Heterogeneous Item Delivery Problem

IOE 591 Project Report

Shuangwei Yu, Tianming Liu, Ziying Wang

December 18, 2021

Abstract

This paper focuses on the Vehicle Routing Problem that is essential for the e-commerce industry, explicitly taking into account heterogeneous items. The objective is to determine the minimum total logistics cost. We present a formal model, an integer program formulation, and a benchmark suite of 10 instances, and implement a solver suite of heuristic components. We add a penalty term to the multi-depot scenario based on real-life and customer experience considerations. The results suggest that the proposed method can find good solutions in a reasonable amount of time for a given dataset.

Keyword: Vehicle Routing, Heterogeneous Items, Integer Programming

1 Introduction

1.1 Motivation and problem description

The rapid development of the e-commerce industry has driven an explosive growth of logistics demand. In the fiscal year of 2020, USPS realized 80.94 billion total deliveries, increasing from 48.27 billion in FY 2019. Thus, saving costs plays an important role in logistics issues. The classical vehicle routing problem (VRP) assumes that customers have the same demands and vehicles have the same capacities. However, in real life, the volume of products ordered by each consumer is different, and the shapes and capacities of vehicles used in logistics are also different.

In this paper, we address a variant vehicle routing problem with heterogeneous packages. In the problem, each customer can order a list of products with different sizes and there could be multiple depots with inventories of all products. Vehicles are located at each depot, they are packed with different products to save packing space and then deliver products to customers on a daily basis. They can only be packed once and need to come back to the depots at the end of each day.

1.2 Literature review

The traditional vehicle routing problem (VRP) was proposed by Dantzig and Ramser (1959), dispatching vehicles from a single depot to service the demands of a set of customers [1]. On this basis, the multiple vehicle routing problem (MVRP) studies situations of multiple depots. It has been studied extensively and has considerable variants due to its wide application in logistics distribution problems. For instance, Ho *et al*. (2008) considered the grouping problem prior to the routing and scheduling problems, since there are additional depots for storing the products and which customers are served by which depots [2]. Wu *et al*. (2002) formulated the multi-depot location routing problem (MDLRP), determining simultaneously the number and locations of depots, assignment of customers to depots, vehicle types to routes, and the corresponding delivery routes [3]. Hadjiconstantinou and Baldacci (1998) presented a heuristic algorithm for the Multi-depot period vehicle routing problem (MDPVRP), they decomposed the problem into 4 levels [4]. Firstly, they decided the boundaries of each depot's service area. Then they solved PVRP for each depot. The third level was to solve a classical VPR for each depot in a given period. And the last step was to solve a classical TSP for each route. Salhi *et al.* (2014) considered the multi-depot heterogeneous vehicle routing problem (MDHFVRP), they developed a variable neighborhood search approach that incorporates new features in addition to the adaptation of several existing neighborhoods and local search operators to explore this dormant but practical logistical problem [5]. Derigs *et al.* (2010) considered heterogeneous products packed in the vehicles with compartments and formulated the vehicle routing problem with compartments (VRPC) [6]. In this model, in addition to the vehicles' capacity limitations, each compartment of the vehicle also has its own capacity restrictions. Avella *et al.* (2004) proposed a Set Partitioning formulation and presented a Branch-and-Price algorithm to solve the multi-period vehicle routing problem (MPVRP) where each tank must travel either completely full or completely empty [7]. Muyldermans and Pang (2010) discussed the multi-compartment vehicle routing problem (MCVRP) where clients have demands for different products and vehicles have several compartments to co-transport these commodities [8]. They presented a local search procedure and exploited the mechanisms of neighbor lists and marking to speed up the searches.

1.3 Organization of the paper

To solve the problem, we formulate an integer program formulation and implement a portfolio of different heuristic components. The rest of the paper is organized as follows. In Section 2, we formulate MILP models to compute total logistics cost for the one-depot routing problem and multi-depot routing problem respectively. We test our model on a dataset and visualize the results in section 3. In section 4, we conduct a sensitivity analysis on some model parameters for the multi-depot case. We conclude our report in section 5.

2 Problem Formulation

2.1 Single depot case

First, let us introduce the background of the problem and the notations.

There are L locations in the transportation network. $L = 0, 1, ..., n$. In the locations, $L_0 = 0$ is the depot and $L_c = L\backslash\{0\}$ are the customer locations. The vehicle travel cost from location i to location j is c_{ij}

Each customer may place multiple orders. The set of all placed orders is denoted by O. For each order $o \in O$, the customer who placed it is $customer(o)$, the product type of the order is $product(o)$, the product shape of the order is $shape(o)$, and the quantity of the order is $quantity(o)$. For each customer l in L_c, the set of orders he or she has placed is $ordCust(l)$

The company has a fleet V. Each vehicle $v \in V$ is homogeneous in size and capacity. The capacity of the vehicle is q and the shape of the vehicle is s.

The decision variables in this problem are:

- Binary variable b_{ijv} which denotes whether vehicle v traverse from location i to j in the solution.
- Binary variable x_{ov} which denotes whether order o is delivered by vehicle j
- Integer variable u_{iv} which marks the location of position i in the route of vehicle v. This is rather new but cannot be omitted.

The VRPC can be formulated as integer program:

$$\min_{b,x,u} \sum_{v\in V}\sum_{i\in L}\sum_{j\in L} c_{ij}b_{ijv} \tag{1a}$$

$$\text{subject to} \sum_{j\in L_c} b_{0jv} \leq 1 \qquad v \in V \tag{1b}$$

$$\sum_{i\in L} b_{ikv} = \sum_{j\in L} b_{kjv} \qquad v \in V, k \in L \tag{1c}$$

$$u_{iv} - u_{jv} + |L|b_{ijv} \leq |L_c| \qquad \forall v \in V, i \in L, j \in L_c \tag{1d}$$

$$u_{l_0 v} = 1 \qquad \forall v \in V \tag{1e}$$

$$\sum_{v\in V}\sum_{c\in C} x_{ov} = 1 \qquad o \in O \tag{1f}$$

$$\sum_{o\in ordCust(j)} x_{ov} \leq |O| \sum_{i\in L} b_{ijv} \qquad \forall v \in V, j \in L_c \tag{1g}$$

$$\sum_{o\in O} quantity(o)x_{ov} \leq q \qquad \forall v \in V \tag{1h}$$

$$\sum_{o\in O} shape(o)x_{ov} \leq s \qquad \forall v \in V \tag{1i}$$

$$x_{ov} = \{0,1\} \qquad \forall o \in O, v \in V \tag{1j}$$

$$b_{ijv} = \{0,1\} \qquad \forall i \in L, j \in L, v \in V \tag{1k}$$

$$u_{iv} = \{1, ..., |L|\} \qquad \forall i \in L, v \in V \tag{1l}$$

The objective function (1a) minimizes the total logistics cost. Constraint (1b) ensures that each vehicle v could be packed only once in the depot 0. Constraint (1c) implies that each vehicle v arriving at location k then departs from k. Constraints (1b) and (1c) together ensure that each vehicle v if packed must depart from and arrive at depot 0. Constraint (1d) states that all tours depart from depot 0, by imposing that the position of customer j is higher than the position of location i, if the vehicle v travels from i to j. Constraint (1e) guarantees the depot 0 to be at position 1 to eliminate duplicates. Constraint (1f) ensures that each order could be delivered by exactly one vehicle. Constraint (1g) implies that if customer j's order o is assigned to vehicle v, then vehicle v must visit customer j. Here $ordCust(j) = \{o \in O|customer(o) = j\}$ denotes the orders of customer j. Constraints (1h) and (1i) guarantee that goods packed on vehicle v could not exceed the capacity q and the shape s. Constraints (1j)-(1l) define the decision variables, x_{ov} and b_{ijv} are binary variables and uiv is an integer variable.

2.2 Multi-depot case

We keep as many variables as possible. There are L locations in the transportation network. There are two kinds of locations: depot and customer. For simplicity, the depot node and customer node do not overlap. $L_d = 0, 1, ..., d$ are the depots. $L_c = 0, 1, ..., l$ are the customers. The vehicle travel cost from location i to location j is c_{ij}

Each customer may place multiple orders. The set of all placed orders is denoted by O. For each order $o \in O$, the customer who placed it is $customer(o)$, the product type of the order is $product(o)$, the product shape of the order is $shape(o)$, and the quantity of the order is $quantity(o)$. For each customer l in L_c, the set of orders he or she has placed is $ordCust(l)$

The company has a fleet V. Each vehicle $v \in V$ is homogeneous in size and capacity. The capacity of the vehicle is q and the shape of the vehicle is s.

The decision variables in this problem are:

- Binary variable b_{ijv} which denotes whether vehicle v traverse from location i to j in the solution.
- Binary variable x_{ov} which denotes whether order o is delivered by vehicle j
- Integer variable t_{iv} denotes the sequence in the route when vehicle v reaches node i. This is for sub-tour elimination.

The VRPC can be formulated as integer program:

$$\min_{b,x,u} \sum_{v\in V}\sum_{i\in L}\sum_{j\in L} c_{ij}b_{ijv} \tag{2a}$$

$$\text{subject to} \sum_{i\in L_d}\sum_{j\in L_c} b_{ijv} \leq 1 \qquad v \in V \tag{2b}$$

$$\sum_{i\in L} b_{ikv} = \sum_{j\in L} b_{kjv} \qquad v \in V, k \in L \tag{2c}$$

$$\sum_{v\in V}\sum_{c\in C} x_{ov} = 1 \qquad o \in O \tag{2d}$$

$$\sum_{o\in ordCust(j)} x_{ov} \leq |O| \sum_{i\in L} b_{ijv} \qquad \forall v \in V, j \in L_c \tag{2e}$$

$$\sum_{o\in O} quantity(o)x_{ov} \leq q \qquad \forall v \in V \tag{2f}$$

$$\sum_{o\in O} shape(o)x_{ov} \leq s \qquad \forall v \in V \tag{2g}$$

$$\sum_{i\in L_d}\sum_{v\in V} t_{iv} = 0 \tag{2h}$$

$$s_{iv} + 1 \leq s_{jv} + 10^9(1 - x_{ijv}) \qquad \forall i \in L, j \in L_c, v \in V \tag{2i}$$

$$x_{ov} = \{0, 1\} \qquad \forall o \in O, v \in V \tag{2j}$$

$$b_{ijv} = \{0, 1\} \qquad \forall i \in L, j \in L, v \in V \tag{2k}$$

$$t_{iv} \in \mathbf{N} \qquad \forall i \in L, v \in V \tag{2l}$$

In this model, the objective function 2a minimizes the total vehicle travel cost. Constraint 2b ensures each vehicle is at most packed at one depot once. Constraint 2c ensures flow conservation in the solution. Constraint 2d ensures each order is delivered by one vehicle. Constraint 2e ensures the vehicle visits the customer it is required to serve in the routing solution. Constraint 2f and 2g ensures the capacity limit and packing shape limit of each vehicle are not violated. Constraint 2h ensures that the depot is the start of each vehicle's route by fixing the depot at the start of the driver's node visiting sequence. Constraint 2i ensures that if vehicle v travels from node i to customer node j in the route solution, then node j must appear later than node i in the vehicle's route sequence. This constraint eliminates sub-tours by not allowing one vehicle to visit one node twice and create a sub-tour. Constraint 2j and 2k define the order assignment variable x_{ov} and route traverse variable b_{ijv} be binary. The final constraint 2l defines the sequence variable t_{iv}, which denotes the position of node i in the route of vehicle v, to be non-negative.

2.3 Multi-depot Case with Penalty term

Naturally, the e-commerce company would want less operating cost for its fleet by a compact packing scene and optimal planning of vehicle routes. However, when one customer's order is delivered multiple times, the customer's experience might be reduced as they might have to retrieve the package multiple times. Also, packing all the customer's items together might save packing materials, which is beneficial to environmental sustainability.

Thus, when making the optimal routing decision, the company may also want to limit how many times the customers have been served. However, one customer's order might be too large to be packed into a certain number of vehicles. Thus, we introduce a penalty term to reduce the number of customer visits.

The formulation of the problem is similar to the multi-depot vehicle routing formulation of 2. The constraints stay the same but the objective function becomes:

$$\min_{b,x,u} \sum_{v\in V}\sum_{i\in L}\sum_{j\in L} c_{ij} b_{ijv} + \beta \sum_{v\in V}\sum_{i\in L}\sum_{j\in L_c} b_{ijv} \tag{3}$$

In the penalty term:

$$\beta \sum_{v\in V}\sum_{i\in L}\sum_{j\in L_c} b_{ijv} \tag{4}$$

$\sum_{v\in V}\sum_{i\in L}\sum_{j\in L_c} b_{ijv}$ is the total number of times the customer is visited by the fleet, which is the same as the number of total deliveries made to serve all orders. β is a regularization parameter that controls the importance of the total delivery number. This formulation does not introduce high-order equations in the problem. Thus, the new problem can be solved using the same solver and have similar complexity to the old problem.

3 Computational Result

We used a total of 10 instances in a benchmark set to find good parameter values and analyze the performance of our algorithm. These instances vary in size, with the capacity of vehicles ranging from 25 to 35, in between and at most 5 vehicles. Our method was implemented using Python and executed on an Apple M1 chip with 8GB memory.

3.1 Single Depot

In the case of single depot, we evaluated the performance of our method for using the instances of single depot with 3 dispatchable vehicles, each vehicle has an identical capacity of 30 units. For each order, the weight of it is ranged between 1 unit to 5 units and each customer has 1-3 orders that need to be delivered and there are 11 customers in total that need to be delivered.

The run-time of our base case is roughly 178s to solve and the minimum cost of fulfilling all the requirements is 47.6.

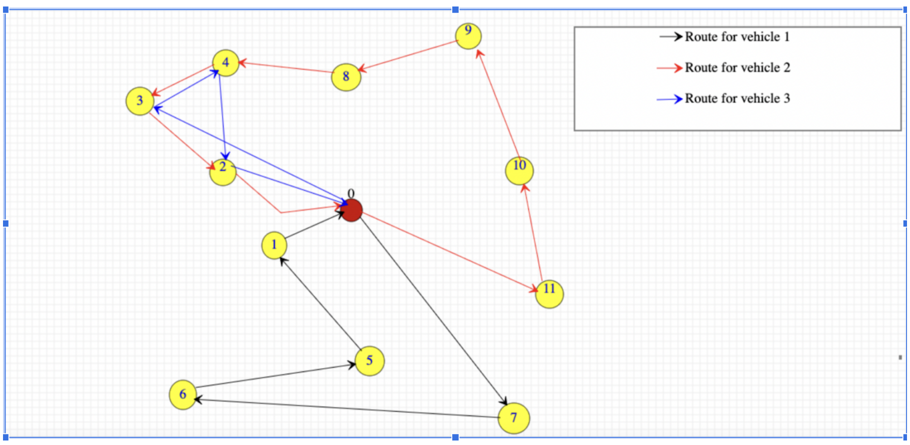


Figure 1: VRP for single depot case

Above is a visualization of the result that we've got. As you can see from the graph, each color of lines represents a route of one vehicle and each circle denotes a node. For nodes 2,3 and 4, there are multiple lines passing through them which means that their orders have been processed by different vehicles. The order assignment table is shown below:

Table 1: Comparison of Order Assignment

Order ID	Customer Node	Vehicle Assignment
1	2	0
2	3	2
3	3	2
4	4	1
5	4	0
6	5	1
7	5	0
8	5	1
9	6	0
10	6	0
11	7	0
12	7	0
13	7	0
14	8	0
15	8	0
16	8	0
17	9	1
18	10	1
19	10	1
20	10	1
21	11	1
22	11	1
23	12	1

Comparing the order assignment table and the optimal route, we can see that all the orders have been delivered by the assigned vehicle in the optimal route. This confirms the validity of our solution. Although node 2 appears 3 times in the optimal routes, it is because that passing through node 2 is included in the shortest path from depot 0 to node 3 and 4. So the solution's optimal route remains the same as from 0 directly to 3 and 4. Thus, the solution is still valid.

3.2 Effect of Penalty Term on Delivery Number

This solution is optimal given our current construction of code but it is definitely a scenario that real-life companies wanted to avoid since delivering the same order with different vehicles isn't financially beneficiary and also not user-friendly. So by adding a penalty term in the objection function, we re-run the code, trying to minimize the cross-delivery incidents, and here is the result.

By comparison, we can see that the cost of performing the optimal route with penalty increases. This doesn't make the case with penalty inferior to the case without penalty since we've altered the objective function by adding a penalty item. The average distance per vehicle increases from 15.8 to 17.6 while a penalty item is added. This is coherent with the result that the total visit of customers decreases from 13 to 11. By adding penalty term, all customer has been served by exactly one vehicle with the cost increasing the average traveling distance for all vehicles. Another important parameter change we need to spot on is the minimum usage of

Table 2: Effect of Penalty Term on the Optimal Solution

Case	No Penalty Term	With Penalty Term
Computation Time	178.6	549.7
Total Cost	47.6	52.6
Average Vehicle Cost	15.8	17.6
Minimum Vehicle Load	2	15
Total Customer Visit	13	11

Table 3: Effect of Penalty Term on Vehicle Routes

Case	No Penalty Term	With Penalty Term
Route of Vehicle 0	$0 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 0$	$0 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 1 \rightarrow 0$
Route of Vehicle 1	$0 \rightarrow 2 \rightarrow 0$	$0 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 0$
Route of Vehicle 2	$0 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 0$	$0 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 0$

vehicles increase from 2 to 15. This is very crucial to the real-life delivery system since we want to avoid the in-balance of items on each vehicle. The comparison of order assignment is shown in appendix figure 5. As shown in the figure, comparing with the penalty term case, customer 4 and 5 receives one additional delivery under the non-penalty case.

3.3 Base Case for Multi-Depot Scenario

After experiencing the single depot case with our code, we have run the optimization model for the base case scenarios for multi-depots. For the base case of multi-depot, we test our model with 2 depots and in a total of 3 vehicles distributed across the depots. The capacity of each vehicle is predetermined at the level of 30 and all other parameters are coherent with the base case of a single depot scenario.

The computation time for the multi-depot base case is 697.11s. The total delivery cost is 47.1. This is lower than the single-depot case. Because an additional depot that is closer to the problem is added.

The optimal route and order assignment for the multi-depot base case is shown below:

Table 4: Effect of Penalty Term on Vehicle Routes

Vehicle	0	1	2
Route	$0 \rightarrow 2 \rightarrow 12 \rightarrow 11 \rightarrow 0$	$0 \rightarrow 3 \rightarrow 10 \rightarrow 9 \rightarrow 8 \rightarrow 4 \rightarrow 0$	$1 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 1$
Customer Served	2,11,12	3,4,8,9,10	5,6,7
Orders Served	1,21,22,23	2-5,14-20	6-13

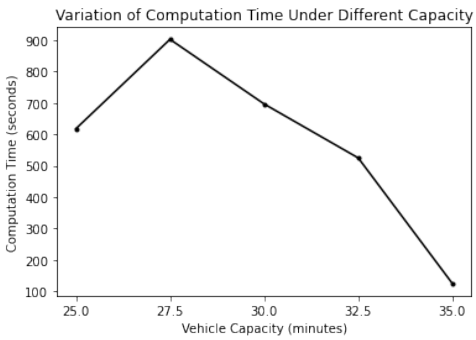
4 Sensitivity Analysis

After testing the validity of our model, we alter some parameters and conduct sensitivity analysis on vehicle capacity and total vehicle number.

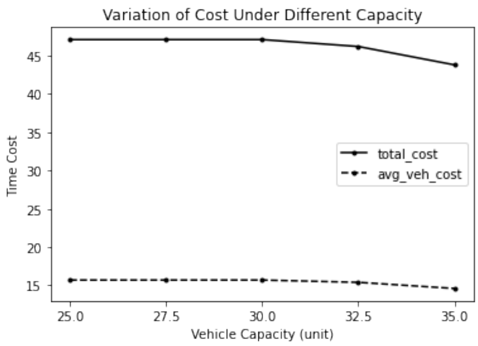
4.1 Sensitivity Analysis on Capacity

We analyzed the effect of the capacity of vehicles by changing the capacity from 25 to 35 and keeping other variables constant.

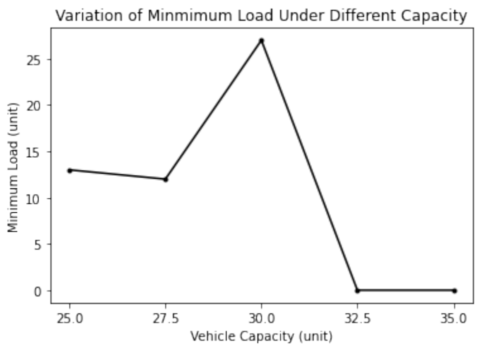
Figure 3a shows that the computation time reaches the highest when the capacity is 27.5, and then decreases rapidly as the capacity increases. The minimum delivery cost stays the same when the capacity is between 25 to 30 and then decreases as shown in figure 3b. This is because when the capacity reaches 32.5, the number of vehicles used is reduced, and with a capacity of 35, there is even no need for two vehicles to deliver the same order. This is shown in appendix table 5. Apparently, delivering an order repeatedly increases the total distance and total delivery times as shown in figure 3d. The increase in capacity reduces the chance of repeated deliveries. However, when the capacity is 32.5, due to the reduction of the number of vehicles, the situation of repeated deliveries reappears, so the total delivery times increase. Figure 3c illustrates that when the capacity is 30, the minimum vehicle load is extremely large. The disappearance of repeated deliveries could explain the phenomenon. And the minimum vehicle load decreases to zero with a capacity of 32.5 and above, because of an idle vehicle.



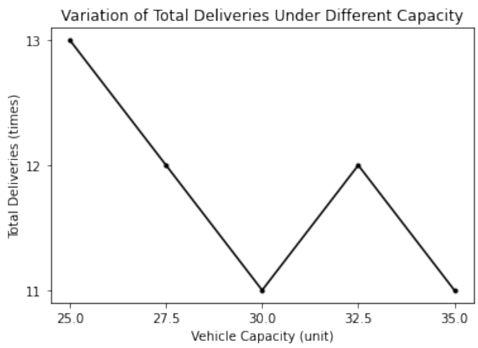
(a) Computation Time



(b) Delivery Cost



(c) Minimum Vehicle Load



(d) Total Delivery Times

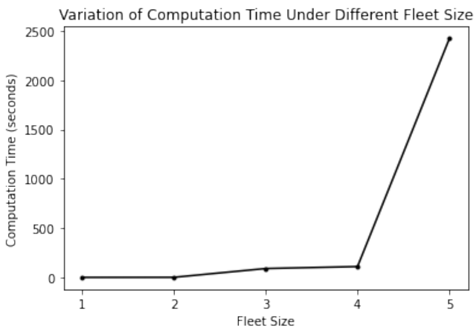
Figure 2: Sensitivity Analysis on Vehicle Capacity

4.2 Sensitivity Analysis on Vehicle Number

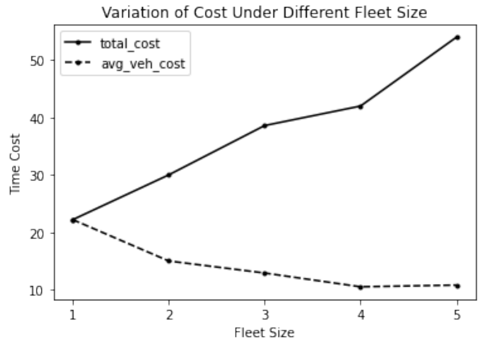
After a sensitivity analysis on vehicle capacity, we proceed by analyzing the effect of vehicle number on the delivery solution. In this analysis, we keep the total capacity of the fleet as 45 and we also keep every vehicle homogeneous. The size of the vehicle is fixed at a large number so different fleets can always meet the shape constraint.

As the computational cost is rather high, we studied a simplified case. There are 7 customers and 2 depots in the item delivery problem we study. The total demand of the customer is 40. Our method was implemented using Python and executed on an Intel Xeon W-2235 CPU with

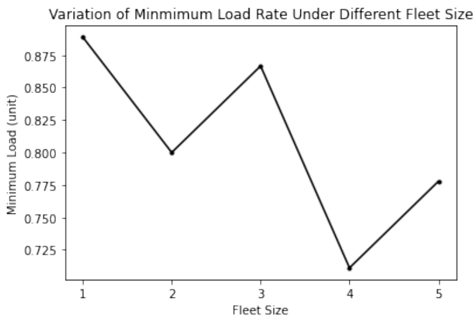
32GB memory. The result of our sensitivity analysis is shown in figure 3.



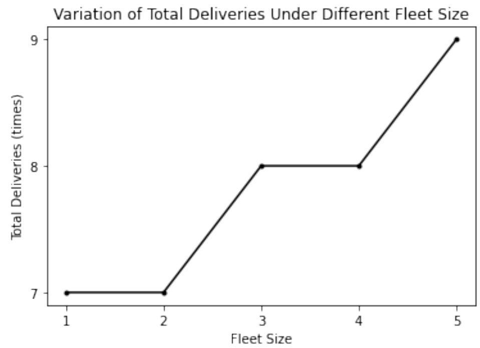
(a) Computation Time



(b) Delivery Cost



(c) Minimum Vehicle Load Rate



(d) Total Delivery Times

Figure 3: Sensitivity Analysis on Vehicle Numbers

As shown in figure 3a, the dimension of the problem increases with the number of vehicles. When there are few vehicles, the computation time is very small. However, as the number of vehicles grows, the solver needs much more time to solve the problem. Solving the 5 vehicle instance takes more than 40 minutes for a PC.

The total cost is lower when the number of vehicles is small. This is shown in figure 3b. Having fewer and large vehicles means everything can be packed together and no repetitive visits to the customers. This reduces the total cost and also reduces the total number of delivery shown in figure 3d. However, having more vehicles means more vehicles split the miles of the delivery task and the average vehicle delivery cost decreases with the number of vehicles. These vehicles may not take a customer's order at once for capacity constraint or efficiency. This can be seen in the optimal routes in appendix table 6. As the orders are packed in a more dispersed fashion, there are more delivery attempts when there are more vehicles, as shown in figure 3d.

For the minimum vehicle load rate illustrated in figure 3c, in this case, as the demand-capacity ratio is high, all cases have a rather high minimum vehicle load rate (above 70%). The minimum load rate varies with fleet size and does not have a fixed trend. This is due to the efficient loading a decision is highly dependent on how the order demand can be matched with vehicle capacity. This match is not linear and thus the minimum load rate does not change monotonically. This can be shown in the order assignments in appendix table 7. We can notice that when there are 2 or 4 vehicles, customer 5's orders are packed together. But at 3 vehicles, the optimal decision is to deliver them separately.

5 Conclusion

In this article, we motivated the vehicle routing problem with heterogeneous products and presented a problem statement covering scenarios from different capacities and numbers of vehicles. We formulated an integer program formulation for single-depot and multi-depot problems respectively and discussed their extensibility towards further scenario enrichment.

A benchmark suite of 10 instances has been created and it contains instances with varying problem characteristics, such as the number and capacity of the vehicle. We have implemented a portfolio of different heuristic components for solving the VRP. After analyzing the results, we further conducted a series of sensitivity analyses on some model parameters.

There are several opportunities for further research regarding the vehicle routing problem with heterogeneous items. Our algorithm runs relatively slowly with small data size and this should be improved. Also, with the popularisation of the e-commerce industry, the real order data will become more complex. Incorporating real-life data would be another promising improvement and would certainly increase the realism of our model.

References

[1] George B Dantzig and John H Ramser. The truck dispatching problem. *Management science*, 6(1):80–91, 1959.

[2] William Ho, George TS Ho, Ping Ji, and Henry CW Lau. A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Engineering applications of artificial intelligence*, 21(4):548–557, 2008.

[3] Tai-Hsi Wu, Chinyao Low, and Jiunn-Wei Bai. Heuristic solutions to multi-depot location-routing problems. *Computers & Operations Research*, 29(10):1393–1415, 2002.

[4] Eleni Hadjiconstantinou and Roberto Baldacci. A multi-depot period vehicle routing problem arising in the utilities sector. *Journal of the Operational Research Society*, 49(12):1239–1248, 1998.

[5] Said Salhi, Arif Imran, and Niaz A Wassan. The multi-depot vehicle routing problem with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research*, 52:315–325, 2014.

[6] Ulrich Derigs, Jens Gottlieb, Jochen Kalkoff, Michael Piesche, Franz Rothlauf, and Ulrich Vogel. Vehicle routing with compartments: applications, modelling and heuristics. *OR spectrum*, 33(4):885–914, 2011.

[7] Pasquale Avella, Maurizio Boccia, and Antonio Sforza. Solving a fuel delivery problem by heuristic and exact approaches. *European Journal of Operational Research*, 152(1):170–179, 2004.

[8] Luc Muyldermans and Gu Pang. On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm. *European Journal of Operational Research*, 206(1):93–103, 2010.

Appendix A Comparison of the Assignment of Order in Single Depot

order	customer node	Single depot	Single depot with penalty
1	1	0	0
2	2	2	1
3	2	2	1
4	3	1	1
5	3	0	1
6	4	1	1
7	4	0	1
8	4	1	1
9	5	0	0
10	5	0	0
11	6	0	0
12	6	0	0
13	6	0	0
14	7	0	0
15	7	0	0
16	7	0	0
17	8	1	2
18	9	1	2
19	9	1	2
20	9	1	2
21	10	1	2
22	10	1	2
23	11	1	2

Figure 4: Comparison of Order Assignment with Penalty Added (Correction: there is a small mistake, actual customer node is the number in the figure +1)

Appendix B Comparison of Solutions in Sensitivity Analysis on Vehicle Capacity

Table 5: Comparison of Vehicle Routes

Vehicle Capacity	25	27.5	30	32.5	35
Route of Vehicle 0	$0 \to 3 \to 10 \to 9 \to 8 \to 4 \to 2 \to 0$	$0 \to 3 \to 10 \to 9 \to 8 \to 4 \to 2 \to 0$	$0 \to 2 \to 12 \to 11 \to 0$		
Route of Vehicle 1	$0 \to 11 \to 12 \to 2 \to 0$	$0 \to 11 \to 12 \to 2 \to 0$	$0 \to 3 \to 10 \to 9 \to 8 \to 4 \to 0$	$0 \to 8 \to 9 \to 10 \to 11 \to 12 \to 0$	$0 \to 3 \to 9 \to 10 \to 12 \to 11 \to 2 \to 0$
Route of Vehicle 2	$1 \to 5 \to 6 \to 7 \to 1$	$1 \to 5 \to 6 \to 7 \to 1$	$1 \to 5 \to 6 \to 7 \to 1$	$1 \to 5 \to 7 \to 6 \to 8 \to 3 \to 4 \to 1$	$1 \to 6 \to 7 \to 8 \to 4 \to 5 \to 1$

Order ID	Customer Node	Capacity 25	Capacity 27.5	Capacity 30.0	Capacity 32.5	Capacity 35.0
1	1	1	0	0	0	1
2	2	1	0	0	1	1
3	2	1	0	0	1	1
4	3	0	0	0	1	2
5	3	0	0	0	1	2
6	4	2	2	2	2	2
7	4	2	2	2	2	2
8	4	2	2	2	2	2
9	5	2	2	2	2	2
10	5	2	2	2	2	2
11	6	2	2	2	2	2
12	6	2	2	2	2	2
13	6	2	2	2	2	2
14	7	0	0	0	2	2
15	7	0	0	0	2	2
16	7	0	0	0	2	2
17	8	0	0	0	1	1
18	9	0	0	0	1	1
19	9	0	0	0	1	1
20	9	0	0	0	1	1
21	10	1	1	1	1	1
22	10	1	1	1	1	1
23	11	1	1	1	1	1

Figure 5: Comparison of Order Assignment (Correction: there is a small mistake, actual customer node is the number in the figure +1)

Appendix C Comparison of Solutions in Sensitivity Analysis on Vehicle Number

Table 6: Comparison of Vehicle Routes

Vehicle Number	2	3	4	5
Route of Vehicle 0	$1 \to 7 \to 6 \to 5 \to 1$	$1 \to 8 \to 6 \to 1$	$1 \to 5 \to 1$	$1 \to 7 \to 1$
Route of Vehicle 1	$0 \to 8 \to 4 \to 3 \to 2 \to 0$	$1 \to 7 \to 5 \to 1$	$1 \to 6 \to 7 \to 1$	$1 \to 5 \to 1$
Route of Vehicle 2		$0 \to 2 \to 3 \to 4 \to 5 \to 0$	$1 \to 7 \to 8 \to 1$	$1 \to 6 \to 8 \to 1$
Route of Vehicle 3			$0 \to 2 \to 3 \to 4 \to 0$	$1 \to 8 \to 5 \to 1$
Route of Vehicle 4				$0 \to 2 \to 3 \to 4 \to 0$

Table 7: Comparison of Order Assignment

Order ID	Customer Node	2 Vehicle Assignment	3 Vehicle Assignment	4 Vehicle Assignment	5 Vehicle Assignment
1	2	1	2	3	4
2	3	1	2	3	4
3	3	1	2	3	4
4	4	1	2	3	4
5	4	1	2	3	4
6	5	0	1	0	3
7	5	0	2	0	1
8	5	0	2	0	1
9	6	0	0	1	2
10	6	0	0	1	2
11	7	0	1	2	0
12	7	0	1	1	0
13	7	0	1	1	0
14	8	1	0	2	2
15	8	1	0	2	3
16	8	1	0	2	2