

Discrete Optimization and Decision Making Project

Last-mile Delivery Problem

Patrick Hamzaj
VR474246

Federico Leonardi
VR479719

1. Introduction

This report presents the solution to the project assignment for the course Discrete Optimization and Decision Making. In particular, the central objective of this project is to devise a solution that efficiently assigns a fleet of delivery vans to serve a known set of customers, each requiring delivery of packages with known weights and scheduled delivery times. Vehicles depart from a common depot, deliver packages while respecting their maximum carrying capacity, and must complete all deliveries within a specified maximum time.

To model this scenario, we consider a mathematical optimization problem structured around a complete directed graph, representing customers and the depot as nodes and the feasible paths between them as arcs. Each arc has an associated travel time that satisfies the triangle inequality. The main challenge involves determining routes for each vehicle that minimize the total completion time of deliveries, adhering to constraints such as vehicle capacity, customer-specific delivery times, and operational limits.

This report presents the mathematical formulation of the described optimization problem, completing in particular the baseline problem (Module 1) and two extensions—namely Module 2, which requires to restrain deliveries within a delivery time window along with a newly introduced set of triplets of incompatible customers served by a single vehicle and Module 4, that introduces an additional objective function focused on fairness and equity among drivers.

2. Problem Description

We consider a set of customers $C = \{1, \dots, \bar{c}\}$ and the depot (node 0). Each customer requires delivery of a package with weight $w = \{w_1, w_2, \dots, w_c\}$ and a delivery time $s = \{s_1, s_2, \dots, s_c\}$. Deliveries are performed by a fleet $K = \{1, \dots, \bar{k}\}$ of vehicles each with maximum capacity W , which must start and end at the depot, returning within a maximum allowed time t_{\max} . The delivery network is modeled as a complete directed graph $G = (V, A)$, with:

- Nodes $V = C \cup \{0\}$ representing customers and depot.
- Arcs $A = \{(i, j) \mid i, j \in V, i \neq j\}$ indicating direct travel possibilities with associated travel times $t_{i,j}$.

Therefore, the parameters of the problem are:

- A set of customers $C = \{1, \dots, \bar{c}\}$.
- The total set of nodes comprising the depot $V = C \cup \{0\}$.
- A set of packages weight $w = \{w_1, w_2, \dots, w_c\}$ for each customer $c \in C$.
- A set of service times $s = \{s_1, s_2, \dots, s_c\}$ for each customer $c \in C$.
- A set of homogeneous vehicles $K = \{1, 2, \dots, \bar{k}\}$ with capacity W .
- A travel time $t_{i,j}$ associated with each arc $(i, j) \in A$.
- A maximum time allowed t_{\max} .

The company's goal is to minimize the total time needed to complete all the deliveries, defined as the total travel times of each vehicle.

3. Mathematical Formulation

3. 1. Decision Variables:

1. Deciding which arcs to travel.

$$x_{i,j,k} = \begin{cases} 1 & \text{if vehicle } k \text{ traverses the arch } (i,j) \in A \\ 0 & \text{otherwise} \end{cases}$$

with $i \neq j \in V$ and $k \in K$.

$x_{i,j,k}$ is a binary variable that equals 1 if vehicle k travels from customer i to customer j (with $i \neq j$); it takes 0 if it does not.

2. Starting time of the service at a customer.

$$y_{i,k} \geq 0, \forall i \in V, \forall k \in K$$

$y_{i,k}$ is a continuous variable representing the time at which vehicle k starts the service at customer i . This variable serves as a labeling system in order to avoid subtours.

3. Overall route duration.

$$d_{j,k} \geq 0, \forall j \in V, \forall k \in K$$

It is a continuous variable that tracks the total duration for vehicle k when departing from node j toward the depot, only if j is the last node before returning.

3. 2. Objective function

In this problem, the task is to optimize the overall service time. This can be done by minimizing the sum over the travel times of the routes travelled by the vehicles.

$$\min \sum_{k \in K} \sum_{(i,j) \in A} t_{i,j} x_{i,j,k}$$

where:

- $t_{i,j}$ is the travel time over the arc (i,j)
- $x_{i,j,k} \in \{0,1\}$ indicates if vehicle k uses $(i \rightarrow j)$

3. 3. Constraints

1. Each customer is served exactly once.

$$\sum_{\substack{j \in V \\ j \neq i}} \sum_{k \in K} x_{j,i,k} = 1, \forall i \in C$$

This formulation ensures that every customer i in the set of customers C is visited exactly once by summing over all vehicles $k \in K$ and all potential preceding nodes $j \in V$ (excluding i itself).

2. Every vehicle must depart.

$$\sum_{j \in C} x_{0,j,k} = 1, \forall k \in K$$

It guarantees that each vehicle $k \in K$ departs from the depot (node 0) exactly once by defining the sum of flows from the depot to all customers $j \in C$ is 1.

3. Flow conservation: incoming arch=outgoing arch.

$$\sum_{\substack{j \in V \\ j \neq i}} x_{j,i,k} = \sum_{\substack{j \in V \\ j \neq i}} x_{i,j,k}, \forall i \in C, \forall k \in K$$

For every customer i , the number of arcs entering i equals the number of arcs leaving. In other words, if a node is visited, it must have both an incoming and an outgoing arc, thereby maintaining the consistency of the vehicle route.

4. Depot flow: each vehicle starts and ends at depot.

$$\sum_{j \in C} x_{0,j,k} = \sum_{j \in C} x_{j,0,k}, \forall k \in K$$

This constraint enforces that for every $k \in K$, the total flow from the depot (node 0) to all customers $j \in C$ equals the total flow from all customers $j \in C$ back to the depot, ensuring a balanced departure and return for each vehicle.

5. Vehicle capacity constraint.

$$\sum_{i \in C} w_i \left(\sum_{\substack{j \in V \\ j \neq i}} x_{j,i,k} \right) \leq W, \forall k \in K$$

With this constraint, we ensure that for each vehicle $k \in K$, the sum of the weights w_i of the customers $i \in C$ it serves does not exceed the vehicle's capacity. The inner summation determines whether customer i is visited by vehicle k —with a value of 1 if visited, 0 otherwise—therefore including w_i only if the customer is served by that vehicle.

6. Time constraint with progression of the vehicles to avoid subtours.

$$y_{j,k} \geq y_{i,k} + t_{i,j} + s_i - M(1 - x_{i,j,k}), \forall i \in V, \forall j \in C, \forall k \in K \text{ with } i \neq j$$

This constraint avoids subtours $i \rightarrow j \rightarrow i$ by forcing a time progression when vehicle k travels from customer i to customer j (with j not being the depot). It ensures that if arch (i, j) is traversed (so $x_{i,j,k} = 1$) the starting time at the next customer $y_{j,k}$ must be at least the sum of the start time at the current customer $y_{i,k}$ plus the travel time from i to j and the service time at i . The Big-M notation is used to deactivate the constraint when $x_{i,j,k} = 0$, which is defined as:

$$M = \sum_{(i,j) \in A} t_{i,j} + \sum_{i \in C} s_i, \text{ with } j \neq 0$$

The choice of the Big-M is large enough to completely deactivate a constraint, which is the maximum possible difference between the right and left sides of the constraint, bounded by the maximum possible route duration.

7. The starting time from the depot is always 0.

$$y_{0,k} = 0, \forall k \in K$$

This sets the initial departure time at the depot to be always 0 for every vehicle $k \in K$, establishing a common starting point for all routes.

8. Route duration constraint.

$$d_{j,k} \geq y_{j,k} + s_j + t_{j,0} - M(1 - x_{j,0,k}), \forall j \in C, \forall k \in K$$

If vehicle k returns to the depot from customer j —so this is the last node of the route—then the route duration $d_{j,k}$ is at least the sum of the vehicle's starting time at j ($y_{j,k}$) plus the service time and the travel time from j to the depot. Again, the Big-M constraint deactivates this constraint when the arc in question is not the last travel for vehicle k .

9. Maximum route duration.

$$d_{j,k} \leq t_{\max} x_{j,0,k}, \forall j \in C, \forall k \in K$$

If a vehicle k returns to depot from customer j , then the route duration $d_{j,k}$ must not exceed the maximum duration t_{\max} . If the arc $j \rightarrow 0$ is not chosen, the constraint is effectively deactivated.

4. Time Windows and Incompatible Triplets (Module 2)

4. 1. Problem Description

The second module extends the previous scenario by adding some constraint to enhance service quality and operational efficiency. Each customer specifies a preferred time window $[a_c, b_c]$, within which the delivery must start: drivers cannot wait at customer's location.

Additionally, to ensure balanced workload among drivers, it has introduced constraints defining incompatibilities between customers, in the form of a set of tuples $R = \{(i, j, l) | i, j, l \in C, i \neq j, j \neq l\}$. Each tuple indicates a restriction that a customer l can not be served by the same delivery vehicle if customers i and j are also served by that vehicle.

The goal remains minimizing total delivery time.

4. 2. Decision Variables

A customer assignment variable has been introduced, in order to later being able to build the constraint of the incompatible tuples.

$$z_{i,k} = \begin{cases} 1 & \text{if customer } i \text{ is served by vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

$\forall i \in C$ and $\forall k \in K$.

This binary decision variable is equal to 1 if customer i is served by vehicle k , 0 otherwise.

4. 3. Constraints

10. **Link customer assignment variables to route variables.**

$$z_{i,k} = \sum_{\substack{j \in V \\ j \neq i}} x_{j,i,k}, \forall i \in C, \forall k \in K$$

The purpose of this constraint is to link the customer assignment variable $z_{i,k}$ with the routing variables $x_{i,j,k}$, for which if customer i is assigned to the route of vehicle k .

11. **Time window constraint.**

$$a_i z_{i,k} \leq y_{i,k} \leq b_i z_{i,k} + M(1 - x_{i,j,k}), \forall i \in C, \forall k \in K$$

If customer i is served by vehicle k ($z_{i,k} = 1$) it enforces the starting time of the delivery at customer i to be within the time window $[a_i, b_i]$.

12. **Incompatible triplets.**

$$z_{i,k} + z_{j,k} + z_{l,k} \leq 2, \forall (i, j, l) \in R, \forall k \in K$$

Finally, this constraint makes use of the choice of serving customers i, j, l : it ensures that at most two out of three customers of given incompatible triples (i, j, l) are served by vehicle k .

In summary, we added a binary variable indicating the choice of serving a customer by a specific vehicle and the imposed that at most two customers can be chosen out of the triplets defined in the restricted set, along with enforcing the starting time to be within a set time window restrain the domain of baseline problem, with the objective function remaining unaltered.

5. Distributing Workload across Drivers (Module 4)

5. 1. Problem Description

In addition to satisfying customer-specific time windows and routing restrictions, Module 4 introduces an additional objective, focused on fairness and equity among delivery drivers. This secondary objective function aims to minimize the difference in workload (delivery times) assigned to each driver, trying to balance the distribution of service.

To formally incorporate this goal, a hierarchical optimization approach is applied, with the primary objective—minimizing total delivery time—still considered as the priority: the secondary objective (workload equity) is addressed only after the primary objective has been optimized. A trade-off is permitted, allowing for up to a 5% degradation from the optimal solution of the primary objective, as stated below:

$$Z_B \leq (1 + 0,05)Z_A$$

with Z_A being the optimal value of the first objective function and Z_B the one of the second objective function.

Thus, the problem becomes a multi-objective optimization scenario prioritizing efficient deliveries while explicitly accounting for fairness in driver workloads, defined as the minimization of the Mean Absoute Deviation (MAD) measure:

$$\text{MAD} = \frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}|$$
$$\text{where } \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

5. 2. Decision Variables

To account for the newly introduced objective function and to be linked with the model built so far, three variables have been added.

5. Workload (i.e., total route duration).

$$T_k \geq 0, \forall k \in K$$

A continuous variable representing the workload assigned to driver (vehicle) k : it quantifies the individual total route duration.

6. Average workload over all drivers.

$$T_{\text{avg}} = \frac{1}{|K|} \sum_{k \in K} T_k$$

T_{avg} is defined as the average workload over all vehicles, calculated by summing the workload of all vehicles and dividing by the total number of vehicles $|K|$. It serves as a benchmark to balance the workload distribution.

7. Absolute deviation from the average.

$$\text{dev}_k = |T_k - T_{\text{avg}}|, \text{ for each } k \in K$$

It denotes the absolute deviation of vehicle k 's workload from the average workload, providing a measure of imbalance that can be incorporated into the objective function that will be minimized.

5. 3. Objective Function

The second objective function introduced is:

$$\min \frac{1}{|K|} \sum_{k \in K} \text{dev}_k$$

where $\text{dev}_k = |T_k - T_{\text{avg}}|$

as stated earlier, the task of the second objective function is to minimize the average deviation of the total route duration across all drivers, thus distributing the workload.

5. 4. Constraints

13. Link between drivers and total route.

$$T_k = \sum_{j \in C} d_{j,k} x_{j,0,k}, \forall k \in K$$

This constraint assigns the total route duration T_k to each vehicle (driver) k , as the sum of the route durations $d_{j,k}$ which hold a value if j is the last node visited before the depot, and thus for which $x_{j,0,k} = 1$. Essentially, it accumulates the duration of the leg that closes the route, ensuring that T_k reflects the duration corresponding to the customer from which the vehicle finally returns to the depot.

14. Calculation over the average route duration.

$$T_{\text{avg}} = \frac{1}{|K|} \sum_{k \in K} T_k$$

It establishes that the average duration T_{avg} is the arithmetic mean of the individual durations T_k .

15. Linearization of the absolute deviation.

$$\begin{aligned} \text{dev}_k &\geq T_k - T_{\text{avg}}, \forall k \in K \\ \text{dev}_k &\geq T_{\text{avg}} - T_k, \forall k \in K \end{aligned}$$

These two constraints ensure that for each vehicle k , the deviation variable is between the difference between T_k and T_{avg} , accounting for which is larger and thus defining the absolute deviation $|T_k - T_{\text{avg}}|$.

6. Conclusions

In this report, we have presented a solution for efficiently assigning delivery vehicles to serve a set of customers with predefined demands, delivery times, and specific constraints. We formulated the Mixed Integer Linear Programming model, which addresses both operational efficiency and fairness among drivers.

The baseline problem (Module 1) requested optimal delivery routes while respecting constraints related to vehicle capacities, service times, and maximum route durations. The model was subsequently enhanced through Module 2, which incorporated customer-specific time windows and incompatibility constraints. Finally, Module 4 extended the optimization by introducing a secondary objective function focused on equitable distribution of workload among delivery drivers. This hierarchical approach allowed us to balance efficiency and fairness, with a trade-off up to 5% degradation in primary objective value to improve workload equity.