sustainability

MDPI

Article

Designing a New Shuttle Service to Meet Large-Scale Instantaneous Peak Demands for Passenger Transportation in a Metropolitan Context: A Green, Low-Cost Mass Transport Option

Han Zheng, Junhua Chen *, Xingchen Zhang and Zixian Yang

School of Traffic and Transportation, Beijing Jiaotong University, No. 3 Shang Yuan Cun, Hai Dian District, Beijing 100044, China; hanzheng@bjtu.edu.cn (H.Z.); xczhang@bjtu.edu.cn (X.Z.); yangzx@bjtu.edu.cn (Z.Y.)
* Correspondence: cjh@bjtu.edu.cn; Tel.: +86-134-2602-5309

Received: 8 August 2019; Accepted: 10 September 2019; Published: 13 September 2019

Abstract: Currently, the green, sustainable development of metropolises is hindered by problems caused by Large-scale Instantaneous Peak-demands for Passenger-transportation (LIPP), such as traffic congestion and air pollution. To mitigate these problems, we propose a new type of demand-responsive service as an alternative to inefficient "door-to-door" service. The proposed service is based on service units designed to aggregate passengers for shuttle service. To guarantee service quality and efficiency, a maximum passenger walking time constraint, a request rejection mechanism and a scheme for ensuring solution feasibility are considered. Through numerical experiments, we prove the following: (i) the proposed transport option exhibits better performance (by 40.37% for passengers and by 35.79% for operators) than the door-to-door transport option for solving real cases. (ii) By testing different datasets, we prove that the proposed service is more suitable for the request distributions that are spatiotemporally concentrated. (iii) Regarding the individual components of the proposed clustering-first, routing-second solution framework, the proposed soft clustering algorithm exhibits better performance than the classical hard clustering method (by 8%), and the proposed routing algorithm is 1.5 times more efficient than the commercial solution software GAMS.

Keywords: column generation; demand-responsive shuttle service; fuzzy c-means clustering; pickup and delivery problem with time windows; metropolitan passenger transportation

1. Introduction

1.1. Background

As very common phenomena in the development of metropolises, Large-scale Instantaneous Peak-demands for Passenger-transportation (LIPP) have led to a number of problems related to mobility management and environmental concerns, such as traffic congestion, air pollution, and traffic accidents [1–3]. These phenomena are of great concern to the managers of metropolises because of the characteristics of the corresponding demands. These characteristics are markedly different from those of the transportation demands in general cities in the following ways.

(i) The peak demands in metropolises are unpredictable and instantaneous. Multiple sources (e.g., commuters and tourism traffic) and many factors (e.g., weather [4], festivals and special events) can cause demands to fluctuate irregularly and without warning.

(ii) The high population densities and clear boundaries of urban functional areas result in peak demands with certain spatiotemporal similarities in terms of origins and destinations (e.g., demands of commuters who live in the same residential areas and work in the same business districts).

This study is motivated by the desire to seek a new type of demand-responsive service, called large-scale instantaneous-peak-demand-responsive express shuttle service (LIPDRS), that can accommodate the LIPP of a metropolis, while balancing the benefits to operators and passengers. As an alternative to "door-to-door" transportation or frequency services, LIPDRS requires passengers to gather at certain locations, from which they are served by shuttles. Therefore, the LIDPRS design process involves first assigning passengers to certain service units and then transporting them along certain routes. LIPDRS has the advantages of a large capacity (ability to accommodate occasional massive demands), high flexibility (planning based on dynamic phone/Internet requests), low cost (affordability for passengers) and an acceptable level of convenience (passengers are required to gather at certain locations to be served in accordance with their requests). With these advantages, LIPDRS is an innovative means of increasing accessibility between high-circulation areas in a metropolis, such as residential districts and central business districts. As a complement to passenger transit systems, LIPDRS can help solve mobility and accessibility problems in a metropolitan context. In addition, this new transport option can encourage people to reduce their proportions of travel by private cars and thus reduce the cost of and pollution due to urban transportation, thereby improving the sustainability of metropolitan development.

However, the realization of LIPDRS comes at the cost of additional complexity. Since passengers are required to gather at certain locations for service, certain challenges arise that must be solved to make LIPDRS possible. These challenges are as follows:

(i) For a fully coordinated transportation service, involving large numbers of passengers and vehicles, the resulting optimization problem needs to be addressed using a holistic approach enabling the joint determination of the vehicle service locations, service time windows, routes, and passenger-to-vehicle assignments subject to rather restrictive travel demand requests. [5]

(ii) The service must strike a suitable balance between operator profitability and passenger convenience/accessibility. The balance among efficiency, effectiveness and equity of service is a key concern in the LIPDRS design process. It is clear that a demand-responsive transportation service does not need to serve all received requests if doing so will result in inefficiency. However, the problem of how to design an appropriate rejection strategy then arises, and the criteria for rejection are difficult to determine. [6]

(iii) Unlike a general demand-responsive bus service, LIPDRS abandons the door-to-door service model. Two additional major challenges that consequently arise are whether the requesters can accept this change and how to describe the threshold for passenger acceptance of LIPDRS from a general cost perspective.

(iv) The relevant control parameters, such as the optimal number of service locations, are difficult to determine [7].

(v) It must be possible to solve real-scale problems using the developed approach.

In this study, we build a mixed-integer programming model with dual objectives to optimize the service from the perspectives of both passengers and operators, following the principle of clustering first, routing second, to solve the LIPDRS design problem. The contributions of this study are four-fold:

(i) We propose a new transport option along with a complete system of modeling and solution methods. A maximum passenger walking time constraint, a request rejection mechanism and a scheme for ensuring solution feasibility are all considered in this scheduling system.

(ii) The proposed transport option is proven to achieve better performance (by 40.37% for passengers and by 35.79% for operators) than the door-to-door transport option when faced with real-world demands.

(iii) By testing on different datasets, we prove that the proposed service is more suitable for the request distributions that are spatiotemporally concentrated. We find that when the ratio of requests that are spatiotemporally concentrated is greater than 50%, LIPDRS is the more suitable option.

(iv) The performance advantages of the proposed clustering-first, routing-second solution framework are proven through experiments. Regarding the individual components of this framework,

the proposed soft clustering algorithm is proven to be superior to the k-means hard clustering method (by 8%), and the proposed routing algorithm is proven to be 1.5 times more efficient than the commercial solution software GAMS.

1.2. Literature Review

Demand-responsive services are an important means of coping with special passenger demands. Existing passenger transportation systems include demand-responsive services such as customized buses and dial-a-ride services [8]. However, among the existing public transport options, neither door-to-door services (e.g., dial-a-ride services or carpooling) nor frequency services (e.g., customized buses) can accommodate LIPP; the existing methods face problems of capacity (insufficient vehicles), efficiency (too many boarding and alighting actions [9]) and quality (e.g., overcrowding and excessive waiting intervals [10]). We present a detailed comparison of LIPDRS with other urban public transport options in Table 1. By listing the main features (including schedule frequency, target demand, decision basis, request method, service locations, price and convenience) of each transport options, we find that none of the classical services can accommodate the characteristic of LIPP (the peak demands in metropolises are unpredictable and instantaneous). Taxis do not offer prescheduling before operating, and their decisions are based on the driver, making taxis limited and uncertain. Additionally, taxis do not offer advantages in terms of price. Thus, they can meet occasional individual demands but not LIPP. The other options with detailed scheduling also cannot accommodate the LIPP. School buses need to guarantee the safety of students, so their decisions are based on registered information from schools, and they travel between homes and schools directly along fixed routes. Conventional shuttles focus on serving passengers with related fixed travel demands, and they have poor adaptability to instantaneous peak demands. Although dial-a-ride services and car-sharing services are responsive to flexible demands, their capabilities are limited, and they cannot reduce the number of cars on the road network; thus, they cannot meet the demand for transporting mass passengers. Moreover, the prices of these two options of transportation are relatively high. Comparing these methods reveals that it is necessary to propose new transportation options.

Table 1. Comparison of various urban public transport options.

Name [1]	Schedule Frequency	TD [2]	Decision Basis	Request Method	Service Locations	P [3]	C [4]	R [5]
Taxi	N/A	OID	N/A	Encounters	Uncertain	H	H	[11–13]
DARS	Flexible	OID	OD information, preferred arrival/departure times	Phone/Internet bookings	OD points	H	H	[14,15]
TCS	Fixed frequency	RMD	Historical information	Surveys/consultation	Bus stations	L	L	[16,17]
CB	Flexible frequency	RMD	Estimated OD information	Surveys/historical data	Bus stations	L	M	[1,18–22]
SB	Fixed routes	RMD	Registered information	School records	Home and school	L	H	[9,23]
CS	Flexible	OID	OD information, preferred arrival/departure times	Phone/Internet bookings	Fixed B&A[6]	L	M	[24–27]
LIPDRS	Determined based on dynamic requests	OMD	Aggregated OD information and preferred arrival/departure times	Phone/Internet bookings	Dynamic B&A[6]	L	M	This study

[1] DARS: Dial-a-ride service; TCS: Traditional conventional shuttle; CB: Customized bus; SBRP: School bus; CS: Car-sharing service; LIPDRS: Large-scale instantaneous-peak-demand-responsive express shuttle service. [2] TD: Target demands; OID: Occasional individual demands; RMD: Regular massive demands; OMD: Occasional massive demands. [3] P: Price; H: High; L: Low. [4] C: Convenience: H: High; M: Moderate; L: Low. [5] R: Reference. [6] B&A: Boarding and alighting locations.

Regarding methodologies, combinations of request clustering and routing methods based on geographical similarities have been the focus of existing research. Cullen et al. [28] proposed a clustering-first, routing-second heuristic model. Bodin and Sexton [29] also constructed a clustering-first, routing-second heuristic model for the pickup and delivery problem with time windows (PDPTW). In this model, passengers are first partitioned into clusters. Then, the clusters are assigned to routes using the algorithm developed by Sexton and Bodin [30,31]. Finally, the algorithm swaps passengers between routes for route re-optimization. Dumas et al. [6] introduced the concept of mini-clusters, in which passengers with high spatiotemporal proximity are clustered together. In this algorithm, a heuristic method is used to produce a set of mini-clusters. Then, a column generation (CG) algorithm is used to optimally combine the mini-clusters into vehicle routes. Finally, the algorithm performs route re-optimization to obtain the optimal schedule for each vehicle. Desrosiers et al. [32] proposed another approach to mini-cluster construction by means of a parallel insertion method based on the spatiotemporal proximity of the requests. Ioachim et al. [33] applied an optimization-based technique instead of a heuristic method for the construction of mini-clusters. In addition to these studies, Pankratz [34], Bard and Jarrah [35], Qu and Bard [36], and Masson et al. [37] also applied clustering algorithms for the PDPTW. The results of these studies show that high-quality vehicle plans can be obtained only through the integrated consideration of clustering and routing.

In the field of operations research, a routing problem for demand-responsive service is typically formulated, as either a vehicle routing problem with time windows (VRPTW) [38,39] or a PDPTW [40,41]. In most VRPTWs or PDPTWs, the number of passengers served, the operating cost, and the service quality are the three major objectives to be optimized, either separately or simultaneously [42]. Typical vehicle-to-passenger matching and routing constraints include time window constraints, capacity constraints, coupling constraints, and maximum ride time constraints, which typically give rise to an NP-hard problem [43,44]. Many researchers have focused on studying corresponding solution techniques. CG algorithms [45,46], branch-and-price algorithms [15], branch-and-cut algorithms [47], branch-and-cut-and-price algorithms [48], tabu search algorithms [49], and dynamic-programming-based metaheuristics [14] can be used to solve problems of this type. The proposed LIPDRS problem can be regarded as an offline multivehicle routing problem because all the requests are assumed to be known beforehand.

Although significant improvements have been achieved in solving demand-responsive service design problems, LIPDRS design still poses several challenges, as discussed in Section 1.1. To overcome these challenges, we model the LIPDRS decision process using a mixed-integer programming model with dual objectives to optimize the service from the perspectives of both passengers and operators. This model includes a carefully designed passenger walking time constraint and request rejection mechanism to ensure that the resulting service plan can guarantee not only good quality of service, but also improved operational efficiency. Additionally, this paper proposes a clustering-first, routing-second framework for solving the proposed model. A modified version of the fuzzy c-means clustering algorithm and a CG algorithm are utilized in this framework to determine the service units and perform vehicle assignment and routing, respectively. In addition, to overcome the challenges caused by relaxing the model variables, a branching scheme is proposed.

The rest of the paper is organized as follows. Section 2, presents details of the methodology for solving the LIPDRS design model, including the (first) clustering phase, i.e., the determination of service units (DSU), and the (second) routing phase, i.e., fleet assignment and routing (FAR). Section 3 presents a numerical experiment conducted to analyze the performances of the proposed services on different request datasets. In addition, we present comparisons between the proposed LIPDRS option and the existing door-to-door transport option based on various artificial datasets to evaluate the application scope of our proposed method.

2. Methodology

In this section, the methodology of the present study is presented. We first introduce the whole architecture for LIPDRS design, including the overall service decision process and the construction of the LIPDRS design model (LIPPD). Second, we introduce the corresponding clustering-first, routing-second algorithm. Finally, we introduce the criteria for evaluating the results.

2.1. Framework for LIPDRS Design

2.1.1. Overall Service Decision Process

To enable an effective and feasible shuttle service, a seven-step process in which the requesters can be actively involved is designed, as shown in Figure 1.

1. *Request Collection.* Potential passengers submit their travel requests (each consisting of a preferred location tuple with boarding and alighting locations and a time window tuple with the expected boarding and alighting times) through mobile phones or other devices. The data collection and planning period is designated as one day; that is, in one day, the demands for the next day are collected and planned. Any request that is submitted before the deadline (either a specified moment, such as 12:00 p.m., or some specified number of hours ahead of the service time) will be pushed into the request pool to wait for stage 2. Other requests will be marked as overdue requests and will wait for stage 5.
2. *Determination of Service Units (DSU).* Requests in the request pool are aggregated into a set of service units (which specify the locations at which and time windows in which the requesters will be served) using the approach described in Section 2.3. The set of service units will be sent to the service pool to wait for stage 3. Isolated requests (i.e., requests that do not belong to any service unit) from the request pool are also detected in this stage. These isolated requests will wait for stage 4.
3. *Fleet Assignment and Routing (FAR).* Vehicles are assigned to specific service units, and the fleet is routed, subject to capacity constraints, spatiotemporal constraints, procedure constraints, etc. Through this process, the operators obtain the service routes and scheduled service plan (timetable) for the LIPDRS.
4. *Economic Evaluation.* The operators use evaluation metrics to compare the generated plan with other transport options and evaluate whether the plan will be economically acceptable to passengers. These evaluation metrics are discussed in Section 2.5.
5. *Request Rejection.* To ensure the quality and efficiency of demand-responsive transport, reasonable rejection of certain transport requests is an integral part of designing a transport plan. Three types of requests need to be rejected according to different rejection rules: overdue requests, isolated requests, and uneconomical requests. These three types of rejection occur in different parts of the planning process, as shown in Figure 1. The specific rejection mechanism will be discussed in Section 2.
6. *Information Feedback.* This stage consists of two parts: feedback to requesters who have been assigned to service units, with scheduled boarding and alighting information, and feedback to rejected requesters.
7. *Service Implementation.* After receiving their feedback information, passengers will walk or bike to their specified boarding locations within their specified time windows for service.

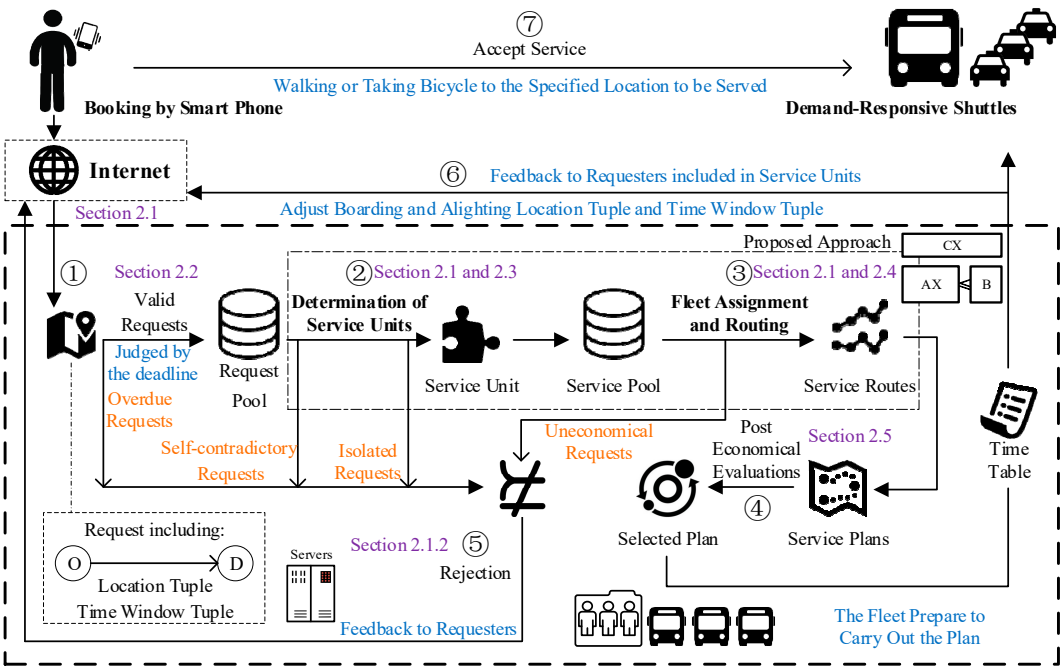


Figure 1. The Planning Process of a Demand-Responsive Express Shuttle Service. In this figure, the critical steps are labeled by numbers that are corresponding to the content of Section 2.1.1. Additionally, the subsections (that present detailed expressions about elements in this figure) are labeled to help understand the entire methodology structure.

2.1.2. LIPDRS Design Model (LIPPD)

The main idea on which the formulation of the LIPPD is based is that it is necessary to first determine service units based on the received requests and then assign and route the fleet in accordance with those service units. We formally formulate the LIPPD as follows:

Objectives:

$$\text{Minimize} \sum_{k\in M}\left(\sum_{i\in V\cup De}\sum_{j\in V\cup De} c_{ij}x_{ij}^{k} + F\sum_{j\in V} x_{De^{+}j}^{k}\right) + \sum_{i\in N} P\left(1-\sum_{k\in M} z_i^k\right) \tag{1a}$$

$$\text{Maximum } L(c) = \frac{\sum_{j=0}^{n-1}\sum_{i=0}^{c-1} u_{ji}^{m}\|\bar{r}-v_i\|^2/(c-1)}{\sum_{j=0}^{n-1}\sum_{i=0}^{c-1} u_{ji}^{m}\|r_j-v_i\|^2/(n-c)} \tag{1b}$$

subject to

$$\left\|r_j - v_i\right\|_W^{At} \le mwt\left(i\in N, v_i\in S, r_j\in R\right) \tag{2}$$

$$v_i \in Cad = \left\{v\prime_j \middle| 0 \le j < c\prime\right\} \tag{3}$$

$$\sum\nolimits_{k\in M} z_i^k \le 1 (i\in N) \tag{4}$$

$$\sum\nolimits_{j\in V\cup De} x_{lj}^k = \sum\nolimits_{j\in V\cup De} x_{jl}^k = z_i^k (i\in N, l\in V, k\ \in M) \tag{5}$$

$$\sum\nolimits_{j\in V\cup De^{-}} x_{De^{+}j}^k \le 1 (k\in M) \tag{6}$$

$$\sum\nolimits_{i\in V\cup De^{+}} x_{iDe^{-}}^k \le 1 (k\in M) \tag{7}$$

$$\sum\nolimits_{j\in V\cup De^{-}} x_{De^{+}j}^k = \sum_{i\in V\cup De^{+}} x_{iDe^{-}}^k (k\in M) \tag{8}$$

$$e_i \le D_i \le l_i (i \in V \cup De) \tag{9}$$

$$D_{i^+} \le D_{i^-} (i \in N) \tag{10}$$

$$x_{ij}^k = 1 \Rightarrow D_i + t_{ij} + op_j \le D_j\left(i, j \in V \cup De^+, k \in M\right) \tag{11}$$

$$x_{iDe^-}^k = 1 \Rightarrow D_i + t_{iDe^-} + op_{De^-} \le l_{De^-} (i \in N^-, k \in M) \tag{12}$$

$$y_{De^+} = 0 \tag{13}$$

$$y_l \le \sum\nolimits_{k \in M} Qz_i^k (i \in N, l \in V) \tag{14}$$

$$x_{ij}^k = 1 \Rightarrow y_i + q_i \le y_j (i, j \in V \cup De, k \in M) \tag{15}$$

$$\sum\nolimits_{i \in V, j \in V} x_{ij}^k d_{ij} \le d_{max} (k \in M) \tag{16}$$

$$\sum\nolimits_{i \in V, j \in V} x_{ij}^k t_{ij} \le t_{max} (k \in M) \tag{17}$$

$$u_{ji} \left(\sum\nolimits_{j=1}^{c-1} u_{ji} = 1 \,,\ 0 \le u_{ji} \le 1 \right) \tag{18}$$

$$x_{ij}^k \in \{0, 1\} (i, j \in V \cup De, k \in M) \text{ and } z_i^k \in \{0, 1\} (i \in N, k \in M) \tag{19}$$

$$D_i \ge 0 (i \in V \cup De) \text{ and } y_i \ge 0 (i \in V \cup De) \tag{20}$$

The related objective functions, constraints, variables, sets and parameters are defined as follows.

- Objective Functions

Equation (1a) represents the cost of operation, including the fixed and variable costs of using vehicles and the penalties for the cancelation (rejection) of service units. A smaller objective function value is better. Equation (1b) guides the DSU process. Unlike in the case of taxi and door-to-door services, LIPDRS requesters will be served at specified boarding/alighting locations (i.e., the locations of the service units) instead of the requesters' home/work locations [50]. The number of service units impacts the efficiency, effectiveness and equity of passenger service and vehicle operation. If the service units are too densely distributed (meaning that the clusters are small in size), the speed of service will not be effectively accelerated; if the service units are too sparsely distributed, more requesters will have to walk farther. Therefore, we select the ratio between compactness and divergence as one of the objective functions of the LIPPD, where (a) the divergence drives the distances between service units to be as large as possible and (b) the compactness drives the walking distances (times) between requesters and service units to be as small as possible. In this case, a larger objective function value is better. To extend the problem considered in this paper to other fields, different objective functions can be chosen from existing studies [7,51,52].

- Constraints

Equation (2) is a spatiotemporal constraint, namely, a maximum walking time constraint, where *mwt* denotes the maximum walking time; the norm $\cdot_{W}^{At}$ will be explained in Section 2.2.2. For example, in Figure 2, there are 17 requests and 6 candidate service units. As a result of the DSU process, 2 of the candidate service units are selected, and the 17 requests are grouped into service unit 2 and service unit 5. Within each service unit, the time windows for boarding and alighting satisfy the intracluster operation time constraint. Service unit 2 consists of 14 close requests, and the walking time between each requester and the service location is less than the predefined maximum walking time *mwt*. Equation (3) ensures that the service units are selected from among the candidate service units because the vehicles should not park at arbitrary locations. Equation (4) ensures that each service unit is assigned to exactly one vehicle or is canceled. Equation (5) enforces that a vehicle must visit both the boarding and alighting points of every service unit assigned to it. Equations (6)–(8) ensure that the vehicles either perform assigned tasks or remain idle. Equations (9)-(12) together form the

precedence constraints. Equations (13)–(15) together form the capacity constraints. Equations (16) and (17) represent the maximum distance constraint and the maximum driving time constraint, respectively.

- Decision Variables

The variables of this model are as follows. $x_{ij}^k\left((i,j) \in (V \times V) \cdot \{(De^+, j) \middle| j \in V\} \cdot \{(j, De^-) \middle| j \in V\}, k \in M\right)$ is a binary variable that is equal to 1 if vehicle k is scheduled to travel from location i to location j and is equal to 0 otherwise. z_i^k $(i \in N, k \in M)$ is a binary variable that is equal to 1 if service unit i is assigned to vehicle k and is otherwise equal to 0. c, a positive integer variable, represents the number of clusters (i.e., the number of service units). The u_{ji} $(\sum_{j=1}^{c-1} u_{ji} = 1, 0 \le u_{ji} \le 1)$ are positive variables. $U_{n \times c}$ is the matrix of membership values, and $u_{ji} \in U_{n \times c}$ is the membership value of the jth request with respect to service unit i. D_i $(i \in V \cup De)$ is an integer variable that specifies the departure time at vertex i. y_i $(i \in V \cup De)$ is an integer variable that specifies the load of the vehicle when arriving at vertex i. We define $q_{De^+} = 0$ for all $k \in M$.

- Sets

In this model, N denotes the set of service units. For each service unit $i \in N$, a request group of size $q_i \in \mathbb{N}$ needs to be transported from boarding point i^+ to alighting point i^-. It is known that $q_i = q_{i^+} = -q_{i^-}$, where positive quantities correspond to pickups and negative quantities correspond to dropoffs. We define $N^+ \triangleq \{i^+ \middle| i \in N\}$ as the set of boarding points and $N^- \triangleq \{i^- \middle| i \in N\}$ as the set of alighting points. Let $V \triangleq N^+ \cup N^-$. Furthermore, M represents the set of vehicles. Each vehicle $k \in M$ has a capacity $Q \in \mathbb{N}$, a starting location De^+, and an ending location De^-. We set $De \triangleq De^+ \cup De^-$.

- Parameters

For all $i, j \in V \cup De$, c_{ij} denotes the travel cost. F represents the fixed cost of a vehicle, and P represents the penalty for cancelling serving a service unit. Then, m is a fuzzifier with a value greater than 1, where a higher value indicates a higher degree of ambiguity [51]. n is the number of requests. $\bar{r}$ represents the average of the spatiotemporal features of all requests. r_j represents the spatiotemporal features of the jth request, and v_i represents the spatiotemporal features of service unit i. These spatiotemporal features consist of location and time window information. We represent a request as an eight-dimensional vector $r_j = \left(lo_{j+}, tw_{j+}, lo_{j-}, tw_{j-}\right) = \left(lng_{j+}, lat_{j+}, e_{j+}, l_{j+}, lng_{j-}, lat_{j-}, e_{j-}, l_{j-}\right)$, where '+' indicates features corresponding to the origin and '–' indicates features corresponding to the destination. Similarly, each service unit is associated with an eight-dimensional feature vector of the same form, $v_i = (lo_{i+}, tw_{i+}, lo_{i-}, tw_{i-}) = (lng_{i+}, lat_{i+}, e_{i+}, l_{i+}, lng_{i-}, lat_{i-}, e_{i-}, l_{i-})$, where '+' indicates boarding information and '–' indicates alighting information. Each candidate service unit is similarly represented in the form $v\prime_i = (lo_{i+}, tw_{i+}, lo_{i-}, tw_{i-}) = (lng_{i+}, lat_{i+}, e_{i+}, l_{i+}, lng_{i-}, lat_{i-}, e_{i-}, l_{i-})$. The mwt denotes the maximum walking time. Then, we denote the set of requests by $R = \left\{r_j \middle| 0 \le j < n\right\}$, the set of service units by $S = \{v_i | 0 \le i < c\}$ (where c is the number of service units) and the set of candidate service units by $Cad = \{v\prime_i | 0 \le i < c\prime\}$, where n, c and $c\prime$ are the numbers of elements of S, R and Cad, respectively. $[e_i, l_i]$ denotes the time window, and op_i denotes the operation time. For all $i, j \in V \cup De$, d_{ij} denotes the travel distance, and t_{ij} denotes the travel time. For each service unit $i \in N$, the number of requests $q_i \in \mathbb{N}$ is given. Q is the capacity of vehicle. Additionally, each vehicle has an operation distance range $[0, d_{max}]$ and an operation time range $[0, t_{max}]$ (representing the driver's working time).

Based on the description given in Figure 2, we further illustrate the model by means of a simple example of the LIPDRS design process. Objective function (1.2) and constraints (2) and (3) are used to select service units from among the candidate service units and assign requests to them in accordance with the spatiotemporal features of the requests. The bottom of Figure 2 shows the process of selecting service units. In this example, the requesters submit requests that contain spatiotemporal origin and destination information, represented by oblique-textured and cross-textured circles, respectively. The candidate service units are represented by untextured squares. Ultimately, two service units are selected, represented by textured squares; they are identified as service unit 2 and service unit 5. These two service units serve two branches of requests with different destinations. Each service unit has two components: the boarding component (oblique-textured squares) and the alighting component (cross-textured squares). Both components contain the corresponding locations and time windows

for the requesters and vehicles. Requesters walk to the boarding location indicated by the boarding component, are transported to the alighting location indicated by the alighting component, and then proceed to their destinations.

In this example, five service units are selected, and the operator assigns them to vehicles in the fleet and designs routes for the fleet. Objective function (1.1) and constraints (3)-(20) are used for assignment and routing. Once the service routes have been determined, the fleet departs from the depot(s), visits each service unit within the corresponding service time window, and then returns to the depot. Generally, the LIPDRS solution comprises several service routes, such as the two example routes (0-1-1′-2-2′-0 and 0-4-5-4′-5′-3-3′-0) shown in Figure 2.

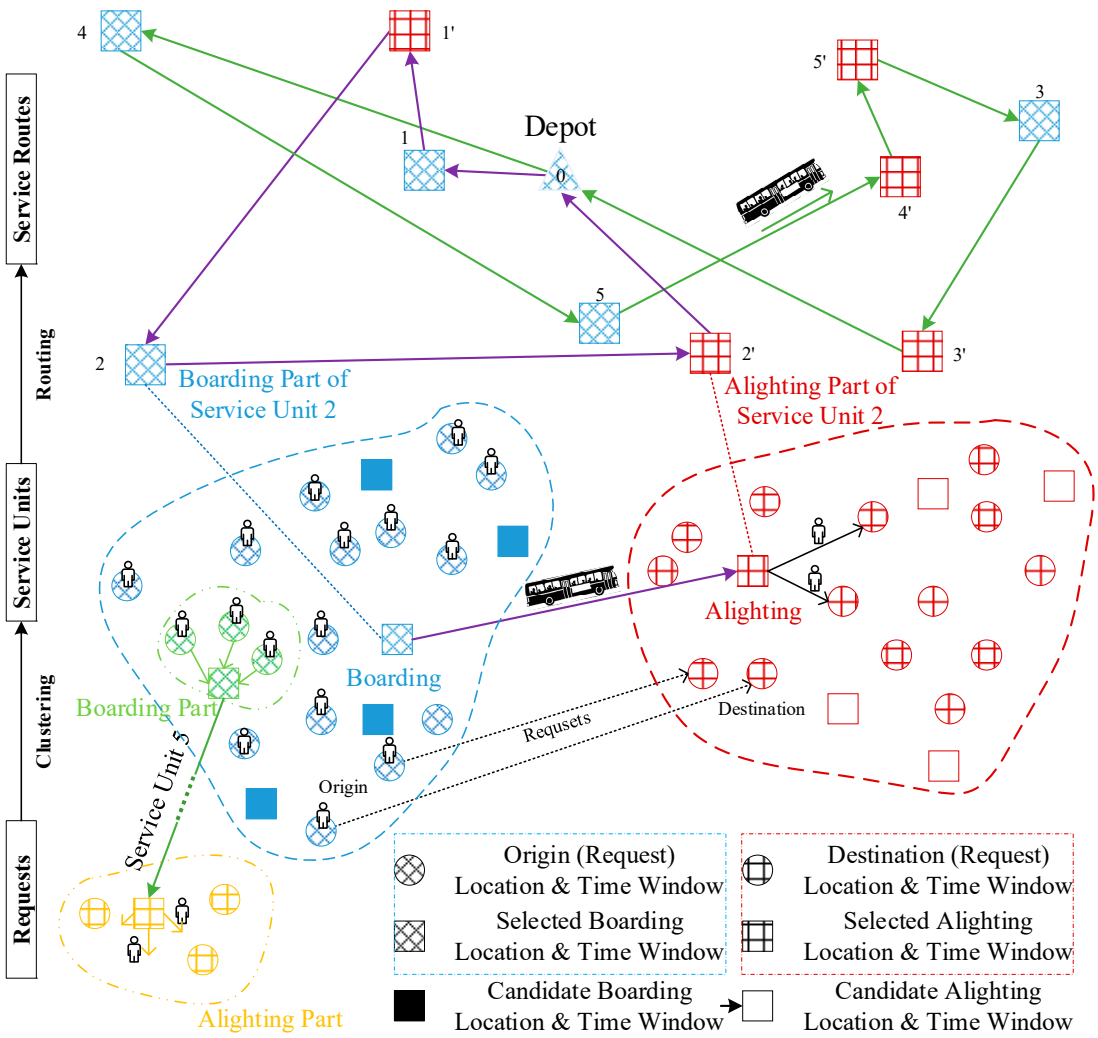


Figure 2. Illustration of the LIPDRS design model (LIPPD). This figure demonstrates the main optimization content of the proposed model, from the bottom to the top. There are three types of nodes in this figure, where the cycles represent requests (the diagonal striped circles represent origins, and the striped circles represent destination); the squares represent boarding parts and alighting parts of service units (the diagonal striped squares represent selected boarding parts, the striped squares represent selected alighting parts, the filled squares represent the candidate boarding parts, and the blank squares represent the candidate alighting parts). Both the boarding part and the alighting part include information about location and time window. The figure shows the process of design service units and service routes from requests that exists in the optimization model. In the middle and the bottom of the figure, requests are assigned to service units according to spatiotemporal features of the requests and constraints. Then, in the top of the figure, the boarding parts and alighting parts of service units are assigned to service routes and to be served by the vehicles in the fleet.

2.1.3. Clustering-First, Routing-Second Algorithm for Solving the LIPPD

The solution process for the LIPPD is the process of finding the optimal service units and service routes in accordance with the spatiotemporal features of the requests. Ideally, we wish to find an optimal solution to the LIPPD that considers the trade-off between passenger costs and vehicle costs [53]. However, from the model, we can see that it is not easy to obtain solutions to this problem: the objective functions are multitarget functions with nonlinear structures. Therefore, instead of looking for the globally optimal solution, we use a two-stage algorithm to find a near-optimal solution. This section introduces the clustering-first (DSU), routing-second (FAR) solution approach. This solution framework implies the following assumption about the priority of the targets: the burden placed on the passengers by this new service mode is minimized first, and the operational cost is then minimized. In other words, we pay more attention to the requesters' costs than to the operators' costs.

To facilitate clearer description throughout the rest of the paper, we use *RM* to denote the submodel comprising Equations (1a), (4)–(17), (19) and (20) and use *CM* to denote the submodel comprising Equations (1b), (2), (3) and (18). The DSU process focuses on solving *CM*, and the FAR process focuses on solving *RM*.

2.2. Preprocessing and Distance Measurement

2.2.1. Preprocessing

The preprocessing phase consists of two tasks: (i) the detection and deletion of self-contradictory requests and (ii) the detection and deletion of abnormal requests.

Since the service considered here needs to be feasible in an actual scenario, it is necessary to exclude user-submitted requests with self-contradictions. In general, such requests are not spatiotemporally feasible; for example, the separation between the expected time windows for boarding and alighting may be too small, meaning that a vehicle cannot complete the specified task, while obeying the speed limit in the corresponding road network. Such self-contradictory requests should be deleted to avoid infeasible solutions to the model. For this purpose, we use formula (21) to detect self-contradictory requests:

$$\|v\prime_i\|^{Am} \geq \varsigma, \ \forall v\prime_i \in Cad \tag{21}$$

where $\|v\prime_i\|^{Am}\ \forall v\prime_i \in Cad$ is a norm defined as $\|v\prime_i\|^{Am} \triangleq \|l_{i+}, l_{i-}\|_R^{lt} - (e_{i-} - e_{i+})$, with ς representing the operation time redundancy for a vehicle and a, b_R^{lt} representing the vehicle travel time (obtained from a digital map) between locations a and b. The definition of the norm $\|a, b\|_R^{lt}$ will be discussed in Section 2.2.2.

In addition to self-contradictory requests, abnormal requests should also be detected and removed from the request pool. There are two main types of abnormal requests: (i) requests caused by misoperation, such as requests whose starting and ending locations are too close or repeated requests with the same ID number, and (ii) spatially invalid requests, such as requests corresponding to locations that are significantly beyond the scope of the service. The screening for such abnormal requests relies mainly on simple heuristic rules.

2.2.2. Distance Measurement

As a service operated in a metropolitan context, LIPDRS requires a reasonably accurate distance calculation method. The temporal and spatial distances between two points cannot be directly calculated using the Euclidean distance but should instead be calculated in accordance with the conditions of the road network, speed limits, etc. Measuring the distance by means of the path planning function of an electronic map is a viable option. Here, we define two temporal distance norms, $\|(l_{i+}, l_{j+})\|_W^{lt}$ and $\|(tw_{i-}, tw_{j-})\|^{twt}$, and one spatial distance norm, $\|(l_{i+}, l_{j+})\|_W^{ls}$, where $\|(l_{i+}, l_{j+})\|_W^{lt}$ and $\|(l_{i+}, l_{j+})\|_W^{ls}$ are obtained from a digital map (e.g., AMAP).

In addition, we define $\|x_i - v_j\|_W^{At}$ as a measure of the temporal distance between a request and a service unit, as expressed by Equation (3). This measure is vitally important to the process of assigning requests to service units:

$$\|x_i - v_j\|_W^{At} = \|(l_{i+}, l_{j+})\|_W^{lt} + \|(tw_{i+}, tw_{j+})\|^{twt} + \|(l_{i-}, l_{j-})\|_W^{lt} + \|(tw_{i-}, tw_{j-})\|^{twt} \quad (22)$$

where $\|(a,b)\|_W^{lt}$ represents the passenger walking time (obtained from a digital map) between locations a and b and $\|(a,b)\|^{twt}$ represents the time difference between time windows a and b.

2.3. Clustering (First Stage): Determination of Service Units (DSUs)

2.3.1. Main DSU Flows

We introduce the DSU solution process in the form of the flow chart shown in Figure 3. During the DSU solution process, on one hand, the locations and time windows of the service units must be selected to allow the requests to be assigned to service units, and on the other hand, as in other clustering-based problems, determining the number of clusters c is also an important task [7]. Therefore, a cluster number determination method is required. To this end, the structure of this flow chart is composed of two loops and corresponding initialization modules. The outer loop is designed to determine the optimal number of service units, and the inner loop applies a clustering method to determine the service units in accordance with a given number of clusters.

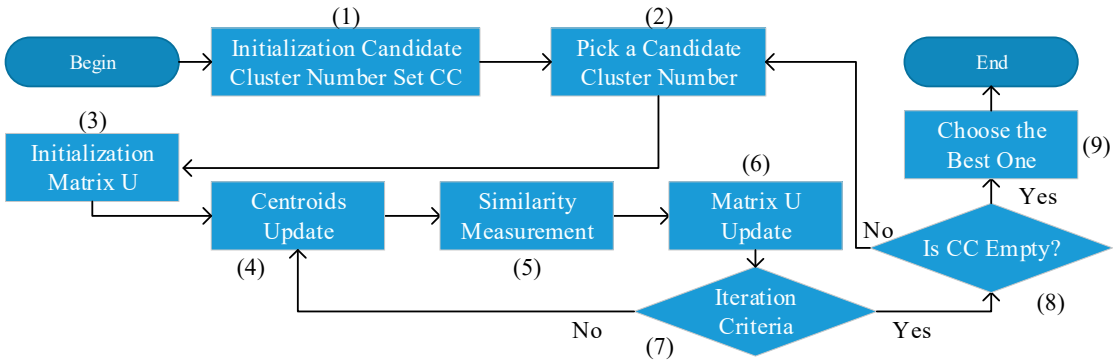


Figure 3. Determination of service units (DSU) flows.

In Figure 3, modules (3)–(7) compose the inner loop for the clustering task. The purpose of clustering, an unsupervised machine learning method, is to aggregate the requesters by generating centroids (which can be used as the basis for selecting service units) based on similarities among the spatiotemporal features of the requests. Among the many clustering methods available, we choose fuzzy c-means clustering (FCM) as the basic method of identifying service units and assigning requests. FCM is a soft clustering method. Unlike hard clustering methods (e.g., k-means), FCM simultaneously considers the possibilities that a data point may belong to multiple clusters by means of membership degrees, resulting in a much finer degree of data modeling that yields numerical results that can be used for further analysis [7]. However, the original FCM method cannot accommodate the walking time constraint, which is vitally important to our research. We therefore propose a new similarity measure to modify the original FCM to consider a limit on the walking time.

The outer loop, which is composed of the inner loop and modules (1), (2), (8) and (9), selects a number of candidate clusters from a set and passes this cluster number to the inner loop. This is an enumeration process. At the end of the loop, the number of clusters that results in the best objective function value is selected as the final number of service units. The number of service units impacts the efficiency, effectiveness and equity of passenger service and vehicle operation. If the service units are too densely distributed (meaning that the clusters are small in size), the speed of service will not

be effectively accelerated; if the service units are too sparsely distributed, more requesters will have to walk farther. Thus, the number of service units is a key factor in striking this balance. We will introduce each module in detail throughout the rest of Section 2.3.

2.3.2. Initialization

There are two initialization modules in Figure 3. Module (1) is for the initialization of the set of the candidate numbers of clusters, *cc*, wherein we construct a set-in increments of one within a given range. The other initialization module is module (3). The elements in the matrix of membership values $U_{n\times c}$ for FCM are randomly initialized in accordance with the following constraints: the elements u_{ji} must satisfy $\sum_{j=1}^{c-1} u_{ji} = 1$ and $0 \le u_{ji} \le 1$.

2.3.3. Measurement of Similarity between a Request and a Service Unit

The norm for measuring the similarity of the *jth* request to the centroid of cluster *i* is vitally important because it directly influences the objective function (1.2). In the flow of the DSU solution process, the similarities are calculated in module (5). A typical norm is the A-norm proposed by Bezdek et al. [54]. However, the A-norm can measure only geographical proximity and thus is not suitable for measuring spatiotemporal proximity, as is necessary for the walking time constraint. Therefore, we propose a Gaussian-kernel-based norm that allows the maximum walking time to be considered, as presented in (23):

$$\|r_j - v_i\|_W^{GAt} = \begin{cases} \|r_j - v_i\|_W^{At} / exp\left(-\|r_j - v_i\|_W^{2At} / 2\sigma^2\right), \left(for\ any\ r_j, \exists v_i : \|r_j - v_i\|_W^{At} \le mwt\right) \\ \|r_j - v_i\|_W^{At}, \left(for\ any\ r_j, \forall v_i : \|r_j - v_i\|_W^{At} > mwt\right) \end{cases} \tag{23}$$

In (23), we use the temporal distance $\|r_j - v_x\|_W^{At}$ to represent the distance between two locations instead of calculating the Euclidean distance. The definition of $\|r_j - v_x\|_W^{At}$ can be found in Section 2.2.2.

A Gaussian kernel, a kernel type that is widely used in machine learning [55], can make objective function (1.2) valid only in a local area (i.e., if a requester's walking time is greater than the maximum passenger walking time *mwt*, then the similarity to the corresponding service unit will be very low). However, as a result of the mechanism of generating the initial solution for FCM [7,56], the initially selected service units might be far away from any request. In other words, a service unit might have minimal similarity to all requests. Therefore, we design (23) as a piecewise function to avoid empty clusters (i.e., when there is at least one request in a service unit, the Gaussian kernel is used). In (23), σ is a width parameter that controls the range in the radial direction, whereby we can control *mwt*.

2.3.4. Iteration Method and Termination Criteria

In modules (4) and (6), the service units v_i are reselected, and the membership values u_{ji} are updated. For this purpose, we propose Equations (24) and (25). The norm $\lfloor a \rfloor_{mCad}$ can be used to enumerate the candidate service units and find the nearest one to the centroid a_i of the requests, where the centroid a_i is calculated as $a_i = \sum_{j=0}^{n-1}\left(u_{ji}\right)^m r_j / \sum_{j=0}^{n-1}\left(u_{ji}\right)^m$.

$$u_{ji} = 1 / \sum_{s=0}^{c-1} \left(\frac{\left|r_j - v_i\right|}{\left|r_j - v_s\right|} \right)^{\frac{1}{m-1}} \tag{24}$$

$$v_i = \left\lfloor \sum_{j=0}^{n-1} \left(u_{ji}\right)^m x_j / \sum_{j=0}^{n-1} \left(u_{ji}\right)^m \right\rfloor_{mCad} \tag{25}$$

Following Bezdek [56], $m = 2$ is adopted in this study. Two criteria for terminating the iterative process are designed for module (7). One is a threshold ε. If $\left|U_{p+1} - U_p\right| < \varepsilon$, then the iterative process terminates, where U_p is the membership matrix in the *pth* iteration. The other is a maximum number

of iterations, *mi*. After either termination criterion is triggered, any remaining requests with very low similarities (close to 0) to any service unit become isolated requests.

2.4. Routing (Second Stage): Fleet Assigning and Routing (FAR)

We develop a model based on the pickup and delivery problem with time windows (PDPTW) for formulating the FAR problem. Considering the complexity of the search space in a large-scale network, we use a column generation (CG) algorithm in the framework for solving the model. In addition, to address the influence of variable relaxation, a branching scheme is used to generate a binary solution tree to enable the selection of a feasible and optimal solution.

2.4.1. Main FAR Flows

The FAR process converts the service units determined via the DSU process into service routes, which represent the specific times and order of the service units served by the fleet. We introduce the FAR solution process in the form of the flow chart shown in Figure 4. The structure of this flow chart is composed of two loops and corresponding initialization modules. These two loops are designed to calculate the model and to guarantee that there is no infeasible solution, namely, no noninteger solution, which can occur as a side effect of the relaxation of the model.

In Figure 4, the inner loop (denoted by A) applies the CG algorithm, iteratively solving the master problem and the subproblem. This loop is composed of modules (4)-(9). The outer loop (denoted by B) applies a branching scheme to avoid noninteger solutions. It is composed of loop A and modules (10)–(15). Modules (1)–(3) are the initialization modules for the two loops. Focusing on these two loops and the modules they include, we will elaborate on the details of the algorithm in Sections 2.4.2–2.4.6.

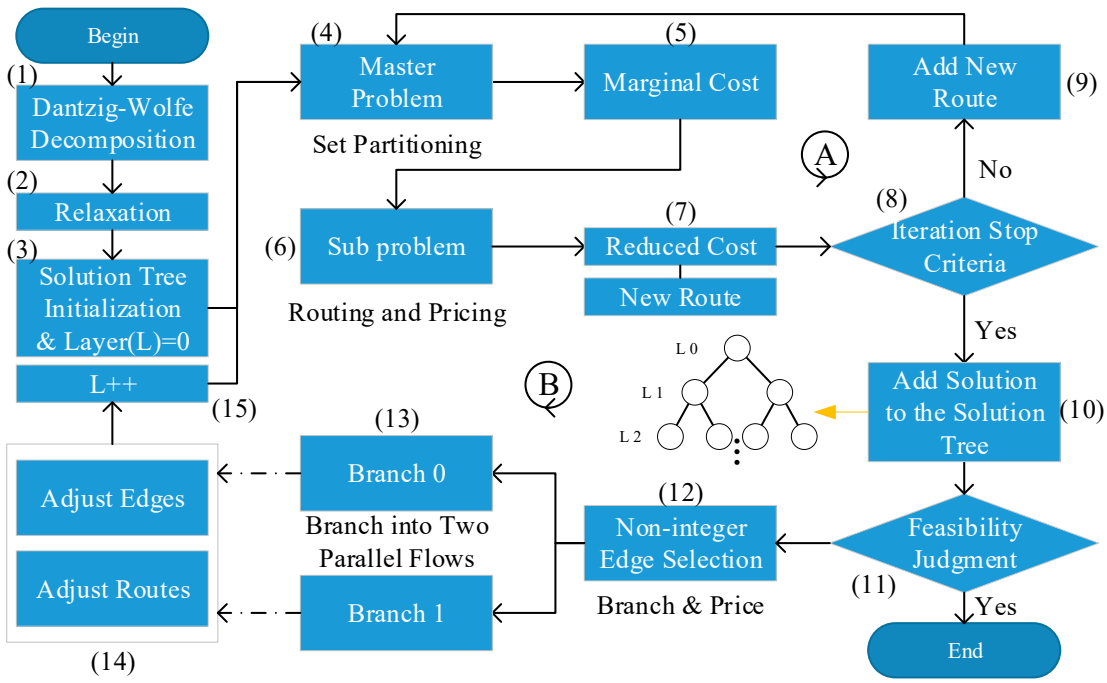


Figure 4. FAR Flows.

2.4.2. Initialization

(1)–(2): Dantzig-Wolfe Decomposition and Relaxation of the Master Problem

To circumvent the difficulties facing such large-scale problems [57], we decompose the model introduced in Section 2.1.2 via Dantzig-Wolfe (DW) decomposition. After DW decomposition, the original model has been decomposed into an equivalent combination of two smaller problems: a master problem (the fleet assignment problem) and a subproblem (the routing and pricing problem).

The master problem is the set coverage problem, and the subproblem is the route planning and pricing problem. The decomposed model can be solved via a CG algorithm. To obtain the marginal cost of the master problem, we relax the integer decision variables of the master problem to positive variables.

(3): Solution Tree

A side effect of the relaxation of the master problem is that the route selection process may yield noninteger solutions. A mature approach for overcoming this problem of noninteger solutions is the branching approach [57]. To obtain information about when to start and end each branch, we design a solution tree with index L and a feasibility judgment method to store solutions and judge whether the process should terminate or continue.

As shown in Figure 4, the solution tree is a layered binary tree. The original solution is stored as the root. The feasibility judgment method is used only to judge the solutions in the *Lth* layer. If the feasibility judgment method detects that a noninteger route has been selected in the solution, then the branching method described in [57] is applied. Otherwise, no noninteger route has been selected in the solution, and the algorithm ends. Finally, we choose the best solution in the solution tree as the output.

2.4.3. Master Problem (Relaxed Set Partitioning Problem, MP)

The master problem *MP* is a set partitioning problem, which can be formulated as follows.

Objective:

$$\begin{aligned} \text{Minimize } & \textstyle\sum_{r_k \in \Omega} c_k \theta_k + \sum_{i \in N} P\left(1 - \sum_{r_k \in \Omega} \delta_i^k \theta_k\right) \\ & = \textstyle\sum_{i \in N} P + \sum_{r_k \in \Omega} \theta_k \left(c_k - \sum_{i \in N} P \delta_i^k\right) \end{aligned} \tag{26}$$

subject to

$$\sum_{r_k \in \Omega} \delta_i^k \theta_k \leq 1 (i \in N) \tag{27}$$

$$\sum_{r_k \in \Omega} \theta_k \leq |M| \tag{28}$$

$$\theta_k \in [0, 1] (r_k \in \Omega) \tag{29}$$

Here, M denotes the set of available vehicles, Ω denotes the set of all feasible pickup and drop-off routes (r_k) served by vehicle k, and c_k denotes the cost of route r_k.

The primary objective of *MP* is to minimize the number of vehicles needed, and the secondary objective is to minimize the total cost. Therefore, c_k is calculated as $c_k = FC + VCL_k$, where L_k denotes the length of route r_k, VC denotes the unit cost of a vehicle, and FC denotes the fixed cost of a vehicle, which should be relatively large. This cost structure can be achieved by defining the travel costs $c_{ij} = d_{ij} (i \neq De)$ and $c_{Dej} = FC + d_{Dej} (j \neq De)$.

Furthermore, let parameters δ_i^k be defined as follows:

$$\delta_i^k \triangleq \begin{cases} 1 \; if \; i \in N \; is \; served \; on \; route \; r_k \in \Omega, \\ 0 \; otherwise. \end{cases} \tag{30}$$

Let the θ_k be positive variables of the master problem:

$$\theta_k \in [0, 1] \tag{31}$$

If $\theta_k = 1$, then route $r_k \in \Omega$ is selected, and if $\theta_k = 0$, then route $r_k \in \Omega$ is not selected; otherwise, if $\theta_k \in (0, 1)$, the solution is infeasible. The process of replacing the θ_k with positive variables rather than integer variables is called relaxation, which is performed in module (2). Equation (27) ensures that each service unit will be either served or canceled, and the penalty for canceling service units is represented by $\sum_{i \in N} P\left(1 - \sum_{r_k \in \Omega} \delta_i^k \theta_k\right)$ in objective function (26). Equation (28) ensures that the number of selected routes cannot exceed the scale of the fleet.

2.4.4. Subproblem (Routing and Pricing Problem, S_k)

Let the graph of the subproblem be denoted by $G = (N, E)$, and let the cost of each arc $(i, j) \in E$ be denoted by c_{ij}^G. Then, the subproblem S_k is formulated as follows:

Objective:

$$\text{Minimize} \sum_{i \in V} \sum_{j \in V} c_{ij}^G x_{ij} \tag{32}$$

subject to

$$\sum_{j \in V} x_{i^+ j} = \sum_{j \in V} x_{ji^+} = z_i (i \in N) \tag{33}$$

$$\sum_{j \in V} x_{i^- j} = \sum_{j \in V} x_{ji^-} = z_i (i \in N) \tag{34}$$

$$\sum_{j \in V} x_{De^+ j} = 1 \tag{35}$$

$$\sum_{i \in V} x_{iDe^-} = 1 \tag{36}$$

$$D_{i^+} \leq D_{i^-} (i \in N) \tag{37}$$

$$D_i + t_{ij} - D_j + op_j + x_{ij} M \leq M\left(i, j \in V \cup De^+\right) \tag{38}$$

$$D_i + t_{iDe^-} - l_{De^-} + op_{De^-} + x_{iDe^-} M \leq M (i \in N^-) \tag{39}$$

$$y_{De} = 0 \tag{40}$$

$$y_j - y_i - q_i + x_{ij} M \leq M (i, j \in V) \tag{41}$$

$$y_j - y_i - q_i - x_{ij} M \geq -M (i, j \in V) \tag{42}$$

$$\sum_{i \in V, j \in V} x_{ij} d_{ij} \leq d_{max} \tag{43}$$

$$\sum_{i \in V, j \in V} x_{ij} t_{ij} \leq t_{max} \tag{44}$$

$$e_i \leq D_i \leq l_i (i \in V) \tag{45}$$

$$0 \leq y_i \leq Q (i \in V) \tag{46}$$

$$x_{ij} \in \{0, 1\} (i, j \in V) \tag{47}$$

$$z_i \in \{0, 1\} (i \in N) \tag{48}$$

The target of S_k is to find routes that minimize the reduced cost for the master problem. In objective function (32), c_{ij}^G is composed of $c_{i^+ j}^G \triangleq c_{i^+ j} - P - u_i$, $c_{Dej}^G \triangleq c_{Dej} - w$ and $c_{ij}^G \triangleq c_{ij}$ $(i \notin N^+ \cup De)$, where u_i and w belong to $\left(\vec{u}, w\right)$, which is the marginal cost of the master problem MP.

In S_k, the following variables are significant: z_i $(i \in N)$, a binary variable, is equal to 1 if service unit $i \in N$ is served by the vehicle and is equal to 0 otherwise. x_{ij} $((i, j) \in V \times V)$, a binary variable, is equal to 1 if the vehicle travels from location $i \in V$ to location $j \in V$ and is equal to 0 otherwise. D_i $(i \in V)$, an integer variable, denotes the departure time at location $i \in V$. y_i $(i \in V)$, an integer variable, denotes the load of the vehicle when arriving at location $i \in V$.

Regarding the constraints, Equations (33) and (34) ensure that the vehicle visits either both the boarding and alighting points of a service unit or neither of them. Equations (35) and (36) ensure that the vehicle starts and ends at its depot. Equations (37)–(39) and (45) together form the precedence constraints. Equations (40)–(42) and (46) together form the capacity constraints. Equations (43) and (44) ensure that the vehicle will return within a limited travel distance and duration.

2.4.5. Iteration Termination Criteria

The solution process for MP and S_k is embedded in loop A. After the input data and initial routes are obtained, MP is solved first, yielding a marginal cost $\left(\vec{u}, w\right)$ for S_k. Then, S_k is solved using a labeling method based on dynamic programming [58], and the solution is a new route for MP. The new route is the one that incurs the minimal reduced cost for MP. This process does not stop until the solution to S_k meets the following termination criterion: the reduced cost of the returned route is equal to or greater than 0.

The individual termination criteria of the two loops are also important. In module (8), if the reduced cost obtained from the subproblem is greater than 0, then the loop stops. Otherwise, the loop continues. In module (11), if the solutions in the Lth layer are feasible, then the loop stops. Otherwise, the loop continues.

2.4.6. Branching Scheme

When a solution appears infeasible, namely, noninteger in nature, a branching scheme should be applied, as in modules (12)–(14). The rules of the branching scheme are as follows:

(i) For each arc (i, j), calculate the values $f_{ij} = \sum_{r_k \in \Omega} b_{ij}^k \theta_k$ $(0 \leq f_{ij} \leq 1, \sum_j b_{ij}^k = \delta_i^k (i, j \in N))$ and select an arc that satisfies $0 < f_{ij} < 1$.

(ii) Derive a branch on which the selected arc cannot belong to the solution (enforce $f_{ij} = 0$ and delete the corresponding edges and routes).

(iii) Derive a branch on which the arc must be included in the solution (enforce $f_{ij} = 1$ and delete the corresponding edges and routes).

After branching, the index L is updated, and the algorithm flow continues until the termination criterion is satisfied.

2.5. Evaluation Metrics

Evaluation metrics are needed to (i) provide a basis for the comparison of different transport options with different objective functions and (ii) evaluate whether a plan is economically feasible, i.e., whether passengers will be economically willing to accept this travel option.

To evaluate the proposed transport option, the difference between it and other existing methods must be quantified. However, the objective functions of different models are different; therefore, it is not possible to directly compare the quality of different service methods based on their objective function values. Here, we use several indicators to evaluate different modes of transport. Economic indicators are often the best basis for comparison. This article proposes indicators for evaluating services from two perspectives: (i) fleet operating costs and (ii) passenger travel costs. The former perspective is the focus of the operators, and the latter is the focus of both the operators and the passengers and can be considered to directly determine whether a particular transport option is acceptable to passengers. These indicators are related to the objective function but provide more detailed information about the solution.

Here, we define two sets of notations for evaluation metrics. From the perspective of the passenger travel costs, we define (i) C_{PW}: the cost that passengers expect to incur due to additional walking time (CNY/min); (ii) T_{TW}: the total walking time (min); (iii) D_{TPV}: the total distance traveled by all passengers on vehicles (km); (iv) C_G: the unified generalized cost (in terms of time and comfort) of passengers using the service; (v) T_{TPV}: the total time spend by all passengers on vehicles; (vi) P_F: the fixed price for using the service; (vii) P_V: the unit variable price for using the service; and (viii) N_S: the number of requests served. From the perspective of the fleet operating costs, we define (ix) N: the number of vehicles used; (x) D_{TTV}: the total distance traveled by vehicles (km); (xi) C_F: the unit fixed cost of using a vehicle; (xii) C_V: the unit variable cost of using a vehicle; (xiii) P_{RR}: the penalty cost for a rejecting request; and (xiv) N_{RR}: the number of rejected requests. Then, we calculate two costs for evaluation, as shown below.

α. The passenger cost, which is composed of (i) the walking cost c_{tw}, $c_{tw} = C_{PW} \cdot T_{TW}$; (ii) the ticket cost c_{tt}, $c_{tt} = P_F \cdot N_S + P_V \cdot D_{TPV}$; and (iii) the general cost c_{gt} incurred due to travel time, $c_{gt} = C_G \cdot T_{TPV}$. The formula for α is given in Equation (49):

$$\begin{aligned} \alpha &= c_{tw} + c_{tt} + c_{gt} \\ &= C_{PW} \cdot T_{TW} + P_F \cdot N_S + P_V \cdot D_{TPV} + C_G \cdot T_{TPV} \end{aligned} \tag{49}$$

β. The fleet cost, which is composed of (i) the fixed cost c_{ff} of the fleet, $c_{ff} = N \cdot C_F$; (ii) the variable cost c_{vf} of the fleet, $c_{vf} = VC_V \cdot D_{TTV}$; and (iii) the rejection penalty p_r, $p_r = N_{RR} \cdot P_{RR}$. The formula for β is given in Equation (50):

$$\begin{aligned} \beta &= c_{ff} + c_{vf} + p_r \\ &= N \cdot C_F + C_V \cdot D_{TTV} + N_{RR} \cdot P_{RR} \end{aligned} \tag{50}$$

Intuitively, by comparing α and β for different transport options, we can judge which option is economically superior.

3. Numerical Experiment

3.1. Data Preparation

Consistent with the process discussed in Section 2.1.1, we used records from a booking system for collecting real-world requests as the basic dataset (RRD) for our numerical experiment. We focused on requests located in northwest Beijing, that is, in the Changping and Haidian districts, because obvious LIPP phenomena arise in these districts as a result of their functional differences: Changping consists of large-scale residential areas, whereas Haidian consists of many business, educational and scientific areas. Many people work in Haidian and live in Changping; therefore, they require transportation between these two districts. The dataset used in this experiment contains 712 requests scattered throughout northwest Beijing during the period of 6:00–9:00 (the morning peak period) on 2019/04/21.

To further explore the performances of the proposed option and other options, we would like to consider different types of datasets (different request distributions) to test the application scopes of the different models. However, it would be difficult to obtain arbitrary request distributions without forcing users to make particular choices. Therefore, we artificially generated a series of datasets based on RRD (i.e., we simulated the booking process by randomly generating requests based on RRD in accordance with a series of rules). By using these datasets, we could test the model performances in various contexts. We defined the following artificial datasets: (i) ARR, a dataset of artificial random requests, contains 700 artificially generated requests that are evenly and randomly distributed in time and space. (ii) ACR, a dataset of artificial clustered requests, contains 700 artificially generated requests separated into 50 clusters that are relatively concentrated in time and space. (iii) ARCR25%, a dataset consisting of a mix of artificial requests with random and clustered structures, contains 700 artificially generated requests, 25% of which are relatively concentrated in time and space and the remaining 75% of which are not. (iv) ARCR50%, another dataset consisting of a mix of artificial requests with random and clustered structures, contains 700 artificially generated requests, 50% of which are relatively concentrated in time and space and the remaining 50% of which are not. (v) ARCR75%, the final dataset consisting of a mix of artificial requests with random and clustered structures, contains 700 artificially generated requests, 75% of which are relatively concentrated in time and space and the remaining 25% of which are not. The preprocessing method discussed in Section 2.2.1 was applied to obtain requests that were suitable for scheduling. Then, we conducted our experiment, as described in Section 3.2.

3.2. Experimental Scheme

The numerical experiment consisted of four steps. In step (i), we used the proposed clustering algorithm to determine the optimal service units for the real-world dataset RRD. Then, we compared the

performance of the proposed clustering method with that of the classical k-means method. In step (ii), we compared the performance of the proposed routing algorithm with that of GAMS (a commercial modeling and solution platform) based on cases with 13–25 service units. In step (iii), we evaluated the performance of the proposed LIPDRS on real requests and analyzed the request rejection situation as well as the economic feasibility on the real-world dataset RRD in comparison to the existing door-to-door transport option. In addition, by combining the results of this step with those of step (i), we selected the optimal plan for RRD. In step (iv), we extended the experiment to an evaluation of the performances of the proposed transport option and the existing door-to-door option on the various artificial request datasets to determine the scopes of application of the different options.

The parameters that needed to be prespecified for this experiment are listed in Table 2. In addition, we used information on the real road network in Beijing, as obtained from 'AMAP', to obtain distance measurements using the method discussed in Section 2.1.3.

Table 2. Parameter settings.

ID	Parameter Name	Value	Unit	Source
1	Maximum walking time, mwt	5	min	[59]
2	Set of candidate numbers of service units, cc	$\left\{ 30, 31, \ldots, 53, 54 \right\}$		Estimated
3	Operation time redundancy, ς	15	min	Survey
4	Threshold, ε	0.001		[54]
5	Maximum number of iterations, mi	200		[7]
6	Unit cost of a vehicle, C_V	1 [4]	unit/km	[57]
7	Fixed cost of a vehicle, C_F	90 [4]	units	[57]
8	Penalty value for canceling a request, p_r	180 [4]	units	Survey
9	Cost that passengers expect to incur for additional walking time (CNY/min), C_{PW}	5	CNY/min	Survey
10	Maximum operating distance, d_{max}	240	km	Survey
11	Maximum driving time, t_{max}	360	min	Survey
12	Vehicle capacity, Q	56		Survey
13	Fleet size, $\|M\|$	50		Setting
14	Vehicle operating speed	ESRN [3]	km/h	Digital map
15	Fixed price for using the service, P_F	1	CNY	Survey [1]
16	Variable price for using the service, P_V	0.25	CNY/km	Survey [1]
17	Generalized cost of using the service, C_G	400	CNY/h	[60]
18	Depot location [2]	Lat: 40.0521, Lng: 116.3085		Survey

[1] According to the carpooling price for Didi in Beijing. [2] We used a currently operating bus hub as the depot. [3] Expected speed in the road network (ESRN), obtained from 'AMAP', a digital map. [4] In the context of our study, we regard the travel distance as being proportional to the travel cost. However, the proportionality between travel distance and cost is not a key element of our study. Therefore, we use a conceptual unit to bypass the proportion determination problem, following reference [57]. According to survey data, the average fuel consumption of a large bus is approximately 0.3–0.35 L/km. Based on the price of diesel in Beijing, one unit corresponds to approximately 2 CNY.

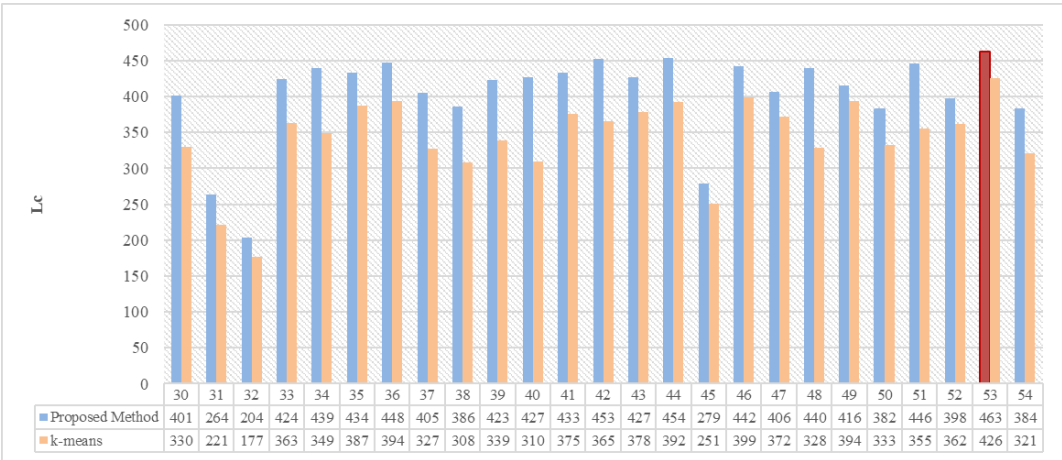
3.3. Results and Analyses

3.3.1. Results and Discussion of Step (i)

The objective function values $L(\mathrm{c})$ (Equation (1b)) corresponding to different numbers c of service units were calculated in this step via enumeration for two clustering methods (the proposed fuzzy clustering method and the classical k-means hard clustering method), and the results are shown in Figure 5. For both the proposed method and the classical method, the $L(c)$ values are high for c values of $\{36, 42, 44, 51, 53\}$ and reach a peak when c is set to 53.

Additionally, the performance of k-means is lower than that of the proposed fuzzy clustering method. For example, when c is set to 53, the $L(\mathrm{c})$ value (Equation (1b)) obtained with the proposed clustering method is 8% higher than that obtained with k-means. The difference is even larger (approximately 2%) when c is set to 40. These findings suggest that the proposed soft clustering method is superior to the hard-clustering method.

We selected the number of clusters resulting in the highest $L(\mathrm{c})$, as an example to illustrate the distribution of the service units, as shown in Figure 6. However, this is not necessarily the most appropriate number of service units. After calculating all plans in their entirety, including the service units and service routes, we can evaluate the statistical characteristics of each plan based on the metrics discussed in Section 2.5 to select the optimal number of service units. In our discussion of step (iii), we will further consider the optimal plan for RRD.



	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
Proposed Method	401	264	204	424	439	434	448	405	386	423	427	433	453	427	454	279	442	406	440	416	382	446	398	463	384
k-means	330	221	177	363	349	387	394	327	308	339	310	375	365	378	392	251	399	372	328	394	333	355	362	426	321

Figure 5. Values of $L(c)$ with various c values. This figure shows the comparison of the proposed method with the classical k-means. It can be clearly observed from the figure that the $L(c)$ values of the two methods have similar trends, but the proposed method has obvious advantages. This advantage is related to the application of the Gaussian kernel function. In addition, the figure expresses the changes in the index $L(c)$ with the value of c. When c is less than 34, the $L(c)$ value tends to decrease monotonically. When c is greater than 34, the L value fluctuates and reaches a maximum at $c = 53$.

3.3.2. Results and Discussion of Step (ii)

To validate the proposed routing method, problem instances with 13-55 service units were solved to obtain two types of solutions, SOGs and SCGs, where the SOG for a particular problem instance is the solution to *RM* (discussed in Section 2.1.3) as obtained by GAMS (a commercial solution software), whereas the SCG is the solution to the models *MP* and S_k obtained using the CG algorithm. We regard the SOG as the standard solution to the problem. By comparing the SOGs and SCGs in terms of their objective values and calculation times, we can evaluate whether the proposed method is feasible and efficient.

Figure 7 presents a comparison of the SOGs and SCGs from the perspectives of the calculation times and objective values. The proposed routing method obtains the same results as GAMS does; thus, the proposed method is feasible. Regarding the calculation time, as the number of service units increases, the calculation times for the SCGs exhibit acceptable linear growth when the number of service units is less than 35. As the number of service units increases greater than 35, the calculation time increases significantly. When the number of service units reaches 55, the calculation time is 21,612 s (approximately 6 h), which is essentially the maximum acceptable limit. In contrast, the calculation times for the SOGs indicate that GAMS cannot address large-scale problems (involving approximately 50 service units). With more than 21 service units, the SOG calculation times exhibit much more rapid growth than the SCG calculation times do; when the number of service units is 39, the calculation time

for the SOG is 33,541 s (approximately 9.3 h, at least 1.55 times as long as that of our proposed method). These observations demonstrate the effectiveness of the proposed method.

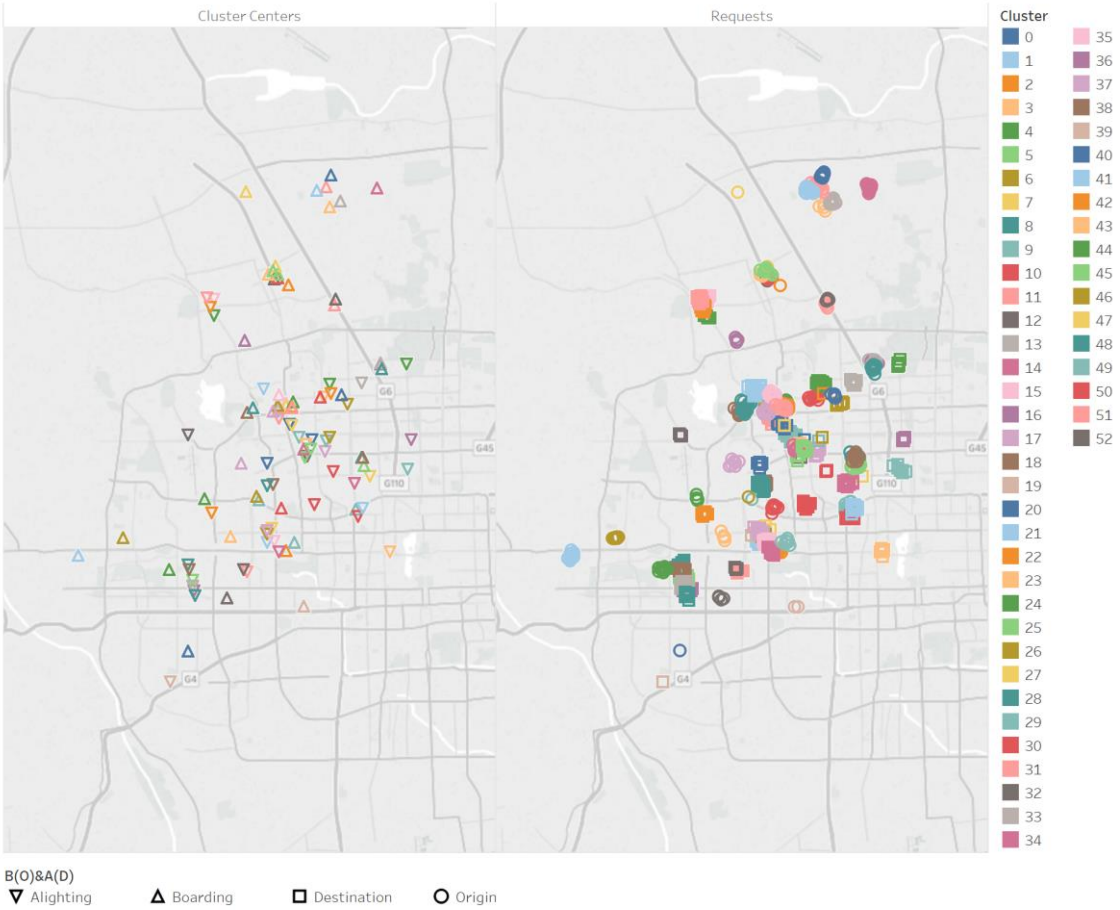


Figure 6. Distribution of 53 service units. This figure illustrates the distribution of requests and service units. To present a readable illustration of the distribution, the cluster centers are drawn on the left, and the requests are drawn on the middle, on the same space range. The legend of clusters is shown on the right. From this figure, it can be observed that the service units obtained by the algorithm are reasonable. This observation can help us set the proper service units.

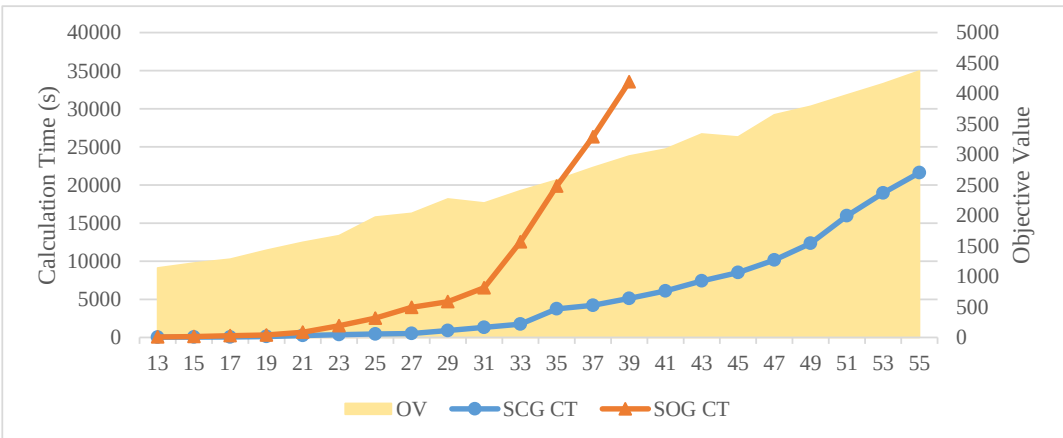


Figure 7. Comparison of SOGs and SCGs: OV represents the objective value, and CT represents the calculation time.

Now, let us consider a case with 13 service units as an example to illustrate how the branching scheme works during the FAR solution process. The edges in this case are indexed by numbers from 0 to 156. The relationship between the objective value and the iteration number throughout the entire solution process is shown in Figure 8. In the 15th iteration, the termination criterion is met. The objective value of the solution is 1097.4995 (with 10 vehicles), and the calculation time is 14,230 ms.

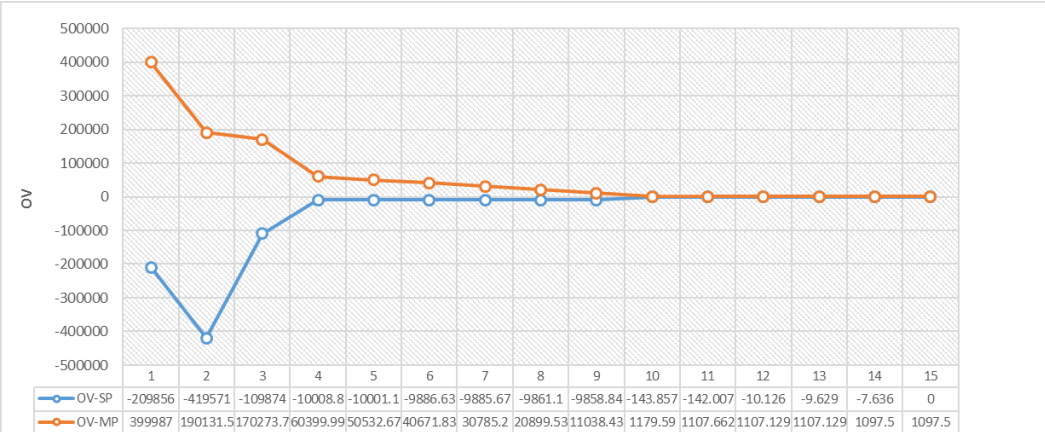


Figure 8. Convergence curve of the proposed CG-based algorithm.

Table 3 presents the details of the original solution obtained with the proposed algorithm. Routes 14, 17, and 22 are fractional; therefore, the solution is not feasible. Consequently, the branching scheme [57] must be applied to guarantee the feasibility of the final solution.

Table 3. Original solution obtained with the proposed CG-based algorithm (Routes and Edges).

Route ID	14	15	17	18	19	20	21	22	23	25	
Value	0.5	1	0.5	1	1	1	1	0.5	1	1	
Edge ID	2	101	144	111	50	98	67	108	56	142	71
Value	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
Calculation time	14,230 ms										
Objective value	1097.4995										
Iterations	15										

We choose edge 101 as the branching point. After branching, we obtain a left branch and a right branch. The left branch eliminates edges 162, 5, 23, 56, 70, 80, 162, and 193 as well as routes including those edges. By contrast, the right branch forces edge 101 to be selected. From the objective values, we find that the left branch (1139.103) is superior to the right branch (1157.919). The objective value of the left branch is equal to that of the SOG (1139.103).

3.3.3. Results and Discussion of Step (iii)

Based on the various service units generated in step (i), we further planned the routes and evaluated the results using the metrics proposed in Section 2.5 to choose the optimal service plan. The results based on different numbers of service units ($c = \{50, 51, 52, 53, 54\}$) are shown in Table 4.

From the perspective of the total passenger cost, as the number of service units increases, the walking time of passengers (and the corresponding generalized cost, c_{tw}) gradually decreases. This trend is also roughly reflected in the passengers' travel time (and the corresponding ticket cost, c_{tt}) and ride time (and the corresponding general cost due to travel time, c_{gt}), but the following difference is found: when the number of service units is 53, the passengers spend the least time and distance traveling (the corresponding c_{tt} and c_{gt} values are the smallest). The reason for this phenomenon is that

these two indicators are jointly determined by the distribution of the service units and the designed routes. In a real-world road network, even a slight change in the service units can result in a change in the routes and, thus, the costs.

From the perspective of the total fleet cost, when the number of service units is in the range of 50-54, the number of rejected requests is not high; in fact, only one request is rejected ($p_r = 180$). When the number of service units is 50-53, the number of vehicles used remains almost constant, being either 19 or 20 ($c_{ff} = 1710$ *or* 1800). When the number of clusters is 54, however, the number of vehicles used increases to 23 ($c_{ff} = 2070$), which is much greater than the numbers of vehicles used in the other cases. A similar trend is seen in terms of the variable cost (c_{vf}), and when the number of service units is 53, the cost is optimal. The reason for this finding may be that the routes corresponding to the specific distribution of the service units in the real-world road network are relatively simple when the number of service units is 53.

In summary, for RRD, the best plan is obtained when the number of service units is 53. In this plan, requesters must walk for approximately 6.8 min ($c_{tw} = 34.00$) on average, and approximately 13.43 requests are served per service unit.

Table 4. Statistical analysis of plans with $c = \{50, 51, 52, 53, 54\}$

c	Total Passenger Cost (CNY)			Total Fleet Cost (Units)		
	c_{tw} [1]	c_{tt} [2]	c_{gt} [3]	c_{ff} [4]	c_{vf} [5]	p_r [6]
50	30263.68	7152.60	208046.58	1710.00	2405.50	180.00
51	28127.17	6917.87	201217.97	1800.00	2453.49	180.00
52	26703.37	6932.35	201639.24	1800.00	2498.54	180.00
53	24211.44	6624.00	192669.20	1800.00	2389.43	180.00
54	23498.54	6829.69	198652.70	2070.00	2502.57	180.00
c	Average and Standard Deviation of Passenger Cost (CNY)			Average and Standard Deviation of Fleet Cost (units)		
	c_{tw}	c_{tt}	c_{gt}	c_{ff}	c_{vf}	p_r
50	42.51±6.15	10.05±2.78	292.20±14.89	90.00±0	126.61±21.37	180.00±0
51	39.50±4.69	9.72±3.74	282.61±12.64	90.00±0	122.67±20.14	180.00±0
52	37.51±4.05	9.74±4.29	283.21±9.96	90.00±0	124.93±18.48	180.00±0
53	34.00±5.59	9.30±2.88	270.60±12.31	90.00±0	119.47±15.12	180.00±0
54	33.00±5.08	9.59±3.21	279.01±14.87	90.00±0	108.81±14.15	180.00±0

[1] c_{tw} Walking cost, $c_{tw} = C_{PW} \cdot T_{TW}$, where C_{PW} represents the cost that passengers expect to incur for additional walking time (CNY/min) and T_{TW} represents the total walking time (min). [2] c_{tt}: Ticket cost, $c_{tt} = P_F \cdot N_S + P_V \cdot D_{TPV}$, where P_V is the unit variable price for using the service, P_F is the fixed price for using the service, N_S is the number of requests served, and D_{TPV} represents the total distance traveled by all passengers in vehicles (km). [3] c_{gt}: General cost due to travel time, $c_{gt} = C_G \cdot T_{TPV}$, where C_G represents the unified generalized cost (in terms of time and comfort) of passengers using the service and T_{TPV} represents the total time spent by all passengers in vehicles. [4] c_{ff}: Fixed cost of the fleet, $c_{ff} = N \cdot C_F$, where N is the number of vehicles used and C_F is the unit fixed cost of using a vehicle. [5] c_{vf}: Variable cost of the fleet, $c_{vf} = C_V \cdot D_{TTV}$, where C_V is the unit variable cost of using a vehicle and D_{TTV} represents the total distance traveled by vehicles (km). [6] p_r: Rejection penalty, $p_r = N_{RR} \cdot P_{RR}$, where N_{RR} is the number of rejected requests and P_{RR} is the penalty cost for rejecting a request.

3.3.4. Results and Discussion of Step (iv)

In this step, we compared the proposed LIPDRS with the existing door-to-door service (based on the model presented in [57]) on various datasets (discussed at the beginning of Section 3). The results are reported in Tables 5 and 6.

Table 5. Comparison of different transport options on various datasets (Total Value).

Dataset	Transport Option	Total Passenger Cost (CNY)			Total Fleet Cost (Units)		
		c_{tw} 1	c_{tt} 2	c_{gt} 3	c_{ff} 4	c_{vf} 5	p_r 6
RRD	LIPDRS	17409.55	4763.66	138550.25	1800.00	2389.43	180.00
	DtD	0.00	8959.04	260597.63	2790.00	4014.38	0.00
ARR	LIPDRS	58562.81	8503.62	247348.89	2520.00	3657.60	3960.00
	DtD	0.00	8635.94	251198.25	2700.00	4069.15	0.00
ARCR 25%	LIPDRS	44964.53	8327.44	242223.61	2340.00	3212.11	1800.00
	DtD	0.00	8900.32	258889.43	2610.00	3956.89	0.00
ARCR 50%	LIPDRS	31222.60	7005.99	203781.57	2160.00	3015.32	1440.00
	DtD	0.00	7901.70	229838.41	2700.00	3957.64	0.00
ARCR 75%	LIPDRS	23392.24	6682.68	194376.27	1890.00	2489.66	360.00
	DtD	0.00	9091.06	264438.16	2970.00	4014.27	0.00
ACR	LIPDRS	13457.35	6418.37	186687.16	1620.00	2401.47	180.00
	DtD	0.00	9282.13	269996.44	2610.00	3981.22	0.00

[1] c_{tw}: Walking cost, $c_{tw} = C_{PW} \cdot T_{TW}$, where C_{PW} represents the cost that passengers expect to incur for additional walking time (CNY/min) and T_{TW} represents the total walking time (min). [2] c_{tt}: Ticket cost, $c_{tt} = P_F \cdot N_S + P_V \cdot D_{TPV}$, where P_V is the unit variable price for using the service, P_F is the fixed price for using the service, N_S is the number of requests served, and D_{TPV} represents the total distance traveled by all passengers in vehicles (km). [3] c_{gt}: General cost due to travel time, $c_{gt} = C_G \cdot T_{TPV}$, where C_G represents the unified generalized cost (in terms of time and comfort) of passengers using the service and T_{TPV} represents the total time spent by all passengers in vehicles. [4] c_{ff}: Fixed cost of the fleet, $c_{ff} = N \cdot C_F$, where N is the number of vehicles used and C_F is the unit fixed cost of using a vehicle. [5] c_{vf}: Variable cost of the fleet, $c_{vf} = C_V \cdot D_{TTV}$, where C_V is the unit variable cost of using a vehicle and D_{TTV} represents the total distance traveled by vehicles (km). [6] p_r: Rejection penalty, $p_r = N_{RR} \cdot P_{RR}$, where N_{RR} is the number of rejected requests and P_{RR} is the penalty cost for rejecting a request.

Table 6. Comparison of different transport options on various datasets (Average and Standard Deviation Value).

Dataset	Transport Option	Average and Standard Deviation of Passenger Cost (CNY)			Average and Standard Deviation of Fleet Cost (units)		
		c_{tw} 1	c_{tt} 2	c_{gt} 3	c_{ff} 4	c_{vf} 5	p_r 6
RRD	LIPDRS	34.00 ± 5.59	9.30 ± 3.88	270.60 ± 12.31	90.00 ± 0	119.47 ± 15.12	180.00 ± 0
	DtD	0.00	12.58 ± 3.25	366.01 ± 11.69	90.00 ± 0	129.50 ± 16.27	0.00
ARR	LIPDRS	82.25 ± 15.14	11.94 ± 2.69	347.40 ± 18.74	90.00 ± 0	130.63 ± 12.48	180.00 ± 0
	DtD	0.00	12.13 ± 2.28	352.81 ± 21.67	90.00 ± 0	135.64 ± 14.27	0.00
ARCR 25%	LIPDRS	63.15 ± 10.48	11.70 ± 2.99	340.20 ± 18.12	90.00 ± 0	123.54 ± 13.64	180.00 ± 0
	DtD	0.00	12.50 ± 3.47	363.61 ± 18.97	90.00 ± 0	136.44 ± 13.78	0.00
ARCR 50%	LIPDRS	43.85 ± 8.79	9.84 ± 3.47	286.21 ± 15.36	90.00 ± 0	125.64 ± 14.27	180.00 ± 0
	DtD	0.00	11.10 ± 2.49	322.81 ± 21.48	90.00 ± 0	131.92 ± 14.68	0.00
ARCR 75%	LIPDRS	32.85 ± 6.01	9.39 ± 2.18	273.00 ± 12.21	90.00 ± 0	118.56 ± 14.96	180.00 ± 0
	DtD	0.00	12.77 ± 1.98	371.40 ± 13.74	90.00 ± 0	121.64 ± 17.01	0.00
ACR	LIPDRS	18.90 ± 5.57	9.01 ± 2.01	262.20 ± 14.27	90.00 ± 0	133.42 ± 14.12	180.00 ± 0
	DtD	0.00	13.04 ± 2.57	379.21 ± 20.13	90.00 ± 0	137.28 ± 15.21	0.00

[1] c_{tw}: Walking cost, $c_{tw} = C_{PW} \cdot T_{TW}$, where C_{PW} represents the cost that passengers expect to incur for additional walking time (CNY/min) and T_{TW} represents the total walking time (min). [2] c_{tt}: Ticket cost, $c_{tt} = P_F \cdot N_S + P_V \cdot D_{TPV}$, where P_V is the unit variable price for using the service, P_F is the fixed price for using the service, N_S is the number of requests served, and D_{TPV} represents the total distance traveled by all passengers in vehicles (km). [3] c_{gt}: General cost due to travel time, $c_{gt} = C_G \cdot T_{TPV}$, where C_G represents the unified generalized cost (in terms of time and comfort) of passengers using the service and T_{TPV} represents the total time spent by all passengers in vehicles. [4] c_{ff}: Fixed cost of the fleet, $c_{ff} = N \cdot C_F$, where N is the number of vehicles used and C_F is the unit fixed cost of using a vehicle. [5] c_{vf}: Variable cost of the fleet, $c_{vf} = C_V \cdot D_{TTV}$, where C_V is the unit variable cost of using a vehicle and D_{TTV} represents the total distance traveled by vehicles (km). [6] p_r: Rejection penalty, $p_r = N_{RR} \cdot P_{RR}$, where N_{RR} is the number of rejected requests and P_{RR} is the penalty cost for rejecting a request.

First, building on the results of steps (i) and (iii), we compared the plans designed for the LIPDRS and door-to-door (DtD) options. Regarding the passenger cost, LIPDRS results in a cost of approximately 160,723.46 (= 17409.55 + 4763.66 + 138550.25) CNY, and DtD results in a cost of approximately 269,556.67 (= 0 + 8959.04 + 260597.63) CNY. Obviously, from the passenger's point of view, LIPDRS costs 40.37% less than DtD. The difference in the general cost due to travel time is the main reason. To meet the LIPP in the real-world road network, the DtD service will require passengers to spend more time on vehicles after boarding, and the corresponding cumulative waiting time will constitute a significant difference in the passengers' travel expenses. Regarding the fleet cost, LIPDRS costs approximately 4369.43 units, and DtD costs approximately 6804.38 units. Thus, LIPDRS costs approximately 35.79% less than DtD. This difference is due to the establishment of the service units, which greatly reduces the transportation burden placed on the vehicles and thus reduces operating costs. Therefore, in this real-world case, the proposed transport option has an economic advantage.

Next, we analyzed the scope of application of the proposed solution by comparing the two different transport options on the artificial datasets. From ARR to ACR (including ARCR25–75%), the spatiotemporal concentration of requests gradually increases. For LIPDRS, the walking time of passengers (and the corresponding generalized cost, c_{tw}) is relatively high on ARR. In addition, the c_{tt} and c_{gt} values are high, almost the same as those for the DtD option. In contrast, DtD does not require passengers to walk. Consequently, the total passenger cost (259,834.19 CNY) is lower than that for LIPDRS (314,414.32 CNY). Moreover, LIPDRS also costs more than DtD from the perspective of the fleet cost. These results indicate that LIPDRS lacks adaptability to random and decentralized distributed requests such as those in ARR.

As the level of demand centralization increases (i.e., as the dataset changes from ARR to ARCR25%, ARCR50% and ARCR75%), the c_{tw} value of LIPDRS decreases, and the c_{tt} and c_{gt} values show the same trend. In terms of the fleet cost, LIPDRS begins to cost less than DtD with ARCR50%. At this point, the advantage of LIPDRS begins to emerge. When the ratio of requests that are spatiotemporally concentrated is above 50%, LIPDRS is the more suitable option. Therefore, we believe that the proposed LIPDRS is much more suitable for accommodating the LIPP in a metropolitan context. For example, on ARCR75%, LIPDRS can reduce the cost for passengers by approximately 17.94% and the cost for operators by approximately 32.14%.

4. Conclusions

To alleviate the problems caused by LIPP in metropolises, this paper proposes an innovative type of demand-responsive public transport service called LIPDRS. By providing an attractive transportation option for passengers who typically drive or use conventional public transport, this type of demand-responsive service can potentially reduce car usage and expand passenger accessibility in congested metropolitan areas.

To obtain the optimal operation plan for LIPDRS, we model the LIPDRS decision process using a mixed-integer programming model with dual objectives to optimize the service from the perspectives of both passengers and operators. This model includes a carefully designed passenger walking time constraint and request rejection mechanism to ensure that the resulting service plan can guarantee not only good quality of service but also improved operational efficiency. Additionally, this paper proposes a clustering-first, routing-second framework for solving the proposed model. A modified version of the fuzzy c-means clustering algorithm and a column generation algorithm are utilized in this framework to determine the service units and perform vehicle assignment and routing, respectively. In addition, to overcome the problems caused by relaxing the model variables, a branching scheme is proposed.

Based on the designed methodology, a four-step experiment was conducted to investigate different transport options on various types of datasets. The results demonstrate the advantages of the proposed method and illustrate its application scope. The contributions of this paper can be summarized as follows. (i) We propose a new transport option along with a complete system of modeling and solution methods. A maximum passenger walking time constraint, a request rejection mechanism

and a scheme for ensuring solution feasibility are all considered in this scheduling system. (ii) The proposed transport option has been proven to achieve better performance (by 40.37% for passengers and by 35.79% for operators) than the existing "door-to-door" transport option does when faced with LIPP. (iii) By testing on different datasets, we prove that the proposed service is more suitable for the request distributions that are spatiotemporally concentrated. We have found that when the ratio of requests that are spatiotemporally concentrated is above 50%, LIPDRS is the more suitable option. (iv) The performance advantages of the proposed clustering-first, routing-second solving framework have been proven through experiments. Regarding the individual components of the framework, the proposed soft clustering algorithm has been proven to be superior to the k-means hard clustering method (by 8%), and the proposed routing algorithm has been proven to be 1.5 times more efficient than the commercial solution software GAMS.

Overall, the proposed methodology has the following attractive features for solving the issues faced due to LIPP in metropolitan contexts: (i) The service considers the trade-off between operator profit and requester cost. (ii) At the expense of increasing the passengers' walking times, the proposed service can effectively reduce the passengers' ride times and the operating costs of the fleet relative to existing door-to-door services. This service is suitable for use in cases of insufficient capacity. The LIPDRS approach can also be used for the scheduling of liquefied natural gas (LNG) buses or pure electric buses. Additionally, it can be used for transport between residential areas and central business districts and for acyclic transportation related to holidays and large events. The use of this mode of transport can help to support green and sustainable metropolitan development.

In this study, only one type of vehicle is considered for LIPDRS planning. However, for a metropolis with diverse demands, this assumption is not appropriate. In future research, we will consider hybrid fleets with different types of vehicles (e.g., with different speeds and capacities) to improve both operator profits and passenger costs. In addition, the whole LIPDRS design process requires some preliminary preparation time for finding the optimal c value, which limits the capability of online implementation. Therefore, we will seek improvements in computational efficiency in future research. For example, some intelligent optimization methods [61–63] or cloud computing techniques [64] can be used to obtain solutions efficiently.

Author Contributions: Conceptualization, H.Z.; Methodology, H.Z.; Project administration, X.Z.; Resources, X.Z.; Software, H.Z.; Supervision, J.C.; Visualization, Z.Y.; Writing—original draft, H.Z.

Funding: The research is supported by the National Key R&D Program of China (Grant No. 2018YFB1201401) and the National Natural Science Foundation of China (Grant No. U1734204).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, T.; Ceder, A.; Bologna, R.; Cabantous, B. Commuting by customized bus: A comparative analysis with private car and conventional public transport in two cities. *J. Public Transp.* **2016**, *19*, 55–74. [CrossRef]
2. Zhao, P.; Hu, H. Geographical patterns of traffic congestion in growing megacities: Big data analytics from beijing. *Cities* **2019**, *92*, 164–174. [CrossRef]
3. Wu, W.; Wang, M.X.; Zhang, F. Commuting behavior and congestion satisfaction: Evidence from Beijing, China. *Transp. Res. Part D Transp. Environ.* **2019**, *67*, 553–564. [CrossRef]
4. Saneinejad, S.; Roorda, M.J.; Kennedy, C. Modelling the impact of weather conditions on active transportation travel behaviour. *Transp. Res. Part D* **2012**, *17*, 129–137. [CrossRef]
5. Lu, T.; Zhou, L.; Liu, J.; Zhou, X. Customized bus service design for jointly optimizing passenger-to-vehicle assignment and vehicle routing. *Transp. Res. Part C Emerg. Technol.* **2017**, *85*, 451–475.
6. Dumas, Y.; Desrosiers, J.; Soumis, F. *Large Scale Multi-Vehicle Dial-a-Ride Problems*; École des Hautes Études Commerciales: Paris, France, 1989.
7. Zhang, K.; Sun, D.; Shen, S.; Zhu, Y. Analyzing spatiotemporal congestion pattern on urban roads based on taxi gps data. *J. Transp. Land Use* **2017**, *10*, 675–694. [CrossRef]

8. Martínez, L.M.; Viegas, J.M.; Eiró, T. Formulating a new express minibus service design problem as a clustering problem. *Transp. Sci.* **2014**, *49*, 85–98. [CrossRef]
9. Park, J.; Kim, B.I. The school bus routing problem: A review. *Eur. J. Oper. Res.* **2010**, *202*, 311–319. [CrossRef]
10. Cain, A.; Darido, G.; Baltes, M.; Rodriguez, P.; Barrios, J. Applicability of transmilenio bus rapid transit system of bogotá, colombia, to the united states. *Transp. Res. Record.* **2007**, *2034*. [CrossRef]
11. Pan, G.; Qi, G.; Zhang, D.; Zhang, D.; Li, S. Land-use classification using taxi gps traces. *IEEE Trans. Intell. Transp. Syst.* **2013**, *14*, 113–123. [CrossRef]
12. Idris, H.; Clarke, J.P.; Bhuva, R.; Kang, L. Queuing model for taxi-out time estimation. *Air Traffic Control Q.* **2001**, *10*, 1–22. [CrossRef]
13. Lee, D.H.; Wang, H.; Cheu, R.L.; Teo, S.H. A taxi dispatch system based on current demands and real-time traffic conditions. *Transp. Res. Rec. J. Transp. Res. Board* **2004**, *1882*. [CrossRef]
14. Ritzinger, U.; Puchinger, J.; Hartl, R.F. Dynamic programming based metaheuristics for the dial-a-ride problem. *Ann. Oper. Res.* **2016**, *236*, 341–358. [CrossRef]
15. Parragh, S.N.; Jorge, P.D.S.; Almada-Lobo, B. The dial-a-ride problem with split requests and profits. *Transp. Sci.* **2014**, *49*, 311–334. [CrossRef]
16. Clarens, G.C.; Hurdle, V.F. An operating strategy for a commuter bus system. *Transp. Sci.* **1975**, *9*, 1–20. [CrossRef]
17. Larrain, H.; Muñoz, J.C. When and where are limited-stop bus services justified? *Transp. A Transp. Sci.* **2016**, *12*, 811–831. [CrossRef]
18. Ren, Y.; Chen, G.; Han, Y.; Zheng, H. *Extracting Potential Bus Lines of Customized City Bus Service Based on Public Transport Big Data*; IOP Publishing: Bristol, UK, 2016; Volume 46, p. 012017.
19. Ma, J.; Yang, Y.; Guan, W.; Wang, F.; Liu, T.; Tu, W.; Song, C. Large-scale demand driven design of a customized bus network: A methodological framework and beijing case study. *J. Adv. Transp.* **2017**, *2017*. [CrossRef]
20. Walteros, J.L.; Medaglia, A.L.; Riaño, G. Hybrid algorithm for route design on bus rapid transit systems. *Transp. Sci.* **2015**, *49*, 66–84. [CrossRef]
21. Ma, J.; Zhao, Y.; Yang, Y.; Liu, T.; Wei, G.; Wang, J.; Song, C. A model for the stop planning and timetables of customized buses. *PLoS ONE* **2017**, *12*, e0168762. [CrossRef]
22. Sun, D.J.; Xu, Y.; Peng, Z.R. Timetable optimization for single bus line based on hybrid vehicle size model. *J. Traffic Transp. Eng.* **2015**, *2*, 179–186.
23. Schittekat, P.; Kinable, J.; Sörensen, K.; Sevaux, M.; Spieksma, F.; Springael, J. A metaheuristic for the school bus routing problem with bus stop selection. *Eur. J. Oper. Res.* **2013**, *229*, 518–528. [CrossRef]
24. Shaheen, S.A. Commuter-based carsharing: Market niche potential. *Transp. Res. Rec. J. Transp. Res. Board* **2001**, *1760*, 178–183. [CrossRef]
25. Katzev, R. Car sharing: A new approach to urban transportation problems. *Anal. Soc. Issues Public Policy* **2003**, *3*, 65–86. [CrossRef]
26. Kek, A.G.H.; Cheu, R.L.; Meng, Q.; Fung, C.H. A decision support system for vehicle relocation operations in carsharing systems. *Transp. Res. Part E* **2009**, *45*, 149–158. [CrossRef]
27. Shaheen, S.A.; Cohen, A.P. Carsharing and personal vehicle services: Worldwide market developments and emerging trends. *Int. J. Sustain. Transp.* **2013**, *7*, 5–34. [CrossRef]
28. Cullen, F.H.; Jarvis, J.J.; Ratliff, H.D. Set partitioning based heuristics for interactive routing. *Networks* **1981**, *11*, 125–143. [CrossRef]
29. Bodin, L.D.; Sexton, T. The multi-vehicle subscriber dial-a-ride problem. *Tims Stud. Manag. Sci.* **1986**, *26*, 73–86.
30. Sexton, T.R.; Bodin, L.D. Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transp. Sci.* **1985**, *19*, 378–410. [CrossRef]
31. Sexton, T.R.; Bodin, L.D. Optimizing single vehicle many-to-many operations with desired delivery times: II. Routing. *Transp. Sci.* **1985**, *19*, 411–435. [CrossRef]
32. Desrosiers, J.; Dumas, Y.; Soumis, F.; Taillefer, S.; Villeneuve, D. An Algorithm for Mini-Clustering in Handicapped Transport. *Cah. Du Gerad*. 1991. Available online: https://www.researchgate.net/publication/245275603_An_algorithm_for_mini-clustering_in_handicapped_transport (accessed on 12 September 2019).
33. Ioachim, I.; Desrosiers, J.; Dumas, Y.; Solomon, M.M.; Villeneuve, D. A request clustering algorithm for door-to-door handicapped transportation. *Transp. Sci.* **1995**, *29*, 63–78. [CrossRef]

34. Pankratz, G. A grouping genetic algorithm for the pickup and delivery problem with time windows. *Or. Spectr.* **2005**, *27*, 21–41. [CrossRef]
35. Bard, J.F.; Jarrah, A.I. Large-scale constrained clustering for rationalizing pickup and delivery operations. *Transp. Res. Part B Methodol.* **2009**, *43*, 542–561. [CrossRef]
36. Qu, Y.; Bard, J.F. A grasp with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Comput. Oper. Res.* **2012**, *39*, 2439–2456. [CrossRef]
37. Masson, R.; Lehuédé, F.; Péton, O. An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transp. Sci.* **2013**, *47*, 344–355. [CrossRef]
38. Toth, P.; Vigo, D. *An Overview of Vehicle Routing Problems*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2001; pp. 1–26.
39. Toth, P.; Vigo, D. *The Vehicle Routing Problem*; Tsinghua University Press: Beijing, China, 2011; pp. 127–153.
40. Parragh, S.N.; Doerner, K.F.; Hartl, R.F. A survey on pickup and delivery problems. *Manag. Rev. Q.* **2008**, *58*, 21–51.
41. Paquette, J.; Cordeau, J.F.; Laporte, G.; Pascoal, M.M.B. Combining multicriteria analysis and tabu search for dial-a-ride problems. *Transp. Res. Part B Methodol.* **2013**, *52*, 1–16. [CrossRef]
42. Cordeau, J.F.; Laporte, G.; Potvin, J.Y.; Savelsbergh, M.W.P. Transportation on demand. *Handb. Oper. Res. Manag. Sci.* **2007**, *14*, 429–466.
43. Fisher, M.L.; Jaikumar, R. A generalized assignment heuristic for vehicle routing. *Networks* **1981**, *11*, 109–124. [CrossRef]
44. Fisher, M.L.; Jörnsten, K.O. Vehicle routing with time windows: Two optimization algorithms. *Oper. Res.* **1997**, *45*, 488–492. [CrossRef]
45. Sol, M. *Column Generation Techniques for Pickup and Delivery Problems*; Technische Universiteit Eindhoven: Eindhoven, The Netherlands, 1994.
46. Parragh, S.N.; Schmid, V. Hybrid column generation and large neighborhood search for the dial-a-ride problem. *Comput. Oper. Res.* **2013**, *40*, 490–497. [CrossRef]
47. Ropke, S.; Cordeau, J.F.; Laporte, G. *Models and Branch-and-Cut Algorithms for Pickup and Delivery Problems With Time Windows*; Wiley-Interscience: Hoboken, NJ, USA, 2007; pp. 258–272.
48. Masson, R.; Ropke, S.; Lehuédé, F.; Péton, O. A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *Eur. J. Oper. Res.* **2014**, *236*, 849–862. [CrossRef]
49. Cordeau, J.F.; Laporte, G. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transp. Res. Part B Methodol.* **2003**, *37*, 579–594. [CrossRef]
50. Stiglic, M.; Agatz, N.; Savelsbergh, M.; Gradisar, M. The benefits of meeting points in ride-sharing systems. *Transp. Res. Part B* **2015**, *82*, 36–53. [CrossRef]
51. Li, Y.; Yu, F. A new validity function for fuzzy clustering. In Proceedings of the International Conference on Computational Intelligence and Natural Computing, Wuhan, China, 6–9 June 2009; pp. 462–465.
52. Fahad, A.; Alshatri, N.; Tari, Z.; Alamri, A.; Khalil, I.; Zomaya, A.Y.; Foufou, S.; Bouras, A. A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *Emerg. Top. Comput. IEEE Trans.* **2014**, *2*, 267–279. [CrossRef]
53. Savas, E.S. On equity in providing public services. *Manag. Sci.* **1978**, *24*, 800–808. [CrossRef]
54. Bezdek, J.C.; Ehrlich, R.; Full, W. Fcm: The fuzzy c-means clustering algorithm. *Comput. Geosci.* **1984**, *10*, 191–203. [CrossRef]
55. Luxburg, U.V. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416. [CrossRef]
56. Bezdek, J.C. A convergence theorem for the fuzzy isodata clustering algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **1980**, *2*, 1–8. [CrossRef]
57. Feillet, D. A tutorial on column generation and branch-and-price for vehicle routing problems. *4OR* **2010**, *8*, 407–424. [CrossRef]
58. Dumas, Y.; Desrosiers, J.; Soumis, F. The pickup and delivery problem with time windows. *Eur. J. Oper. Res.* **1991**, *54*, 7–22. [CrossRef]
59. O'Sullivan, S.; Morrall, J. Walking distances to and from light-rail transit stations. *Transp. Res. Rec. J. Transp. Res. Board* **1996**, *1538*, 19–26. [CrossRef]
60. Kumar, C.V.P.; Basu, D.; Maitra, B. Modeling generalized cost of travel for rural bus users: A case study. *J. Public Transp.* **2004**, *7*, 4. [CrossRef]

61. Gao, K.; Cao, Z.; Zhang, L.; Chen, Z.; Han, Y.; Pan, Q.J. A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. *IEEE/CAA J. Autom. Sinica* **2019**, *6*, 904–916. [CrossRef]
62. Zhao, J.; Liu, S.; Zhou, M.; Guo, X.; Qi, L. Modified cuckoo search algorithm to solve economic power dispatch optimization problems. *IEEE/CAA J. Autom. Sinica* **2018**, *5*, 794–806. [CrossRef]
63. Bi, J.; Yuan, H.; Tan, W.; Zhou, M.; Fan, Y.; Zhang, J.; Li, J. Application-aware dynamic fine-grained resource provisioning in a virtualized cloud data center. *IEEE Trans. Autom. Sci. Eng.* **2015**, *14*, 1172–1184. [CrossRef]
64. Li, W.; Xia, Y.; Zhou, M.; Sun, X.; Zhu, Q. Fluctuation-aware and predictive workflow scheduling in cost-effective infrastructure-as-a-service clouds. *IEEE Access* **2018**, *6*, 61488–61502. [CrossRef]

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).