The truck-and-freighter routing problem

ORO - Optimization in Transportation and Logistics

Seydou COULIBALY & Hector GATT

November 16th 2018

1 Problem description

The truck-and-freighter routing problem consists in delivering goods starting from a principal distribution center to the multiple customers in a city. The specificity of the problem is that the delivery of goods is performed by an innovative type of truck[1], where a small electric truck (called city freighter) can travel inside a larger truck (called truck in the following). The city implements an innovative logistics delivery policy and some streets are forbidden to traditional trucks. As a result, some customers can be served only by the city freighter.

In the Truck&Freighter Routing Problem (TFRP), customers are all delivered from a depot located outside of the city. To travel to the city center, the large truck leaves this depot, carrying the city freighter. Both vehicles can separate and join again at dedicated parking areas. In terms of capacity, the large truck can be considered as having an infinite capacity. The city freighter has a more limited capacity, but it can be resupplied several times by the large truck at parking areas. The day is considered to start at time 0 and end at the end of the time horizon T. All trucks should be back at the depot by this time.

Each customer has a location, a quantity and a time window. All customer should be served before the end of their time window. If a truck arrives at a customer before the opening of the time window, it has to wait until this time.

The routing costs are considered to be equal to the sum of vehicle routing traveling times. Note that no cost is accounted for when the small city freighter travels within the large truck.

The problem consists of designing the routes of the truck and city freighter, including determining when they travel together and separately and where they separate and join, such that all routes start and end at the depot, customers are served within their respective time windows, vehicles capacity are respected, and traveling costs are minimized.

This problem is illustrated on Figure 1.

[1] `http://bil.libner.com`

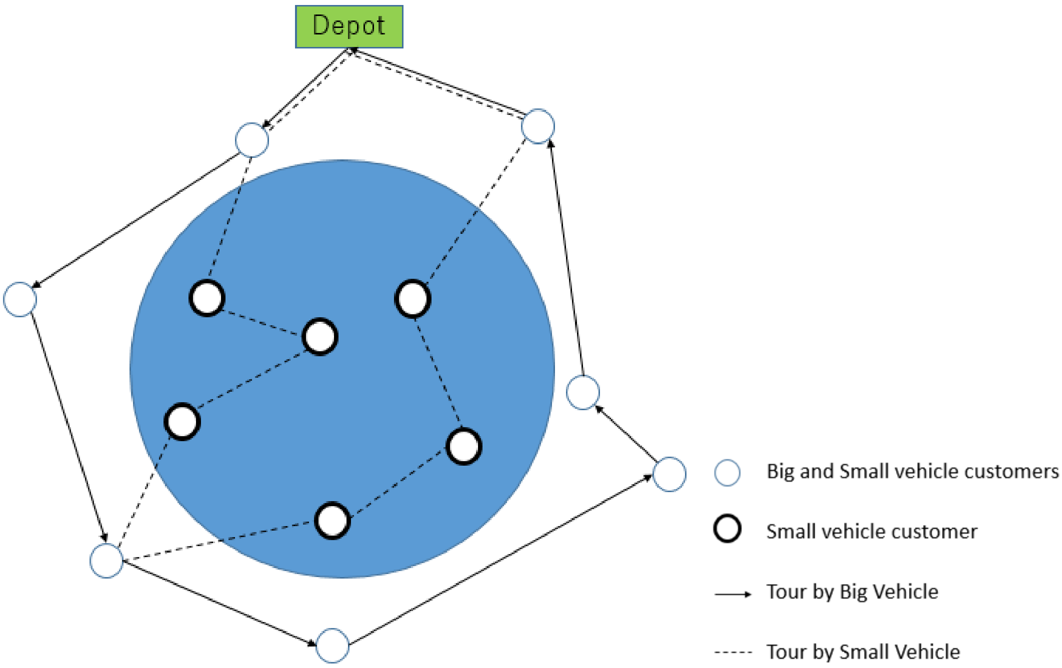


Figure 1: Typical solution of the truck and freigter routing problem

2 Notation

To model the problem, the following notation is given

2.1 Sets and data

Sets:

- $J = \{1, ..., n\}$: set of customers.
- vertex 0 and $n + 1$ model the depot.
- $J_S \subset J$: nodes which can be served by the small freighter only
- $J_L \subset J$: nodes which can be served by the big truck only
- $J_{LS} \subset J$: nodes which can be served by the big truck or the small freighter
- P: set of transfer nodes (parking areas)
- $V = \{0; n + 1\} \cup J \cup P$

Data:

- c^B_{ij}: cost of going from node $i \in V$ to node $j \in V$ with the big truck
- c^S_{ij}: cost of going from node $i \in V$ to node $j \in V$ with the small freighter alone
- q_j demand of customer $j \in J$
- Q capacity of the small freighter
- t_{ij}: time of traveling from node $i \in V$ to node $j \in V$

- s_j: duration of service at customer $j \in J$
- $[a_j, b_j]$: customer $j \in J$ time window
- T time horizon.

In this project we assume that $c_{ij}^B = t_{ij}$. For the city freighter, a ratio α is given such that the city freighter travelling time on an arc (i, j) is $\alpha \times t_{ij}$ and $c_{ij}^S = \alpha \times t_{ij}$. Thus, $c_{ij}^S = \alpha \times c_{ij}^B$. Ferthermore the duration of service at each transfer nodes and depot equal 0.

2.2 Model

Decision variables

- x_{ij}^S: binary variable that is equal to 1 if the edge (i,j) in V^2 is traversed by the small freighter
- x_{ij}^B: binary variable that is equal to 1 if the edge (i,j) in V^2 is traversed by the big truck
- w_i^S: beginning of service of the small freighter at vertex i $\in V$
- w_i^{BI}: beginning of service of the big truck at vertex i $\in V$
- w_i^{BO}: end of service of the big truck at vertex i for i $\in P$
- u_i: load of small freighter while entering vertex i $\in V$

$$min\ z = \sum_{i \in V} \sum_{j \in V} c_{ij}^B \times x_{ij}^B + \sum_{i \in V} \sum_{j \in V} c_{ij}^S \times x_{ij}^S$$

such that:

$$\sum_{j \in V} x_{ij}^S + \sum_{j \in V} x_{ij}^B = 1 \qquad \forall i \in J_{LS} \tag{1}$$

$$\sum_{j \in V} x_{ij}^S = 1 \qquad \forall i \in J_S \tag{2}$$

$$\sum_{j \in V} x_{ij}^B = 1 \qquad \forall i \in J_L \tag{3}$$

$$x_{ii}^S = 0 \qquad \forall i \in V \tag{4}$$

$$x_{ii}^B = 0 \qquad \forall i \in V \tag{5}$$

$$x_{ij}^B + x_{ij}^S \leq 1 \qquad \forall i, j \in V \tag{6}$$

$$\sum_{j \in V} x_{ij}^B = \sum_{j \in V} x_{ji}^B \qquad \forall i \in (J \cup P) \tag{7}$$

$$\sum_{j \in V} x_{ij}^S = \sum_{j \in V} x_{ji}^S \qquad \forall i \in J \tag{8}$$

$$\sum_{i \in V} \sum_{j \in P} x_{ij}^S = \sum_{i \in V} \sum_{j \in P} x_{ji}^S \tag{9}$$

$$\sum_{i \in (J \cup P)} x_{0\,i}^S = \sum_{i \in (J \cup P)} x_{i\,n+1}^S = 0 \tag{10}$$

$$\sum_{i \in (J \cup P)} x_{0\,i}^B = \sum_{i \in (J \cup P)} x_{i\,n+1}^B = 1 \tag{11}$$

$$w_i^S + s_i + c_{ij}^S \leq w_j^S + (1 - X_{ij}^S) \times T \qquad \forall i, j \in V \qquad (12)$$
$$w_i^{BI} + s_i + c_{ij}^B \leq w_j^{BI} + (1 - X_{ij}^B) \times T \qquad \forall i, j \in V \qquad (13)$$
$$w_i^{BO} \leq w_j^S + (1 - X_{ij}^S) \times T \qquad \forall i \in P, j \in V \qquad (14)$$
$$w_i^{BO} + c_{ij}^B \leq w_j^{BI} + (1 - X_{ij}^B) \times T \qquad \forall i \in P, j \in V \qquad (15)$$
$$u_j \leq u_i - q_i + (1 - x_{ij}^S) \times Q \qquad \forall i, j \in (J_S \cup J_{LS}) \qquad (16)$$
$$u_j = Q \qquad \forall j \in P \qquad (17)$$
$$w_1^S \leq w_i^S \leq w_n^S \qquad \forall i \in (J \cup P) \qquad (18)$$
$$w_1^{BI} \leq w_i^{BI} \leq w_n^{BI} \qquad \forall i \in (J \cup P) \qquad (19)$$
$$w_1^{BI} = w_1^S \qquad (20)$$
$$w_n^{BI} = w_n^S \qquad (21)$$
$$a_i \leq w_i^S \qquad \forall i \in (J_S \cup J_{LS}) \qquad (22)$$
$$a_i \leq w_i^{BI} \qquad \forall i \in (J_L \cup J_{LS}) \qquad (23)$$
$$\sum_{j \in V} x_{ij}^S \leq 1 \qquad \forall i \in P \qquad (24)$$
$$w_i^S \leq w_i^{BO} \qquad \forall i \in P \qquad (25)$$
$$w_i^{BI} \leq w_i^{BO} \qquad \forall i \in P \qquad (26)$$
$$T \times (1 - X_{ij}) + w_i^{BO} \geq w_j^S - c_{ij}^S \qquad \forall i \in P, j \in J \qquad (27)$$
$$x_{ij}^S \leq \sum_{k \in V} x_{ik}^B \qquad \forall i \in P, j \in V \qquad (28)$$
$$x_{ij}^S \leq \sum_{k \in V} x_{kj}^B \qquad \forall i \in V, j \in P \qquad (29)$$
$$x_{ij}^S \in \{0, 1\} \qquad \forall i, j \in V \qquad (30)$$
$$x_{ij}^B \in \{0, 1\} \qquad \forall i, j \in V \qquad (31)$$
$$w_i^S \geq 0 \qquad \forall i \in V \qquad (32)$$
$$w_i^{BI} \geq 0 \qquad \forall i \in V \qquad (33)$$
$$u_i \geq 0 \qquad \forall i \in V \qquad (34)$$

2.3 Comments

The objective function is the sum of edges costs taken by small freighter and big truck.

Constraint (1) states that the clients that are in the entire area have to be visited at least by the small freighter or the big truck.

Constraint (2) states that the clients that are in the area of the small freighter have to be served by the small freighter.

Constraint (3) states that the clients that are in the area of the big truck area have to be served by the big truck.

Constraint (4) and (5) state that the small freighter and the big truck can't chose an edge going from one node to the same one.

Constraint (6) states that the big truck and the small freighter are not allowed to take the same edge each of its part.

Constraint (7), (8), (9), (10), (11) are flow conservation constraints for the big truck, for the small freighter and on the depot.

Constraint (12) and (13) are constraints on the time window for the small freighter and the big truck.

Constraint (14) and (15) are used to create an upper bound time window for the parkings.

Constraint (16) (17) state that the load of the small vehicle is full if the small vehicle just leaves a parking where the big truck was, and decreases of the amount of the client if it leaves a client node.

Constraint (18), (19), (20) and (21) are used to create the time windows of the nodes.

Constraint (22) and (23) state that the small and the big truck cannot enter a node before the beginning of the window of this client.

Constraint (24) states that the small freighter can join at most one node $j \in V$ after leaving a parking $i \in P$.

Constraint (25) states that the small freighter can't leave a parking before the big truck is arrived.

Constraint (26) states that the big truck can't leave a parking before to be arrived.

Constraint (27) is used to to know when the big truck leaves the parking.

Constraint (28) and (29) state that the the big truck has to go to the parking i if the small freighter goes to it.

Constraint (30), (31), (32) and (33), (34) are variables definitions.

3 Valid inequality

We can add a valid inequality for the customers $j \in J_{LS}$.
$j \in J_{LS}$, means that the customer j can be served by the small freighter or by the big truck. We create a subset $W_1 \subset J_{LS}$ of customers that can not be visited by the big truck due to time window restrictions and a subset $W_2 \subset J_{LS}$ of customers that can not be visited by the small vehicle due to capacity or time window restrictions. We now know that the small freighter has to serve the customers of the area $(J_S \cup W_1)$ and the big truck has to serve the area $(J_B \cup W_2)$. The area that can be served by the two is now reduced and becomes $(J_{LS} \backslash (w_1 \cup w_2))$

We can now add the three following constraints to our model:

- $\sum_{j \in V} x_{ij}^S = 1, \ \forall i \in (w_1 \cap J_S)$

- $\sum_{j \in V} x_{ij}^B = 1, \ \forall i \in (w_2 \cap J_B)$

- $\sum_{j \in V} x_{ij}^S + \sum_{j \in V} x_{ij}^B = 1, \ \forall i \in (J_{LS} \backslash (w_1 \cap w_2))$

4 Experimentations

4.1 Instance description

12 instances are provided with denominations `T_N_P.txt` where T denotes the instance type with respect to the geographical distribution of points (clustered (C) or random (R)), N denotes the number of customers and P denotes the number of parking areas in the problem. Make sure to read the `readme.txt` file for instances description.

The shared instance parameters are the small freighter capacity, the speed ratio α, the length of the time horizon, a common time window width, and the service duration at each customer. Note that all customers are considered to have the same time window width, hence, only the opening time is given in instances. The default values to be taken are the following:

- $Q = 400$
- $\alpha = 2$
- $T = 32400sec.$
- Time window width $\delta = 7200sec.$
- Service duration $s_i = 300sec., \forall i \in J.$

4.2 Evaluation of modeling options

After looking for some limits of our problem, we observed that our model is not complete if we want to modelize some alternatives:

First Alternative:
We want to modelise the fact that the small freighter is allowed to go more than one time to the same parking. In that case, the problem in our model is that, if we name i the parking where the small freighter goes two times, the variable w_i^S will not be able to take the two values of the coming of the small freighter. And then, we have a problem with the interection of the time window of the big truck in this parking and the two time windows when the small freighter comes to this parking.

Second Alternative: In our model, the small freighter has no constraint on its autonomy. So if we want to modelize the fact that the small freighter has an electricity capacity and can only filled in the big truck, we have to had a value on every edge (i,j) taken by the small freighter, meaning how many electricity it takes to go from i to j. In that case, it is not very difficult to add this option. But it becomes really more difficult, if the small freighter can be filled in some station in the city. Then we have to think completely differently our model.

4.3 Instance and parameter analysis

The time spent by the solvers on our instances, with the default values, is defined bellow:

	GLPK	Cbc	Gurobi	CPLEX
C1-2-8	1.29"	12.06"	0.24"	0.21"
C1-3-10	> 30"	> 30"	2.17"	4.32"
C1-3-12	> 30"	> 30"	1.87"	2.55"
C2-2-8	4.90"	> 30"	0.47"	0.32"
C2-3-10	> 30"	> 30"	11.46"	20.77"
C2-3-12	> 30"	> 30"	2.77"	4.84"
R1-2-8	Infeasible	Infeasible	Infeasible	Infeasible
R1-3-10	1.57"	> 30"	0.29"	0.26"
R1-3-12	> 30"	> 30"	2.16"	2.69"
R2-2-8	1.19"	5.31"	0.18"	0.21"
R2-3-10	9.45"	> 30"	1.48"	0.73"
R2-3-12	2.99"	> 30"	1.40"	0.44"

We can see that solvers GLPK and Cbc are limited once we pass to instances with 3 parkings.
Gurobi seems to be better with instances of type "C-.." and CPLEX seems to be better with instances of type "R-..".
The table defined bellow gives the value of the objective function for every instance.

	Objective Function Value
C1-2-8	3587.1
C1-3-10	3880.3
C1-3-12	3936.8
C2-2-8	3538.2
C2-3-10	3768.4
C2-3-12	4117.3
R1-2-8	Infeasible
R1-3-10	5963.0
R1-3-12	6251.4
R2-2-8	5444.7
R2-3-10	4400.7
R2-3-12	5027.0

The two following tables show respectively the tour of the big and of the small freighter for every instance.

	Tour of the big truck
C1-2-8	1 – 30 – 87 – 32 – 42 – 20 – 66 – 1
C1-3-10	1 – 38 – 86 – 36 – 20 – 46 – 1
C1-3-12	1 – 84 – 23 – 93 – 87 – 91 – 97 – 82 – 30 – 1
C2-2-8	1 – 84 – 86 – 40 – 38 – 17 – 1
C2-3-10	1 – 36 – 82 – 93 – 87 – 91 – 97 – 41 – 1
C2-3-12	1 – 66 – 46 – 30 – 77 – 32 – 87 – 97 – 78 – 33 – 34 – 1
R1-2-8	Infeasible
R1-3-10	1 – 40 – 96 – 87 – 41 – 16 – 11 – 73 – 1
R1-3-12	1 – 64 – 83 – 38 – 44 – 7 – 42 – 41 – 1
R2-2-8	1 – 17 – 95 – 34 – 59 – 92 – 77 – 1
R2-3-10	1 – 33 – 59 – 36 – 98 – 97 – 53 – 15 – 1
R2-3-12	1 – 38 – 45 – 44 – 15 – 42 – 41 – 32 – 92 – 98 – 83 – 1

	Tour of the small truck
C1-2-8	30 – 63 – 76 – 25 – 61 – 20
C1-3-10	36 – 33 – 31 – 35 – 25 – 12 – 6 -53 – 20
C1-3-12	84 – 80 – 74 – 33 – 62 – 69 – 76 – 30
C2-2-8	84 – 33 – 35 – 25 – 12 – 9 – 17
C2-3-10	36 – 48 – 6 – 12 – 25 – 14 – 41
C2-3-12	30 – 6 – 12 – 35 – 25 – 34
R1-2-8	-
R1-3-10	41 – 63 – 80 – 69 – 9 – 22 – 73
R1-3-12	64 – 48 – 33 – 52 – 69 – 3 – 13 – 14 – 41
R2-2-8	17 – 21 – 14 – 63 – 35 – 34
R2-3-10	36 – 63 – 31 – 65 – 6 – 14 – 15
R2-3-12	15 – 45 – 13 – 9 – 18 – 74 – 41

The tableau defined bellow shows the values of the objective fonction value, according to some changements on the parameter values.

	C1-2-8
α=1	2682.1
α=1.5	3159.35
α=2	3587.1
α=2.5	4014.85
α=3	4435.59
α=3.5	4859.59
α=4	5275.6
α=4.5	5695.6
α=5	6115.6

	C1-2-8
Q=100	Infeasible
Q=200	3587.1
Q=300	3587.1
Q=400	3587.1
Q=500	3587.1

	C1-2-8
T = 12400	Infeasible
T = 22400	Infeasible
T = 32400	3587.1
T = 44400	3587.1
T = 54400	3587.1

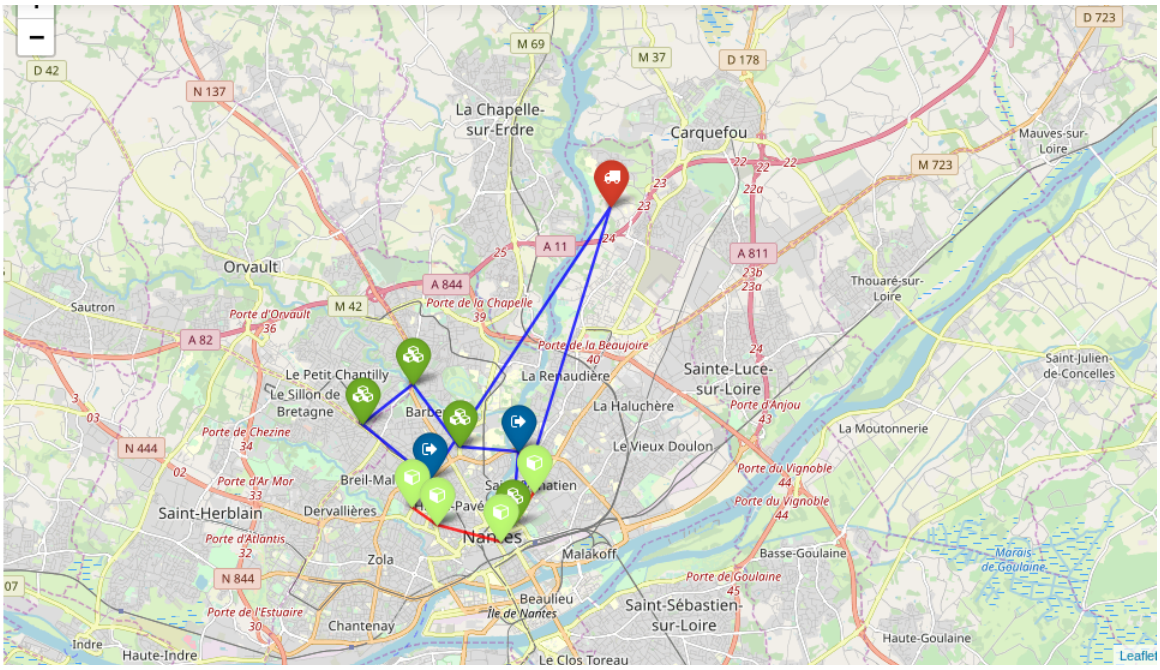
	C1-2-8
Δ=3200	3587.1
Δ=5200	3587.1
Δ=7200	3587.1
Δ=9200	3587.1
Δ=11200	3587.1

According to the values of this table we can make some small observations:

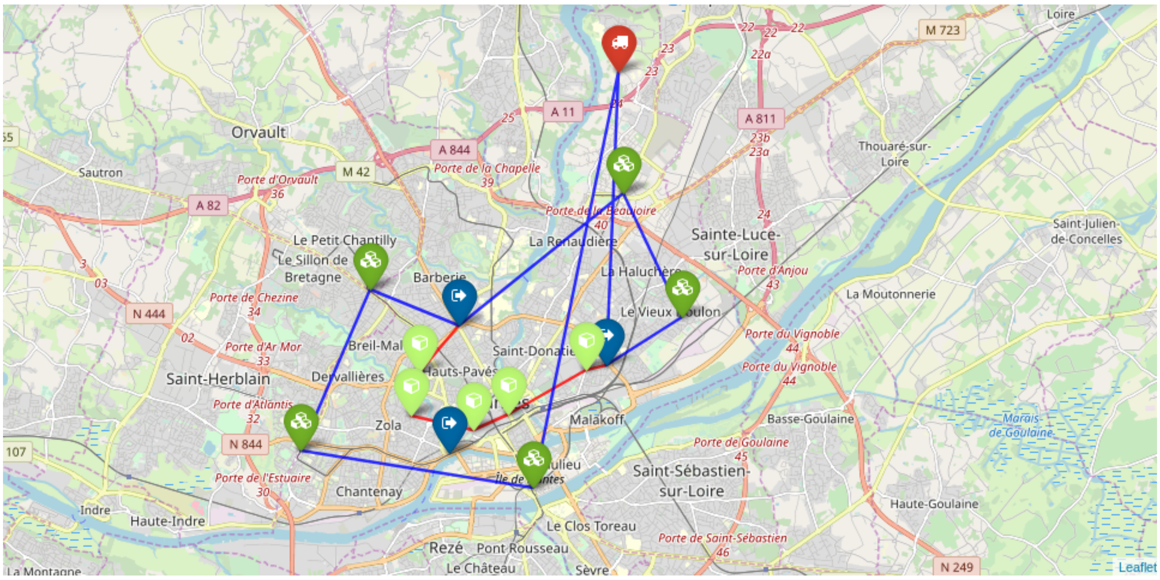
- The value of the objection fonction increases at the same time the parameter α increases
- Reducing or increasing the capacity of the small freighter doesn't change the objective function value, but we can get some infeasible solution if we reduce it too much and there is not enough parking to refill the small freighter.
- Reducing too much the time horizon can create some infeasible solution. By increasing it, the objective function value is not changed.
- Changing the time window of the clients has no impact on the objective function value.

4.4 Visualisation with Jupyter

Here is an example of the visualization on the instance C1-2-8.



And here, on the instance R1-3-10.



The blue points are the parking areas, the green points are the customers served by the big truck and the light green points are the customers served by the small freighter.