mathematics

Article

An Improved Genetic Algorithm for the Granularity-Based Split Vehicle Routing Problem with Simultaneous Delivery and Pickup

Yuxin Liu *, Zihang Qin and Jin Liu

College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China; 202130310160@stu.shmtu.edu.cn (Z.Q.); jinliu@shmtu.edu.cn (J.L.)
* Correspondence: liuyx@shmtu.edu.cn

Abstract: The Split Vehicle Routing Problem with Simultaneous Delivery and Pickup (SVRPSDP) consists of two subproblems, i.e., the Vehicle Routing Problem with Simultaneous Delivery and Pickup (VRPSDP) and the Split Delivery Vehicle Routing Problem (SDVRP). Compared to the subproblems, SVRPSDP is much closer to reality. However, some realistic factors are still ignored in SVRPSDP. For example, the shipments are integrated and cannot be infinitely subdivided. Hence, this paper investigates the Granularity-based Split Vehicle Routing Problem with Simultaneous Delivery and Pickup (GSVRPSDP). The characteristics of GSVRPSDP are that the demands of customers are split into individual shipments and both the volume and weight of each shipment are considered. In order to solve GSVRPSDP efficiently, a Genetic-Simulated hybrid algorithm (GA-SA) is proposed, in which Simulated Annealing (SA) is inserted into the Genetic Algorithm (GA) framework to improve the global search abilities of individuals. The experimental results indicate that GA-SA can achieve lower total costs of routes compared to the traditional meta-algorithms, such as GA, SA and Particle Swarm Optimization (PSO), with a reduction of more than 10%. In the further analysis, the space utilization and capacity utilization of vehicles are calculated, which achieve 86.1% and 88.9%, respectively. These values are much higher than those achieved by GA (71.2% and 74.8%, respectively) and PSO (60.9% and 65.7%, respectively), further confirming the effectiveness of GA-SA. And the superiority of simultaneous delivery and pickup is proved by comparing with separate delivery and pickup. Specifically, the costs of separate delivery and pickup are more than 80% higher than that of simultaneous delivery and pickup.

Keywords: vehicle routing problem; granularity; split; simultaneous delivery and pickup; simulated annealing; genetic algorithm

MSC: 90-10

check for updates

Citation: Liu, Y.; Qin, Z.; Liu, J. An Improved Genetic Algorithm for the Granularity-Based Split Vehicle Routing Problem with Simultaneous Delivery and Pickup. *Mathematics* **2023**, *11*, 3328. https://doi.org/10.3390/math11153328

Academic Editors: Chao Gao, Zhen Wang and Peican Zhu

Received: 7 July 2023
Revised: 21 July 2023
Accepted: 24 July 2023
Published: 28 July 2023

Copyright: © 2023 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 (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The Vehicle Routing Problem (VRP) [1–3] is a classic NP-hard problem, which has wide applications in the real world. Since it was first proposed by Dantzig and Ramser [4] in 1959, VRP has been a hot topic in the operations research literature. Recently, more and more researchers focus on variation VRPs, such as the VRP with time windows [5,6], the VRP with simultaneous delivery and pickup [7,8] and the VRP with heterogeneous fleet [9,10]. These VRPs come from reality and have high practical application values [11–13].

The Vehicle Routing Problem with Simultaneous Delivery and Pickup (VRPSDP) was proposed by Min [14] in 1989. The traditional VRP is equivalent to the delivery problem, i.e., transportation from one or more warehouse points to different customer points. The pickup problem, on the other hand, is equivalent to the real-life return demands from different customers. Simultaneous pickup and delivery means that delivery and pickup are considered simultaneously with the aim of reducing transportation costs and increasing efficiency. VRPSDP is the foundation of the urban logistics distribution problem [15–18].

The Split Delivery Vehicle Routing Problem (SDVRP) was proposed by Dror and Trudeau [19] in 1990. Its original intention was to solve the problem that the demands of customers are larger than the capacity of vehicles in practice. In SDVRP, transportation is be split according to customer demands, with multiple vehicles working together to complete the transportation task. If the service is not completed at one time, the vehicle needs to return to the depot point and restart the service. Research has proved that both the number of vehicles and the total costs can be optimized by allowing the demands of a customer to be split between any number of vehicles [20]. Afterwards, many researchers have also begun to conduct extensive research based on SDVRP [21–24].

The Split Vehicle Routing Problem with Simultaneous Delivery and Pickup (SVRPSDP) [25,26] is a combination of VRPSPD and SDVRP, in which delivery and pickup are conducted simultaneously and meanwhile demands of customers can be divided. However, it splits the demands of customers infinitely, without taking into account the integration of shipments. That is, the sizes and weights of shipments are actually fixed in reality, and each shipment should be transported as a whole [27,28]. Addressing this issue, this paper establishes a new model called the Granularity-based SVRPSDP, shorted as GSVRPSDP. In GSVRPSDP, each shipment of the customer is considered, rather than a total demand value. Each shipment has two features: a volume and a weight. Meanwhile, vehicles have volume and weight constraints. It is obvious that GSVRPSDP can be better adapted to realistic transportation situations in the face of different kinds of shipments, such as large-volume but relatively light shipments.

The main algorithms currently used to solve the traditional VRP and its variants are exact solution, meta-heuristic and super-heuristic [29–32]. Exact solution algorithms take too long to solve, super-heuristic algorithms are more complex and involve a greater workload, while meta-heuristic algorithms can obtain satisfactory solutions relatively quickly. Moreover, hybrid meta-heuristic [33–35] has become a popular and effective method in recent years, one which can compensate well for the shortcomings of a single algorithm and allow the strengths of algorithms to complement each other [36], such as Genetic Algorithm (GA) with strong global search capability versus Simulated Annealing (SA) with strong local search capability. Hence, this paper designed a Genetic–Simulated hybrid algorithm (GA-SA) to solve the GSVRPSDP. Through experiments, comparisons with SA, GA and Particle Swarm Optimization (PSO) are performed to show that GA-SA can obtain better objective optimization values and generate better vehicle travel paths in the scenario of GSVRPSDP.

The rest of this paper is organized as follows. Section 2 describes the problem established in this paper. Section 3 introduces the proposed GA-SA, including the solution representation and individuals updating based on SA. Section 4 shows the experimental design, including experimental data and parameters. Section 5 presents the results and analytical discussion of the comparison experiments. Finally, Section 6 summaries the paper and provides the future work.

2. Problem Definition

In this section, we first give the definition of the basic VRP, and then introduce the GSVRPSDP that we constructed in this paper.

2.1. The Basic VRP Model

The basic VRP can be represented with a graph $G = \{V, E\}$. $V = \{0\} \cup V_c$ is the set of vertices, in which 0 represents the distribution center and $V_c = \{1, ..., N\}$ represents the customers. Each customer $i \in V_c$ has a non-zero demand m_i. $E = \{(i, j)|i, j \in V, i \neq j\}$ is the set of edges. A fleet of vehicles with a limited capacity is located at the center to deliver shipments to all the customers. Each vehicle must start and end its route at the center 0, and the total demands served by each vehicle cannot exceed its capacity. If the capacity is smaller than the total demands of all customers, then several vehicles are needed, and each takes responsibility for several customers. For example, as shown in Figure 1, four vehicles are needed to meet the demands of all customers, which are shown as four routes. The objective of the VRP is to minimize the total costs of vehicles.

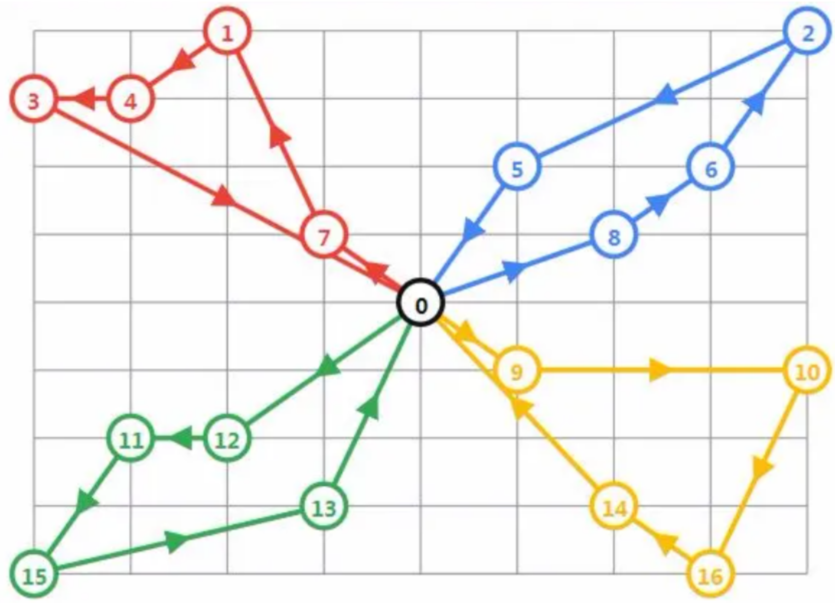


Figure 1. An example instance of the basic VRP model. 0 represents the distribution center, and 1–16 represent the customers. The example includes four routes, each of which is assigned a different color.

2.2. The Proposed GSVRPSDP Model

The proposed GSVRPSDP model is modified based on the basic VRP from the following three aspects:

1. The customer not only has a delivery demand, but also has a pickup demand. To meet the two demands efficiently, vehicles can deliver and pick up shipments simultaneously.
2. The demands of customers are composed of individual shipments, and each shipment has a weight feature and a volume feature. Meanwhile, the vehicle not only has a weight constraint, but also has a volume constraint.
3. The demands of customers can be split, which means that vehicles can deliver part of the shipments and pick up part of the shipments from each customer.

The definition of notations in GSVRPSDP can be seen in Table 1, followed by decision variables, and then the mathematical model of GSVRPSDP.

Table 1. The definition of notations in GSVRPSDP.

Parameter	Description
C_0	The fixed costs of vehicle operation
C_1	The variable costs per unit distance travelled
N	The number of customers
V	The set of nodes ($V = 0, 1, 2, 3, \ldots, N$, where 0 denotes the distribution center; $1, 2, \ldots, N$ denotes customer points)
K	The number of vehicles to be used
SV	The set of vehicles ($SV = 1, 2, 3, \ldots, K$)
d_{ij}	The distance from the customer i to the customer j
M_W	The maximum load of the vehicle
M_V	The maximum volume of the vehicle

Table 1. *Cont.*

Parameter	Description
${m_i}^d$	The weight of the shipments of the *i*th customer to be delivered.
${v_i}^d$	The volume of the shipments of the *i*th customer to be delivered.
${m_i}^p$	The weight of the shipments to be picked up from the *i*th customer.
${v_i}^p$	The volume of the shipments to be picked up from the *i*th customer.
m_{ijk}	The weight of shipments on the vehicle *k* before it is sent to the customer *j* after finishing pickup and delivery at the customer *i*.
v_{ijk}	The volume of shipments on the vehicle *k* before it is sent to the customer *j* after finishing pickup and delivery at the customer *i*.

Decision variables:

$$x_{ijk} = \begin{cases} 1, & \text{The vehicle } k \text{ travels from the customer } i \text{ to } j \\ 0, & otherwise \end{cases}$$

$$y_{ik} = \text{The fraction of the number of shipments delivered to the } i\text{th customer by the } k\text{th vehicle}$$

$$z_{ik} = \text{The fraction of the number of shipments picked up from the } i\text{th customer by the } k\text{th vehicle}$$

Mathematical model:

$$MinZ = C_0K + C_1 \sum_{i=0}^{N} \sum_{j=0}^{N} \sum_{k=1}^{K} d_{ij} x_{jjk} \tag{1}$$

$$\sum_{k=1}^{K} y_{ik} = 1 (\forall i = 1, 2, \ldots, N) \tag{2}$$

$$\sum_{k=1}^{K} z_{ik} = 1 (\forall i = 1, 2, \ldots, N) \tag{3}$$

$$\sum_{k=1}^{K} \sum_{j=1}^{N} x_{0jk} = \sum_{k=1}^{K} \sum_{j=1}^{N} x_{j0k} = K \tag{4}$$

$$\sum_{i=1}^{N} x_{ijk} = \sum_{i=1}^{N} x_{jjk} (k \in m, \forall j = 1, 2, \ldots, N) \tag{5}$$

$$\sum_{k=1}^{K} \sum_{i=0}^{N} m_{ijk} - \sum_{k=1}^{K} \sum_{c}^{C^d} \mathrm{y}_{jck}^{d} m_{jc}^{d} + \sum_{k=1}^{K} \sum_{c}^{C^p} \mathrm{y}_{jck}^{p} m_{jc}^{d} = \sum_{k=1}^{K} \sum_{i=0}^{N} m_{jik}, \forall j = 1, 2, \ldots, N \tag{6}$$

$$\sum_{k=1}^{K}\sum_{i=0}^{N} v_{ijk} - \sum_{k=1}^{K}\sum_{c}^{C^d} y_{jck}^{d} v_{jc}^{d} + \sum_{k=1}^{K}\sum_{c}^{c^p} y_{jck}^{p} v_{jc}^{d} = \sum_{k=1}^{K}\sum_{i=0}^{N} v_{jik}, \forall j = 1,2,\ldots,N \tag{7}$$

$$m_{ijk} \le M, \quad i \in n, \mathrm{j} \in n, k \in SV \tag{8}$$

$$v_{ijk} \le V, \quad i \in n, \mathrm{j} \in n, k \in SV \tag{9}$$

$$\sum_{k=1}^{K}\sum_{i=0}^{N} x_{ijk} \ge 1 (j = 1,2,\ldots,N) \tag{10}$$

$$K \ge \max\left\{ \frac{\sum_{i=1}^{N}\sum_{c=1}^{c_i^d} v_{ic}^{d}}{V}, \frac{\sum_{i=1}^{N}\sum_{c=1}^{C_i^d} m_{ic}^{d}}{M} \right\}, \quad K \in N^* \tag{11}$$

$$K \ge \max\left\{ \frac{\sum_{i=1}^{N}\sum_{c=1}^{c_i^p} v_{ic}^{p}}{V}, \frac{\sum_{i=1}^{N}\sum_{c=1}^{c_i^p} m_{ic}^{p}}{M} \right\}, \quad K \in N^* \tag{12}$$

Among them, Equation (1) is the objective function, which indicates that the goal of problem is to minimize the sum of the vehicle fixed costs and the vehicle variable costs. Constraint (2) insures that each customer receives its full shipments. Constraint (3) insures that each customer sends its full shipments. Constraint (4) is the distribution center constraint. All vehicles depart from the distribution center and return to the distribution center after completing all distribution tasks. Constraint (5) indicates that the number of vehicles entering and leaving the customer point is balanced. Constraint (6) is weight constraint at the node. The weights of shipments change after the vehicle serves the customer j. Constraint (7) is volume constraint on the node. The volumes of shipments change after the vehicle serves the customer j. Constraint (8) is vehicle maximum load constraint. The total weight of shipments loaded by the vehicle on any section of the distribution path is less than the maximum vehicle load. Constraint (9) is the vehicle maximum volume constraint. The total volume of shipments loaded in the vehicle in any section of the distribution path is less than the maximum volume of the vehicle. Constraint (10) is the customer point service constraint. Each customer point is served at least once. Constraints (11) and (12) represent the required number of vehicle constraints.

3. A Genetic-Simulated Hybrid Algorithm for Solving the GSVRPSDP

For solving the proposed GSVRPSDP efficiently, a hybrid meta-heuristic (called GA-SA) that combines the global search ability of GA and the local search ability of SA is proposed. The overall procedure of GA-SA is shown in Algorithm 1. It can be seen that, GA-SA follows the basic framework of evolutionary computation, and the basic idea of SA is used to update individuals further after the offspring is generated based on GA operators (i.e., crossover, mutation and reproduction). The individuals updating based on the SA is a loop that has three substeps: (1) rescheduling customers, (2) optimizing loading and (3) optimizing routes. The loop is controlled by a temperature, it has an initial value that is bigger than 0. And each time the three steps proceed, the temperature declines. When the temperature is no bigger than 0, it stops, and the new population is generated. However, if all individuals in the offspring are updated based on SA, the time consumed could be very high. Hence, a certain proportion of offspring (i.e., $\alpha \times POPSIZE$) is selected to be updated. In the experiments, parameter analysis is conducted to give a proper value of α.

Algorithm 1: GA-SA for solving the GSVRPSDP

Randomly initialise *POPSIZE* individuals
Initialize the temperature T
Calculate the fitness of the individuals based on Equation (1) **while** *termination condition not satisfied* **do**
  Generate offsprings using GP operators
  Randomly select $\alpha \times POPSIZE$ individuals
  for *each selected individual* T_i **do**
    $Fit_{best}=fit(T_i)$ /* recording the best fitness during updating T_i */
    $T_{best}=T_i$ /* recording the best individual during updating T_i */
    while $T \geqslant 0$ **do**
      Rescheduling customers
      Optimizing loading
      Optimizing routes
      Evaluate the new individual T_i'
      if $fit(T_i') < fit(T_i)$ **then**
        $Fit_{best}=fit(T_i')$
        $T_{best}=T_i'$
      $T = T - 10$
    $T_i \leftarrow T_{best}$
Return best solution

In the following section, we first introduce the solution representation, and then three substeps of SA in detail.

3.1. Solution Representation and Evaluation

Adopting the route-first split-second idea for solving routing problems [37], the chromosome in GA-SA is a permutation of N customers without delimiters. Each customer is represented by its number, from 1 to N. It implies that the vehicle will follow the shortest paths between consecutive customers. Then greedy strategy is used to split the chromosome into several feasible routes that meet the weight and volume constraints of vehicles.

For a given chromosome $c = (\tau_1, \tau_2, \ldots, \tau_N)$, the first vehicle starts from the depot and heads to serve the customer τ_1, then it follows the customer permutation in the chromosome to serve as many customers as possible. Then, a second vehicle will start from the depot and go to serve the customer afterwards. The procedure is repeated until the final customer in the chromosome is totally distributed to one vehicle. In this process, the vehicle will deliver or pick up shipments according to the order of numbering from small to large and adhere to the capacity and volume constraints. Furthermore, in the real world, it is common for customers to have no pickup demands. In such cases, the vehicle simply skips over that customer and continues on its route. If a former vehicle cannot meet the delivery or pickup demands of one customer, it will serve as many as possible and the unserved demands are fulfilled by the next vehicle. Along with each route, there are two permutations that provide the delivery and pickup orders of shipments for each vehicle.

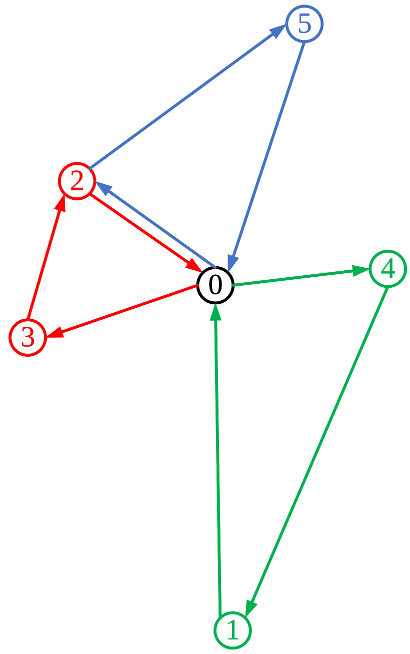
One example is shown in Figure 2, the chromosome is "[3, 2, 5, 4, 1]". After serving customer 3, the first vehicle can pick up part of the shipments of customer 2, then customer 2 is fulfilled by two vehicles. Details about the actual routes based on the chromosome and the delivery and pickup lists can be seen in Figure 2.

Chromosome: [3, 2, 5, 4, 1]

Route: **[[0, 3, 2, 0], [0, 2, 5, 0], [0, 4, 1, 0]]**

Delivery list: $[[d_3^1, d_3^2, d_3^3, d_3^4, d_2^1, d_2^2, d_2^3], [d_5^1, d_5^2, d_5^3], [d_4^1, d_4^2, d_1^1, d_1^2]]$

Pickup list: $[[p_2^1, p_2^2, p_2^3], [p_2^4, p_5^1], [p_4^1]]$



M_W=15 kg
M_V=15 m^3

Data of the instance example									
Serial No.	X	Y	No. of Delivery	Volume Set (m^3)	Weight Set (kg)	No. of Pickup	Volume Set (m^3)	Weight Set (kg)
0	25	25	--	--	--	--	--	--
1	26	5	2	{4, 5}	{5, 5}	0	--	--
2	17	31	3	{1.2, 1.8, 1}	{2, 3, 2}	4	{1, 8.5, 4.2, 2}	{5, 4, 5.6, 2}
3	14	22	4	{2, 2, 1, 4}	{2, 3, 0.5, 2.5}	0	--	--
4	35	26	2	{1, 4}	{2, 1}	1	{4}	{1}
5	30	42	3	{7, 3, 2}	{3, 9, 1}	1	{7}	{3}

Figure 2. An example of the solution representation. Each route is denoted by a different color. In the delivery list, d_i^j represents delivery the jth shipment of the customer i, and in the pickup list, p_i^j represents pick up the jth shipment of the customer i.

The costs of the routes are calculated based on Equation (1), which is defined as the fitness of the chromosome. The lower fitness indicates the better performance.

3.2. Initial Population and Termination Condition

Chromosomes in the initial population are generated either by the greedy strategy or at random. The greedy strategy starts with an empty route. It randomly chooses a customer first and adds it into the route. In each iteration, the nearest customer to the last customer in the route is added. The step is repeated until all the customers are added.

The GA-SA stops when a maximal number of generations is reached.

3.3. Individuals Updating Based on SA

The individuals that are chosen will be updated based on the following three steps in one loop.

3.3.1. Rescheduling Customers

First, the individual is viewed as a chromosome without delimiters. Then, the Metropolis rule in SA is used to reschedule customers to vehicles. That is, if a randomly generated value is bigger than the probability $P(T)$, the vehicle will serve the next customer based on the permutation in the chromosome. Otherwise, the vehicle will go back to the depot, and the customer will be served by a new vehicle. The goal of this process is to improve the diversity of the population.

The acceptance probability $P(T)$ is an equation relating to the temperature T, as shown in Equation (13). As the evolution process proceeds, the temperature T gradually decreases, and the probability of abandoning the next customer gradually decreases, which effectively increases the convergence speed of the algorithm while enhancing the global convergence of the algorithm.

$$P(T) = 1 - \frac{T}{100} \tag{13}$$

3.3.2. Optimizing Loading

The loading optimization mainly acts on the customers that are served by two vehicles. The picked-up or delivered shipments are reassigned to the first vehicle, and the remaining shipments will be served by the second vehicle. Meanwhile, the second vehicle will be re-evaluated for how many customers it can serve.

Taking the instance in Figure 2 as an example, customer 2 has three delivery shipments (represented as d_2^1, d_2^2 and d_2^3) and four pickup shipments (represented as p_2^1, p_2^2, p_2^3 and p_2^4). Before loading optimization, vehicle 1 picks up shipments p_2^1, p_2^2 and p_2^3. The remaining pickup shipments (i.e., p_2^4) will be served by vehicle 2, which can be seen in the pickup list in Figure 2. The loading optimization will randomly reassign shipments that are delivered and picked up by vehicle 1 under the control of weight and volume constraints. Hence, the new pickup list could be [[p_2^1, p_2^2, p_2^4], [p_2^3, p_5^1], [p_4^1]].

3.3.3. Optimizing Routes

This process aims to change the serving order of customers within the service scope of a vehicle, which could further reduce the vehicle variable costs. The specific adjustment method is determined by the number of customers served by a vehicle. (1) If the number of customers is one or two, the serving order will not be changed because it has no influence on the final costs. (2) If the number of customers is three (e.g., 1 2 3), there are six combination patterns, i.e., 1 2 3, 1 3 2, 2 1 3, 2 3 1, 3 1 2 and 3 2 1. The algorithm first checks whether these routes are feasible or not, that is, whether they can satisfy the weight and volume constraints. Then, the feasible routes that have the minimal costs will be selected. (3) If the number of customers is bigger than three, the algorithm will first randomly select ten combination patterns and then select one feasible minimal costs route from them.

4. Experiment Setup

This section first presents the designed datasets that are consistent with the problem model proposed in the paper. Then, the parameter settings are given. Finally, the key parameter α, which represents the number of individuals being updated based on SA, is analyzed, and the optimal values are recommended. The algorithms were implemented via Spyder (Python 3.9) programming on a computer configured with AMD Ryzen 72,700 U, 2.20 GHz, 8 GB RAM, running under Windows 10 OS.

4.1. Experimental Data

Since there are no standard datasets matching the problem modeled in this paper, we modified the benchmark SDVRP datasets (http://neumann.hec.ca/chairedistributique/data/sdvrp/ (accessed on 15 July 2022)) to GSVRPSDP dataset. The datasets consist of four groups of instances, which have 8, 16, 32 and 64 customers. For the instance with eight customers, the total number of both pickup and delivery shipments is 50. For the instance with 16 customers, the total numbers of both pickup and delivery shipments are 100. For the instance with 32 customers, the total number of both pickup and delivery shipments is 200. And for the instance with 64 customers, the total number of both pickup and delivery shipments is 400. Table 2 shows the instance with eight customers, in which 1–8 denote customers, and 0 is the depot. X and Y represent the x-coordinates and y-coordinates of depot and customers. Each customer has a different number of shipments to be delivered and picked up. And each shipment has a specific volume and weight. For example, line 4 in Table 2 shows that the coordinate of customer 2 is (17, 31); it has three shipments that need to be delivered, the volumes of these goods are 1.3 m^3, 1.1 m^3 and 0.1 m^3 and their corresponding weights are 1.2 kg, 2.2 kg and 0.2 kg. Meanwhile, customer 2 has five shipments that need to be picked up, whose volumes are 0.9 m^3, 0.9 m^3, 1.1 m^3, 1.2 m^3 and 1.7 m^3, and the corresponding weights are 0.3 kg, 2.0 kg, 0.6 kg, 0.5 kg and 1.7 kg.

Table 2. VRPSPSP data with 8 customers.

Serial No.	X	Y	No. of Delivery	Volume Set (m^3)	Weight Set (kg)	No. of Pickup	Volume Set (m^3)	Weight Set (kg)
0	25	25	-	-	-	-	-	-
1	26	5	12	{2.5, 2.0, 1.1, 0.5, 0.8, 2.0, 1.7, 0.7, 1.7, 0.8, 2.1, 1.6}	{1.6, 1.2, 0.5, 1.9, 1.4, 1.5, 1.4, 1.0, 1.8, 0.7, 1.7, 2.1}	6	{1.7, 1.9, 2.5, 1.2, 1.3, 1.2 }	{1.9, 1.2, 2, 1.4, 1.1, 2.1}
2	17	31	3	{1.3, 1.1, 0.1}	{1.2, 2.2, 0.2}	5	{0.9, 0.9, 1.1, 1.2, 1.7}	{0.3, 2.0, 0.6, 0.5, 1.7}
3	14	22	4	{1.5, 1.7, 0.1, 1.9}	{0.1, 1.5, 1.6, 0.8}	12	{0.8, 1.6, 2.3, 1.4, 0.7, 1.2, 0.4, 1.9, 1.2, 2.2, 1.7, 2.3}	{1.7, 0.7, 2.1, 1.1, 2.2, 2.2, 0.2, 2.3, 1.4, 1.7, 1.7, 1.1}
4	35	26	5	{1.3, 0.7, 1.2, 2.1, 1.1}	{1.7, 1.1, 1.4, 1.9, 1.4}	8	{1.1, 2.1, 0.6, 2.5, 2.5, 1.8, 1.1, 2.2}	{0.7, 1.5, 2.5, 1.2, 1.8, 0.3, 2.4, 1.5}
5	30	42	10	{1.2, 0.9, 0.1, 1.0, 1.2, 1.7, 1.4, 2.3, 0.1, 0.5}	{2.1, 0.4, 0.2, 1.4, 0.9, 0.8, 1.6, 1.1, 0.6, 0.9}	4	{1.0, 0.5, 2.3, 2.1}	{1.9, 2.3, 0.9, 1.5}
6	10	11	8	{2.4, 0.8, 1.1, 1.7, 1.5, 0.5, 1.6, 0.7}	{1.8, 2.1, 1.5, 0.9, 2.2, 1.1, 0.6, 0.9}	3	{1.1, 2.0, 1.5}	{2.1, 0.8, 1.1}
7	41	21	3	{1.5, 1.1, 1.3}	{1.2, 1.9, 0.8}	5	{2.5, 1.8, 1.1, 1.3, 1.9}	{1.8, 1.3, 0.4, 1.6, 1.2}
8	44	12	5	{0.9, 2.5, 1.2, 1.4, 1.1}	{1.6, 1.9, 1.5, 2.0, 1.3}	7	{0.4, 0.9, 2.2, 0.6, 1.2, 1.3, 2.4}	{0.5, 0.8, 2.2, 1.2, 1.5, 0.6, 0.1}

4.2. Parameter Settings

The superiority of the proposed GA-SA is shown by comparing it with the traditional meta-algorithms, i.e., GA and PSO, in experiments. Therefore, the parameter settings can be divided into three categories: (1) the vehicle parameters such as vehicle load and volume, (2) the GA-SA parameters, which include the settings for the traditional GA and (3) the PSO parameters. The settings of all parameter values are consistent with the standard algorithms, and the same parameters that appeared in different algorithms have the same values. Please see Table 3 for details.

Table 3. The parameter settings of the compared algorithms.

Parameter	Description	Value
M	Vehicle load	15 kg
V	Vehicle volume	15 m^3
C_0	Vehicle start-up costs	6
C_1	Vehicle change costs	1

Table 3. *Cont.*

Parameter	Description	Value
P_c	Crossover probability	0. 9
P_m	Mutation probability	0. 1
POPSIZE	Population size	40
Generations	Evolutionary algebra	50
tournament_size	Select operation	5
Tend	Termination temperature	1
w	Inertia factor	0.2
C_1	Self-perception factor	0.4
C_2	Social perception factor	0.4

4.3. Sensitivity Analysis (α)

In order to find a good α value that can balance the run costs and the searching ability of GA-SA, we investigated seven α values, i.e., 0, 0.1, 0.2, 0.4, 0.6, 0.8 and 1.0. $\alpha = 0$ means that no individuals are updated based on SA, which degenerates into the traditional GA, and $\alpha = 1.0$ means that all individuals are updated based on SA. The experiments are taken on GSVRPSDP dataset with 8 customers. And based on our preliminary experiments, results on other instances show the same pattern. The results are shown in Figure 3, which illustrate that with the increment of α, the performance of GA-SA (i.e., the costs) has a high improvement, but the run time rises sharply. For example, when α = 0, the averaged cost is 302.56 and the run time is 536 s. When α = 0.6, the cost is 288.57 and the run time is 2423 s. It means that if 60% individuals are updated based on SA, the run time has almost 5 times higher than the traditional GA. However, when $\alpha > 0.4$, with the increment of α, the improvement of the costs is slight. Hence, we recommend $\alpha \in [0.2, 0.4]$ for GA-SA. Specifically, we set $\alpha = 0.2$ in the following experiments.

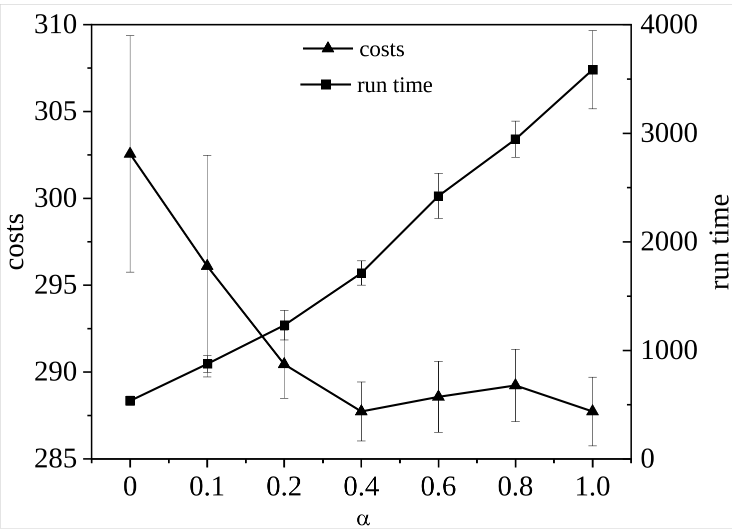


Figure 3. The costs and run time of GA-SA with different values of α. The results are the average values based on 30 independent runs.

5. Results and Analysis

In the new algorithm, GA-SA, we proposed three substeps to improve the performance of individuals in GA. In order to test the efficiency of each step specially, we run the algorithm with rescheduling customers only (called GA-SA$_R$) in the experiments. The other compared algorithms are the traditional GA, PSO and SA. Each algorithm was run 30 times independently for each instance, and the best, worst and average total costs were taken.

5.1. Results

Table 4 shows the total costs of GA-SA, GA-SA$_R$, the traditional GA, PSO and SA on four VRPSPSD instances. Bold values indicate the superiority of the data. The percentages are calculated as ($CAlg$.-(GA-SA))/$CAlg$., where $CAlg$. represents the costs obtained by the compared algorithm (e.g., GA, PSO). From the table, we have the following observations:

- GA-SA$_R$ can obtain lower total costs than the traditional GA in terms of best, worst and average. It shows that rescheduling customers based on a probability can help GA get out of the local optimal and obtain better routes.
- GA-SA outperforms GA-SA$_R$ in four instances, which means that the loading optimization and route optimization can further improve the diversity of individuals and achieve much better results.
- Among all five compared algorithms, GA-SA obtains the lowest costs in terms of best, worst and average in four instances. Specifically, the average total costs of GA-SA in four instances are 4.7%, 10.4%, 19.2% and 13.4% lower than that of GA-SA$_R$, GA, PSO and GA, respectively. It shows the efficiency of the proposed strategy.

Table 4. The total costs comparison among GA-SA, GA-SA$_R$, GA, PSO and SA for four GSVRPSDP instances.

Customer No.	Algorithm	Best	Average	Worst
8	GA-SA	**286.8**	**288.2**	**290.9**
	GA-SA$_R$	286.8	289.8	291.8
	GA	294.6	297.3	305.3
	PSO	300.7	307.8	318.2
	SA	294.7	301.2	307.2
16	GA-SA	**542.4**	**561.1**	**571.3**
	GA-SA$_R$	547.6	565.1	583.6
	GA	558.4	573.8	587.6
	PSO	581.4	589.5	601.5
	SA	566.2	578.5	592.1
32	GA-SA	**1161.8**	**1204.5**	**1251.9**
	GA-SA$_R$	1187.9	1221.3	1321.6
	GA	1265.7	1289.2	1339.6
	PSO	1422.6	1476.7	1573.1
	SA	1314.1	1325.9	1364.5
64	GA-SA	**2639.9**	**2787.7**	**2994.6**
	GA-SA$_R$	2897.3	3002.3	3276.2
	GA	3194.5	3240.3	3348.4
	PSO	3512.1	3620.5	3766.4
	SA	3340.3	3384.9	3452.1
Average of four instances	GA-SA	**1157.7**	**1210.4**	**1277.2**
	GA-SA$_R$	1229.9 (5.9%)	1269.6 (4.7%)	1368.3 (6.7%)
	GA	1328.3 (12.8%)	1350.2 (10.4%)	1395.2 (8.5%)
	PSO	1454.2 (20.4%)	1498.6 (19.2%)	1564.8 (18.4%)
	SA	1378.8 (16.0%)	1397.6 (13.4%)	1429.0 (10.6%)

Figure 4 shows the convergence curves of the average costs of the compared algorithms for four instances, which shows the superiority of GA-SA more visually. It can be summarized that PSO is obviously worse than the other algorithms. GA is easy to fall into a local optimal solution, which leads to program interruption, and GA-SA$_R$ can jump out of the local optimal solution and achieve a better result due to the introduction of a random factor in rescheduling customers, which accepts a solution worse than the current one with a certain probability when iteratively updating the feasible solution. GA-SA shows further better convergence performance than GA-SA$_R$.

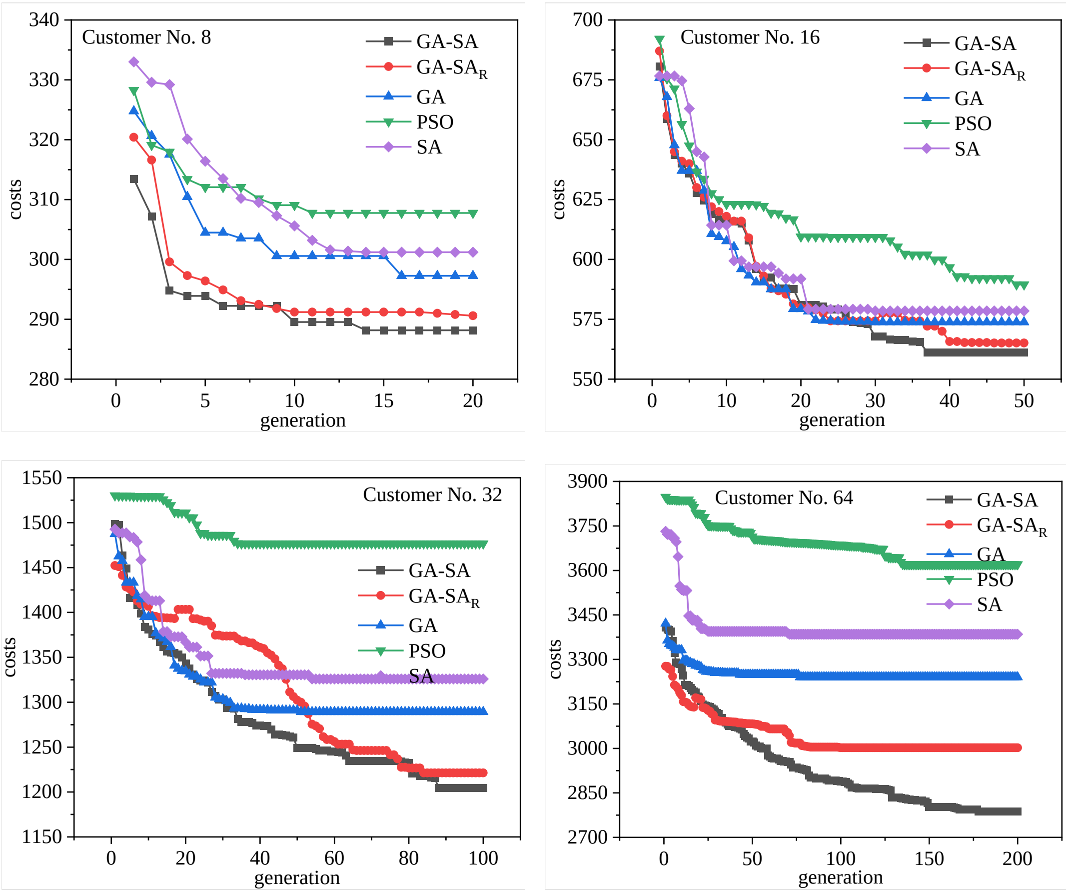


Figure 4. The convergence curves of the average costs of compared algorithms on four instances.

5.2. Effectiveness Analysis

For analyzing the routes generated by GA-SA in detail, we randomly choose a solution of GA-SA on the instance with 16 customers and calculate the space utilization and capacity utilization of vehicles. The results are shown in Table 5. It can be seen that the average space utilization and the average capacity utilization of GA-SA is 86.1% and 88.9%, respectively, which are much better than that of GA and PSO.

As presented in Section 3.1, if the shipments of a customer cannot be delivered or picked up fully by one vehicle, GA uses a direct method, according to the order of numbering from small to large, to assign shipments to the first vehicle, and the remaining shipments are assigned to the second vehicle. Meanwhile, in GA-SA, loading optimization is introduced to break the numbering constraints and randomly reassigns shipments that are delivered and picked up by the first vehicle under the control of weight and volume constraints. This is helpful to fully utilize the loading capacity of the first vehicle, and the second vehicle can have more capacity to serve more customers. For example, the routes show that customers 16 and 12 are served twice, which indicates that it is very common that the customers are served by two vehicles and reassigning shipments among them could result in a better way to load shipments. To look at each route further, we found that vehicle 2 has the lowest capacity utilization, which is 72.7%, but its space utilization rate reaches 93.3%. This indicates that a large proportion of goods transported by vehicle 2 are light goods.

Table 5. An example of routes generated by GA-SA at the instance with 16 customers.

	Routes	Space Utilization	Capacity Utilization
Vehicle 1	[0, **16**, 8, 10, 0]	92.7%	98.7%
Vehicle 2	[0, **16**, 7, 0]	**93.3%**	**72.7%**
Vehicle 3	[0, **12**, 4, 0]	78.7%	81.3%
Vehicle 4	[0, **12**, 3, 0]	90.7%	96.7%
Vehicle 5	[0, 13, 3, 2, 0]	71.3%	77.3%
Vehicle 6	[0, 11, 1, 13, 0]	97.3%	98.7%
Vehicle 7	[0, 1, 15, 0]	82.7%	94.7%
Vehicle 8	[0, 6, 15, 0]	79.3%	99.3%
Vehicle 9	[0, 5, 14, 0]	95.3%	93.3%
Vehicle 10	[0, 9, 14, 0]	79.8%	76.7%
GA-SA Average		**86.1%**	**88.9%**
GA Average		71.2%	74.8%
PSO Average		60.9%	65.7%

5.3. *Analysis of the Superiority of Simultaneous Pickup and Delivery*

Separate pickup plus separate delivery means pickup and delivery are two independent tasks. In this section, we conduct experiments to compare the simultaneous pickup and delivery and separate pickup plus separate delivery.

From Table 6, we found that the costs of separate pickup and separate delivery for the same transportation task are more than 80% higher than the costs of simultaneous pickup and delivery. This proves the high efficiency and low costs of simultaneous pickup and delivery. Hence, it is necessary to include simultaneous pickup and delivery in the vehicle-routing-related problems.

Table 6. Comparison of simultaneous pickup and delivery results.

	Delivery	Pickup	Mean
8 Customers Instance			
pickup and delivery	-	-	288.2
delivery + pickup	254.3	271.4	525.7
Growth multiplier	-	-	**82.4%**
16 Customers Instance			
pickup and delivery	-	-	561.1
delivery + pickup	515.6	500.7	1016.3
Growth multiplier	-	-	**81.1%**

6. Conclusions

In this paper, the Granularity-based Split Vehicle Routing Problem with Simultaneous Delivery and Pickup (GSVRPSDP) was formulated. In GSVRPSDP, the demands of customers consist of the number of shipments that have specific volumes and weights. This was suitable for various types of shipments in the real application, for example bulk items and industrial shipments in business transportation, packages including food, clothing, electronics and medications in e-commerce transportation, and even workers and tools in line maintenance. Meanwhile, simultaneous delivery and pickup is very common in real-world transportation. For example, in e-commerce transportation, delivery companies need to deliver multiple packages to different destinations, while also requiring pickups of returned packages from them. In line maintenance, utility companies need to dispatch multiple workers or tools to different work sites for line maintenance, while also requiring pickups along the way.

A hybrid meta-heuristic algorithm called GA-SA was implemented for solving the GSVRPSDP. GA-SA proposed three substeps, including rescheduling customers, optimizing loading and optimizing routes, to help individuals escape from the local optima and achieve further improvements. The effectiveness of GA-SA was verified by comparing it with GA-SA$_R$ (i.e., GA-SA with rescheduling customers only), GA, PSO and SA on four instances, with the number of customers ranging from 8 to 64. The results demonstrated that GA-SA could obtain the lowest transportation costs among the compared algorithms, with reductions of 4.7%, 10.4%, 19.2% and 13.4% compared to GA-SA$_R$, GA, PSO and SA, respectively. Moreover, further analysis revealed that the vehicle space utilization and capacity utilization of the routes obtained using GA-SA were 15–25% higher than those obtained using GA and PSO.

For future work, in order to meet more complex requirements in the real world, many other factors could be added into GSVRPSDP: (1) Uncertainty: in reality, some information about the customers (e.g., the exact weights and volumes of shipments) and roads (e.g., whether it can be travelled or not) is often unknown in advance, and is revealed dynamically while the services are being conducted. (2) Time windows: customers may have different requirements for the service time, and some customers may request a delivery as soon as possible, while others may need to receive the goods on a specific time. (3) Multi-objective optimization: it may be necessary to optimize for multiple objectives simultaneously, such as minimizing the total distance traveled while also minimizing the number of vehicles used. (4) Multiple depots: there may be multiple depots or warehouses that need to be serviced by the vehicles.

Author Contributions: Methodology, Y.L.; validation, Z.Q.; formal analysis, Y.L.; writing—original draft, Y.L.; writing—review & editing, Y.L. and J.L.; supervision, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key Research and Development Program of China (No. 2021YFC2801001) and the Shanghai Sailing Program (No. 21YF1416700).

Data Availability Statement: The data presented in this study are available from https://yxinliu.github.io/ (accessed on 6 July 2023).

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

References

1. Adewumi, A.; Adeleke, O. A Survey of Recent Advances in Vehicle Routing Problems. *Int. J. Syst. Assur. Eng. Manag.* **2018**, *9*, 155–172. [CrossRef]
2. Pillac, V.; Gendreau, M.; Guéret, C.; Medaglia, A.L. A Review of Dynamic Vehicle Routing Problems. *Eur. J. Oper. Res.* **2013**, *225*, 1–11. [CrossRef]
3. Sluijk, N.; Florio, A.M.; Kinable, J.; Dellaert, N.; Van Woensel, T. Two-echelon Vehicle Routing Problems: A Literature Review. *Eur. J. Oper. Res.* **2023**, *304*, 865–886. [CrossRef]
4. Dantzig, G.B.; Ramser, J.H. The Truck Dispatching Problem. *Manag. Sci.* **1959**, *6*, 80–91. [CrossRef]
5. Worawattawechai, T.; Intiyot, B.; Jeenanunta, C.; Ferrell, W.G. A learning enhanced golden ball algorithm for the vehicle routing problem with backhauls and time windows. *Comput. Ind. Eng.* **2022**, *168*, 108044. [CrossRef]
6. Wu, Y.; Cai, Y.; Fang, C. Evolutionary Multitasking for Bidirectional Adaptive Codec: A Case Study on Vehicle Routing Problem with Time Windows. *Appl. Soft. Comput.* **2023**, *145*, 110605. [CrossRef]
7. Chen, J.F.; Wu, T.H. Vehicle Routing Problem with Simultaneous Deliveries and Pickups. *J. Oper. Res. Soc.* **2006**, *57*, 579–587. [CrossRef]
8. Wu, H.; Gao, Y. An Ant Colony Optimization Based on Local Search for the Vehicle Routing Problem with Simultaneous Pickup-delivery and Time Window. *Appl. Soft Comput.* **2023**, *139*, 110203. [CrossRef]
9. Subramanian, A.; Penna, P.H.V.; Uchoa, E.; Ochi, L.S. A Hybrid Algorithm for the Heterogeneous Fleet Vehicle Routing Problem. *Eur. J. Oper. Res.* **2012**, *221*, 285–295. [CrossRef]
10. Stavropoulou, F. The Consistent Vehicle Routing Problem with heterogeneous fleet. *Comput. Oper. Res.* **2022**, *140*, 105644. [CrossRef]
11. Muñoz-Herrera, S.; Suchan, K. Local Optima Network Analysis of Multi-Attribute Vehicle Routing Problems. *Mathematics* **2022**, *10*, 4644. [CrossRef]
12. Zhang, J.; Zhu, Y.; Li, X.; Ming, M.; Wang, W.; Wang, T. Multi-Trip Time-Dependent Vehicle Routing Problem with Split Delivery. *Mathematics* **2022**, *10*, 3527. [CrossRef]
13. Alshraideh, M.; Mahafzah, B.; Al-Sharaeh, S.; Hawamdeh, Z. A Robotic Intelligent Wheelchair System Based on Obstacle Avoidance and Navigation Functions. *J. Exp. Theor. Artif. Intell.* **2015**, *27*, 471–482. [CrossRef]

14. Min, H. The Multiple Vehicle Routing Problem with Simultaneous Delivery and Pickup. *Transp. Res. Part A Gen.* **1989**, *23*, 377–386. [CrossRef]
15. Gajpal, Y.; Abad, P. An Ant Colony System (ACS) for Vehicle Routing Problem with Simultaneous Delivery and Pickup. *Comput. Oper. Res.* **2009**, *36*, 3215–3223. [CrossRef]
16. Polat, O.; Kalayci, C.B.; Kulak, O.; Günther, H.O. A Perturbation Based Variable Neighborhood Search Heuristic for Solving the Vehicle Routing Problem with Simultaneous Pickup and Delivery with Time Limit. *Eur. J. Oper. Res.* **2015**, *242*, 369–382. [CrossRef]
17. Xu, D.; Li, K.; Zou, X.; Liu, L. An Unpaired Pickup and Delivery Vehicle Routing Problem with Multi-visit. *Transp. Res. Part E Logist. Trans. Rev.* **2017**, *103*, 218–247. [CrossRef]
18. Bouanane, K.; Amrani, M.E.; Benadada, Y. The Vehicle Routing Problem with Simultaneous Delivery and Pickup: A Taxonomic Survey. *Int. J. Logist. Syst. Manag.* **2022**, *41*, 77–119. [CrossRef]
19. Dror, M.; Trudeau, P. Split Delivery Routing. *Nav. Res. Logist.* **1990**, *37*, 383–402. [CrossRef]
20. Khmelev, A.; Kochetov, Y. A Hybrid VND Method for the Split Delivery Vehicle Routing Problem. *Electron. Notes Discret. Math.* **2015**, *47*, 5–12. [CrossRef]
21. Archetti, C.; Bianchessi, N.; Speranza, M.G. A Column Generation Approach for the Split Delivery Vehicle Routing Problem. *Networks* **2011**, *10*, 241–254. [CrossRef]
22. Wilck, J.H., IV; CAavalier, T.M. A Genetic Algorithm for the Split Delivery Vehicle Routing Problem. *Am. J. Oper. Res.* **2012**, *2*, 207–216.
23. Berbotto, L.; Garcia, S.; Nogales, F. A Randomized Granular Tabu Search Heuristic for the Split Delivery Vehicle Routing Problem. *Ann. Oper. Res.* **2014**, *222*, 153–173. [CrossRef]
24. Bortfeldt, A.; Yi, J. The Split Delivery Vehicle Routing Problem with Three-Dimensional Loading Constraints. *Eur. J. Oper. Res.* **2020**, *282*, 545–558. [CrossRef]
25. Nowak, M.; Ergun, O.; White, C.C., III. An Empirical Study on the Benefit of Split Loads with the Pickup and Delivery Problem. *Eur. J. Oper. Res.* **2009**, *198*, 734–740. [CrossRef]
26. Chen, Q.; Li, K.; Liu, Z. Model and Algorithm for an Unpaired Pickup and Delivery Vehicle Routing Problem with Split Loads. *Transp. Res. Part E Logist. Transp. Rev.* **2014**, *69*, 218–235. [CrossRef]
27. Nowak, M.; Ergun, Ö.; White, C.C., III. Pickup and Delivery with Split Loads. *Transp. Sci.* **2008**, *42*, 32–43. [CrossRef]
28. Salazar-González, J.J.; Santos-Hernández, B. The Split-Demand One-Commodity Pickup-and-Delivery Travelling Salesman Problem. *Transp. Res. Part B Methodol.* **2015**, *75*, 58–73. [CrossRef]
29. Şahin, M.; Çavuşlar, G.; Öncan, T.; Şahin, G.; Aksu, D.T. An Efficient Heuristic for the Multi-Vehicle One-To-One Pickup and Delivery Problem with Split Loads. *Transp. Res. Part C Emerg. Technol.* **2013**, *27*, 169–188. [CrossRef]
30. Li, Z.P.; Zhao, F.; Liu, H.W. Intelligent Water Drops Algorithm for Vehicle Routing Problem with Multiple Time Windows. *Oper. Res. Manag. Sci.* **2015**, *24*, 1–10.
31. Wolfinger, D. A Large Neighborhood Search for the Pickup and Delivery Problem with Time Windows, Split Loads and Transshipments. *Comput. Oper. Res.* **2021**, *126*, 105110. [CrossRef]
32. Ren, T.; Luo, T.; Jia, B.; Yang, B.; Wang, L.; Xing, L. Improved Ant Colony Optimization for the Vehicle Routing Problem with Split Pickup and Split Delivery. *Swarm Evol. Comput.* **2023**, *77*, 101228. [CrossRef]
33. Bortfeldt, A. A Hybrid Algorithm for the Capacitated Vehicle Routing Problem with Three-Dimensional Loading Constraints. *Comput. Oper. Res.* **2012**, *39*, 2248–2257. [CrossRef]
34. Yu, V.F.; Aloina, G.; Susanto, H.; Effendi, M.K.; Lin, S.-W. Regional Location Routing Problem for Waste Collection Using Hybrid Genetic Algorithm-Simulated Annealing. *Mathematics* **2022**, *10*, 2131. [CrossRef]
35. Jiang, H.; Wu, Y.; Zhang, Q. Optimization of Ordering and Allocation Scheme for Distributed Material Warehouse Based on IGA-SA Algorithm. *Mathematics* **2020**, *8*, 1746. [CrossRef]
36. Su, X.X.; Wang, H.W.; Qin, H.; Wang, K. Hybrid Heuristic Algorithm for Solving Multi-Distributor Vehicle Path Problem. *Oper. Res. Manag.* **2022**, *31*, 42–47.
37. Lacomme, P.; Toussaint, H.; Duhamel, C. A GRASP × ELS for the Vehicle Routing Problem with Basic Three-Dimensional Loading Constraints. *Eng. Appl. Artif. Intell.* **2013**, *26*, 1795–1810. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.