15.093 Optimization Methods

Fall 2023

Drone Delivery Optimization

Lucas Goh, Xidan Xu

Contents

1 Purpose and Scope

The landscape of online shopping has been transformed significantly by the internet and AI. Customer expectations for purchase delivery have shortened from weeks to days, and even to just a few hours. The emergence of delivery drones introduces an entirely new set of challenges that demand data science solutions. In this project, we aim to develop an efficient and optimized solution for a simulation of managing and scheduling a fleet of delivery drones to fulfill customer orders as quickly as possible.

2 Data

Our data is sourced from a competition on Kaggle organized by Google. The data is provided as a plain text file containing exclusively ASCII characters with lines terminated with a single '\n' character at the end of each line (UNIX-style line endings).

2.1 Data Description

In the dataset, we are provided with four categories of information:

- Parameters of the simulation, which includes number of drones available, 24 drones, a drone's capacity, 200, map's grid size, etc.
- 400 products, which includes product types and product weights.
- 8 warehouses, which includes locations of warehouses and availability of product types at each warehouse.
- 671 orders, which includes the delivery location $[a, b]$ and the quantity of products in each order.

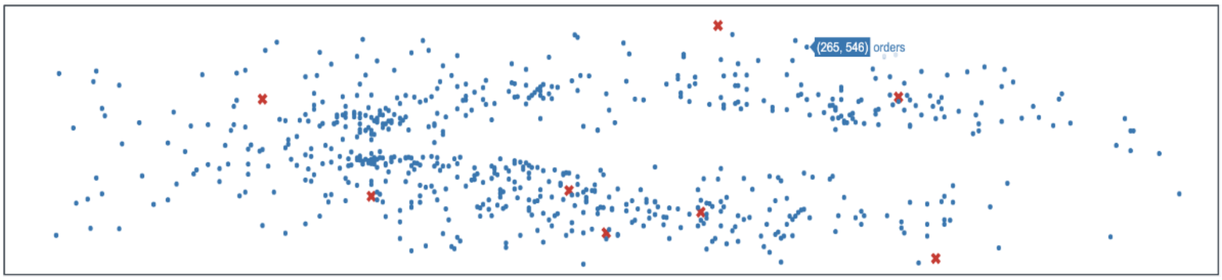

Figure 1: Plot of Orders and Warehouses

2.2 Simulation Rules

The simulation takes place in a two-dimensional grid. Each cell is identified by a pair of integer coordinates $[a, b]$. All drones must depart and return to the warehouse at the end of the simulation. In the simulation, we define a cost variable measured as time T, which equals the time lapsed when the drone returns to the warehouse. All drones take one unit of T to travel one unit of Euclidean distance, and many drones can travel at the same time and can overlap each other since they can fly at different altitudes.

3 Optimization Objective

Our objective function is defined as:

$$\min T_{\max}, \tag{1}$$

where $T_{\max}$ represents the time when the last drone returns to the warehouse. This approach indirectly ensures that all orders are fulfilled as quickly as possible. Alternatively, it can be formulated as:

$$\min \max_{k,l}(T_{k,l}), \tag{2}$$

where $T_{k,l}$ is the time spent by vehicle k in round l.

4 Exploratory Data Analysis

Our Exploratory Data Analysis (EDA) reveals that our network comprises 8 warehouses, managing 671 orders and encompassing 400 different product types.

For products, the mean weight is 64.59, the maximum weight is 148, and the minimum weight is 2. The distribution is shown in Figure 3.

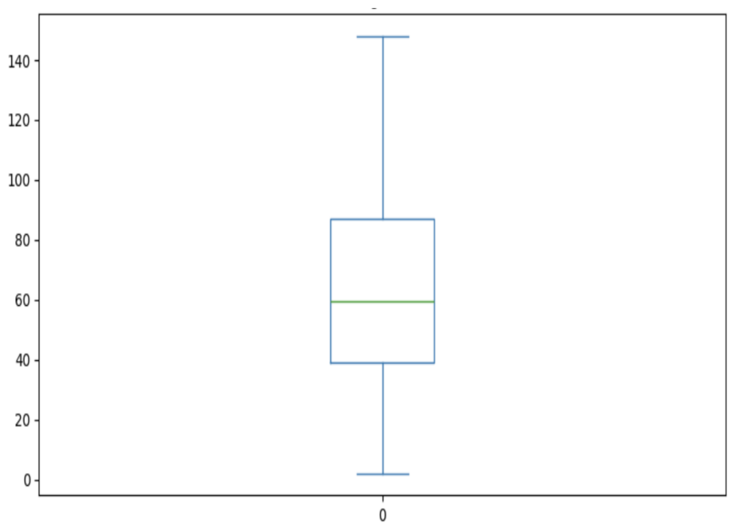

Figure 2: Product weight distribution

For orders, the total order weight distribution is shown in Figure 2. The red line represents the load of 1 drone, with the heaviest orders taking up to seven drones to deliver in a single trip. All orders to the left of the red line can be satisfied with the corresponding load. For instance, the total weight of orders before the first red line is less than the load of one drone.

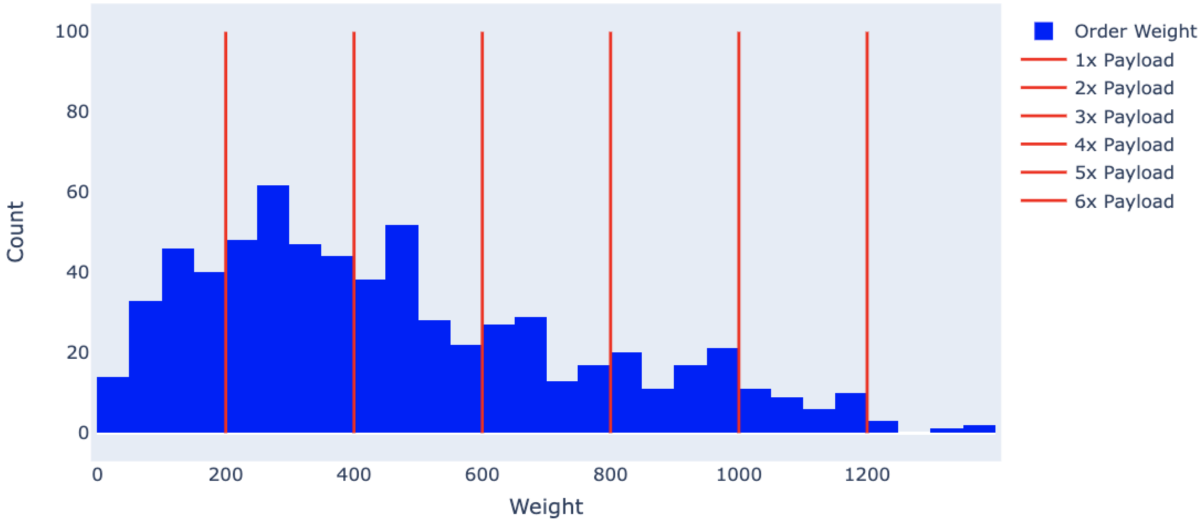

Figure 3: Orders Weights Distribution

It's also important to note that each product within these orders can be fully supplied, ensuring supply meets or exceeds demand (Figure 4). From Figure 4, the supply of all products exceeds the demand. However, not all orders can be efficiently fulfilled by the nearest warehouse. This necessitates an additional step for optimal order assignment to ensure efficient distribution.

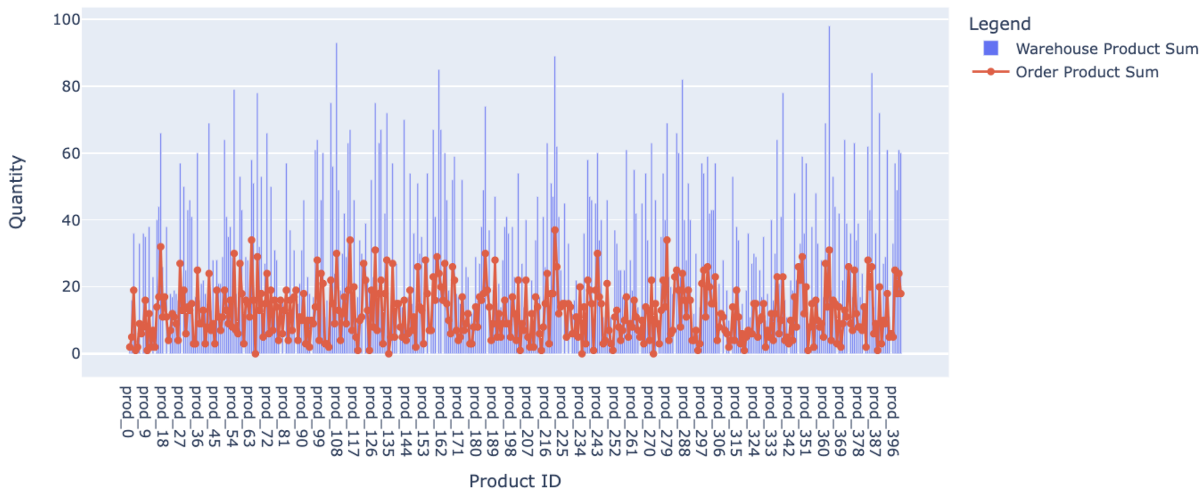

Figure 4: Demand Supply Visualization

5 Baseline Model

Our baseline model is a naive approach where each order is completed at one time. The objective value for this approach is calculated as the following:

1. Assign each order to warehouse(s) (see Section 6.1).
2. Define d_{iw} as the distance between order i and warehouse w.

Assuming that each order is fulfilled at one time, the total time required to visit all orders is simply $\sum_{i\in I,w\in W} 2 \times d_{iw}$. We have also created a video to demonstrate how the process works. You can view it here. We also note that this approach assumes trivially that there are sufficient drones that can return to the warehouses for each order and be readied before the next order starts. The baseline T_{max} using this approach is 89892.71.

6 Solution Approach

As this is a large-scale problem, solving a single formulation is difficult and can be computationally expensive. Inspired by methodologies used in the bus station assignment and vehicle routing problems taught in class, we shifted our focus to a three-step modular problem. It allows us to simplify the original problem by breaking it down into more manageable sub-problems.

1. Optimally Assign Orders to Nearest Warehouses.
2. Capacitated Vehicle Routing Problem (CVRP).
3. Assign Drones to Each Warehouse.

We first assign orders to individual warehouses, subjecting to demand and supply constraints. Then, we solve the capacitated vehicle routing problem for each warehouse.

6.1 Assignment of Orders to Warehouses

First, we want to allocate orders to warehouses optimally by minimizing the distance traveled for each product, subjected to demand and supply constraints.

Formulation of Order-Warehouse Assignment:

Objective:

$$\min \quad \sum_{p=0}^{P}\sum_{i=0}^{n}\sum_{w=0}^{W} \text{dist}[i][j] \times x_{piw}$$

where:

- P is the number of products.
- n is the number of orders.
- W is the number of warehouses.
- $\text{dist}[i][w]$ is the distance between order i and warehouse w.
- x_{piw} is a non-positive integer indicating the amount of product p from warehouse w assigned to order i.

Constraints:

1. Supply Constraints: Ensures that the total number of each product assigned from all warehouses does not exceed the available stock of that product in all warehouses.
For each warehouse w and each product p:

$$\sum_{i \in n} x_{piw} \leq \text{stock}[w, p]$$

where stock$[w, p]$ is the stock of product p in warehouse w.

2. Demand Constraints: Ensures that the demand for each product in each order is exactly met. For each order i and each product k:

$$\sum_{w \in W} x_{piw} = \text{demand}[i, p]$$

where demand$[i, p]$ is the number of product p demanded in order i.

6.2 Capacitated Vehicle Routing Problem

In this section, we solve the Capacitated Vehicle Routing Problem for each warehouse independently. Take, for instance, Warehouse 1: we evaluate it along with all the orders allocated to it in the initial step. This scenario is treated as a distinct capacitated vehicle routing problem (CVRP). This process is then replicated for every other warehouse.

This particular CVRP differs from the formulations discussed in our classes. It incorporates unique elements such as demand constraints, capacity limitations, and the possibility of multiple rounds. Notably, drones are capable of executing the same route over several rounds. For this formulation, we will provide a general formulation for all warehouses.

Formulation of CVRP

Parameters:

- dist$[i, j]$: Distance matrix – distance between node i and node j
- Q: Capacity of each vehicle
- $PW[p]$: Product weight of product p
- d$[j, p]$: Demand of order j for product p

Sets and Indices:

- W: Set of warehouses (indexed by w)
- I: Set of locations, including orders and warehouses (indexed by i and j)
- K: Set of drones (indexed by k)
- L: Set of possible rounds (indexed by l), arbitrarily set to 7 (maximum rounds required for the heaviest order if only using a single drone) multiply the largest number of orders received by a single warehouse.
- P: Set of products (indexed by p)

Decision Variables:

- $x_{ijkl} \in \{0, 1\}$, 1 if drone k travels directly from node i to node j in round l, 0 otherwise.
- $q_{jpkl} \in \mathbb{N}$: Quantity of product p delivered to order i by drone k in its l'th round.

Objective:

$$\min \max_{k \in K, l \in L} (T_{k,l}), \tag{3}$$

Subject to:

1. **Time Initialization:** At the start, the time spent for all drones are 0.

$$T_{k0} = 0, \forall k \in K$$

2. **Time Accumulation:** The time for each vehicle accumulates over rounds.

$$T_{kl} = T_{k,l-1} + \sum_i \sum_j (\text{dist}[i, j] \cdot x_{ijk,l-1}), \forall k \in K, l > 0$$

3. **Depot Degree Constraints:** Each vehicle departs from and arrives at the depot exactly once per round.

$$\sum_j x_{wjkl} = 1, \forall w \in W, j \in I, k \in K, l \in L$$

$$\sum_i x_{iwkl} = 1, \forall w \in W, i \in I, k \in K, l \in L$$

4. **Flow Conservation:** For each vehicle and round, the incoming flow equals the outgoing flow for each non-depot location.

$$\sum_i x_{ijkl} = \sum_i x_{jikl}, \forall j \notin W, k \in K, l \in L$$

5. **Drone Capacity Constraint:** The weight of products carried by each drone should not exceed its capacity in each round.

$$\sum_i \sum_j PW[p] \cdot q_{ijkl} \leq Q, \forall k \in K, l \in L$$

6. **Demand Satisfaction:** Ensure each product's demand is satisfied across all rounds and vehicles.

$$\sum_k \sum_l q_{ipkl} = \text{d}[i, p], \forall i \in n, p \in P$$

7. **Linking q_{jpkl} and x_{ijkl} Variables:** Deliver products to a customer only if the vehicle is visiting that customer.

$$q_{jpkl} \leq \text{d}[j, p] \cdot \sum_i x_{ijkl}, \forall, j \in I, p \in P, k \in K, l \in L$$

6.3 Assignment of Drones to Each Warehouse

It is important to understand that after obtaining the routes for each drone and each round from Section 6.2, It is possible for us to reassign the routes after using an initial starting number of drones.

Here, we looked at two approaches:

1. **Assign drones evenly across all warehouses, with 3 drones per warehouse**
2. **Assign drones according to distance required to deliver all orders assigned to a warehouse**

We will explore the result in the next section.

7 Results

In this section, we will lay out the results that we obtained by solving our approach in Section 6.

7.1 Results for Assignment of Orders to Warehouses

First, using the formulation in Section 6.1, we optimally assigned orders to the warehouse where the product demand can be fulfilled. As observed in Figure 5, warehouse 4 has the most orders to fulfill, followed by warehouse 0, warehouse 7, and warehouse 3. On the contrary, warehouse 6 and warehouse 1 has the least numbers of orders to fulfill.

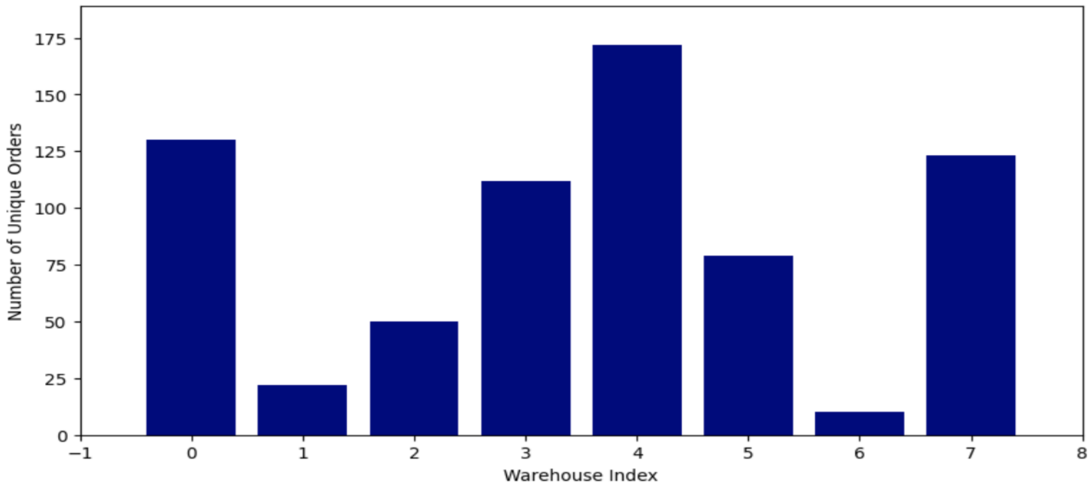

Figure 5: Orders distribution

Beyond this, we also observed that, out of the total 671 orders, there are 35 orders that requires deliveries from 2 warehouses. There also exists 1 order that requires orders from 3 warehouses. To account for these special scenario, we manually split these orders into multiple orders that contain fractional product demands of the original orders. This is so that our formulation in Section 6.2 can be fulfilled.

No. of Assigned Warehouses	No. of Orders
1	635
2	35
3	1

Table 1: Number of Assigned Warehouse per Order

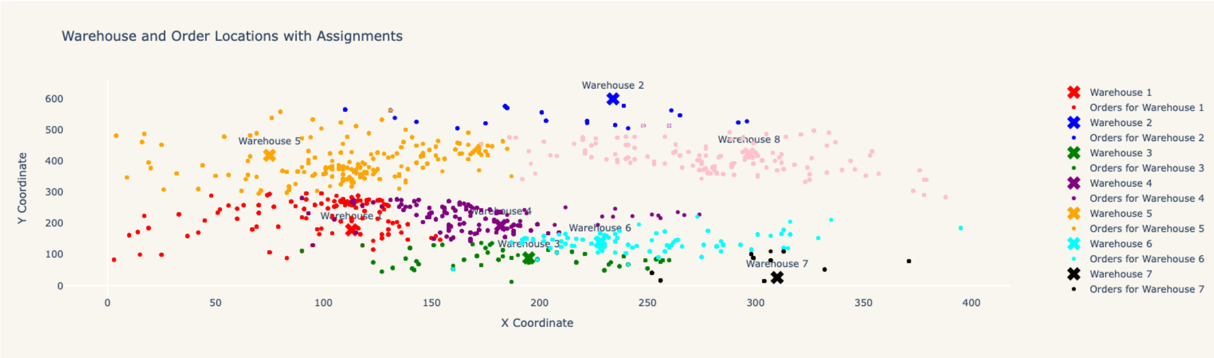

Figure 6: Visualization result for assignment problem

7.2 Results for CVRP

Having obtained the orders assigned to each warehouse, we solve the CVRP outlined in Section 6.2. As shown in the results below, the total distance required to visit deliver all the orders are similar to the number of orders assigned. However, as there are more orders to deliver, it increases the likelihood that a drone has to travel far away from a warehouse, therefore increasing the distance required to travel for Warehouse 4 significantly.

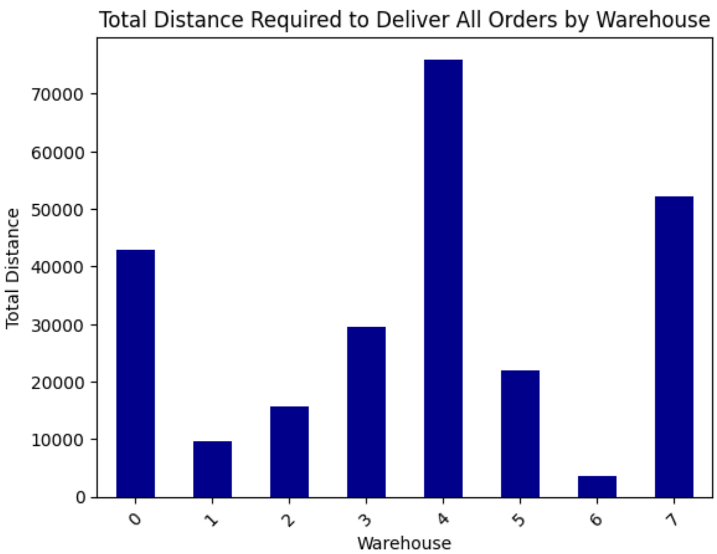

Figure 7: Visualization result for assignment problem

7.3 Results for Assignment of Drones

7.3.1 Assign All Drones Evenly

Now that we have the routes to visit all the orders for each warehouse, we need to assign drones to each warehouse. If we assign all 24 drones evenly (with 3 drones per warehouse), the result is as the following:

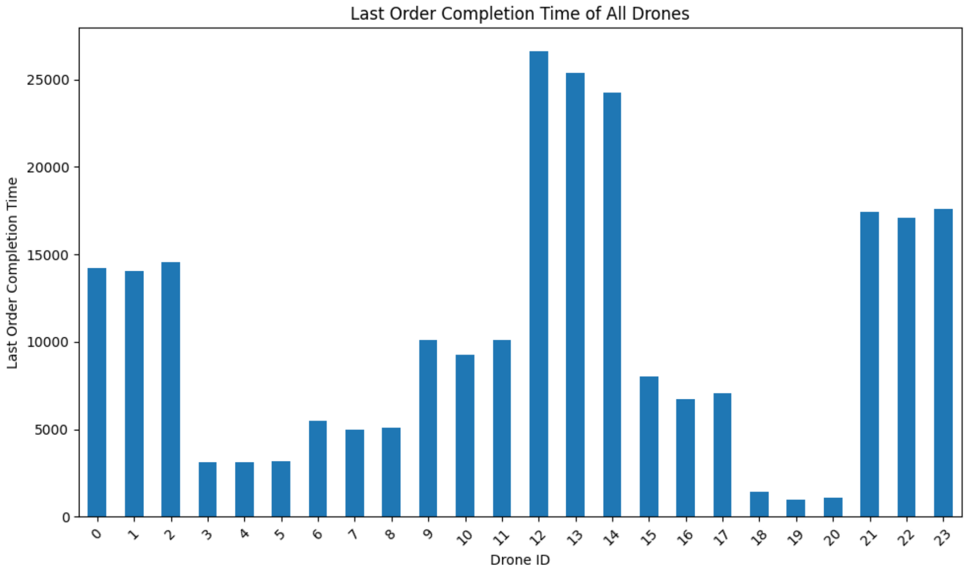

Figure 8: Last Order Completion Time of All Drones

The result is sub-optimal here, as we observe that there is a large discrepancy in the last order completion times between different warehouses. We observe that drones in Warehouse 6, which has the least number of orders, completed their last orders very early. On the contrary, we observe that drones in Warehouse 4 took the longest to complete all its orders. As a result, the T_{max} for this approach is 26618.34.

7.3.2 Assign Drones Using Total Distance Required To Travel All Orders Per Warehouse

To find a better way to assign the drones, we can assign drones according to the total distance required to visit all orders assigned to a warehouse (shown in Figure 7). This gives us the following drone assignment:

Warehouse	0	1	2	3	4	5	6	7
No. of Drones Assigned	4	1	2	3	7	2	1	4

Table 2: Number of Drones Assigned to Each Warehouse

The result is as follows:

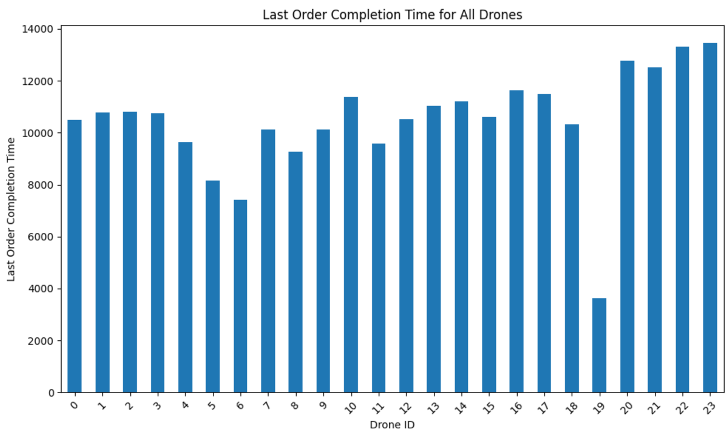

Figure 9: Routes for Warehouse 0

We observed that the last completion times are more evenly distributed across different drones. We acknowledge that Drone 19 is the least utilized. However, since it is the only drone assigned to Warehouse 6, it has to be maintained to meet delivery demands.

Here is a link to the Warehouse 0 Routes Animation: Warehouse 0 Routes Animation.

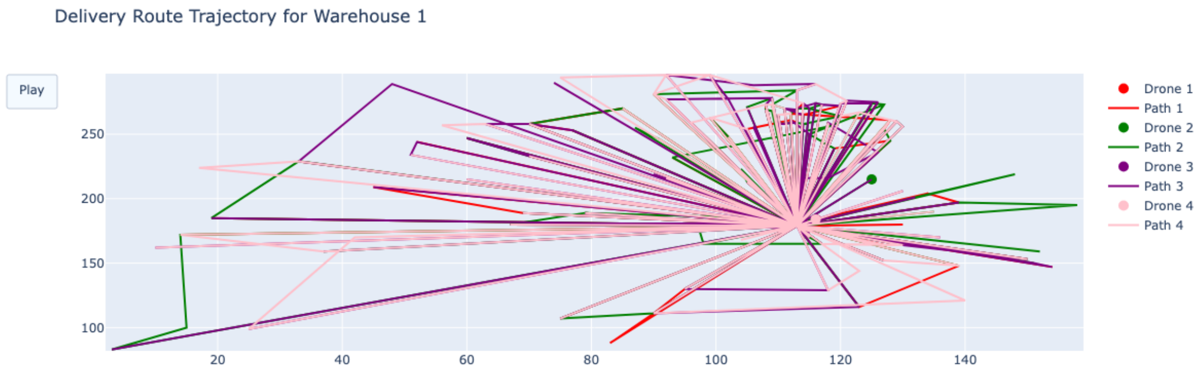

Figure 10: Routes for Warehouse 0

Figure 11: Routes for Warehouse 0

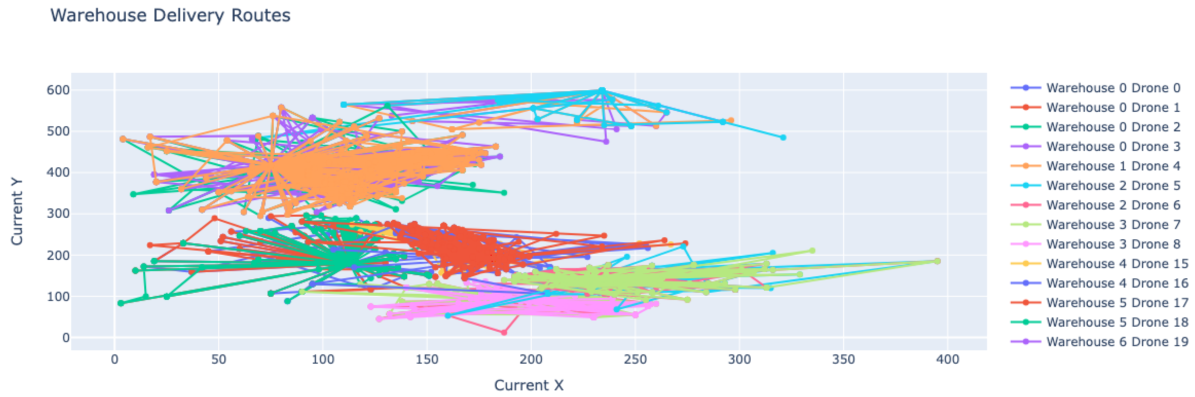

Figure 12: Visualization result for CVRP

In this figure, we observe the final delivery times for each warehouse. As Warehouse 7 has the last order to be delivered among all warehouses, this gives us a final T_{max} of 13459.79.

Strategy	T_{max}	**Improvement % (Compared to Baseline)**
Baseline	89892.71	-
Assign Drones Evenly	26618.34	70.39%
Optimal Drone Assignment	13459.79	85.03%

Table 3: Comparison of Different Strategies for Drone Assignment

8 Conclusion

In this paper, we address a large-scale problem by dividing it into three distinct steps: initially, we assign orders to warehouses; subsequently, for each warehouse, we solve a Capacitated Vehicle Routing Problem (CVRP); and finally, upon determining all routes, we optimize the assignment of drones to each warehouse. This allow us to achieve an optimal $T_{max}^{Optimal}$ of 13459.79, representing an 85.03% improvement from $T_{max}^{Baseline}$.

For the next step, one can choose to implement an even more optimal solution by relaxing the constraint that drones have to operate at a single warehouse. To implement this, one would need to make modifications to formulations in Section 6.2) and Section 6.3. This should enable drones, following each delivery, to restock at the nearest warehouse that has unfulfilled demand, thereby improving efficiency. This expanded approach will also require consideration of the initial positioning of all drones initially.

9 Appendix

9.1 Formulation. minimize total distance

Sets and Indices:

- I: Set of locations, including the depot (indexed by i and j)
- K: Set of vehicles (indexed by k)
- L: Set of possible rounds (indexed by l), with a maximum of 50 rounds
- P: Set of products (indexed by j)

Parameters:

- $\mathrm{dis}[i, j]$: Distance matrix – distance between location i and location j
- Q: Capacity of each vehicle
- $PW[p]$: Product weight of product p
- $\mathrm{d}[j, p]$: Demand of order j for product p

Decision Variables:

- x_{ijkl}: Binary variable, 1 if vehicle k travels directly from node i to node j in round l, 0 otherwise.
- q_{jpkl}: Quantity of product p delivered to customer j by vehicle k in its l'th round.

Objective: Minimize the total distance traveled across all vehicles and rounds.

$$\text{Minimize} \sum_i \sum_j \sum_k \sum_l (\mathrm{dis}[i, j] \cdot x_{ijkl}) \tag{4}$$

Subject to:

1. **Depot Degree Constraints:** Each vehicle departs from and arrives at the depot exactly once per round.

$$\sum_j x_{1jkl} = 1, \forall k, l$$

$$\sum_i x_{i1kl} = 1, \forall k, l$$

2. **Flow Conservation:** For each vehicle and round, the incoming flow equals the outgoing flow for each non-depot location.

$$\sum_i x_{ijkl} = \sum_i x_{jikl}, \forall j \neq 1, k, l$$

3. **Vehicle Capacity Constraint:** The weight of products carried by each vehicle should not exceed its capacity in each round.

$$\sum_i \sum_j PW[p] \cdot q_{ijkl} \leq Q, \forall k, l$$

4. **Demand Satisfaction:** Ensure each product's demand is satisfied across all rounds and vehicles.
$$\sum_k \sum_l q_{jpkl} = \mathrm{d}[j, p], \forall j, p$$
5. **Linking q_{jpkl} and x_{ijkl} Variables:** Deliver products to a customer only if the vehicle is visiting that customer.
$$q_{jpkl} \leq \mathrm{d}[j, p] \cdot x_{mikl}, \forall j, m, p, k, l$$