.46	348.44	**356.06**
S50_O300	50	44	10044.84	41.84	**15.43**	415.07	408.57	**418.56**
S55_O330	52	41	10016.17	48.35	**16.65**	403.02	396.70	**407.29**
S60_O360	65	37	10040.55	47.94	**15.14**	534.03	529.49	**540.26**

Table 11
Comparison of algorithm performances for large-scale examples.

Simulation examples	Number of time slices	Summation of objective function values in all time slices (Formula (43))		Average computing time of all time slices (seconds)	
		SPA	SPACL	SPA	SPACL
S100_O600	26	6286.88	**6618.39**	11.04	**3.01**
S110_O660	25	7029.30	**7371.41**	15.07	**3.93**
S120_O720	25	7901.25	**8232.93**	18.41	**4.57**
S130_O780	27	9445.48	**9878.94**	18.04	**3.96**
S140_O840	27	8195.45	**8724.90**	22.70	**4.64**
S150_O900	29	10061.43	**10637.58**	24.24	**4.23**
S160_O960	28	10395.75	**11138.64**	26.75	**4.96**
S170_O1020	28	10241.04	**10942.21**	32.86	**5.93**
S180_O1080	28	11052.41	**11847.49**	33.83	**6.19**
S190_O1140	28	10391.50	**11351.90**	43.05	**7.27**
S200_O1200	26	12267.05	**13197.67**	51.01	**9.34**

Table 12
Performances of solution approaches in obtaining mechanism results in the time slice with the largest-scale problem (large-scale examples).

Simulation examples	The largest problem scale among all time slices		Computing time (seconds)			Objective function value (Formula (43))		
	Number of passengers	Number of vehicles	CPLEX (for optimization only)	SPA	SPACL	CPLEX	SPA	SPACL
S100_O600	87	31	10039.08	51.75	**9.16**	662.66	621.53	**692.25**
S110_O660	103	50	10090.17	107.61	**26.44**	Failure	827.36	**866.32**
S120_O720	103	54	10097.07	124.99	**26.92**	Failure	802.78	**832.58**
S130_O780	104	51	10095.10	94.18	**17.71**	Failure	826.71	**855.84**
S140_O840	107	48	10093.67	136.68	**20.03**	Failure	835.89	**888.35**
S150_O900	135	45	10197.00	172.22	**20.96**	Failure	919.76	**1069.98**
S160_O960	143	53	10315.57	207.29	**35.17**	Failure	1043.25	**1182.45**
S170_O1020	155	62	10720.36	246.38	**37.45**	Failure	1105.57	**1253.25**
S180_O1080	139	54	10288.52	201.35	**30.30**	Failure	1026.67	**1154.91**
S190_O1140	161	58	10620.93	260.99	**40.40**	Failure	1138.53	**1303.59**
S200_O1200	152	62	10738.39	255.35	**45.76**	Failure	1123.82	**1257.26**

obtaining the best solutions compared with CPLEX and SPA. The algorithm SPA has the lowest frequency in obtaining the best solutions for all examples, because it is either outperformed by CPLEX for relatively small-scale examples or outperformed by SPACL for relatively large-scale examples. This conclusion can also be reflected by the summation of objective function values presented in Table 9. We should note that when the scales of the simulation example become large enough, the solver CPLEX is unable to obtain a feasible solution within a reasonable amount of time. That is why the table shows the number of time slices in which CPLEX fails to obtain a feasible solution in the column "summation of objective function values in all time slices". In the simulation examples, S50_O300, S55_O330, and S60_O360, CPLEX has 2, 2, and 3 failures to obtain feasible solutions, respectively. In Table 9, we also compare the three solution approaches in terms of the average computing times of all time slices for all simulation examples. We find that SPACL has the fastest computing speed compared with CPLEX and SPA. The longest average computing time of SPACL is only 4.81 s, which is sufficiently prompt for on-demand ridesharing. In contrast, CPLEX's computing time is so long that it is not practical for implementation.

Table 13
Comparison between the online hybrid mechanism and the offline static mechanism (medium- and large-scale problems).

Simulation examples	Total number of served passengers		Total number of dispatched vehicles		Total collected prices ($)		Total transportation cost ($)		Total profit ($)	
	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM	ONHBM	OFFSM
S15_O90	**102**	**102**	**40**	51	581.51	**594.66**	**134.90**	153.95	**446.61**	440.71
S20_O120	**131**	**131**	**52**	60	730.21	**754.31**	**170.41**	195.50	**559.80**	558.81
S25_O150	**169**	165	**66**	77	**955.75**	926.61	**215.81**	237.37	**739.94**	689.24
S30_O180	**204**	**204**	**77**	92	1103.41	**1113.04**	**237.32**	270.13	**866.09**	842.91
S35_O210	**245**	242	**91**	101	**1336.06**	1295.47	**282.69**	298.99	**1053.37**	996.48
S40_O240	**272**	268	**104**	118	**1453.28**	1433.78	**306.99**	336.23	**1146.29**	1097.55
S45_O270	**308**	305	**116**	126	**1680.07**	1664.92	**345.62**	370.54	**1334.45**	1294.38
S50_O300	**340**	338	**122**	142	**1771.69**	1769.82	**351.52**	389.01	**1420.17**	1380.81
S55_O330	**378**	374	**139**	149	**1996.39**	1983.26	**393.96**	418.49	**1602.43**	1564.77
S60_O360	**413**	**413**	**148**	168	2206.73	**2213.69**	**433.69**	475.33	**1773.04**	1738.36
S100_O600	**661**	652	**235**	241	**3412.68**	3356.09	**698.05**	702.66	**2714.63**	2653.43
S110_O660	**718**	716	**259**	269	3723.91	**3732.58**	**745.87**	767.20	**2978.04**	2965.38
S120_O720	**805**	794	**292**	303	**4128.31**	4080.33	**821.01**	832.68	**3307.30**	3247.65
S130_O780	**868**	855	**308**	316	**4470.99**	4428.96	**866.58**	883.03	**3604.41**	3545.93
S140_O840	**941**	934	**335**	345	4835.45	**4859.57**	**938.63**	982.17	**3896.82**	3877.40
S150_O900	**1010**	991	**357**	371	**5258.74**	5139.42	**1003.46**	1014.97	**4255.28**	4124.45
S160_O960	**1038**	1018	**367**	376	**5372.38**	5290.93	**1035.34**	1057.18	**4337.04**	4233.75
S170_O1020	**1127**	1124	**402**	417	5775.72	**5808.23**	**1104.01**	1158.49	**4671.71**	4649.74
S180_O1080	**1195**	1169	**427**	436	**6159.51**	6022.49	**1189.47**	1190.34	**4970.04**	4832.15
S190_O1140	**1267**	1244	**446**	454	**6528.99**	6421.65	**1230.35**	1232.72	**5298.64**	5188.93
S200_O1200	**1327**	1318	**464**	477	**6837.29**	6815.57	**1256.19**	1305.60	**5581.10**	5509.97

ONHBM: the online hybrid mechanism.
OFFSM: the offline static mechanism.

In each simulation example, the problem scale varies in different time slices. We select the time slice with the largest-scale problem in each simulation example for additional analysis. The maximum computing time of the CPLEX solver in solving an optimization model is extended to 10,000 s to verify the solution quality obtained by the proposed SPACL algorithm. Table 10 presents the comparison results of the three solution approaches in terms of computing speed and solution qualities (the bold numbers indicate the best performance).

In terms of computational speed, CPLEX is very slow in solving the optimization models (Mts_0) in the time slice with the largest-scale problem for all simulation examples except the example S20_O120, which is solved by CPLEX for 64.34 s. All other examples cannot be exactly solved by CPLEX within 10,000 s. Compared with CPLEX, SPA is much faster in obtaining the mechanism results. The maximum computing time of SPA for all medium-scale examples do not exceed 50 s. However, this amount of time is still not prompt enough for practical application. The SPACL is even much faster than SPA. The maximum computing time of SPACL to get the mechanism result for the largest-scale problem (S55_O330 with 52 passengers and 41 vehicles) is only 16.65 s.

In terms of the solution quality, when the problem scale is relatively small (e.g., examples with scale less than 41 passengers × 38 vehicles), the quality of solutions obtained by CPLEX solver is slightly higher than that of the proposed SPACL, but the gap is extremely small. The largest gap is only 1.18%, which occurs in the example S30_O180. As the problem scale increases (e.g., problem scale greater than 45 passengers × 37 vehicles), SPACL begins to outperform the CPLEX solver in terms of the solution quality. For the comparison between SPA and SPACL in terms of the solution quality, it seems that SPACL always outperforms SPA.

6.2.2. *Results of large-scale simulation examples*

In the large-scale simulation examples, we decrease the number of vehicles to $em_{ts} = max(\lceil en_{ts}/4 \times (1 + rand) \rceil,\ 2)$ (en_{ts} is number of emerging requests within time slice *ts*, and *rand* is a uniform distributed number between 0 and 1) so that we can investigate the algorithm performance under the scenario with limited vehicle recourses. CPLEX is not applied to run each whole simulation example, because the CPLEX cannot obtain a high-quality solution within reasonable amount of time based on the medium-scale experiment results. We only use CPLEX to run for the time slice with the largest-scale problem and set the maximum running time as 10,000 s to solve the optimization model Mts_0. The large-scale simulation examples are run at a more advanced Dell computer with processor Intel (R) Core(TM) i7-10510U CPU @ 1.80 GHz and 32 GB RAM.

Table 11 displays the performance of SPA and SPACL (the bold numbers indicate the best performance) in obtaining the mechanism results for all large-scale simulation examples. It is straightforward that SPACL outperforms SPA in terms of both computational speed and solution quality. In addition, we select the time slice with the largest-scale problem in each simulation example for comparative analysis. The maximum computing time of the CPLEX solver in solving an optimization model is set to 10,000 s. Table 12 presents the comparison results of the three solution approaches to the time slice with the largest problem scale from the 11 large-scale simulation examples (the bold numbers indicate the best performance). From the results, we find that SPACL has a higher performance than CPLEX and SPA in terms of both computational time and solution quality. When the problem scale is sufficiently large (e.g., problem scale greater than 103 passengers × 50 vehicles), the CPLEX solver is unable to return a feasible solution within 10,000 s, while SPACL is able to obtain mechanism results within 50 s for all large-scale problems. We believe that the computing time can be further reduced by using more advanced computational resource and the parallel computing technique.

From the experimental results, we give the recommendation of algorithm adoption based on the three algorithms' computational speed and capability of sustaining the four mechanism properties. For small-scale problems (e.g., problem scale smaller than 10 passengers × 10 vehicles), the CPLEX solver is recommended as the solution approach because CPLEX is efficient in solving small-scale problems and can simultaneously guarantee the four mechanism design properties as CPLEX solver can get exact solutions. For medium-scale problems (e.g., scales between "10 passengers × 10 vehicles" and "30 passengers × 25 vehicles"), SPA is recommended as the solution approach because SPA is fast in solving medium-scale problems and most importantly can sustain the property of "preference-based individual rationality" and "preference-based incentive compatibility" which are the most two important properties in mechanism design theory. For large-scale problems (e.g., scales greater than "30 passengers × 25 vehicles"), the property "incentive compatibility" is very difficult to hold when SPA may not efficiently solve the problem within acceptable time, and thus SPACL is recommended as the solution approach, which is much more efficient compared with SPA and CPLEX for large-scale problems.

At last, we compare the online hybrid mechanism with the offline static mechanism for the medium- and large-scale simulation examples in Table 13 (the bold numbers indicate the better performance). First, the online hybrid mechanism dispatches fewer vehicles to serve no less passengers than the static offline mechanism does. This is because, under the online hybrid mechanism, the vehicle routing plan is adjusted dynamically to accommodate new passengers' requests in real time, while, under the static offline mechanism, the matching plan and routing sequence will never be changed after the vehicle is dispatched, and thus newly occurred passengers will not be served by these dispatched vehicles. The real-time accommodation of newly occurred passengers saves dispatched vehicles and serves more passengers. Therefore, the online hybrid mechanism can achieve a lower vehicle empty seat rate than the static offline mechanism. Second, from the comparison in total collected prices, we cannot easily tell which mechanism collects higher prices. However, the online hybrid mechanism significantly saves the transportation cost, and thus makes more profits compared with the static offline mechanism.

7. Conclusions

This paper designed an online hybrid mechanism with four incentive objectives for the FMR service involving mixed scheduled and on-demand passengers. We proposed and proved four properties, namely "preference-based individual rationality", "preference-based incentive compatibility", "financial sustainability", and "scheduling preferability", to achieve the four incentive objectives, respectively. We proved that the improved SPA algorithm successfully sustains the properties of "preference-based individual rationality", "financial sustainability", and "scheduling preferability" in obtaining large-scale mechanism results. From the simulation results, we can draw the two major conclusions: 1) Compared with the static offline mechanism proposed in our previous work, the proposed online hybrid mechanism can serve more passengers, reduce vehicle dispatches, achieve lower vehicle empty seat rate, save transportation cost, and make more profits; 2) The improved SPA outperforms the commercial solver CPLEX and the original SPA in obtaining the mechanism results in terms of both solution quality and computational speed, especially for large-scale simulation examples.

CRediT authorship contribution statement

Zheyong Bian: Conceptualization, Methodology, Writing – original draft. **Yun Bai:** Methodology. **Xiang Liu:** Conceptualization, Writing – review & editing. **Bijun Wang:** Methodology, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

We would like to express gratitude to graduate students Gladstone Lotsi and Lei Yang from the program of Supply Chain and Logistics Technology, Department of Construction Management, University of Houston for taking the time to review and proofread the paper.

Appendix A. Notation

See Tables 14a–14c.

Table 14a
Notation in the preliminary optimization model.

Sets	
PS	Set of ns scheduled passenger requests (at location nodes), $PS = \{1, 2, \ldots, ns\}$
VS	Set of ms vehicles available for dispatching for scheduled passenger requests, $VS = \{ns+1, ns+2, \ldots, ns+ms\}$
H	The transit hub (node), $H = \{0\}$
Variables	
$xs_{ijk} = \begin{cases} 1 \text{ Vehicle } k \text{ travels from Passenger(s) } i\text{'s location to Passenger } j\text{'s location} \\ 0 \text{ Otherwise} \end{cases} k \in VS, i, j \in PS$	
$ys_{ki} = \begin{cases} 1 \text{ Passenger(s) } i\text{'s request is the first pickup task assigned to Vehicle } k \\ 0 \text{ Otherwise} \end{cases} k \in VS, i \in PS$	
$zs_{ki} = \begin{cases} 1 \text{ Passenger(s) } i \text{ is the last to be served by Vehicle } k \text{ before traveling to the transit hub} \\ 0 \text{ Otherwise} \end{cases} k \in VS, i \in PS$	
$ws_{ki} = \begin{cases} 1 \text{ Passenger(s) } i \text{ is served by Vehicle } k \\ 0 \text{ Otherwise} \end{cases} k \in VS, i \in PS$	
X	$X = \{xs_{ijk}, ys_{ki}, zs_{ki}, ws_{ki}\}$, representing a matching plan and routing sequence
IVT_i	Travel time spent in a vehicle.
NR_i	Number of passengers sharing the trip with Passenger(s) i.
Parameters	
np_i	Number of passengers in request i.
dc_{ki}	The estimated dispatching cost of Vehicle k to pick up Passenger(s) i.
c_{ij}	The transportation cost to travel from node i to node j ($i, j \in PS$)
ch_i	The transportation cost from passenger location i to the transit hub ($i \in PS$)
t_{ij}	The travel time from location i to location j, $i \in VS \cup PS$ and $j \in PS \cup H$.
Q_k	Vehicle k's seat capacity.
θ_i^{AD}	Latest arrival time.
θ_i^{NR}	Tolerable number of shared passengers.
θ_i^{EIVT}	The maximum tolerable detour time (i.e., extra travel time spent in a vehicle beyond the direct shipment time).

Table 14b
Notation in the re-optimization model.

Sets	
P_{ts}	Set of n_{ts} passenger requests, $P_{ts} = \{1, 2, \ldots, n_{ts}\}$. These passengers include all scheduled and on-demand passengers who send requests before the end of the time slice ts but have not been picked up yet, but exclude those whose matching plan will not change anymore because the vehicle cannot pick up more passengers since the vehicle seats are all occupied or some shared riders' maximum tolerable numbers of shared riders are reached.
V_{ts}	Set of m available vehicles to serve passenger requests, $V_{ts} = \{n_{ts}+1, n_{ts}+2, \ldots, n_{ts}+m_{ts}\}$. These vehicles include all three types introduced in Section 3.3.
Variables	
$x_{ijk} = \begin{cases} 1 \text{ Vehicle } k \text{ travels from Passenger(s) } i\text{'s location to Passenger } j\text{'s location} \\ 0 \text{ Otherwise} \end{cases} k \in V_{ts}, i, j \in P_{ts}$	
$y_{ki} = \begin{cases} 1 \text{ Vehicle } k \text{ will serve Passenger(s) } i \text{ next} \\ 0 \text{ Otherwise} \end{cases} k \in V_{ts}, i \in P_{ts}$	
$z_{ki} = \begin{cases} 1 \text{ Passenger(s) } i \text{ is the last to be served by Vehicle } k \text{ before traveling to the transit hub} \\ 0 \text{ Otherwise} \end{cases} k \in V_{ts}, i \in P_{ts}$	
$w_{ki} = \begin{cases} 1 \text{ Passenger(s) } i \text{ is served by Vehicle } k \\ 0 \text{ Otherwise} \end{cases} k \in V_{ts}, i \in P_{ts}$	
Parameters	
dc_{ki}	The dispatching cost of Vehicle k to serve the next passenger (s) i.
ch_i	The transportation cost to directly drive passenger(s) i to the transit hub ($i \in P_{ts}$)
cv_k	If Vehicle k is already dispatched before the current time slice, then cv_k equals the transportation cost of this vehicle to travel from the vehicle location to the transit hub. Otherwise, $cv_k = 0$.
Q_k	The remaining seat capacity of Vehicle k before the re-optimization.
AT_k	The time when Vehicle k will be available.
DLS_k	The latest arrival time of Vehicle k. This deadline is determined only by the passengers who are already picked up by the vehicle before the re-optimization. For example, Vehicle k already has two passengers in it. If these two passengers' arrival deadlines are dl_1 and dl_2, then, $DLS_k = \min(dl_1, dl_2)$. Passengers who are not picked up by the vehicle do not affect DLS_k.
RNR_k	The maximum allowable number of additional riders that Vehicle k can pick up. This parameter is determined by the number of passengers who will be served before re-optimization, as well as the requirements of the passengers, who are already in the vehicle, on the number of co-riders. For example, if one passenger is already picked up by Vehicle k, this passenger's maximum tolerable number of co-riders is 3, and another passenger must be picked up as required in the previous routing plan, then $RNR_k = 2$, indicating that this vehicle can pick up two additional passengers at most.
$NPIV_k$	The number of passengers already picked up by Vehicle k

(continued on next page)

Table 14b (*continued*)

$RIVT_k$	The maximum allowable travel time for Vehicle k to reach the transit hub, which is determined by the maximum tolerable in-vehicle travel times of the passengers who are already in Vehicle k. For example, Vehicle k has already picked up two passengers. The two passengers have already stayed in the vehicle for 5 and 8 min, respectively. Their maximum tolerable travel times spent in vehicles are 15 and 20 min, respectively. Then $RIVT_k = \min(15 - 5, 20 - 8) = 10$ min.
δ_{ki}	The indicator parameter: if Vehicle k is already dispatched to pick up passenger(s) i, $\delta_{ki} = 1$; otherwise $\delta_{ki} = 0$.
α_i^{AD}	Arrival deadline of on-demand request i.
α_i^{NR}	Maximum tolerable number of shared passengers of on-demand request i.
α_i^{EIVT}	Maximum tolerable detour time of on-demand request i.
α_i^P	WTP price of on-demand request i.
VA_i	The value of on-demand request i. If he is served, his value is equal to his maximum WTP price: $VA_i = \alpha_i^P$; otherwise, his value is zero: $VA_i = 0$.

Table 14c
All other notation.

Notation	Explanation
PO_{ts}	The set of passenger requests sent within time slice ts
en_{ts}	Number of passenger requests sent in time slice ts. That is $en_{ts} = \|PO_{ts}\|$
Ts	The time to process all scheduled passengers' requests and to start to receive on-demand requests
ip_g	The intermediate price specified in Formula (1)
cf	The constant initial fee in the intermediate price ip_g
dr	The distance rate to determine the intermediate price ip_g
d_g	Passenger(s) g's travel distance of direct shipment from the origin to the transit hub
DT_g	The train departure time
rt_g	Passenger(s) g's request time
Δt_g	The time difference between the train departure time DT_g and the request time rt_g ($\Delta t_g = DT_g - rt_g$)
$UG(\Delta t_g)$	The urgency coefficient, a monotone increasing function of Δt_g
rt_g^-	Scheduled passenger(s) g's request time: $rt_g = rt_g^-$ if $rt_g < Ts$
rt_g^+	On-demand passenger(s) g's request time: $rt_g = rt_g^+$ if $rt_g \geq Ts$
$ip_g(rt_g^-)$	Scheduled passenger(s) g's intermediate price
$ip_g(rt_g^+)$	On-demand passenger(s) g's intermediate price
ps_g	Scheduled passenger's final price formulated by Formula (40)
po_g	On-demand passenger's final price Formula (43)
MS_0	The optimization model for schedule service, defined by Formulas (3)–(9) and (11)–(17)
MS_g	A series of models used in the scheduled passengers' price calculation formula (Formula (40)), defined by Formulas (3)–(9), (11)–(17), and (39)
Mts_0	The re-optimization model for on-demand service for one time slice (time slice ts), defined by Formulas (18), (22)–(26), and (28)–(38)
Mts_g	A series of models used in the on-demand passengers' price calculation formula (Formula (43)), defined by Formulas (41), (22)–(26), (28)–(38), and (42)
$fs(X)$	Objective function value model MS_0 given the plan X for the scheduled service (Formula (3))
$fo(X)$	Objective function value of model Mts_0 for the on-demand service (Formula (41))
$TS_g(X)$	Obtain the gth transition solution from model MS_0 to Model MS_g, $g \in PS$
$TO_g(X)$	Obtain the gth transition solution from model Mts_0 to Model Mts_g, $g \in PO_{ts}$
$\overline{dc}_g$	The maximum possible dispatching cost to pick up Passenger(s) g
$X^*_{MS_0}$	The theoretical optimal solution of model MS_0
$X^*_{MS_g}$	The theoretical optimal solution of model MS_g
$X^*_{Mts_0}$	The theoretical optimal solution of model Mts_0
$X^*_{Mts_g}$	The theoretical optimal solution of model Mts_g
$\boldsymbol{XS}_0$	The solution pool of model MS_0
$\boldsymbol{XS}_g$	The solution pool of model MS_g
$\boldsymbol{Xts}_0$	The solution pool of model Mts_0
$\boldsymbol{Xts}_g$	The solution pool of model Mts_g
$\boldsymbol{XS}$	The initial solution pool of model MS_0
$\boldsymbol{Xts}$	The initial solution pool of model Mts_0
X_0	A solution for scheduled service, in which all passengers do not share the trip with any other passengers. In other words, passengers take the non-ridesharing service and are shipped to the transit hub directly.
XS_0*	The best feasible solution in solution pool $\boldsymbol{XS}_0$
XS_g*	The best feasible solution in solution pool $\boldsymbol{XS}_g$
Xts_0*	The best feasible solution in solution pool $\boldsymbol{Xts}_0$
Xts_g*	The best feasible solution in solution pool $\boldsymbol{Xts}_g$
YS_g*	The gth transition solution of XS_0* for scheduled service: $YS_g^* = TS_g(XS_0^*)$
Yts_g*	The gth transition solution of Xts_0* for on-demand service: $Yts_g^* = TO_g(Xts_0^*)$
XS_{0i}	ith solution in the solution pool $\boldsymbol{XS}_0$
Xts_{0i}	ith solution in the solution pool $\boldsymbol{Xts}_0$
XS_i	ith solution in the initial solution pool $\boldsymbol{XS}$
XS_{gi}	ith solution in solution pool $\boldsymbol{XS}_g$, which is the transition solution of XS_{i0}: $XS_{gi} = TS_g(XS_{i0})$
Xts_i	ith solution in the initial solution pool $\boldsymbol{Xts}$
Xts_{gi}	ith solution in solution pool $\boldsymbol{Xts}_g$, which is the transition solution of Xts_{i0}: $Xts_{gi} = TO_g(Xts_{i0})$

Appendix B. Proofs of Propositions

Proposition 1. *For scheduled service, the final price is always smaller than or equal to the intermediate price*:

$$ps_g \leqslant ip_g\left(rt_g^-\right)$$

Proof. The scheduled passengers' prices are given by Formula (40):

$$ps_g = ip_g\left(rt_g^-\right) - fs\left(X_{MS_g}^*\right) + fs\left(X_{MS_0}^*\right)$$

Compared with model MS_0, each model MS_g ($g \in PS$) has an additional constraint (Formula (39)). This indicates that the optimal solution ($X_{MS_0}^*$) to model MS_0 has a smaller objective value than that ($X_{MS_g}^*$) of MS_g. That is

$$fs\left(X_{MS_g}^*\right) \geqslant fs\left(X_{MS_0}^*\right)$$

Thus

$$ps_g \leqslant ip_g\left(rt_g^-\right) \quad \square$$

Proposition 4. *Suppose that X is any feasible solution of model MS_0 and $Y_g = TS_g(X)$. Then we have*

$$fs(X) \geqslant fs\left(Y_g\right) - \overline{dc}_g - ch_g$$

where $\overline{dc}_g$ is the maximum possible dispatching cost to pick up Passenger(s) g.

Proof. If $\sum_{i \in PS \backslash g} ws_{ki} np_i = 0$ in X, then $Y_g = X$ based on Algorithm 1, and thus

$$fs(X) = fs\left(Y_g\right) \geqslant fs\left(Y_g\right) - \overline{dc}_g - ch_g$$

If $\sum_{i \in PS \backslash g} ws_{ki} np_i > 0$, we can find k that $ws_{kg} = 1$. In the transition solution Y_g, Vehicle k' is dispatched to pick up Passenger(s) g. Then $\overline{dc}_g \geqslant dc_{k'g}$, since $\overline{dc}_g$ is the maximum possible dispatching cost to pick up Passenger(s) g. Based on triangle inequality assumption, we have the following cases:

- If there exist i and j that $x_{igk} = 1$ and $x_{gjk} = 1$, then $fs(X) - \left(fs\left(Y_g\right) - \overline{dc}_g - ch_g\right) \geqslant fs(X) - \left(fs\left(Y_g\right) - dc_{k'g} - ch_g\right) = c_{ig} + c_{gj} - c_{ij} \geqslant 0$
- If $z_{kg} = 1$ and there exists i that $x_{igk} = 1$, then $fs(X) - \left(fs\left(Y_g\right) - \overline{dc}_g - ch_g\right) \geqslant fs(X) - \left(fs\left(Y_g\right) - dc_{k'g} - ch_g\right) = c_{ig} + ch_g - ch_i \geqslant 0$.
- If $y_{kg} = 1$ and there exists j that $x_{gjk} = 1$, then $fs(X) - \left(fs\left(Y_g\right) - \overline{dc}_g - ch_g\right) \geqslant fs(X) - \left(fs\left(Y_g\right) - dc_{k'g} - ch_g\right) = dc_{kg} + c_{gj} - dc_{kj} \geqslant 0$.

Thus,

$$fs(X) \geqslant fs\left(Y_g\right) - ch_g \quad \square$$

Proposition 6. *If $Y_g = TO_g(X)$, for any $g \in PO_{ts}$, we have*

$$TC\left(Y_g\right) \leqslant TC(X)$$

where the TC(X) is the transportation cost given the plan X.

Proof. If $\sum_{k \in V_{ts}} w_{kg} = 0$ in X, then $Y_g = X$. Thus

$$TC\left(Y_g\right) = TC(X).$$

If $\sum_{k \in V_{ts}} w_{kg} = 1$, there exists k that $w_{kg} = 1$. Let $\Delta TC = TC(X) - TC(Y_g)$. Based on triangle inequality assumption, we have the following cases:

(1) If there exist i and j that $x_{igk} = 1$ and $x_{gjk} = 1$, then $\Delta TC = c_{ig} + c_{gj} - c_{ij} \geqslant 0$.
(2) If $z_{kg} = 1$ and there exists i that $x_{igk} = 1$, then $\Delta TC = c_{ig} + ch_g - ch_i \geqslant 0$.
(3) If $y_{gk} = 1$ and there exists j that $x_{gjk} = 1$, then $\Delta TC = dc_{kg} + c_{gj} - dc_{kj} \geqslant 0$.

(4) If $y_{gk}=1$, $z_{gk}=1$, and Vehicle k has already been dispatched before the current time slice, then $\Delta TC = dc_{kg}+ch_g-cv_k > 0$.
(5) If $y_{gk}=1$, $z_{gk}=1$, and Vehicle k has not been dispatched before the current time slice, then $\Delta TC = dc_{kg}+ch_g > 0$.

Thus,

$$TC(Y_g) \leqslant TC(X). \quad \square$$

Proposition 7. *If $Y_g = TO_g(X)$, for any $g \in PO_{ts}$, we have*

$$\sum_{i\in P_{ts}\backslash g} VA_i(X) = \sum_{i\in P_{ts}} VA_i(Y_g)$$

Proof. If $\sum_{k\in V_{ts}} w_{kg} = 0$, $Y_g = X$, and thus $\sum_{i\in P_{ts}} VA_i(X) = \sum_{i\in P_{ts}} VA_i(Y_g)$. Since $\sum_{k\in V_{ts}} w_{kg} = 0$, $VA_g(X)=0$. Thus

$$\sum_{i\in P_{ts}\backslash g} VA_i(X) = \sum_{i\in P_{ts}} VA_i(Y_g)$$

If $\sum_{k\in V_{ts}} w_{kg} = 1$ in X, then Passenger(s) g is served in X. Then,

$$VA_g(X) = \alpha_g^p$$

Passenger(s) g is not served in Y_g. All of other served passengers in X are still served in Y_g. Thus

$$\sum_{i\in P_{ts}} VA_i(X) = \sum_{i\in P_{ts}} VA_i(Y_g) + \alpha_g^p$$

$$\Rightarrow \sum_{i\in P_{ts}\backslash g} VA_i(X) + \alpha_g^p = \sum_{i\in P_{ts}} VA_i(Y_g) + \alpha_g^p$$

$$\Rightarrow \sum_{i\in P_{ts}\backslash g} VA_i(X) = \sum_{i\in P_{ts}} VA_i(Y_g) \quad \square$$

Proposition 9. *The hybrid mechanism obtained by SPACL is preference-based individual rational.*

Proof. The first three conditions in Table 1 are always satisfied because XS_0* is feasible to the model MS_0 and Xts_0* is a feasible solution of model Mts_0* satisfying the constraints specified by the passengers (Formulas (9), (11)–(12), (28)–(32)).

No constraint in the optimization model is imposed to satisfying Condition (4) , and thus we need to use the following method to prove the validity of Condition (4). We prove it for scheduled and on-demand passengers, separately.

For scheduled service, we discuss two cases based on a passengers' choice to accept the offer or not given the intermediate price.

(1) If the passenger(s) does not accept the choice, then his value and the price are both zero:

$$VA_g = ps_g = 0$$

(2) If the passenger(s) accepts the offer, the passenger(s) will be served, and his value is no less than the intermediate price:

$$VA_g \geqslant ip_g\left(rt_g^-\right)$$

Based on Formula (44), we have

$$ps_g = ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs\left(XS_0^*\right)$$

XS_0* is the best solution in the pool $\boldsymbol{XS}_0$. Based on Algorithm 5, the main algorithm of SPACL to obtain the mechanism for scheduled service, we have XS_g*$\in \boldsymbol{XS}_g \subseteq \boldsymbol{XS}_0$. Thus we have

$$fs\left(XS_g^*\right) \geqslant fs\left(XS_0^*\right)$$

$$ps_g \leqslant ip_g\left(rt_g^-\right)$$

Since $VA_g \geqslant ip_g\left(rt_g^-\right)$ in this case, we have

$$ps_g \leqslant ip_g\left(rt_g^-\right) \leqslant VA_g$$

For on-demand service, based on Formula (45), we have

$$VA_g\left(Xts_0^*\right) - po_g = VA_g\left(Xts_0^*\right) - fo\left(Xts_g^*\right) + fo\left(Xts_0^*\right) - VA_g\left(Xts_0^*\right)$$

$$= fo\left(Xts_0^*\right) - fo\left(Xts_g^*\right)$$

Xts_0* is the best solution in the pool $\boldsymbol{Xts}_0$, and Xts_g* is included in the pool $\boldsymbol{Xts}_0$ based on Algorithm 6. Thus, we have

$$VA_g\left(Xts_0^*\right) - po_g = fo\left(Xts_0^*\right) - fo\left(Xts_g^*\right) \geqslant 0$$

We proved that all passengers', both scheduled and on-demand, mobility preferences are always satisfied under the mechanism obtained by SPACL. Thus, SPACL can hold "preference-based individual rationality". □

Proposition 10. *The scheduled service is financially sustainable if each passenger's intermediate price is no less than* $2(\overline{dc}_g+ch_g)$ *under the mechanism obtained by the SPACL algorithm. If* $ip_g\left(rt_g^-\right) \geqslant 2\left(\overline{dc}_g + ch_g\right)$, *then*

$$\sum_{g\in PS} ps_g \geqslant fs\left(XS_0^*\right)$$

where ps_g *is obtained by Formula (44):*$ps_g = ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs\left(XS_0^*\right)$.

Proof. Let YS_g* be the gth transition solution of XS_0^* : $YS_g^* = TS_g\left(XS_0^*\right)$. Based on the SPACL algorithm, YS_g* is always in the pool $\boldsymbol{XS}_g$. XS_g* is the best solution in $\boldsymbol{XS}_g$. Thus, we have

$$fs\left(YS_g^*\right) \geqslant fs\left(XS_g^*\right)$$

Based on Proposition 4, we have

$$fs\left(XS_0^*\right) \geqslant fs\left(YS_g^*\right) - \overline{dc}_g - ch_g$$

Therefore,

$$fs\left(XS_0^*\right) \geqslant fs\left(XS_g^*\right) - \overline{dc}_g - ch_g$$

Based on Formula (44), we have

$$ps_g = ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs\left(XS_0^*\right)$$

$$\geqslant ip_g\left(rt_g^-\right) - fs\left(XS_g^*\right) + fs\left(XS_g^*\right) - \overline{dc}_g - ch_g$$

$$= ip_g\left(rt_g^-\right) - \overline{dc}_g - ch_g$$

Given the condition $ip_g\left(rt_g^-\right) \geqslant 2\left(\overline{dc}_g + ch_g\right)$, we have

$$ps_g \geqslant ip_g\left(rt_g^-\right) - \overline{dc}_g - ch_g \geqslant \ + ch_g$$

In the proof of Proposition 5, we already proved that

$$fs(X_0) \leqslant \sum_{g\in PS}\left(\overline{dc}_g + ch_g\right)$$

where X_0 is a solution in which all passengers do not share the trip with others. Based on Algorithm 3, X_0's solution quality is not higher than any solution in the pool $\boldsymbol{XS}_0$ while XS_0* is the best solution in $\boldsymbol{XS}_0$. Then we have

$$fs\left(X_{MS_0}^*\right) \leqslant fs(X_0) \leqslant \sum_{g\in PS}\left(\overline{dc}_g + ch_g\right)$$

Again, due to $ps_g \geqslant \overline{dc}_g + ch_g$, we have

$$fs\left(X_{MS_0}^*\right) \leqslant fs(X_0) \leqslant \sum_{g\in PS}\left(\overline{dc}_g + ch_g\right) \leqslant \sum_{g\in PS}\left(ps_g\right)$$

indicating that the scheduled service is financially sustainable under the mechanism obtained by the SPACL algorithm if the condition $ip_g(rt_g^{-}) \geq 2(\overline{dc}_g + ch_g)$ is satisfied. □

Proposition 11. *The mechanism obtained by SPACL is scheduling preferable*

Proof. We prove the inequations in Table 3.

Proof of inequation (a):

Based on Proposition 9, we have

$$Us_g = VA_g - ps_g \geqslant 0 \quad for \quad any \ g \in PS$$

Proof of inequation (b):

If the on-demand passenger request is rejected in Xts_0* obtained by SPACL, then $VA_g(Xts_0^*) = 0$. Thus, based on Formula (45), we have

$$po_g = fo\left(Xts_g^*\right) - fo\left(Xts_0^*\right) + VA_g\left(Xts_0^*\right) = fo\left(Xts_g^*\right) - fo\left(Xts_0^*\right)$$

Since passenger(s) g is not served in Xts_0*, based on Definition 5 (transition solution from model Mts_0 to model Mts_g), we have

$$Xts_0^* = Yts_g^* = TOg\left(Xts_0^*\right)$$

Based on the SPACL algorithm, $Yts_g^* \in \boldsymbol{Xts_g}$. Xts_g* is the best solution in $\boldsymbol{Xts_g}$. Thus, we have

$$fo\left(Yts_g^*\right) \leqslant fo\left(Xts_g^*\right)$$

Moreover, $Xts_g^* \in \boldsymbol{Xts_g} \subseteq \boldsymbol{Xts_0}$ based on the SPACL algorithm, and Xts_0* is the best solution in $\boldsymbol{Xts_0}$. Thus, we have

$$fo\left(Yts_g^*\right) \leqslant fo\left(Xts_g^*\right) \leqslant fo\left(Xts_0^*\right)$$

Since $Xts_0^* = Yts_g^*$, we have

$$fo\left(Yts_g^*\right) = fo\left(Xts_g^*\right) = fo\left(Xts_0^*\right)$$

Thus, we have

$$po_g = fo\left(Xts_g^*\right) - fo\left(Xts_0^*\right) = 0$$

Thus, the utility is

$$Uo_g = VA_g\left(Xts_0^*\right) - po_g = 0 \leqslant Us_g$$

Proof of inequation (c):

As we discussed above, the algorithm SPACL ensures $fo(Yts_g^*) \leq fo(Xts_g^*)$. Thus, based on Formula (45), the price of an on-demand passenger request

$$po_g = fo\left(Xts_g^*\right) - fo\left(Xts_0^*\right) + VA_g\left(Xts_0^*\right)$$

$$\geqslant fo\left(Yts_g^*\right) - fo\left(Xts_0^*\right) + VA_g\left(Xts_0^*\right)$$

in which

$$fo\left(Yts_g^*\right) = \sum_{i \in P_{ts}} VA_i\left(Yts_g^*\right) - TC\left(Yts_g^*\right) + \sum_{i \in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k \in V_{ts}} w_{ki}\right)$$

$$fo\left(Xts_0^*\right) = \sum_{i \in P_{ts}} VA_i\left(Xts_0^*\right) - TC\left(Xts_0^*\right) + \sum_{i \in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k \in V_{ts}} w_{ki}\right)$$

Based on the definition of the transition solution (Definition 5) in on-demand service, except the difference, $\sum_{k \in V_{ts}} w_{kg} = 0$ in $fo(Yts_g^*)$ and $\sum_{k \in V_{ts}} w_{kg} = 1$ in $fo(Xts_0^*)$ (because Passenger(s) g is not served in Yts_g* but is served in Xts_0*), all of other elements in $\sum_{i \in P_{ts}} ip_i\left(rt_i^+\right)\left(1 - \sum_{k \in V_{ts}} w_{ki}\right)$ are identical. Then we have

$$po_g \geqslant fo\left(Yts_g^*\right) - fo\left(Xts_0^*\right) + VA_g\left(Xts_0^*\right)$$

$$= ip_g\left(rt_g^+\right) + \left(\sum_{i \in P_{ts}} VA_i\left(Yts_g^*\right) - \sum_{i \in P_{ts} \setminus g} VA_i\left(Xts_0^*\right)\right) + \left(TC\left(Xts_0^*\right) - TC\left(Xts_0^*\right)\right)$$

From Propositions 6 and 7, respectively, we have

$$TC\left(Xts_0^*\right) - TC\left(Yts_g^*\right) \geqslant 0$$

and

$$\sum_{i\in P_{ts}} VA_i\left(Yts_g^*\right) - \sum_{i\in P_{ts}\backslash g} VA_i\left(Xts_0^*\right) = 0$$

Thus, we have

$$po_g \geqslant ip_g\left(rt_g^+\right)$$

In Proposition 9, we proved that $ps_g \leqslant ip_g\left(rt_g^-\right)$ if ps_g is obtained by Formula (44). In addition, based on Formula (1), $ip_g\left(rt_g^+\right) \geqslant ip_g\left(rt_g^-\right)$. Thus, we have

$$po_g \geqslant ip_g\left(rt_g^+\right) \geqslant ip_g\left(rt_g^-\right) \geqslant ps_g$$

In this case the passenger(s) is served in both of the scheduled and on-demand service. Then $VA_g(Xts_0^*) = \alpha_i^p = VA_g(XS_0^*)$. Thus, we have

$$Uo_g = VA_g\left(Xts_0^*\right) - po_g = \alpha_i^p - po_g \leqslant \alpha_i^p - ps_g = VA_g\left(XS_0^*\right) - ps_g = Us_g \quad \square$$

Appendix C. Pseudocode of algorithms

Algorithm 1. (*Obtain the transition solutions $Y_g = TS_g(X)$ from model MS_0 to model MS_g*)

```
Input a solution X = {xs_ijk, ys_ki, zs_ki, ws_ki};
Duplicate X to Y_g: Y_g = X, and then made the following modifications in Y_g;
Find k that ws_kg = 1;
If ∑_{i∈PS\g} ws_ki np_i > 0
  If ys_kg = 0 or zs_kg = 0
    If ys_kg = 1
      Identify j that xs_gjk = 1;
      Set xs_gjk = 0;
      Set ys_kj = 1;
    Else
    If zs_kg = 1
      Identify i that xs_igk = 1;
      Set xs_igk = 0;
      Set zs_ki = 1;
    Else
      Identify i that xs_igk = 1;
      Set xs_igk = 0;
      Identify j that xs_gjk = 1;
      Set xs_gjk = 0;
      Set xs_ijk = 1;
    End if
  End if
  Set ws_kg = 0, ys_kg = 0, zs_kg = 0;
  k' = argmin_{l∈VS\k}(dc_lg);
  Let Vehicle k' to serve Passenger(s) g: ws_k'g = 1, ys_k'g = 1, zs_k'g = 1;
End if
Output Y_g.
```

Algorithm 2. (*Obtain the transition solutions $Y_g = TO_g(X)$ from model Mts_0 to model Mts_g*)

```
Input a solution X = {x_ijk, y_ki, z_ki, w_ki};
Duplicate X to Y_g: Y_g = X, and then made the following modifications in Y_g;
If ∑_{k∈V_ts} w_kg = 1
  Identify k that w_kg = 1;
  If ∑_{i∈P_ts\g} w_ki np_i > 0
```

(continued on next page)

(*continued*)

```
If y_kg = 0 or z_kg = 0
  If y_kg = 1
    Identify j that x_gjk = 1;
    Set x_gjk = 0;
    Set y_kj = 1;
  Else
  If z_kg = 1
    Identify i that x_igk = 1;
    Set x_igk = 0;
    Set z_ki = 1;
  Else
    Identify i that x_igk = 1;
    Set x_igk = 0;
    Identify j that x_gjk = 1;
    Set x_gjk = 0;
    Set x_ijk = 1;
  End if
 End if
End if
Set w_kg = 0, y_kg = 0, z_kg = 0;
End if
Output Y_g.
```

Algorithm 3. (*Generate an initial solution pool, **XS**, for model MS_0*)

```
Set the total number of iterations (NI), number of neighborhood solutions (CN) for each iteration,
    number of solutions in the pool XS (NS);
Set ni = 0; % The index of the current iteration;
Set X_current = X_0; % Record an initial solution X_0 as the current solution. In X_0, all passengers do not
    share the trip with others;
Set f_current = fs(X_current) ; % Record the objective function value of X_current (Formula (3));
Set X_best = X_0; % Record the best solution found;
Set f_best = fs(X_best) ; % Record the objective function value of X_best;
Set an empty tabu list: TL = ∅;
Set XS = {X_0, X_0, … , X_0}; % Initialize the solution pool XS with NS identical solutions (X_0);
Do while ni < NI
  ni = ni + 1;
  Generate CN X_current's neighborhood solutions X = {X_1, X_2,…, X_CN};
  {X_(1), X_(2),…, X_(CN)} = sort(X, 'ascend');
  If fs(X_(1)) < f_best
    X_best = X_(1);
    f_best = fs(X_(1));
  End if
  k = 1;
  Do while X_(k) ∈ TL
    k = k + 1;
  End do
  X_current = X_(k);
  Put X_(k) into the tabu list TL;
  j = 1;
  Do while fs(X_(j)) < max {fs(XS_i), XS_i ∈ XS}
    XS_max = argmax {fs(XS_i), XS_i ∈ XS};
    XS_max = X_(j); % Replace the worst solution in XS with X_(j)
    j = j + 1;
  End do
End do
Output XS;
```

Algorithm 4. (*Generate an initial solution pool, **Xts**, for model Mts_0*)

```
Set the total number of iterations (NI), number of neighborhood solutions (CN) for each iteration,
    number of solutions in the pool Xts (NS);
Set ni = 0; % The index of the current iteration;
Set X_current = X_initial;
Set f_current = fo(X_current); % Record the objective function value of X_current (Formula (41));
Set X_best = X_initial; % Record the best solution found;
Set f_best = fo(X_best); % Record the objective function value of X_best;
Set the tabu list as empty: TL = ∅;
```

(*continued on next page*)

(*continued*)

Set $\boldsymbol{Xts} = \{X_{initial}, X_{initial}, \dots, X_{initial}\}$ % Initialize the solution pool $\boldsymbol{Xts}$ with NS identical solutions ($X_{initial}$);
Do while $ni < NI$
  $ni = ni + 1$;
  Generate CN $X_{current}$'s neighborhood solutions $\mathbf{X} = \{X_1, X_2, \dots, X_{CN}\}$;
  $\{X_{(1)}, X_{(2)}, \dots, X_{(CN)}\} = \text{sort}(\mathbf{X}, \text{'descend'})$;
  If $fo(X_{(1)}) > f_{best}$
    $X_{best} = X_{(1)}$;
    $f_{best} = fo(X_{(1)})$;
  End if
  $k = 1$;
  Do while $X_{(k)} \in TL$
    $k = k + 1$;
  End do
  $X_{current} = X_{(k)}$;
  Put $X_{(k)}$ into the tabu list TL;
  $j = 1$;
  Do while $fo(X_{(j)}) > \min \{fo(Xts_i), Xts_i \in \boldsymbol{Xts}\}$
    $Xts_{\min} = \text{argmin} \{fo(Xts_i), Xts_i \in \boldsymbol{Xts}\}$;
    $Xts_{\min} = X_{(j)}$; % Replace the worst solution in $\boldsymbol{Xts}$ with $X_{(j)}$
    $j = j + 1$;
  End do
End do
Output $\boldsymbol{Xts}$;

Algorithm 5. (*Main algorithm of SPACL to obtain the mechanism for scheduled service*)

Input $\boldsymbol{XS}$ obtained by Algorithm 3;
For $g \in PS$
  $Y_{gi} = TS_g(XS_i)$, for all $XS_i \in \boldsymbol{XS}$; %Algorithm 1
  $\boldsymbol{XS}_g = \{Y_{gi}, \text{ for all } i\}$;
End for
  $\boldsymbol{XS}_0 = \{\boldsymbol{XS}, \boldsymbol{XS}_g \text{ (for all } g \in PS)\}$; % Combine all of these solution pools into $\boldsymbol{XS}_0$
  $XS_0^* = \text{argmin } fs(XS_{0i})$, for all $XS_{0i} \in \boldsymbol{XS}_0$;
Do while $XS_0^* \notin \boldsymbol{XS}$
  $\boldsymbol{XS} = \{\boldsymbol{XS}, XS_0^*\}$; % Duplicate the best solution XS_0^* to $\boldsymbol{XS}$
  For $g \in PS$
    $\boldsymbol{XS}_g = \{\boldsymbol{XS}_g, TS_g(XS_0^*)\}$; % Put the transition solution $TS_g(XS_0^*)$ into $\boldsymbol{XS}_g$
  End for
  $XS_0^* = \text{argmin } fs(XS_{0i})$, for all $XS_{0i} \in \boldsymbol{XS}_0$;
End do
For $g \in PS$
  $XS_g^* = \text{argmin } fs(XS_{gi})$, for all $XS_{gi} \in \boldsymbol{XS}_g$;
  $ps_g = ip_g(rt_g^-) - fs(XS_g^*) + fs(XS_0^*)$;
End for
Output the best solution XS_0^* and prices $\boldsymbol{ps} = \{ps_1, ps_2, \dots, ps_{ns}\}$;

Table 15
Reduced price amount and discount of the maximum price for the scheduled service.

Simulation examples	Average reduced amount ($)	Average discount
S15_O90	1.93	64.1%
S20_O120	1.50	69.4%
S25_O150	1.86	62.7%
S30_O180	2.01	60.8%
S35_O210	1.87	62.1%
S40_O240	1.81	62.7%
S45_O270	1.85	62.5%
S50_O300	1.96	62.8%
S55_O330	1.77	63.0%
S60_O360	1.78	64.2%
S100_O600	1.87	63.2%
S110_O660	1.87	63.5%
S120_O720	1.72	65.2%
S130_O780	1.74	65.2%
S140_O840	1.82	63.9%
S150_O900	1.71	65.4%
S160_O960	1.80	63.9%
S170_O1020	1.69	65.0%
S180_O1080	1.79	64.5%
S190_O1140	1.74	64.9%
S200_O1200	1.90	63.4%

Algorithm 6. (*Main algorithm of SPACL to obtain the mechanism for on-demand service*)

Input $\boldsymbol{Xts}$ obtained by Algorithm 4;
For $g \in PO_{ts}$
  $Y_{gi} = TO_g(Xts_i)$, for all $Xts_i \in \boldsymbol{Xts}$; %Algorithm 2
  $\boldsymbol{Xts}_g = \{Y_{gi}$, for all $i\}$;
End for
$\boldsymbol{Xts}_0 = \{\boldsymbol{Xts}, \boldsymbol{Xts}_g$ (for all $g \in PO_{ts}$)};% Combine all of these solution pools into $\boldsymbol{Xts}_0$
$Xts_0{}^* =$ argmax $fo(Xts_{0i})$, for all $Xts_{0i} \in \boldsymbol{Xts}_0$;
Do while $Xts_0{}^* \notin \boldsymbol{Xts}$
  $\boldsymbol{Xts} = \{\boldsymbol{Xts}, Xts_0{}^*\}$; % Duplicate the best solution $Xts_0{}^*$ to $\boldsymbol{Xts}$
  For $g \in PO_{ts}$
    $\boldsymbol{Xts}_g = \{\boldsymbol{Xts}_g, TO_g(Xts_0{}^*)\}$; % Put the transition solution $TO_g(Xts_0{}^*)$ in $\boldsymbol{Xts}_g$
  End for
  $Xts_0{}^* =$ argmax $fo(Xts_{0i})$, for all $Xts_{0i} \in \boldsymbol{Xts}_0$;
End do
For $g \in PO_{ts}$
  $Xts_g{}^* =$ argmax $fo(Xts_{gi})$, for all $Xts_{gi} \in \boldsymbol{Xts}_g$;
  $po_g = fo(Xts_g{}^*) - fo(Xts_0{}^*) + VA_g(Xts_0{}^*)$;
End for
Output the best solution $Xts_0{}^*$ and prices $\boldsymbol{po} = \{po_g$, for all $g \in PO_{ts}\}$;

Table 16a
Value (maximum WTP price) assuming that the truthful maximum WTP price of a passenger is \$3.00 and the tolerable number of shared riders is 1.

α_i^P (dollars)	Value (maximum WTP price) (dollars)			
	$\alpha_i^{NR} = 0$ (misreport)	$\boldsymbol{\alpha_i^{NR} = 1}$ **(truthful report)**	$\alpha_i^{NR} = 2$ (misreport)	$\alpha_i^{NR} = 3$ (misreport)
2.3 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.4 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.5 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
2.6 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
2.7 (misreport)	0.00 (not served)	3.00 (number of shared riders = 1)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
2.8 (misreport)	0.00 (not served)	3.00 (number of shared riders = 1)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
2.9 (misreport)	0.00 (not served)	3.00 (number of shared riders = 1)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
3.0 (truthful report)	3.00 (number of shared riders = 0)	**3.00 (number of shared riders = 1)**	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
3.1 (misreport)	3.00 (number of shared riders = 0)	3.00 (number of shared riders = 1)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)
3.2 (misreport)	3.00 (number of shared riders = 0)	3.00 (number of shared riders = 1)	0.00 (number of shared riders = 2)	0.00 (number of shared riders = 2)

α_i^{NR}: the passenger's reported maximum tolerable number of shared riders.
α_i^P: the passenger's reported maximum WTP price if served.

Table 16b
Paid price.

α_i^P (dollars)	Actual paid price (dollars)			
	$\alpha_i^{NR} = 0$ (misreport)	$\boldsymbol{\alpha_i^{NR} = 1}$ **(truthful report)**	$\alpha_i^{NR} = 2$ (misreport)	$\alpha_i^{NR} = 3$ (misreport)
2.3 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.4 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.5 (misreport)	0.00 (not served)	0.00 (not served)	2.46 (served)	2.46 (served)
2.6 (misreport)	0.00 (not served)	0.00 (not served)	2.46 (served)	2.46 (served)
2.7 (misreport)	0.00 (not served)	2.65 (served)	2.46 (served)	2.46 (served)
2.8 (misreport)	0.00 (not served)	2.65 (served)	2.46 (served)	2.46 (served)
2.9 (misreport)	0.00 (not served)	2.65 (served)	2.46 (served)	2.46 (served)
3.0 (truthful report)	2.95 (served)	**2.65 (served)**	2.46 (served)	2.46 (served)
3.1 (misreport)	2.95 (served)	2.65 (served)	2.46 (served)	2.46 (served)
3.2 (misreport)	2.95 (served)	2.65 (served)	2.46 (served)	2.46 (served)

α_i^{NR}: the passenger's reported maximum tolerable number of shared riders.
α_i^P: the passenger's reported maximum WTP price if served.

Table 16c
Utility (value – paid price).

α_i^P (dollars)	Utility (value – paid price) (dollars)			
	$\alpha_i^{NR} = 0$ (misreport)	$\boldsymbol{\alpha_i^{NR} = 1}$ **(truthful report)**	$\alpha_i^{NR} = 2$ (misreport)	$\alpha_i^{NR} = 3$ (misreport)
2.3 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.4 (misreport)	0.00 (not served)	0.00 (not served)	0.00 (not served)	0.00 (not served)
2.5 (misreport)	0.00 (not served)	0.00 (not served)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
2.6 (misreport)	0.00 (not served)	0.00 (not served)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
2.7 (misreport)	0.00 (not served)	0.35 (mobility preferences satisfied)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
2.8 (misreport)	0.00 (not served)	0.35 (mobility preferences satisfied)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
2.9 (misreport)	0.00 (not served)	0.35 (mobility preferences satisfied)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
3.0 (truthful report)	0.05 (mobility preferences satisfied)	**0.35 (mobility preferences satisfied)**	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
3.1 (misreport)	0.05 (mobility preferences satisfied)	0.35 (mobility preferences satisfied)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)
3.2 (misreport)	0.05 (mobility preferences satisfied)	0.35 (mobility preferences satisfied)	−2.46 (mobility preferences not satisfied)	−2.46 (mobility preferences not satisfied)

α_i^{NR}: the passenger's reported maximum tolerable number of shared riders
α_i^P: the passenger's reported maximum WTP price if served

Appendix D. Scheduled service discount

See Table 15.

Appendix E. Preference-based incentive compatibility demonstration

How can an exact algorithm hold "preference-based incentive compatibility"?

Table 17a
Results of SPACL upon misreporting maximum WTP price.

Reported maximum WTP price		Request is rejected	The passenger is served		
			Utility decreases	Utility does not change	Utility increases
Misreport to $8	Frequency out of 100 simulations	76	20	4	0
	Average utility change (dollars)	−9	0.204	0	0
Misreport to $10	Frequency out of 100 simulations	0	89	10	1
	Average utility change (dollars)	−9	−0.264	0	0.068

Table 17b
Results of SPACL upon misreporting maximum tolerable shared riders.

Reported maximum tolerable shared riders		Request is rejected	The passenger is served			
			Mobility preference is not satisfied	Mobility preference is satisfied		
				Utility decreases	Utility does not change	Utility increases
Misreport to 0	Frequency out of 100 simulations	100	0	0	0	0
	Average utility change (dollars)	−9	0	0	0	0
Misreport to 2	Frequency out of 100 simulations	0	91	9	0	0
	Average utility change (dollars)	0	−5.939	−0.172	0	0

Table 17c
Utility change due to misreporting (negative number means utility decrease and positive number means utility increase).

Misreporting strategies		Average utility change for 100 simulations (dollars)
Maximum WTP price	Misreport to \$8	−6.80
	Misreport to \$10	−0.23
Maximum tolerable shared riders	Misreport to 0	−9.00
	Misreport to 2	−5.42

We select a passenger in the numerical example to demonstrate that misreporting certain preferences will not increase the passengers' utilities if the mechanism is obtained by an exact algorithm. We assume that this passenger may misreport maximum tolerable number of shared riders (α_i^{NR}) and/or maximum WTP price (α_i^{P}). His truthful maximum tolerable number of shared riders is 1 and truthful maximum WTP price is \$3.00. The following three tables present the mechanism results based on his reported α_i^{NR} and α_i^{P}.

Table 16a is the passenger's value, which is the maximum WTP price depending on whether he is served and whether the mobility preference is satisfied. If the passenger is severed and the mobility preference is satisfied (here, the maximum tolerable number of shared riders is not exceeded), his maximum WTP price is \$3.00; otherwise, his maximum WTP price is \$0.00.

Table 16b is the passenger's actual paid price. If the passenger is served, he will be charged with a positive price; otherwise, the price is naturally \$0.00.

Table 16c presents the passenger's utility given different combinations of reported α_i^{NR} and α_i^{P}. We can observe that truthful reporting can achieve the largest utility \$0.35, and misreporting will never increase the utility. If the passenger misreports a lower maximum WTP price (α_i^{P}), the passenger may be rejected to take the service, while if he is still served, the price will never be reduced. If the passenger misreports that he does not want to share the trip with any other rider, the paid price increases from \$2.65 to \$2.95, and thus this utility decreases from \$0.35 to \$0.05. If the passenger misreports that he could share the ride with 2 or 3 passengers, then he will share the trip with 2 passengers but his actual tolerable number of shared riders is "1". Thus, the mobility preference is not satisfied and his maximum WTP price is deemed as \$0.00. However, he still needs to pay a positive price \$2.46, and thus his utility becomes negative. Misreporting other mobility preferences, such as the latest arrival time and maximum tolerable detour time have the same results. Based on the discussion above, it is demonstrated that falsifying certain preference will not increase passengers' utilities.

How much does SPACL sacrifice the property "preference-based incentive compatibility"?

We also select a passenger in a numerical example (S100_O600) to quantify how much SPACL could sacrifice the property "preference-based incentive compatibility" and how SPACL prevents passengers from manipulating the mechanism. In this example, the passenger' true maximum WTP price is \$9.00 and true maximum tolerable number of shared riders is 1. We then assume that this passenger misreports the maximum WTP price to \$8 and \$10 or misreports the maximum tolerable number of shared riders to 0 and 2. SPACL is used to obtain the mechanism results for 100 replicated simulations. We record the frequency when the utility increases due to misreport. The results are presented in Tables 17a and 17b.

If the passenger misreports the maximum WTP price to \$10, the passenger gains larger utility for only once out of the 100 simulations. Also, if the passenger misreports the maximum WTP price to \$8 or misreports the maximum tolerable number of shared riders to 0 and 2, we do not observe any times when the passenger's utility increases. Table 17c presents the average change of the utility for the 100 simulation replications when the passenger adopts the four misreporting strategies. We can observe that the four misreporting strategies all lead to utility decrease compared with the strategy of truthful reporting. This indicates that even if SPACL does not hold "preference-based incentive compatibility", passengers cannot anticipate the results upon misreporting the mobility preferences and thus cannot learn to manipulate the mechanism.

References

Agatz, N., et al., 2010. The Value of Optimization in Dynamic Ride-Sharing: a Simulation Study in Metro Atlanta, Research paper, Erasmus Research Institute of Management (ERIM), Report No. ERS-2010-034-LIS. Retrieved from: http://hdl.handle.net/1765/20456.

Agatz, N., Erera, A.L., Savelsbergh, M.W.P., Wang, X., 2011. Dynamic ride-sharing: A simulation study in metro Atlanta. Proc. -Soc. Behav. Sci. 17, 532–550.

Asghari, M., et al., 2016. Price-aware real-time ride-sharing at scale: an auction-based approach. Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM.

Asghari, M., Shahabi, C., 2017. An on-line truthful and individually rational pricing mechanism for ride-sharing. Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM.

BBC news, 2016. http://www.bbc.co.uk/newsbeat/article/37219411/uber-detours-and-how-to-fight-them.

Bian, Z., Liu, X., 2017. Planning the ridesharing route for the first-mile service linking to railway passenger transportation. 2017 Joint Rail Conference. American Society of Mechanical Engineers Digital Collection.

Bian, Z., Xiang, L., 2018. A detour-based pricing mechanism for first-mile ridesharing in connection with rail public transit. ASME/IEEE Joint Rail Conference vol. 50978.

Bian, Z., Liu, X., 2019a. Mechanism design for first-mile ridesharing based on personalized requirements part I: Theoretical analysis in generalized scenarios. Transp. Res. Part B: Methodol. 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. Part B: Methodol. 120, 172–192.

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

Chau, S.-K., Shen, S., Zhou, Y., 2020. Decentralized Ride-Sharing and Vehicle-Pooling Based on Fair Cost-Sharing Mechanisms. IEEE Trans. Intell. Transp. Syst.

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.

Cheng, S.-F., Nguyen, D.T., Lau, H.C., 2014. Mechanisms for arranging ride sharing and fare splitting for last-mile travel demands. Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems. International Foundation for Autonomous Agents and Multiagent Systems.
Clarke, E.H., 1971. Multipart pricing of public goods. Public Choice 11 (1), 17–33.
Daganzo, C.F., Ouyang, Y., 2019. A general model of demand-responsive transportation services: From taxi to ridesharing to dial-a-ride. Transp. Res. Part B: Methodol. 126, 213–224.
Daganzo, C.F., Ouyang, Y., Yang, H., 2020. Analysis of ride-sharing with service time and detour guarantees. Transp. Res. Part B: Methodol. 140, 130–150.
Fielbaum, A., Alonso-Mora, J., 2020. Unreliability in ridesharing systems: Measuring changes in users' times due to new requests. Transp. Res. Part C: Emerg. Technol. 121, 102831. https://doi.org/10.1016/j.trc.2020.102831.
Groves, T., 1973. Incentives in teams. Econometr. J. Econometr. Soc. 41 (4), 617. https://doi.org/10.2307/1914085.
Haferkamp, J., Ehmke, J.F., 2020. An Efficient Insertion Heuristic for On-Demand Ridesharing Services. Transp. Res. Procedia 47, 107–114.
Hsieh, F.-S., Zhan, F.-M., Guo, Y.-H., 2019. A solution methodology for carpooling systems based on double auctions and cooperative coevolutionary particle swarms. Appl. Intell. 49 (2), 741–763.
Hu, Shichun, et al., 2020. Cost-Sharing Mechanism Design for Ride-Sharing. https://cpb-us-e1.wpmucdn.com/sites.usc.edu/dist/0/249/files/2020/10/Ride_Sharing-20201008.pdf.
Hua, S., Qi, M., 2019. An Adaptive Large Neighborhood Search Heuristic for Dynamic Ridesharing Problem. 2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA). IEEE.
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.
Kamar, E., Horvitz, E., 2009. Collaboration and Shared Plans in the Open World: Studies of Ridesharing. IJCAI vol. 9.
Ke, J., Yang, H., Li, X., Wang, H., Ye, J., 2020. Pricing and equilibrium in on-demand ride-pooling markets. Transp. Res. Part B: Methodol. 139, 411–431.
Kleiner, Alexander, Nebel, Bernhard, Ziparo, Vittorio Amos, 2011. A mechanism for dynamic ride sharing based on parallel auctions. Twenty-Second International Joint Conference on Artificial Intelligence.
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. Part C: Emerg. Technol. 122, 102891. https://doi.org/10.1016/j.trc.2020.102891.
Lei, C., Jiang, Z., Ouyang, Y., 2019. Path-based Dynamic Pricing for Vehicle Allocation in Ridesharing Systems with Fully Compliant Drivers. Transp. Res. Procedia 38, 77–97.
Lesh, M.C., 2013. Innovative concepts in first-last mile connections to public transportation. Urban Public Transp. Syst. 2013, 63–74.
Li, Y., Liu, Y., Xie, J., 2020. A path-based equilibrium model for ridesharing matching. Transp. Res. Part B: Methodol. 138, 373–405.
Liu, Y., Li, Y., 2017. Pricing scheme design of ridesharing program in morning commute problem. Transp. Res. Part C: Emerg. Technol. 79, 156–177.
Lloret-Batlle, R., Masoud, N., Nam, D., 2017. P2p Ridesharing with Ride-Back on Hov Lanes: Towards a Practical Alternative Mode for Daily Commuting. No. 17-06253.
Luo, P., 2019. A two-stage approach to ridesharing assignment and auction in a crowdsourcing collaborative transportation platform. Electronic Theses and Dissertations. Paper 3240.
Ma, H., Fang, F., Parkes, D.C., 2018. Spatio-Temporal Pricing for Ridesharing Platforms. arXiv preprint arXiv:1801.04015.
Ma, J., Xu, M., Meng, Q., Cheng, L., 2020. Ridesharing user equilibrium problem under OD-based surge pricing strategy. Transp. Res. Part B: Methodol. 134, 1–24.
Masoud, Neda, Lloret-Batlle, Roger, 2016. Increasing ridership and user permanence in ridesharing systems using a novel peer-to-peer exchange mechanism. 95th Annual Meeting of the Transportation Research Board, Washington, DC. 2016.
Masoud, Neda, Lloret-Batlle, Roger, Jayakrishnan, R., 2017b. Using bilateral trading to increase ridership and user permanence in ridesharing systems. Transportation Research Part E: Logistics and Transportation Review 102, 60–77.
Masoud, N., Nam, D., Yu, J., Jayakrishnan, R., 2017a. Promoting Peer-to-Peer Ridesharing Services as Transit System Feeders. Transp. Res. Record: J. Transp. Res. Board 2650 (1), 74–83.
Matsuo, Tokuro, Ito, Takayuki, Shintani, Toramatsu, 2005. A volume discount-based allocation mechanism in group buying. International Workshop on Data Engineering Issues in E-Commerce. IEEE.
Mu'alem, A., Nisan, N., 2008. Truthful approximation mechanisms for restricted combinatorial auctions. Games Econ. Behav. 64 (2), 612–631.
Nguyen, Duc Thien, 2013. Fair Cost Sharing Auction Mechanisms in Last Mile Ridesharing. https://ink.library.smu.edu.sg/etd_coll/93/.
Nisan, N., Ronen, A., 2007. Computationally Feasible VCG Mechanisms. J. Artif. Intell. Res. (JAIR) 29, 19–47.
O'Toole Randal, 2000. Charting Public Transit's Decline. CATO institute. https://www.cato.org/publications/policy-analysis/charting-public-transits-decline (accessed by 2020).
Parkes, David C., Kalagnanam, Jayant R., Eso, Marta, 2001. Achieving budget-balance with Vickrey-based payment schemes in exchanges.
Perera, T., et al., 2018. Hybrid genetic algorithm for an on-demand first mile transit system using electric vehicles. International Conference on Computational Science. Springer, Cham.
Rückert, Nicolas, Sturm, Daniel, Fischer, Kathrin, 2020. A Heuristic Solution Approach for the Optimization of Dynamic Ridesharing Systems. In: Operations Research Proceedings 2019. Springer, Cham, pp. 733–739.
Santos, D.O., Xavier, E.C., 2015. Taxi and ride sharing: A dynamic dial-a-ride problem with money as an incentive. Expert Syst. Appl. 42 (19), 6728–6737.
Shen, Wen, Lopes, Cristina V., Crandall, Jacob W., 2016. An Online Mechanism for Ridesharing in Autonomous Mobility-on-Demand Systems. arXiv preprint arXiv:1603.02208.
Shen, Y.u., 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.
Shi, Bing, Luo, Yikai, Zhu, Liquan, 2020. Auction-Based Order-Matching Mechanisms to Maximize Social Welfare in Real-Time Ride-Sharing. International Conference on Database Systems for Advanced Applications. Springer, Cham.
Stiglic, M., Agatz, N., Savelsbergh, M., Gradisar, M., 2018. Enhancing urban mobility: Integrating ride-sharing and public transit. Comput. Oper. Res. 90, 12–21.
Taxi calculator, 2018. https://www.taxi-calculator.com/taxi-fare-new%20york%20city/259.
Ungemah, D., Rivers, M., Anderson, S., 2006. The world can be flat: Case study of flat-rate pricing for vanpool operations. Transp. Res. Rec. 1956 (1), 30–33.
Vickrey, W., 1961. Counterspeculation, auctions, and competitive sealed tenders. J. Fin. 16 (1), 8–37.
Wei, Yanhao, Xu, Linli, Zhu, Yi, 2019. Does Surge Pricing Affect Customer Complaint Rate?. Available at SSRN 3437470.
Yan, P., Lee, C.-Y., Chu, C., Chen, C., Luo, Z., 2021. Matching and pricing in ride-sharing: Optimality, stability, and financial sustainability. Omega 102, 102351. https://doi.org/10.1016/j.omega.2020.102351.
Zhang, Chaoli, et al., 2017. Online auctions with dynamic costs for ridesharing. 2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS). IEEE, 2017.
Zhang, Chaoli, Fan, Wu, Bei, Xiaohui, 2018. An Efficient Auction with Variable Reserve Prices for Ridesourcing. Pacific Rim International Conference on Artificial Intelligence. Springer, Cham.
Zhang, C., Xie, J., Wu, F., Gao, X., Chen, G., 2020. Pricing and allocation algorithm designs in dynamic ridesharing system. Theoret. Comput. Sci. 803, 94–104.
Zhao, D., et al., 2014. Incentives in ridesharing with deficit control. Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems. International Foundation for Autonomous Agents and Multiagent Systems.
Zhao, Dengji, Ramchurn, Sarvapali D., Jennings, Nicholas R., 2015. Incentive Design for Ridesharing with Uncertainty. arXiv preprint arXiv:1505.01617.
Zheng, L., Cheng, P., Chen, L., 2019. Auction-based order dispatch and pricing in ridesharing. 2019 IEEE 35th International Conference on Data Engineering (ICDE). IEEE.