systems

Article

Sharing a Ride: A Dual-Service Model of People and Parcels Sharing Taxis with Loose Time Windows of Parcels

Shuqi Xue [1,*], Qi Zhang [1] and Nirajan Shiwakoti [2]

1 School of Modern Posts, Xi'an University of Posts and Telecommunications, Xi'an 710061, China; 13948114968@stu.xupt.edu.cn
2 School of Engineering, RMIT University Carlton, Carlton, VIC 3053, Australia; nirajan.shiwakoti@rmit.edu.au
* Correspondence: shuqixue@xupt.edu.cn

Abstract: (1) Efficient resource utilization in urban transport necessitates the integration of passenger and freight transport systems. Current research focuses on dynamically responding to both passenger and parcel orders, typically by initially planning passenger routes and then dynamically inserting parcel requests. However, this approach overlooks the inherent flexibility in parcel delivery times compared to the stringent time constraints of passenger transport. (2) This study introduces a novel approach to enhance taxi resource utilization by proposing a shared model for people and parcel transport, designated as the SARP-LTW (Sharing a ride problem with loose time windows of parcels) model. Our model accommodates loose time windows for parcel deliveries and initially defines the parcel delivery routes for each taxi before each working day, which was prior to addressing passenger requests. Once the working day of each taxi commences, all taxis will prioritize serving the dynamic passenger travel requests, minimizing the delay for these requests, with the only requirement being to ensure that all pre-scheduled parcels can be delivered to their destinations. (3) This dual-service approach aims to optimize profits while balancing the time-sensitivity of passenger orders against the flexibility in parcel delivery. Furthermore, we improved the adaptive large neighborhood search algorithm by introducing an ant colony information update mechanism (AC-ALNS) to solve the SARP-LTW efficiently. (4) Numerical analysis of the well-known Solomon set of benchmark instances demonstrates that the SARP-LTW model outperforms the SARP model in profit rate, revenue, and revenue stability, with improvements of 48%, 46%, and 49%, respectively. Our proposed approach enables taxi companies to maximize vehicle utilization, reducing idle time and increasing revenue.

Keywords: Share-a-Ride problem; ride-hailing; ALNS; route planning; sharing taxi

check for updates

Citation: Xue, S.; Zhang, Q.; Shiwakoti, N. Sharing a Ride: A Dual-Service Model of People and Parcels Sharing Taxis with Loose Time Windows of Parcels. *Systems* **2024**, *12*, 302. https://doi.org/10.3390/systems12080302

Academic Editor: Mahyar Amirgholy

Received: 19 July 2024
Revised: 9 August 2024
Accepted: 10 August 2024
Published: 14 August 2024

1. Introduction

The rapid development of the logistics industry has catalyzed urban expansion yet concurrently caused environmental and socio-economic detriments, such as increased greenhouse gas emissions and severe traffic congestion [1] Long et al. (2018) [2]. Consequently, the quest for innovative approaches in urban logistics operations aimed at improving efficiency while reducing its negative impacts on urban infrastructure and the environment has emerged as a focal point of the research field. The ride-sharing model has come into the spotlight under such a backdrop. Numerous studies [1–3] have found that shared transportation helps reduce carbon emissions, decreases personal car ownership, and reduces the willingness to purchase new cars. It contributes to environmental sustainability. As a flexible transportation service, shared transportation enhances urban mobility while addressing the first-mile and last-mile shortcomings of public transit. Economically, shared transportation also plays a significant role. It reduces travel costs for residents and alleviates traffic pressure costs for the government. The Share-a-Ride problem of passengers and freight model is thus proposed as a promising solution in the form of integrating passenger and freight transport resources to maximize the utilization of urban passenger transport resources, such as taxis [4] and subways [5], to serve the urban freight.

Li et al. [4] initially introduced the concept and developed a mathematical model for a passenger and parcel Share-a-Ride system (SARP), in which both passengers and parcels are transported concurrently in the same taxi. To preserve the service quality of passengers, they considered two provisions in the model: (a) there is an upper limit on the number of parcels transported within the passenger service time; (b) a taxi cannot simultaneously serve two passengers' orders at different destinations. These provisions respectively safeguard the travel impact and personal safety of passengers enjoying shared services. Based on these provisions, they furtherly proposed a Freight in Passengers Problem (FIP). The main difference between the two models is that all requests are flexible in the SARP, but in the FIP, the routes are partially fixed beforehand based on passenger requests [4], which is expected to alleviate urban traffic congestion while providing new revenue streams for taxi companies. As research on the passenger and parcel Share-a-Ride problem deepened, Beirigo et al. [6] and van der Tholen et al. [7] later removed these two conditions in subsequent studies, making the problem more realistic. Beirigo et al. [6] focuses on modeling a variation of the people and freight integrated transportation problem (PFIT problem) in which both passenger and parcel requests are pooled in mixed-purpose compartmentalized SAVs (shared autonomous vehicles). To adapt to various application scenarios, Enzi et al. [8] made improvements to the SARP, allowing parcels to be temporarily stored in the passenger compartment with passenger consent, thereby achieving more efficient utilization of vehicle capacity. To facilitate the practical application of this model, Ghilas et al. [9] considered the stochastic nature of delivery locations and passenger boarding times in the model. Ren et al. [10] proposed a dynamic routing optimization model of the Share-a-Ride problem by continuously updating parcel information in real time, and the model was resolved with an improved genetic algorithm (GA).

In addition to taxis and online car-hailing, which provide dual-service of passenger and freight, the Freight on Transit (FOT) problem has also gained attention. This method substantially employs subway systems as the primary conveyance for improved passenger and parcel integration [5,11,12]. Ye et al. [11] introduced a last-mile delivery model that uses subways as a backbone network completed by automated service points, crowd-shipping, and backup transfers with zero-emission vehicles to optimize urban delivery efficiency. The model is formulated as a two-stage stochastic problem and resolved with a branch-and-price algorithm. In the work of Azcuy et al. [13], they considered utilizing more widely available urban public transport systems, such as buses and trams. In this model, parcels are transferred from public transit vehicles to last-mile delivery vehicles at designated stations. Traditional fixed-route public transport modes cannot facilitate door-to-door delivery; hence, auxiliary vehicles are typically used for last-mile delivery, with the route planning of these auxiliary vehicles often regarded as a variant of the Pick-up and Delivery Problem (PDP). Pimentel et al. [14] constructed a Mixed-Integer Programming (MIP) model aiming to minimize delivery time, allowing parcels to be unloaded at any bus station but only loaded at specific bus stations. Ehmke et al. [15] building upon Pimentel's research, allowed parcels to be loaded or unloaded at any bus station, enhancing the practical feasibility of the Freight on Transit (FOT) problem. Lu et al. [16] first explored the application of mixed fleets consisting of electric and gasoline vehicles in carrying passengers and transporting parcels. They introduced a grid-based mathematical heuristic algorithm for solving the vehicle scheduling problem for large-scale fleets of electric and gasoline vehicles.

Currently, solution methods for SARP include commercial solvers handling the problem, exact algorithms [17,18], and heuristic algorithms. Researchers have explored the capabilities of CPLEX and Gurobi commercial solvers for small-scale instances of the SARP, discovering that these solvers were effective for limited scopes [4,12,19–22]. As research advances, to enable the SARP model to handle larger-scale instances and be better applied in real-life scenarios, researchers have begun to utilize heuristic algorithms to solve SARP problems. Li et al. [23] proposed an Adaptive Large Neighborhood Search (ALNS) heuristic algorithm to address SARP and compared it to a Mixed Integer Programming (MIP) solver on test instances. The heuristic algorithm outperforms the MIP solver in both solution time and solution quality, especially when CPU time is limited. Beirigo et al. [6] implemented various heuristic algorithms, including genetic algorithms and simulated annealing, to address SARP. Hosni et al. [24] approached the taxi-sharing problem by formulating it as a mixed-integer programming problem and demonstrated that the application of Lagrangian decomposition methods coupled with heuristic algorithms yielded superior results compared to conventional techniques. Lu et al. [16] were the first to consider the application of mixed fleets comprising electric and gasoline vehicles in carrying passengers and transporting parcels. The study proposed a grid-based mathematical heuristic algorithm for solving the vehicle scheduling problem for large-scale fleets of electric and gasoline vehicles. In addition to heuristic algorithms, exact algorithms are also applied to solve the SARP problem. Han et al. [25] proposed an exact solution for the shared ride problem. Firstly, all feasible trips for passengers and parcels are generated efficiently through enumeration. Then, the ε-constraint method is employed to identify all Pareto optimal solutions for the bi-objective problem. This method is not only exact but also transforms the NP-hard problem into a simple vehicle-trip matching problem.

The mode and solution methods for the SARP problem are shown in Table 1.

Table 1. Relevant literature on SARP and its extensions.

Study	Sharing Mode	Transportation Means	Method Type
[4,10,11]	Allow passenger–parcel sharing and parcel–parcel sharing.	Taxis	Exact and heuristic
[5,7,13]	Allow passenger–parcel sharing and parcel–parcel sharing.	Subways	Heuristic
[6,14]	Allow passenger–parcel sharing and parcel–parcel sharing.	SAVs (shared autonomous vehicles)	Exact
[17]	Allow passenger–parcel sharing and parcel–parcel sharing.	Bus	Exact and heuristic
[18]	Allow passenger–parcel sharing and parcel–parcel sharing.	A mixed fleet of electric and gasoline vehicles	Heuristic

In summary, current research primarily focuses on dynamically responding to both passenger and parcel orders, typically by initially planning passenger routes and then dynamically inserting parcel requests. However, in practice, parcel delivery offers greater time flexibility compared to passenger transport. For instance, working individuals generally collect their parcels from courier stations after work in the evening. Because they do not have particularly strict requirements regarding the time window of the parcels, they only hope that the quality of the parcels can be ensured. In light of this, this paper considers the passenger and parcel Share-a-Ride problem under loose time window constraints. In this model, taxis visit parcel centers before the start of work to load a number of parcels and pre-plan an initial route for parcel delivery. At working time, while primarily serving dynamic passenger travel requests, taxis also deliver parcels, ensuring that passenger service quality is not compromised. The parcel delivery operates within a loose service time window; taxis are only required to deliver all carried parcels to their destinations before the end of the workday. This approach significantly reduces the potential delays in passenger service due to parcel delivery within the shared passenger and cargo transport problem while also providing additional revenue for taxis and alleviating the impact of urban logistics on road traffic. Therefore, the problem addressed in this paper is defined as the passenger and parcel-sharing taxis problem with loose time windows for parcels (SARP-LTW). This operating mode enhances the flexibility of traffic flow, increases taxi operational revenue through the integration of transportation resources, and improves the diversity of logistics services [26,27]. To tackle this problem, the study has developed an improved Adaptive Large Neighborhood Search algorithm based on ant colony pheromone, designed to achieve optimal delivery solutions more efficiently.

2. Problem Description

In this section, we will first present the illustration of the proposed problem, followed by a definition of the problem. The model assumptions and the mathematical formulation of the SARP-LTW model are then explained.

2.1. Subsection Illustration of the Proposed Problem

The SARP model significantly enhances the utilization efficiency of taxi resources, increases taxi operational revenue, and alleviates the pressure of urban logistics distribution. Within the SARP framework, taxis provide dual services to passengers and parcels under a series of constraints based on their requests. Taxis generally proactively respond to passenger requests, plan corresponding passenger service routes, and may incorporate parcel delivery during the journey. If the parcel delivery request aligns with the planned passenger service routes, they might be considered for "ride-sharing" with the passenger service. However, the decision to accept/refuse parcel delivery requests is constrained by passenger preferences, as completing parcel deliveries during passenger service may cause delays to the passengers' travel. If the incurred delay exceeds the passengers' acceptance threshold, the system may reject the parcel delivery request or need to provide passengers with certain subsidy incentives [6].

Figure 1 illustrates the operational mode of SARP, showcasing these decision processes. Parcel 1 is completed after serving Passenger 1 without delay to passenger 1, while Parcel 2 is handled during the service of Passenger 2, causing a certain delay to Passenger 2's journey.

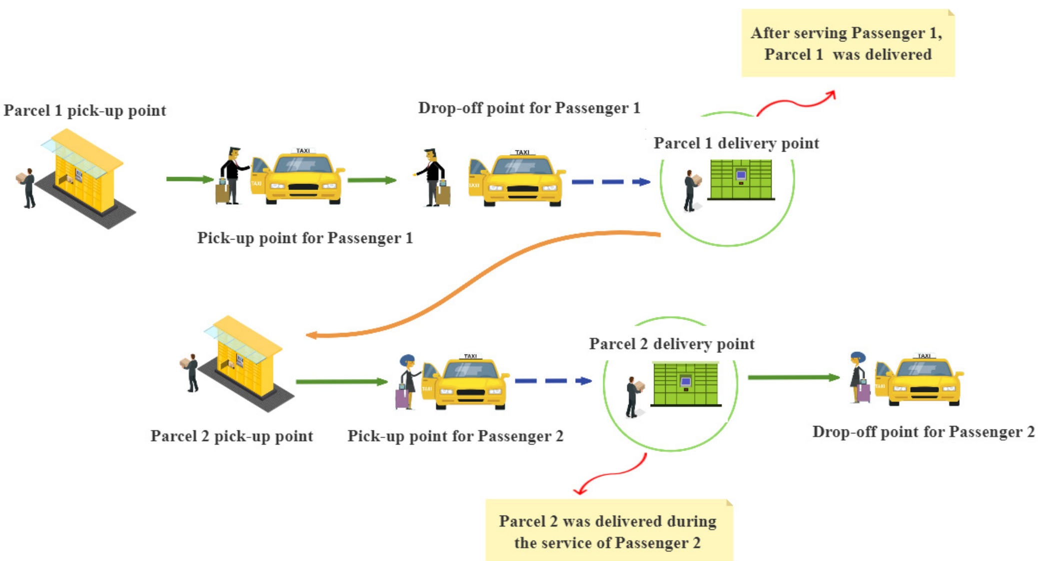


Figure 1. The Share-a-Ride problem.

In scenarios where both passenger travel requests and parcel delivery requests are dynamically updated simultaneously, the complexity of the model and the time required for practical solutions will significantly increase, especially when applied to real-life large-scale instances. Hence, developing new models for implementing SARP with reduced complexity is currently the most promising and worthwhile research direction and consideration. By considering the differences in parcel delivery timeliness and passenger travel timeliness, this paper proposes a novel model of shared-passenger–parcel travel. In this model, the timeliness requirement for parcel delivery is lower than that for passenger travel. Taxis primarily prioritize responding to passenger travel requests during their daily operations, while parcel deliveries only need to be completed before the end of the workday by the taxis, thus not hindering people from retrieving parcels after their work in the evening. Under this model, taxis pick up a certain quantity of parcels at the parcel center before the start of work and predefine some initial parcel delivery routes. Upon commencement of work the next day, taxis prioritize serving passenger travel orders and strive to complete parcel deliveries while minimizing or avoiding passenger travel delays as much as possible. Based on this model, compared to simultaneously dynamically responding to passenger travel requests and parcel delivery requests, the complexity of the problem and the solution algorithms will be significantly reduced while still reaping the benefits of shared travel. Figure 2 illustrates an example of the proposed SARP-LTW.

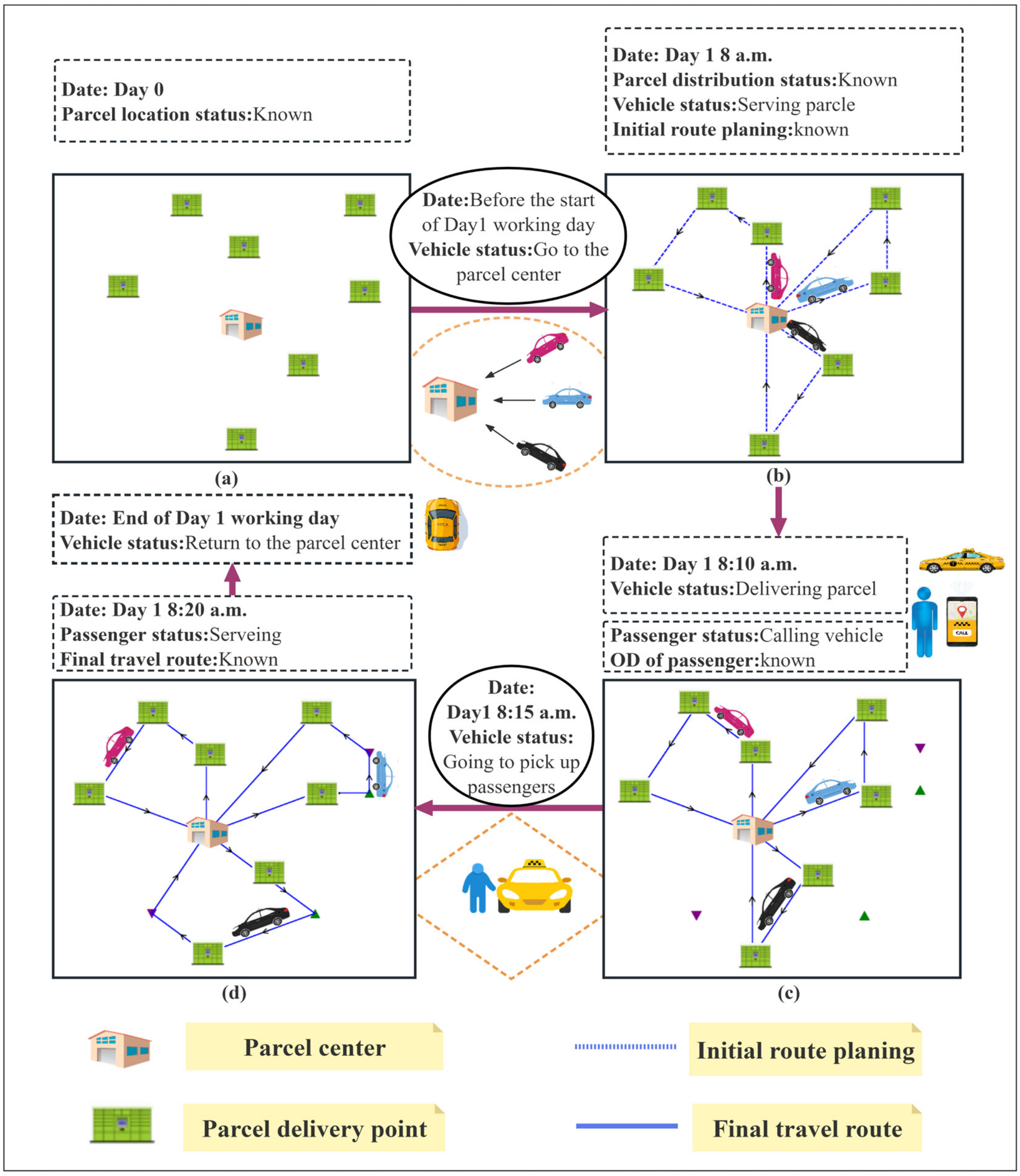


Figure 2. An illustrative example of the SARP-LTW at different times. (**a**) The quantity and location distribution of parcels; (**b**) The initial delivery routes for parcels; (**c**) Taxis deliver parcels and are ready for passenger travel requests. (**d**) Taxis adjust the routes to pick up the passenger.

It is assumed that the parcel center starts operating at 8 a.m. every day. Below is a storyline of Figure 2.

Day 0: At the end of working day 0, taxis receive parcel order requests with certain numbers and destinations, as shown in Figure 2a.

Before 8 a.m. on Day 1: Taxis arrive at the parcel center before the start of work to collect parcels from the parcel center, completing the pick-up process of parcels. In addition, an initial delivery route for parcels will be generated based on a Greedy routing method, as shown in Figure 2b.

At 8 a.m. on Day 1: Taxis begin to deliver the first parcel and, at the same time, will be ready for passenger travel requests.

8:15 a.m. on Day 1: The taxi receives its first passenger travel request on the way to the destination of the parcels; taxis will adjust the route to pick up the passenger. For example, blue and black taxis, while the pink taxi continues to follow the initial parcel delivery route.

8:20 a.m. on Day 1: Taxis are serving both parcel and passenger orders simultaneously and prioritize passenger orders. However, if taxis pass by a parcel destination point during the passenger service route, they may deliver parcels before passengers alight, thus incurring additional costs due to the detour for passengers.

In summary, the people and parcel-sharing taxi problem with loose time windows of parcels can be described as vehicles planning delivery routes for pre-booked parcel delivery requests from the previous working day and providing delivery services on the current working day. During parcel delivery, if passenger orders arise, taxis will re-plan routes and accommodate passenger boarding requests. This Share-a-Ride model reduces vehicle idle time and achieves cost savings and efficiency gains.

2.2. *Definition of the Problem*

Below, we present a detailed description of the passenger and parcel-sharing taxis' problem with loose time windows of parcels.

There exists a designated parcel center denoted as O and K taxis of uniform capacity available for parcel and passenger service. There are N passenger and M parcel orders that need to be served. The information for M parcel orders is with the same pick-up point (parcels center) and random destinations (delivery points), while the information on N passenger orders is stochastic with their boarding and drop-off points. In addition, for each passenger order, there are two requests: a boarding request and a drop-off request. For each parcel order, there are two requests: a pick-up request and a delivery request. We will use "requests" to fulfill them with the total number of requests (including both parcels and passengers) denoted as σ.

In addition to passenger boarding requests, each request has its own service time window $[e_i, l_i]$. If the related service is completed outside the time window, the time window cost for the vehicle will increase. For passenger drop-off requests, the upper time window is defined in this paper as allowing a delay of 10 min beyond the estimated arrival time based on the vehicle's navigation. The lower time window is set as 10 min earlier than the estimated arrival time. Assuming the vehicle's navigation estimates the arrival time at the destination as ϵ, the time window for passenger drop-off is represented as: $[\epsilon - 10, \epsilon + 10]$. The time window for parcel pick-up is relatively flexible, with vehicles only required to collect parcels from the parcel center before the start of the working day. As for the time window for parcel delivery, as long as the vehicle delivers the parcels to the designated delivery points before the end of the current working day, it does not violate the time window requirement.

The SARP-LTW can be described in an ordered graph $G = (V, E)$ with V referred to as vertices and E representing the set of edges between the vertices. Here $V = V^p \cup V^g \cup \{0, 2\sigma + 1\}$. V^p and V^g respectively represent passenger and parcel nodes, and both 0 and $2\sigma + 1$ represent the parcel center. All taxis depart from the parcel center and return to it before the start of the next working day.

2.3. Model of SARP-LTW

The focal point of our investigation in the SARP-LTW lies in optimizing taxi operations. We propose a strategy where the predefined parcel routes are dynamically adjusted to accommodate real-time passenger demands. Taxis undertake parcel pick-ups once and facilitate multiple deliveries thereafter. Our aim is to devise routes that maximize taxi revenue while adhering to the following assumptions:

(a) When simultaneously carrying parcels and passengers, passenger requests are given higher priority than parcels. When taxis depart from the parcel center, parcels are already loaded into the taxis. Parcels only need to be delivered to the designated delivery points before the end of the taxi working day, for example, 6:00 p.m.
(b) Each parcel order or passenger order is assigned to only one taxi to serve, prohibiting transfers between vehicles mid-service.
(c) Vehicles are allowed to at most merge up to two groups of passenger orders simultaneously. For example, while passenger 1 is en route from the origin to the destination, another passenger request is allowed for a ride-sharing arrangement.
(d) The start of work for taxis is at 8:00 a.m. from the parcel center, and they need to return to the parcel center at 6:00 p.m.

Table 2 presents the variables and parameters of the model developed in this study.

The model formulation for SARP-LTW:

Model for initial parcel delivery solution:

$$min\left[\sum_{i\in V}\sum_{j\in V}\sum_{k\in K} d_{ij}X_{ij}^{k}\right] \tag{1}$$

Subject to:

$$\sum_{j\in V}\sum_{k\in K} X_{ij}^{k} \leq 1, \forall i \in V^{g} \tag{2}$$

$$\sum_{i\in V} X_{ij}^{k} = \sum_{i\in V} X_{ij+\sigma}^{k}, \forall j \in V^{g,0}, k \in K \tag{3}$$

$$\sum_{i\in V} X_{0,i}^{k} = \sum_{i\in V} X_{i,2\sigma+1}^{k} = 1, \forall k \in K \tag{4}$$

$$\sum_{j\in V} X_{ij}^{k} = \sum_{j\in V} X_{ji}^{k}, \forall i \in V^{g}, k \in K \tag{5}$$

$$\sum_{k=1} y_{\mathrm{i}}^{k} = 1, \forall i \in V^{g,d}, k \in K \tag{6}$$

$$T_{k} \leq \tau_{2\sigma+1}^{k} - \tau_{0}^{k}, k \in K \tag{7}$$

$$max\{0, q_{i}\} \leq \mathrm{w}_{i}^{k} \leq min\{Q_{k}, Q_{k}+q_{i}\}, \forall i \in V, k \in K \tag{8}$$

$$X_{ij}^{k} \in \{0,1\} \tag{9}$$

Model for share a ride problem:

$$max\left[\left(\sum_{i\in V^{p,o}}\sum_{j\in V}\sum_{k\in K}(\alpha+\gamma_{1}d_{i,i+\sigma})X_{ij}^{k} + \sum_{i\in V^{f,o}}\sum_{j\in V}\sum_{k\in K}\left(\beta+\gamma_{2}d_{i,i+\sigma}+q_{i}^{g}*2\right)X_{ij}^{k} - \gamma_{3}\sum_{i\in V}\sum_{j\in V}\sum_{k\in K} d_{ij}X_{ij}^{k} - \gamma_{4}\sum_{i\in V^{p,o}}\sum_{j\in V}\sum_{k\in K}\left(\frac{\Delta d_{\mathrm{ij}}^{k}}{d_{\mathrm{ij}}^{k}}-1\right)\right)\right] \tag{10}$$

Subject to:

$$\sum_{j\in V}\sum_{k\in K} X_{ij}^{k} \leq 1, \forall_{i} \in V^{p,0} \cup V^{g,0} \tag{11}$$

$$\sum_{i\in V} X_{ij}^{k} = \sum_{i\in V} X_{ij+\sigma}^{k}, \forall_{j} \in V^{p,0} \cup V^{g,0}, k \in K \tag{12}$$

$$\sum_{i\in V} X_{0,i}^{k} = \sum_{i\in V} X_{i,2\sigma+1}^{k} = 1, \forall_{k} \in K \tag{13}$$

$$\sum_{i \in V} X_{i,0}^k = \sum_{i \in V} X_{2\sigma+1,i}^k = 0, \forall_k \in K \tag{14}$$

$$\sum_{j \in V} X_{ij}^k = \sum_{j \in V} X_{ji}^k, \forall_i \in V^p \cup V^g, k \in K \tag{15}$$

$$\tau_j^k = \left(\tau_i^k + t_{ij}\right) X_{ij}^k,\ \forall_{i,j} \in V, k \in K \tag{16}$$

$$w_j^k = \left(w_i^k + q_j\right) X_{ij}^k,\ \forall_{i,j} \in V, k \in K \tag{17}$$

$$r_i^k = \tau_{\sigma+i}^k - \tau_i^k, \forall_i \in V^{p,0} \cup V^{g,0}, k \in K \tag{18}$$

$$T_k \leq \tau_{2\sigma+1}^k - \tau_0^k, k \in K \tag{19}$$

$$e_i \leq \tau_i^k \leq l_i, \forall_i \in V, k \in K \tag{20}$$

$$t_{i,\sigma+i} \leq r_i^k \leq \varpi_i, \forall_i \in V^{p,0}, k \in K \tag{21}$$

$$t_{i,\sigma+i} \leq r_i^k \leq \varpi_i, \forall_i \in V^{g,o}, k \in K \tag{22}$$

$$max\{0, q_i\} \leq W_i^k \leq min\{Q_k, Q_k + q_i\}, \forall_i \in V, k \in K \tag{23}$$

$$P_{i+\sigma} - P_i - 1 \leq \eta, \forall_i \in V^{p,0} \tag{24}$$

$$X_{ij}^k \in \{0,1\}; \tau_j^k, w_j^k, r_i^k \in R_+; P_i \in [0, 2(m+n)] \tag{25}$$

Equations (1)–(8) define the model for the initial parcel delivery solution for taxis. (Equivalent to the Capacitated Vehicle Routing Problem model). Equation (1) is the objective function, which minimizes the total vehicle delivery distance. Equations (2) and (3) specify that each parcel can only be serviced by one taxi. Equation (4) ensures that taxis start and end at the parcel center. Equation (5) ensures the balance of path flow. Equation (6) ensures that all parcels are served. Equation (7) guarantees the maximum driving time of the taxi. Equation (8) represents the capacity constraints of the taxi. Equation (9) defines the decision variables.

Equations (10)–(25) pertain to the SARP model. The objective function formulated in Equation (10) is designed to optimize the profit of taxis by considering four components. The first component represents the revenue generated from passenger orders, including the taxi flag-down fare and mileage fee charged per unit of travel distance. The second part accounts for the revenue from parcel orders, including the base fee (analog to the taxi flag-down fare for passenger orders), parcel mileage fee and overweight fee. The third part quantifies the operational cost associated with the distance traveled by taxis, and the fourth part captures penalty costs associated with passenger orders incurred by any necessary detours for parcel orders. Notably, while the mileage fee for passenger orders is computed based on the actual travel distance, for parcel orders, it is determined by the Euclidean distance from the parcel center to the delivery point due to its loose time windows because calculating the cost of parcel delivery based on the actual delivery distance is an unreasonable billing method for users. In our model, passengers have a higher priority compared to parcels, and taxis only need to deliver parcels to the drop-off point before the end of their shift. If the cost is calculated based on the actual delivery distance, the parcel user would bear a significant and unreasonable detour cost. As shown in Figure 3 below, the cost incurred when the taxi services delivery point 5, based on the actual delivery distance (128.73), is 71.33 more than the cost calculated using the Euclidean distance (57.24). Therefore, the Euclidean distance was considered for the cost of parcel delivery.

Table 2. Description of variables and parameters.

Sets	
n	Number of passengers
m	Number of parcels
V_p	Set of passenger stops
V_g	Set of parcel stops
V	$= V^p \cup V^f \cup \{0, 2\sigma+1\}, 0$ and $2\sigma+1$ represent the parcel center
$V^{\mathrm{p,o}}$	Set of passenger origins $V^{\mathrm{p,\ o}} = \{1, 2, \ldots, n\}$
$V^{\mathrm{p,d}}$	Set of passenger destinations $V^{\mathrm{p,\ d}} = \{\sigma+1, \sigma+2, \ldots, \sigma+n\}$
$V^{g,\mathrm{o}}$	Set of parcel origins $V^{g,\mathrm{o}} = 0$
$V^{g,\mathrm{d}}$	Set of parcel destinations $V^{g,\ \mathrm{d}} = \{\sigma+n+1, \sigma+n+2, \ldots, 2\sigma\}$
H_k	Set of passengers served by taxi k, $H_k = \{1, 2, \ldots h_k\}$
C	Set of pairs (i, j), which (i, j) define a pair of subsequently served requests
Parameters and constants	
q_i^g	Load of parcel i
d_i	Distance from the origin to the destination for the request i, i.e., distance between stops i and $i+\sigma$
$[e_i, l_i]$	Time window for request i
Q_k	Parcel capacity of taxi k
T_k	Maximum duration time for taxi k
h_k	Number of passenger orders served by taxi k
η	Maximum number of requests between one passenger service
d_{ij}	Distance between stops i and j
t_{ij}	Travel time between stops i and j
Δd_{ij}^k	Extra travel distance for taxi k if parcel is delivered between passengers i and $i+\sigma$
Δt_{ij}^k	Extra travel distance for taxi k if parcel is delivered between passengers i and $i+\sigma$: $\Delta t_{\mathrm{ij}}^k = \Delta d_{\mathrm{ij}}^k / \text{average speed}$
ϖ_i	Maximum delivery time for request i
α	Flag-down fare for passenger service
β	Base fare for parcel service
γ_1	Fare charged for delivering one passenger per kilometer
γ_2	Fare charged for delivering one parcel per kilometer
γ_3	Average cost per kilometer for delivering requests
γ_4	Discount factor for exceeding the expected time of passengers
r_i^k	Time spent by request i in taxi k, $r_i^k = \tau_{i+\sigma}^k - \tau_i^k, i \in C$
Auxiliary variables	
τ_i^k	Timepoint when taxi k arrives at stop i
P_i	Index of request i in a service sequence of a taxi
w_i^k	Load of taxi k after visiting stop i
Decision variables	
x_{ij}^k	Binary decision variables equal to 1 if taxi k goes directly from node i to stop j
y_{i}^k	Binary decision variables equal to 1 if taxi k visits stop i

Equations (11) and (12) indicate that each passenger order or parcel order can only be served by one taxi once. Equations (13) and (14) specify that taxis are required to depart from the parcel center and return to it before the start of the next working day. Equation (15) enforces the conservation of taxi flow throughout the network. Equations (16) and (17) respectively calculate the arrival time at point j and the load weight upon departure from j for each taxi. Equation (18) represents the time required for each taxi service request; Equation (19) imposes the constraint on the maximum driving time for vehicles; Equation (20) represents the drop-off time window for passenger orders. Equation (21) ensures that passengers must be served before the upper limit of their drop-off time window, while Equation (22) ensures that parcels must be delivered to their delivery points before the end of the working day. Equation (23) represents the maximum parcel capacity constraint for taxis; Equation (24) specifies that for each taxi, during the fulfillment of passenger requests, it can manage up to η requests from the time of boarding to the point of drop-off (as illustrated in Figure 3), thereby ensuring the priority of passenger service. Equation (25) represents the decision variables.

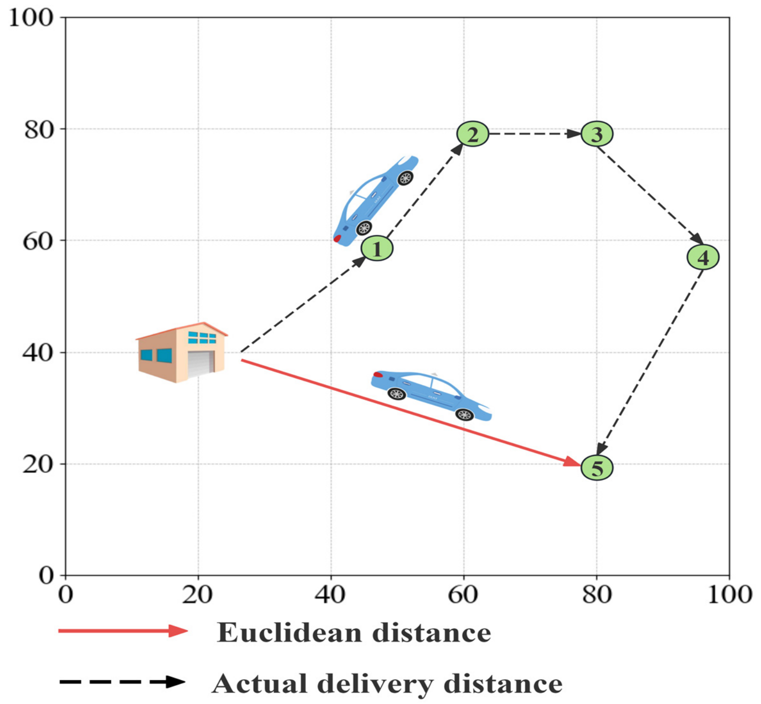


Figure 3. Comparison of Euclidean distance and actual delivery distance.

It is worth noting that in the parameter description, p_i is index the position of passengers. As shown in Figure 4, when the taxi picks up a passenger i from point 4, it will serve other requests during the journey to destination position 8. For example, a taxi may drop off a parcel at position 5, pick up a passenger order $i+1$ at position 6, and deliver the passenger order $i+1$ to the destination at position 7 during this period.

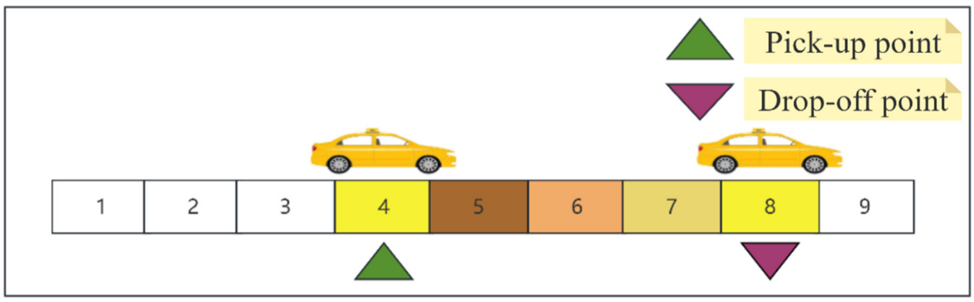


Figure 4. Illustrative diagram of vehicle position index.

In summary, while the taxi is serving the passenger i, it can also attend to other requests, but the total number of other requests served can not exceed η.

3. The Algorithm Design

In this section, we will first introduce the problem-solving strategy for the proposed problem, followed by a description of the algorithms used to solve SARP-LTW, including the execution process of the algorithms.

3.1. The Problem-Solving Strategy

In this study, we design a two-stage strategy to solve the SARP-LTW. In the first stage, an initial parcel service route is determined utilizing the Greedy algorithm, and in the second stage, an improved ALNS algorithm based on ant colony optimization is developed to find the optimal route for the dual service of passengers and parcels. The problem-solving process is illustrated in Figure 5.

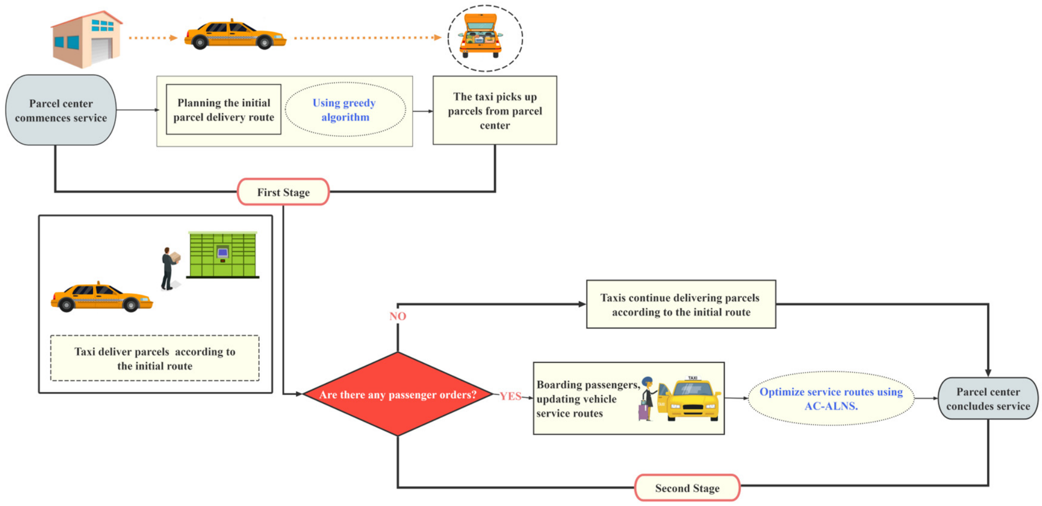


Figure 5. Problem-solving process flowchart.

3.2. Greedy-AC-ALNS Algorithm

ALNS algorithm is an improved large-scale neighborhood search (LNS) algorithm that dynamically adjusts the weights for different destroy and repair operators based on the performance of each operator. When applied to the SARP-LTW, it was found that due to the randomness of the destroy operator, it might remove the nearest adjacent nodes in a segment of the path for repair. This might be disadvantageous to problem resolution, especially for small-scale instances with fewer customer points. Therefore, to prevent the ALNS algorithm from indiscriminately disrupting the nearest adjacent nodes, we introduce a pheromone updating mechanism from the ant colony optimization algorithm to improve the ALNS algorithm, named the Greedy-AC-ALNS (Greedy Adaptive Large Neighborhood Search Algorithm based on Ant Colony Optimization) algorithm. In Greedy-AC-ALNS, pheromones serve as an indicator of path quality, with their concentration of pheromones inversely proportional to the distance between two adjacent nodes. The pheromone levels are dynamically adjusted throughout the search process to guide the algorithm towards more efficient paths. Details of the pheromone dynamics involving concentration and pheromone concentration matrix (p_c and p_m), evaporation rate (E_r) and increments are (d_r), alongside the representation of adjacent nodes as p_1 and p_2 in each path segment are summarized in Algorithm 1.

Algorithm 1 The process of updating pheromones

1: **Input Parameters:** P_c, P_m
2: **Set:** $E_r;\ d_r;\ p1;\ p2$
3: **for** each path n in initial_paths **do**
4: **for** each pair (p_i, p_{i+1}) in path **do**
5: $p_c\left[p_i, p_{i+1}\right]_{(t+1)} += p_{c(t)} * E_r + d_r / distance$
6: **end for**
7: **end for**
8: **Output:** The node pair $(p1,\ p2)$ with the highest concentration of pheromones in each path segment

The basic idea is to construct an initial solution composed of multiple sub-paths using the Greedy algorithm. The concentration of pheromones for each adjacent node pair in each path segment will continue to be updated until all nodes are traversed. At time $t + 1$, the concentration of pheromones is the concentration at the time t multiplied by the evaporation rate plus the increment of pheromones, as explained in the fifth line. This process continues until the node pair with the highest concentration of pheromones is found in each path segment. This also explains why the concentration of pheromones is inversely proportional to the distance between nodes. Since the node pair with the highest concentration of pheromones corresponds to the two nodes with the shortest distance in the sub-path, to ensure a better optimization result and minimize vehicle costs, it is necessary to maintain the relative positions of these two nodes. Subsequently, using a roulette wheel selection mechanism, a pair of destruction and repair operators are applied to the initial solution to create a new solution. The proposed SARP-LTW model is also suitable for solving using commercial solvers. However, as in the experiments of Li et al. (2014) [4], when the number of requests was 12, it took nearly 1.7 h for a commercial solver to obtain results. Therefore, considering the computation efficiency, the heuristic algorithm was designed in this study. Algorithm 2 presents the pseudocode of Greedy-AC-ALNS.

Algorithm 2 Greedy Adaptive Large Neighborhood Search Algorithm based on Ant Colony Optimization

1: **Input Parameters:** $Loc(x, y), tw(e_i, l_i), T, iteration$
2: Generate initial solution s using the Greedy Algorithm, $s_{best} = s$
3: $s' = s$
4: The initial score of operators σ
5: The initial weight of operators ω
6: The number of times an operator is selected b_d
7: Fix the nodes $p1$ and p2 in each subroute using the pheromone concentration
8: **while** $n \leq iteration$ **do**
9: Randomly select a removal operator using a roulette wheel selection method. Draw ξ or ψ request from s'. Remove ξ or ψ from λ
10: Randomly select a repair operator using a roulette wheel selection method to reinsert request ξ or ψ. Update s'
11: **if** $f(s') > f(s_{best})$ **then**
12: $s = s'$ Update the best solution $s_{best} = s'$
13: $\sigma += \sigma_1$
14: **else**
15: **if** $f(s') > f(s)$ **then**
16: $s = s'$
17: $\sigma += \sigma_2$
18: **else**
19: $s = s'$ with probability $p(s, s')$
20: $\sigma += \sigma_3$
21: **end if**
22: $iteration = iteration + 1$
23: **end if**
24: $T \leftarrow \beta T$
25: $\omega = UpdateWeights(\sigma, b_d)$
26: **end while**
27: **Output:** $s_{best} = s'$

In the Greedy-AC-ALNS algorithm, ξ represents the location of the parcel's delivery points, while ψ represents the location of passenger boarding and alighting nodes. Both ξ and ψ belong to λ, which represents unvisited requests. The time window for parcels and passengers is denoted by $tw(e_i, l_i)$. The variable s represents the initial solution, s' the temporary solution, and s_{best} the optimal solution. T represents the temperature of simulated annealing, β represents the cooling coefficient. n denotes the iteration count and *iteration* indicates the maximum iteration count. The algorithm's termination condition is reaching the maximum iteration count. The algorithm terminates when the maximum iteration count is reached.

The Greedy-AC-ALNS algorithm disrupts and repairs paths during the optimization process, as illustrated in Figure 6. Figure 6a depicts the initial feasible solution. Subsequently, feasible paths are subjected to node deletions using the destruction operator. It is important to note that if passengers are removed from the path, the pick-up and drop-off points must be deleted to ensure the integrity of the repaired solution. Finally, the repair operator reinserts the deleted nodes back into the route. It should be noted that Figure 6 does not depict the process of a taxi delivering a parcel to another taxi during its delivery journey. Instead, Figure 6 represents the algorithm optimization that takes place at the moment the taxi receives the request submission. The taxi does not begin the delivery service until the algorithm optimization in Figure 6c is completed.

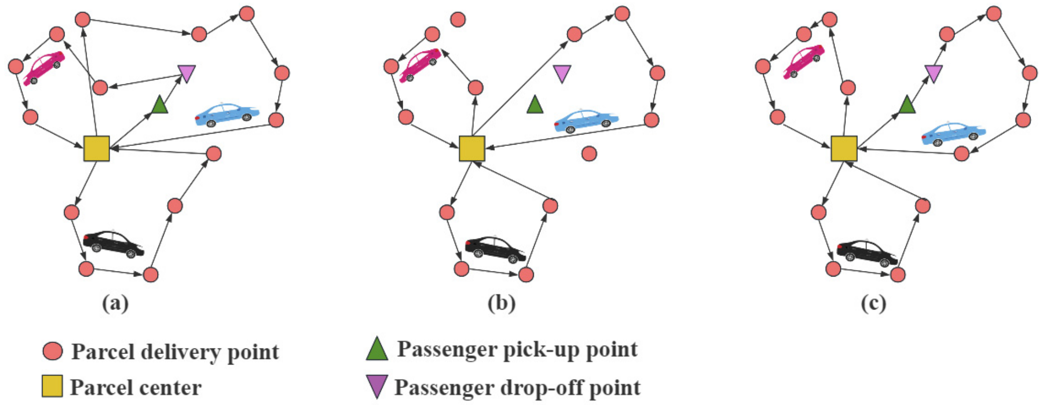


Figure 6. The explanation of destruction and repair operators. (**a**) Initial feasible solution; (**b**) Destruction operator for breaking the initial solution; (**c**) Repair operator for reinserting the initial solution.

Disruption and Repair Operators

The algorithm consists of two major categories of operators: disruption operators and repair operators.

(1) Disruption operators

Upon reaching a feasible solution, this study employs the following two disruption operators:

Random disruption: This operator systematically eliminates a specified number of orders from parcel or passenger requests in a stochastic manner. The removed orders do not require sorting based on other metrics. (e.g., the request submission time and the cost associated with serving the request). It is important to note that the removed orders should be stored in a pool of pending orders for further processing. This operator's operation is relatively simple and aids in diversifying the domain search, breaking out of local optima.

Worst disruption: This operator removes the order from the current solution, which causes the maximum increment in cost. Before executing this operator, each order is sorted in descending order of cost, and then the orders that significantly impact the objective function are deleted. Subsequently, the deleted orders are assigned to other routes one by one in an attempt to obtain a better solution.

(2) Repair operators

Random repair: The deleted orders are randomly inserted into other sub-paths, provided that the capacity constraint of the path is met. If these two conditions are not met, a new path is selected for insertion.

Greedy insertion: This operator employs the Greedy algorithm's concept. It assesses the cost increment of inserting the deleted order into other paths, and the path with the minimal cost increment is chosen to execute the repair operation.

Adaptive adjustment strategy

The AC-ALNS algorithm employs multiple disruption and repair operators, with the probability of each operator being selected depending on its performance in the previous iteration. Consequently, each operator's weight is updated after being used once, indicating its probability of being used in the next iteration.

In the implementation of the algorithm, firstly, the weights and scores of all operators are initialized, and an initial solution is set. In the subsequent iteration process, operators are randomly selected using a roulette wheel selection method. After obtaining a new solution, the operators for disruption and repair are scored, and their weights are updated based on their performance. The score accumulation during iteration is as follows:

σ_1: Score of finding the global optimal solution in this iteration σ.

σ_2: Score of finding a solution better than the current one but not the global optimal solution in this iteration.

σ_3: Score of not finding a solution better than the current one in this iteration, but the solution is accepted by the algorithm.

Then, the formula for updating the weights of the operators is as follows:

$$\omega_d = \begin{cases} r\frac{\sigma}{b_d} + (1-r)\omega_d; & b_d \neq 0 \\ \omega_d \quad ; & b_d = 0 \end{cases} \tag{26}$$

where ω_d represents a certain operator, b_d stands for the number of times the operator is used during iteration, and r represents the weight coefficient.

Acceptance criterion for simulated annealing

After finding a new feasible solution, it needs to be determined whether it will be accepted as a new solution. This study adopts the acceptance criterion of simulated annealing: if the new solution is better than the current solution, it is accepted; if the new solution is worse than the current solution, it is accepted with a probability calculated based on $p = exp\left(\frac{F(S)-F(S')}{T}\right)$ where $F(S')$ and $F(S)$ denotes the fitness of the new solution and the current solution, respectively. The maximum number of iterations for the algorithm is set to 1000, with the termination criterion being no improvement in the current solution over the next 250 iterations.

Here, we compare the initial solutions generated by the Greedy method with those generated by the savings algorithm. To ensure consistency in the experimental setup, both methods are applied to solve the Solomon standard problem instances RC101-10, RC101-25, and RC101-50. The node information in the instances represents delivery points, while the passengers' locations are randomly generated within a (100×100) grid, with the quantity matching the number of initial parcel delivery routes. The comparison of the results obtained from the two initial solution construction methods is presented in Table 3 below.

Table 3. Comparison of Greedy algorithm with savings mileage method for initial parcel delivery plans.

Instance	Greedy Algorithm		Savings Mileage Method	
	Number of Vehicles	Total Distance (km)	Number of Vehicles	Total Distance (km)
RC101-10	3	298.74	5	509.83
RC101-25	7	723.36	12	1144.34
RC101-50	16	1074.66	24	1906.22

As shown in Table 3, during the generation of initial solutions, the Greedy path generation method requires fewer vehicles and less total travel distance compared to the mileage-saving method. Moreover, both methods can deliver parcels to designated delivery points before the end of the working day, greatly enhancing vehicle operational efficiency and setting favorable conditions for subsequent optimization processes. Therefore, this paper chooses the Greedy method for generating initial parcel delivery routes. The following strategies are employed during initial route generation: firstly, encoding all parcels φ to be delivered, with each parcel corresponding to its distance ϑ from the parcel center, represented as $[\varphi, \vartheta]$. Sorting parcels in ascending order based on their distances from the parcel center; then, under the constraints of vehicle travel paths and total capacity, adopting the single-path distance Greedy insertion method to add parcels to the route. When the vehicle reaches maximum travel distance or maximum capacity, it stops receiving parcels, creating a new route to deliver the remaining parcels. This process is repeated until all parcel orders are successfully fulfilled.

4. Numerical Results

The example data in this paper are generated based on the standard Solomon examples. The standard Solomon examples refer to a well-known set of benchmark instances for the Vehicle Routing Problem with Time Windows (Solomon, 1987) [28]. More specifically, the delivery points for each parcel are adapted from the Solomon examples, as shown in Table 4, while passenger requests are randomly generated within a Manhattan network.

Table 4. Partial delivery point information and coordinates.

Id	x_coord (m)	y_coord (m)	Volume (dm^3)	Start Time	End Time
1	95	77	1.54	8	18
2	40	30	1.90	8	18
3	94	14	1.11	8	18
4	66	36	1.48	8	18
5	77	91	1.72	8	18
6	38	49	1.32	8	18
7	20	25	0.57	8	18
8	63	76	1.13	8	18
9	16	38	1.65	8	18
10	50	50	0	8	18

Inspired by the Solomon benchmark [4,25,28], the generation of parcel requests considers three various spatial distributions to represent a diversified portfolio of parcel demand scenarios. For two main reasons, we considered three different scenarios for the spatial distribution of parcels. For example, individual requests are usually scattered, while delivery points from parcel centers to warehouses, factories, hotels, etc., are often clustered. In the final scenario, parcels originate from individual requests or factories. The result of these factors can lead to the following spatial distributions:

Random distribution (R): Destinations are random, corresponding to the pattern of individual parcel requests.

Clustering distribution (C): Destinations are clustered, corresponding to the pattern of factories, offices, hotels, etc.

Mixture of both distributions (RC): The destinations are both random and clustered, corresponding to the spatial distribution of individual and factory parcel demands.

The taxi-related parameters are set as follows: the average driving speed is 40 km/h, the maximum travel distance is 120 km [4], and the maximum volume of parcels that the taxi can accommodate is 20 dm^3 [4], the operational cost of taxi is 2 ¥/km, mileage fee for passenger service is 5 ¥/km, mileage fee for parcels is 3 ¥/km, and unit distance detour cost is 1.5 ¥/km. The scores for operators in the algorithm are set as follows: $\sigma_1 = 30$, $\sigma_2 = 20$, $\sigma_3 = 10$. The initial temperature value is set to 0.2 times the initial solution's objective value, and the temperature decay coefficient is set to 0.9.

In this section, we first select 10 delivery points from the Solomon dataset C101. The initial parcel service routes are derived based on the Greedy algorithm, and the results are depicted in Figure 7.

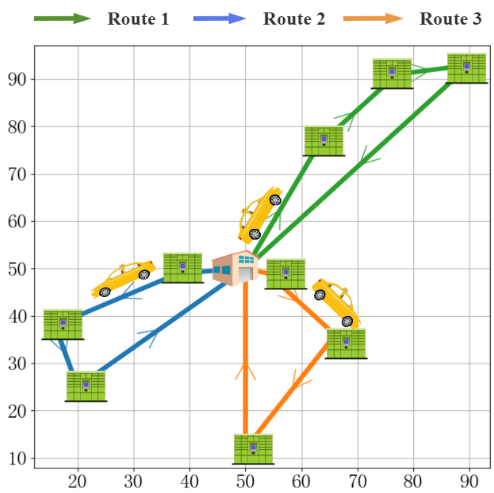


Figure 7. Initial parcel delivery paths.

Once passenger orders occur, they will be processed by sorting these orders according to their requested time windows for pick-up. Priority is given to passengers with earlier time windows, who are then allocated to the closest taxi along its delivery route until all passengers are accommodated. The initial parcel service routes will be updated, as shown in Figure 8a. Subsequently, the routes will be further optimized utilizing the Greedy-AC-ALNS, and the optimal routes for dual service of passengers and parcels are depicted in Figure 8b.

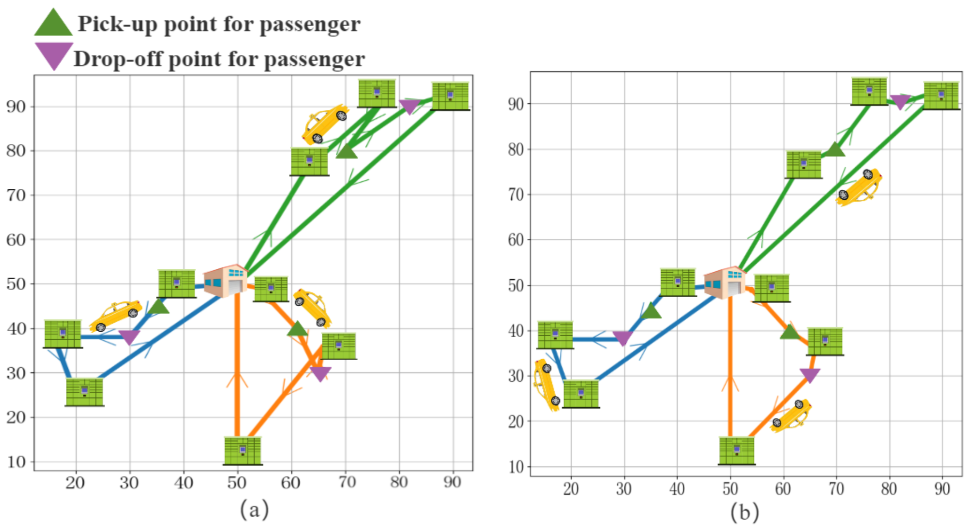


Figure 8. Updated routes after passenger insertion. (**a**) Unoptimized dual-service routes for passengers and parcels; (**b**) Optimized dual-service routes for passengers and parcels.

4.1. Comparison with the SARP

In this section, we first make a comparative study between the two models, SARP-LTW and the original SARP. A consistent set of case studies is employed to ensure consistency. In SARP, all parcels are inserted into predefined passenger service routes, whereas in SARP-LTW, the predefined routes for parcel service will be updated dynamically to promptly respond to passengers' orders.

The comparisons are regarding profitability, runtime, coefficient of variation, and maximum value generated by the two operating modes. The comparative results are presented in Table 5 and Figure 9, in which Re represents revenue, RT represents running time, CV represents coefficient of variation, and MV represents maximum value.

Table 5. Comparison of profitability in different operating models.

Case	SARP				SARP-LTW			
	Re (¥)	RT(h)	CV	MV (¥)	Re (¥)	RT(h)	CV	MV (¥)
C101-25	3258.16	0.10	0.16	3894.17	4399.56	0.09	0.15	5046.18
C101-50	4877.91	0.11	0.14	5537.15	6461.48	0.14	0.13	7054.86
C101-100	6178.84	1.5	0.12	6964.68	10,315.14	0.47	0.08	11,297.61
R101-25	3257.44	0.13	0.13	4045.89	4399.56	0.09	0.15	5146.89
R101-50	5410.64	0.15	0.15	6015.37	6461.48	0.16	0.13	6974.44
R101-100	7065.41	1.3	0.96	7873.16	10,187.85	0.41	0.08	11,963.05
RC101-25	3046.46	0.08	0.08	3517.39	3856.59	0.10	0.05	4315.84
RC101-50	4112.66	0.15	0.13	4689.35	6242.41	0.13	0.12	6945.21
RC101-100	7743.25	1.18	1	8369.23	10,302.96	0.51	0.26	11,703.06

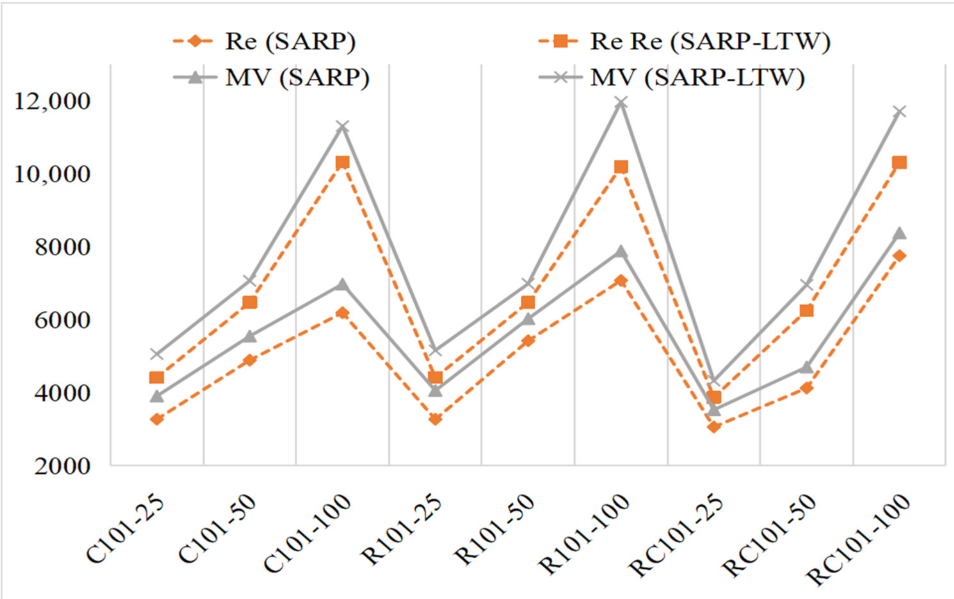


Figure 9. Comparison of profitability in different from SARP and SARP-LTW.

From Table 5, it can be observed that in terms of solution time, the difference is slight between the two models. Even for large-scale instances, both models can achieve optimal solutions in a short time. Regarding profitability, the SARP-LTW consistently outperforms the SARP model, whether gauged by the mean value or the maximal value of 10 experiments. In the SARP mode, after pre-establishing passenger service routes, when dynamic parcel requests occur, vehicles choose whether to carry parcels based on service priority, which may result in some parcels not being serviced, consequently diminishing profitability. However, in the SARP-LTW model, as parcel delivery routes are pre-determined and parcel delivery time windows are relatively flexible, taxis do not need to refuse passenger rides for urgent parcel delivery during weekdays. This operating mode ensures effective parcel delivery and can generate better revenue than the original SARP. Figure 9 provides a comparison of SARP-LTW with SARP in terms of both average and maximum profitability. It can be observed that SARP-LTW outperforms in performance metrics across all instances, which is especially noticeable in larger instances where the performance gap between SARP-LTW and SARP widens.

Then, the detour rate (DR), revenue rate (RR), and passenger service time (PST) of both models are analyzed and compared in Table 6. As shown in Table 6, the SARP-LTW model proposed in this paper outperforms the SARP model in terms of profit rate and average passenger service time, albeit displaying a slightly elevated detour rate. The SARP-LTW is designed to pre-plan routes for parcel service and dynamically insert passengers into the routes during parcel service. Given that parcels have relatively flexible time windows compared to passengers, priority is set for passengers. This will lead to detours between adjacent parcel delivery points due to passenger insertion. However, parcel service is totally accepted in the context of loose time windows for parcels. In the original SARP model, routes are primarily planned for passengers, and if parcel requests arise during passenger service, taxis need to assess whether carrying parcels would impact the passenger experience. If accommodating parcels compromises passenger comfort, taxis might need to provide compensatory measures, potentially leading to parcel rejection to mitigate costs. Consequently, the SARP mode exhibits a lower detour rate than the SARP-LTW. Regarding the profit rate, as the SARP-LTW rarely rejects passenger orders, it operates more efficiently, resulting in higher profitability. Furthermore, in both models, passengers are accorded precedence over parcels, resulting in analogous average passenger service times.

Table 6. Comparison of differences between SARP and SARP-LTW.

Case	**SARP**			**SARP-LTW**		
	DR (%)	**PR (%)**	**PST (h)**	**DR (%)**	**PR (%)**	**PST (h)**
C101-100	4.05	33.6	0.29	6.21	49.8	0.26
R101-100	4.91	31.8	0.32	6.63	48.1	0.29
RC101-100	8.66	26.3	0.41	9.89	45.3	0.38

4.2. Performance of Greedy-AC-ALNS

To validate the efficacy of the Greedy-AC-ALNS, this section conducts a comparative performance evaluation. The Genetic Algorithm (GA) serves as the benchmark algorithm. The Greedy-AC-ALNS designed in this paper is an improvement upon the ALNS algorithm. To verify whether the improvement is significant, the results obtained by the Greedy-AC-ALNS algorithm for three sets of instances are compared with those obtained by the Genetic Algorithm (GA) and the original ALNS algorithm.

To ensure equitable comparison, all three algorithms are subjected to 100 iterations. The constraints such as vehicle travel distance, capacity, unit distance travel cost, and revenue remain consistent. Each algorithm is applied to solve each instance ten times, and the average values are calculated. A total of 270 experiments were conducted. The results are shown in Table 7, where the bold font indicates the best-performing indicator among the three algorithms.

From Table 7, in terms of profit, except for instance C101-25, the Greedy-AC-ALNS algorithm outperforms the other two algorithms in all nine instances, producing the highest average profit. This demonstrates that the improved Greedy-AC-ALNS algorithm exhibits stronger search capabilities in finding optimal solutions. In terms of algorithm runtime, the GA requires relatively more time, while the ALNS algorithm can find the optimal solution in a shorter time. This is related to the structure of the ALNS algorithm itself. Unlike the Greedy-AC-ALNS algorithm, the ALNS algorithm does not require updating information between nodes during the search for the optimal solution, thus saving some time compared to the Greedy-AC-ALNS algorithm.

As shown in Figure 10, among the Genetic Algorithm (GA), ALNS algorithm, and Greedy-AC-ALNS algorithm, the Greedy-AC-ALNS algorithm exhibits the best solving performance across instances of different scales. This indicates that the improvement made in this study on the ALNS algorithm is effective.

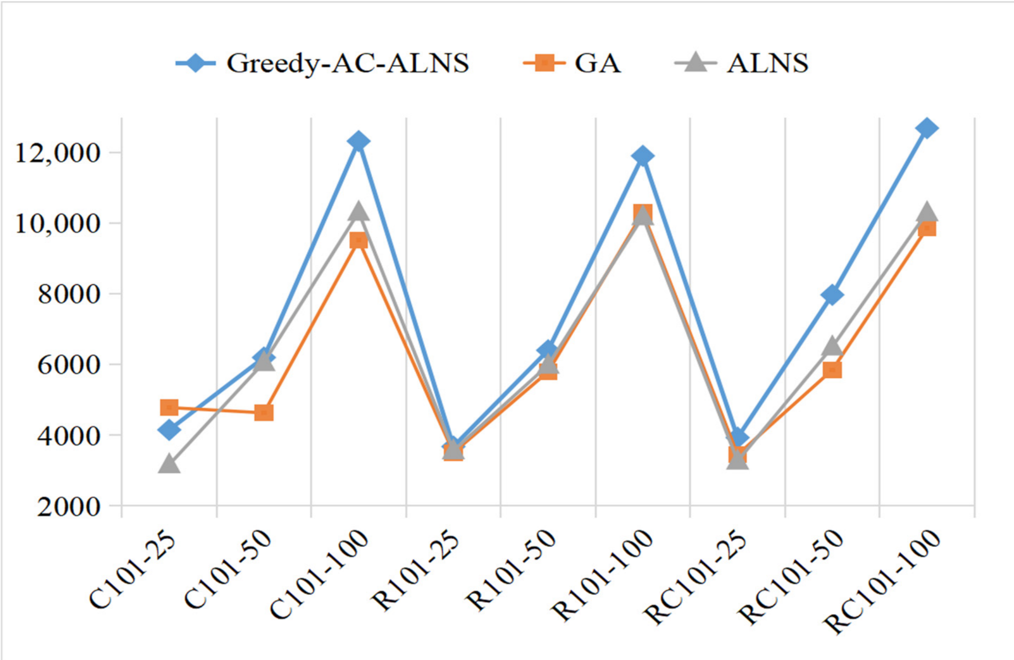


Figure 10. Performance comparison of different algorithms for solving.

In addition to the performance analysis above, we also analyzed the convergence of different algorithms based on the RC-101 instance, in which the spatial distribution of parcels and passengers is the most complex in all scenarios. From Figure 11, it can be seen that the Greedy-AC-ALNS algorithm reaches the optimal solution with fewer iterations compared to the ALNS and GA, which demonstrates the effectiveness of the improved algorithm.

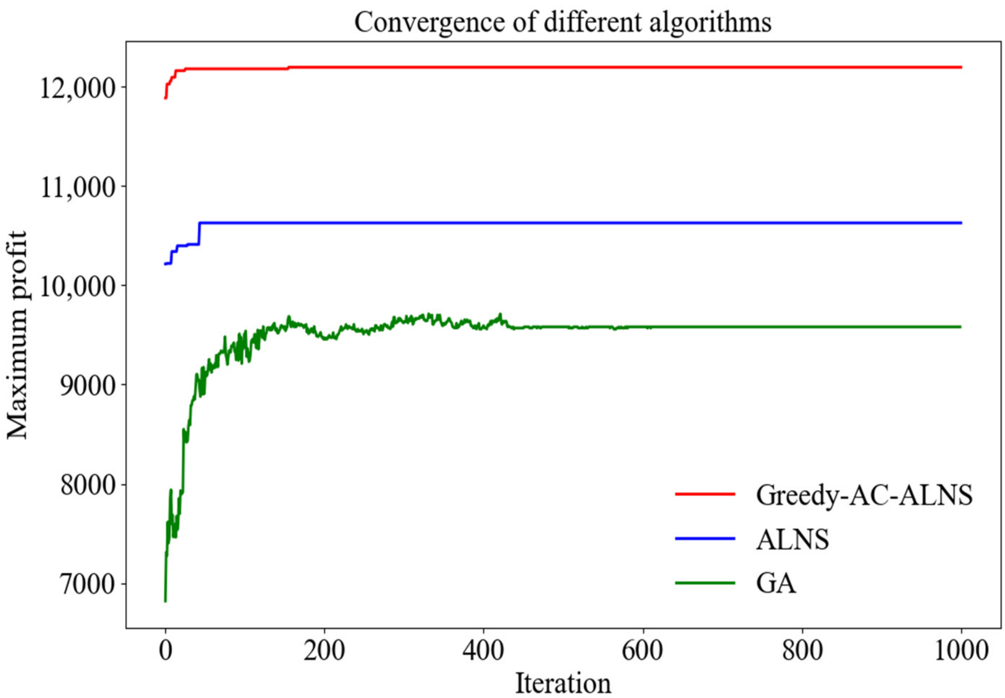


Figure 11. Convergence of different algorithms.

Table 7. Performance comparison results of different algorithms.

Case	GA			ALNS			Greedy-AC-ALNS		
	Re (¥)	RT(s)	CV	Re (¥)	RT(s)	CV	Re (¥)	RT(s)	CV
C101-25	4782.58	0.36	0.1	3167.85	0.09	0.15	4151.88	0.55	0.08
C101-50	4632.91	0.72	0.12	6059.75	0.14	0.13	6200.42	0.46	0.16
C101-100	9518.23	2.5	0.13	10,315.14	0.47	0.08	12,326.55	0.42	0.05
R101-25	3507.46	0.34	0.25	3563.34	0.06	0.66	3679.75	0.35	0.13
R101-50	5798.17	0.84	0.36	5986.37	0.15	0.2	6404.34	0.33	0.16
R101-100	10,298.9	2.75	0.12	10,187.85	0.41	0.04	11,909	0.43	0.03
RC101-25	3447.66	0.25	0.12	3271.24	0.05	0.05	3932.66	0.44	0.06
RC101-50	5851.08	0.91	0.16	6503.55	0.13	0.12	7975.32	0.49	0.13
RC101-100	9863.65	2.5	0.1	10,302.96	0.51	0.05	12,694.29	0.5	0.03

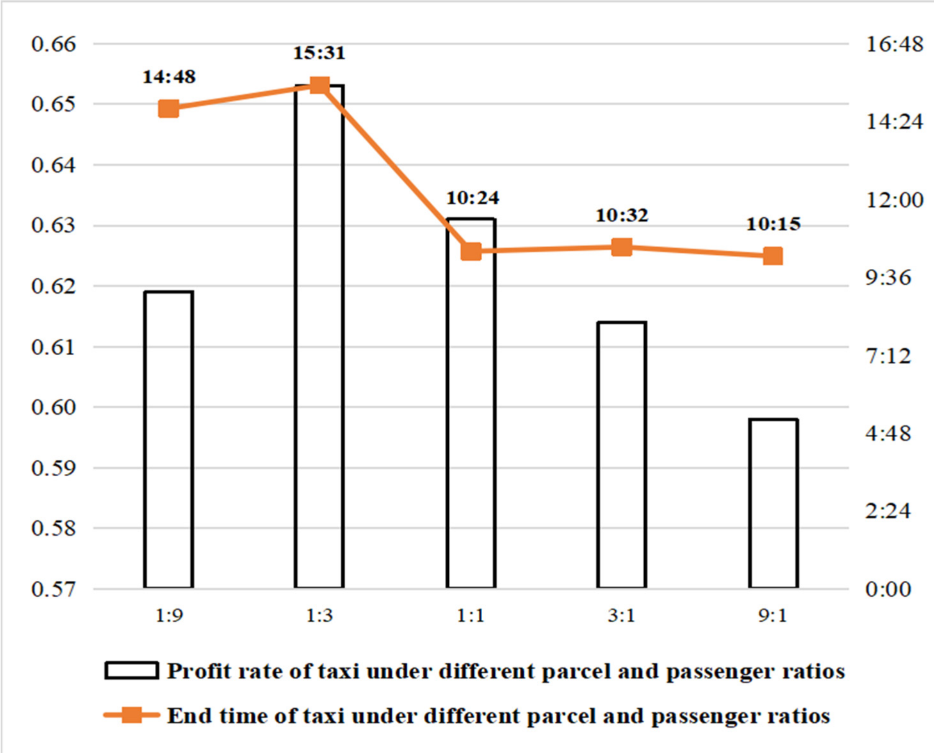
4.3. Effect of Ratio of Parcel to Passenger

In this section, we investigated the impact of different ratios of parcel orders to passenger orders on the performance metrics of SARP-LTW, including the effects on taxis' profitability, vehicle end times, and the maximum system capacity. The results are shown in Figure 12.

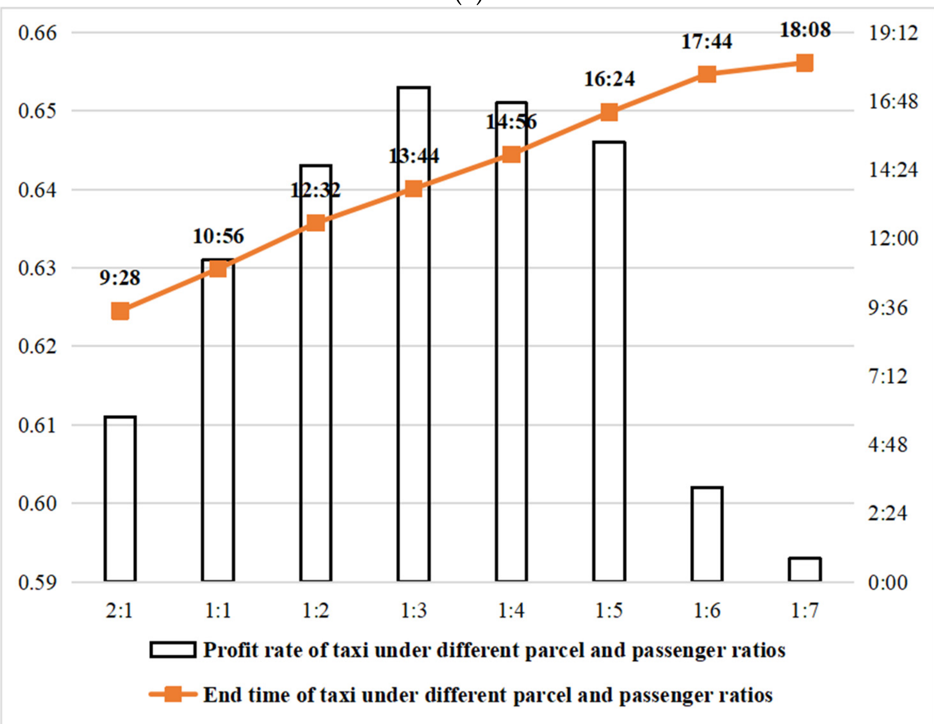
Figure 12a investigates the impact of different ratios on taxis' profitability under the condition of a fixed total number of parcels and passengers. It is important to note that in Figure 12a, it is assumed that there are enough vehicles to serve all requests. Due to variations in revenue generated by different total request counts, we use profitability rate to reflect the influence of different parcel–passenger ratios on taxi profitability in our study.

From Figure 12a, it can be observed that regardless of changes in the parcel–passenger ratio, the vehicle profitability rate remains above 0.5 in the SARP-LTW model. When the parcel–passenger ratio is 1:3, the vehicle profitability rate reaches its maximum value of 0.653, and the vehicles can complete all delivery tasks before the shift ends. When the parcel–passenger ratio is less than 1:3, the profitability rate decreases because the large number of passengers results in additional detour costs for vehicles, leading to a decline in vehicle profitability. Conversely, when the parcel–passenger ratio exceeds 1:3, the increasing number of parcels, but with a constant total request count, results in a relatively lower service unit price for parcels compared to passengers, thereby also leading to a decrease in vehicle profitability rate. Therefore, in the SARP-LTW model, if there are enough vehicles to serve all orders, vehicles will achieve better profitability when the parcel–passenger ratio reaches 1:3.

Figure 12b investigates the maximum parcel–passenger ratio that the SARP-LTW system can accommodate when the number of vehicles is fixed. From Figure 12b, it can be observed that as the number of passenger orders increases, the taxis' end times also become later. This is because using a limited number of vehicles to serve a larger number of passenger requests requires more time. When the parcel–passenger request ratio reaches 1:7, it is found that the SARP-LTW system will be unable to complete all delivery tasks before the taxis finish their shifts. Therefore, when the number of vehicles is fixed, and the number of parcels is already determined, the maximum number of passenger orders that taxis can serve should not exceed six times the number of parcels.



(**a**)



(**b**)

Figure 12. The impact of different ratios of parcels to passengers on various metrics. (**a**) Profit rate and service end time of taxi when serving different ratios of parcel and passenger requests with a sufficient number of vehicles; (**b**) Profit rate and service end time of taxi when serving different ratios of parcel and passenger requests with a fixed number of vehicles.

5. Conclusions and Discussion

This study focuses on the Share-a-Ride Problem (SARP) and discusses its operational mechanism and cost-benefit aspects. It is a significant contribution to the existing research on passenger and freight co-transportation problems, partially addressing the shortcomings of current research and providing an essential reference for the implementation of passenger and freight co-transportation modes. The research findings primarily reveal two points.

(1) Our research clarifies the specific concept and operational mechanism of the people and parcel-sharing taxis problem with loose time windows of parcels and establishes a mathematical model suitable for this operational scenario.

(2) In response to the complexity of the problem and the scale of the instances, this paper designs the Greedy-AC-ALNS algorithm based on the ant colony information updating mechanism. From the results of the instance analysis, the SARP-LTW outperforms the SARP mode in terms of profit rate, revenue, and revenue stability, with improvements of 56%, 46%, and 69%, respectively. The improved Greedy-AC-ALNS algorithm designed in this paper also outperforms the ALNS algorithm in terms of solution stability, with an average solution coefficient of variation lower than that of the ALNS algorithm by 0.43 and a 16% higher ability to search for optimal solutions. The instance analysis shows that the people and parcel-sharing taxis problem with loose time windows of parcels in this paper performs better in terms of operational revenue compared to the SARP mode, indicating potential practical applicability in the future. As for the Greedy-AC-ALNS algorithm proposed in this paper, it effectively improves both solution efficiency and solution quality when dealing with large-scale instances.

(3) In the final numerical analysis, our study also explored the impact of different parcel–passenger ratios on various metrics of the system under different vehicle scales. Ultimately, we found that when there are many vehicles, the highest vehicle profitability rate occurs when the parcel–passenger ratio is 1:3. However, when the number of vehicles is limited, the system cannot complete all delivery tasks before the vehicles finish their shifts when the parcel–passenger ratio reaches 1:7.

The practical implications of our dual-service SARP-LTW model are significant for urban transportation systems. By allowing taxis to handle both passenger and parcel deliveries with loose time windows for parcels, our model can substantially increase the efficiency and profitability of taxi operations. This integrated approach enables taxi companies to maximize vehicle utilization, reducing idle time and increasing revenue. Furthermore, the prioritization of dynamic passenger requests ensures minimal delays for passengers, enhancing customer satisfaction. It is important to note that our SARP-LTW model offers considerable flexibility. Moreover, given that both traditional taxis and ride-hailing taxis have comparable parcel storage space and passenger service capacity, the proposed Share-a-Ride model is applicable to both traditional taxis and ride-hailing taxis.

Implementing our model in urban settings could lead to reduced traffic congestion and lower emissions, as fewer vehicles are required to meet the combined demand for passenger and parcel services. This could also encourage the adoption of sustainable transport practices. Additionally, the use of the Greedy-AC-ALNS algorithm provides a robust and efficient solution method, potentially reducing operational costs and improving service reliability.

However, this study also has certain limitations, primarily manifested in the following aspects: (1) The instance analysis section utilized simulated Solomon instances rather than real-life ones. This might lead to discrepancies between the solution outcomes and real-life scenarios, thereby introducing some bias in the analysis results. (2) The objective function of the SARP-LTW model constructed in this paper only considers profit factors and does not incorporate multiple objectives such as environmental impacts. In subsequent research, environmental impacts could be included as one of the optimization objectives. In the future, it is imperative to expand the use of real-life data to ensure more objective and rational analysis results.

In the future, the implementation of our proposed operating model may encounter various problems and challenges. We have analyzed and discussed these issues from the perspectives of external impacts, policy environment, and operational aspects. Firstly, the externalities of the Share a Ride model are difficult to quantify and evaluate. Currently, there are few scholars working on this aspect. For example, it is challenging to establish a unified standard to effectively empirically evaluate the impact of the Share a Ride model. Therefore, developing a scientific and reasonable evaluation system for shared transportation and comprehensively assessing its external impacts will be a continuous challenge for policymakers and researchers. Secondly, the smooth implementation of the Share a Ride model requires reasonable policy support, and the public sector needs to provide certain policy support to enterprises. However, there are still some detailed issues that need to be properly addressed, such as regulatory measures, safety concerns, and illegal parking. Although current researchers have done a lot of work in terms of policy support, such as funding subsidies, protecting users' personal information, and improving road regulations, there is still much room for further research. Lastly, regarding the operational aspect, ensuring fair profit distribution among operators, reaching a consensus with public transportation departments, and popularizing this operating model to users are all issues that need to be considered in the future.

Author Contributions: Conceptualization, S.X.; methodology, S.X.; software, Q.Z.; validation, S.X. and Q.Z.; formal analysis, S.X.; investigation, S.X.; resources, Q.Z.; data curation, Q.Z.; writing—original draft preparation, Q.Z.; writing—review and editing, S.X. and N.S.; visualization, Q.Z.; supervision, S.X. and N.S.; project administration, S.X.; funding acquisition, S.X. All authors have read and agreed to the published version of the manuscript.

Funding: This project is supported by the National Natural Science Foundation of China under Grant [71901175].

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the article.

Conflicts of Interest: The authors report there are no competing interests to declare.

References

1. Alisoltani, N.; Leclercq, L.; Zargayouna, M. Can dynamic ride-sharing reduce traffic congestion? *Transp. Res. Part B Methodol.* **2022**, *145*, 212–246. [CrossRef]
2. Long, J.; Tan, W.; Szeto, W.Y.; Li, Y. Ride-sharing with travel time uncertainty. *Transp. Res. Part B Methodol.* **2018**, *118*, 143–171. [CrossRef]
3. Sun, Z.; Wang, Y.; Zhou, H.; Jiao, J.; Overstreet, R.E. Travel behaviours, user characteristics, and social economic impacts of shared transportation: A comprehensive review. *Int. J. Logist. Res. Appl.* **2019**, *24*, 1–27. [CrossRef]
4. Li, B.; Krushinsky, D.; Reijers, H.A.; Van Woensel, T. The Share-a-Ride Problem: People and parcels sharing taxis. *Eur. J. Oper. Research.* **2014**, *238*, 31–40. [CrossRef]
5. Wang, R.; Chen, F.; Liu, X.; Liu, X.; Li, Z.; Zhu, Y. A Matching Model for Door-to-Door Multimodal Transit by Integrating Taxi-Sharing and Subways. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 469. [CrossRef]
6. Beirigo, B.A.; Schulte, F.; Negenborn, R.R. Integrating people and freight transportation using shared autonomous vehicles with compartments. *IFAC-Pap. Online* **2018**, *51*, 392–397. [CrossRef]
7. van der Tholen, M.; Beirigo, B.A.; Jovanova, J.; Schulte, F. The share-a-ride problem with integrated routing and design decisions: The case of mixed-purpose shared autonomous vehicles. *Int. Conf. Comput. Logist.* **2021**, *62*, 347–361.
8. Enzi, M.; Parragh, S.N.; Pisinger, D.; Prandtstetter, M. Modeling and solving the multimodal car- and ride-sharing problem. *Eur. J. Oper. Res.* **2021**, *16*, 290–303. [CrossRef]
9. Ghilas, V.; Demir, E.; Van Woensel, T. A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. *Transp. Res. Part B Methodol.* **2016**, *91*, 34–51. [CrossRef]
10. Ren, T.; Jiang, Z.; Cai, X.; Yu, Y.; Xing, L.; Zhuang, Y.; Li, Z. A dynamic routing optimization problem considering joint delivery of passengers and parcels. *Neural Comput. Appl.* **2022**, *33*, 10323–10334. [CrossRef]
11. Ye, Y.; Guo, J.; Yan, L. A Metro Freight Plan for Mixed Passenger and Freight Transportation. *J. Adv. Transp.* **2022**, *2022*, 6602387. [CrossRef]

12. Ye, Y.; Guo, J.; Yan, L. A Mixed Decision Strategy for Freight and Passenger Transportation in Metro Systems. *Comput. Intell. Neurosci.* **2021**, *2021*, 5412016. [CrossRef]
13. Azcuy, I.; Agatz, N.; Giesen, R. Designing integrated urban delivery systems using public transport. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *156*, 102525. [CrossRef]
14. Pimentel, C.; Alvelos, F. Integrated urban freight logistics combining passenger and freight flows mathematical model proposal. *Transp. Res. Procedia* **2018**, *30*, 80–89. [CrossRef]
15. Ehmke, J.F.; Campbell, A.M. Customer acceptance mechanisms for home deliveries in metropolitan areas. *Eur. J. Oper. Res.* **2019**, *233*, 193–207. [CrossRef]
16. Lu, C.C.; Diabat, A.; Li, Y.T.; Yang, Y.M. Combined passenger and parcel transportation using a mixed fleet of electric and gasoline vehicles. *Transp. Res. Part E Logist. Transp. Rev.* **2022**, *157*, 102546. [CrossRef]
17. Shu, S.; Chen, Z.; Yu, Z.; Cao, S.; Wu, G.; Shi, D.; Wang, G.; Liu, Z.; Chen, X.; Na, X.; et al. Modeling Freight-Sharing Platform Operations for Optimal Compensation Strategy Using Markov Decision Processes. *Inst. Electr. Electron. Eng.* **2022**, 1006–1011.
18. Taiebat, M.; Amini, E.; Xu, M. Sharing behavior in ride-hailing trips: A machine learning inference approach. *Transp. Res. Part D Transp. Environ.* **2022**, *103*, 10316. [CrossRef]
19. Vincent, F.Y.; Mareg, M.Z.; Sisay, G.G.; Putu, A.Y.I.; Panca, J. A Metaheuristic algorithm for the share-a-ride problem. *Expert Syst. Appl.* **2023**, *230*, 120569.
20. Yu, V.F.; Indrakarna, P.A.; Redi, A.A.N.P.; Lin, S.W. Simulated annealing with mutation strategy for the share-a-ride problem with flexible compartments. *Mathematics* **2021**, *9*, 2320. [CrossRef]
21. Zhan, X.; Szeto, W.Y.; Shui, C.S.; Chen, X.M. A modified artificial bee colony algorithm for the dynamic ride-hailing sharing problem. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *150*, 102124. [CrossRef]
22. Azi, N.; Gendreau, M.; Potvin, J.Y. An adaptive large neighborhood search for a vehicle routing problem with multiple routes. *Comput. Oper. Res.* **2022**, *41*, 167–173. [CrossRef]
23. Li, B.; Krushinsky, D.; Van Woensel, T.; Reijers, H.A. An adaptive large neighborhood search heuristic for the share-a-ride problem. *Comput. Oper. Res.* **2016**, *66*, 170–180. [CrossRef]
24. Hosni, H.; Naoum-Sawaya, J.; Artail, H. The shared-taxi problem: Formulation and solution methods. *Transp. Res. Part B Methodol.* **2014**, *70*, 303–318. [CrossRef]
25. Ji, W.; Liu, S.; Han, K.; Liu, T. Share-a-Ride Problem with mixed ride-hailing and logistic vehicles. *arXiv* **2024**, arXiv:2403.11944.
26. Cohen, B.; Kietzmann, J. Ride On! Mobility Business Models for the Sharing Economy. *Organ. Environ.* **2014**, *27*, 280–298. [CrossRef]
27. Zhu, J.; Lin, N.; Zhu, H.; Liu, X. Role of sharing economy in energy transition and sustainable economic development in China. *J. Innov. Knowl.* **2023**, *8*, 1–12. [CrossRef]
28. Solomon, M.M. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Oper. Res.* **1987**, *35*, 254–265. [CrossRef]