Freight-on-Transit for urban last-mile deliveries: A Strategic Planning Approach

Diego Delle Donne, Laurent Alfandari, Claudia Archetti, and Ivana Ljubić

Department of Information Systems, Decision Sciences and Statistics, ESSEC Business School, France
{delledonne,alfandari,archetti,ljubic}@essec.edu

Abstract

We study a delivery strategy for last-mile deliveries in urban areas which combines freight transportation with mass mobility systems with the goal of creating synergies contrasting negative externalities caused by transportation. The idea is to use the residual capacity on public transport means for moving freights within the city. In particular, the system is such that parcels are first transported from origins (central distribution centers) to drop-in stations, which are stop locations on public vehicles itineraries. Then, they are transported through public vehicles to drop-out stations, from where they are delivered to destination by freighters using green vehicles (such as bikes, drones, porters, etc.). The system is known as Freight-On-Transit (FOT). In this paper, we focus on the strategic decisions related to defining the public transportation network that will take part to the delivery system, i.e., which public vehicle lines and stop locations will be included (and thus equipped for the service). We propose different formulations for the problem and effective heuristic solution approaches based on column generation. We perform exhaustive tests aimed at providing managerial insights on the performance and the efficiency of the system.

Keywords: freight-on-transit, network flow, column generation, mixed integer linear programming.

1 Introduction

E-commerce is experiencing an explosion of revenues and market share. According to [33], e-commerce market share of retail sales is expected to reach almost 22% in 2024. This phenomenon has unprecedented impacts on people's daily life. Nowadays, and especially as a consequence of the COVID-19 pandemic, consumers are more and more used to buy on the internet instead of going to physical stores. In addition to the daily life alteration, another consequence of e-commerce explosion is related to the new logistic and distribution model: goods are now directly delivered to customers places. The effect is the huge increase of commercial vehicles travelling around urban and rural areas for home deliveries. Given the granularity of customers orders (e-buyers typically place small orders with high frequency, see [21]) and the importance given to fast shipping (see [26]), existing delivery operations related to e-commerce are becoming unsustainable. This is especially true when considering last-mile deliveries in urban areas, where most of online buyers concentrate. In China for example, more than 40 billions of packages were delivered in urban areas in 2017 (+ 28% in one year, see [31]). The main drawbacks associated with the steep increase of the number of (half-empty) vans travelling around urban roads are related to pollution, congestion and risk of accidents. According to [20], negative externalities generated by road freight transportation (mainly related to road accidents, congestion, pollution and climate change) represent a huge loss for society, equivalent to around 6.6% of EU GDP. Most of the economic and social losses are concentrated in the city and are around 2% to 4% of cities' GDP.

Various strategies have been recently developed to face the issue. Autonomous vehicles are seen as the future of last-mile deliveries (see [30]) as they are environmentally friendly and some of them, like drones and

robots, avoid congestion. However, there are issues related to their use. For example, drones raise security issues given the density of population of urban areas. Amazon Prime air program, which uses drones for performing Amazon's deliveries, has been launched in 2016 but is not operative yet. Delivery robots with wheels (e.g., Yape, Starship, Amazon Scout, Fedex's SameDay Bot), using sideways at pedestrian speed, now effectively operate in a number of cities around the world (see [1, 27] for related delivery optimization problems). Other non-autonomous solutions involve different green transportation means, like bicycles. Cargo-bikes are nowadays an established reality in many cities.

Combination of public transport systems with freight transportation

One opportunity that has not yet been fully analysed and exploited is combining mass mobility systems with freight transportation, in particular, using Public Transport Systems (PTS) for performing last-mile deliveries. Indeed, PTS suffer from a chronic inefficiency: they are overloaded on peak-hours and largely underused out of the peak (see [6]). There are thus high potential benefits in exploiting the residual capacity of PTS for last-mile deliveries, to face delivery sustainability and cope with PTS inefficiency. This novel logistics concept of integrating goods and passenger flows to promote higher utilization rates for the public transport network is known as Freight on Transit (FOT) (see, e.g., [28] and [10]). Passenger-and-Package Sharing systems have been recommended by the European Commission as soon as 2007 ([15]).

The interest in FOT is not only scientific, as shown in the literature review section, but also practical. Indeed, involved stakeholders, mainly public authorities, have started the discussion a few years ago (see for example, [12]) and some applications already exist. In a promising FOT pilot project launched by Monoprix in Paris (2007-2017), the company used the RER line D in Paris to transport goods between their distribution center in Combs-la-Ville and Bercy station in the city boundary of Paris [4]. In addition, Amazon has recently won a patent for a mobile package pickup system using public transport [2]. The system calls for installing storage compartments on buses or other vehicles, and letting riders with the authorised codes unlock a designated compartment and pick up their item. Other recent interesting initiatives include the municipality of Vienna, where the delivery service is in place combining FOT with cargo bikes. Packages are delivered to the bus garage where they are sorted on bike containers and delivered through cargo bikes (see [9]). A similar initiative is taking place in Barcelona [13].

In this paper, we investigate a system which is close to the one proposed by Amazon. Specifically, we consider a last-mile delivery service where parcels move from origin points, called Central Distribution Centers (CDC), to destinations (customers places) through a delivery service making use of PTS. In particular, parcels are first transported from CDC to PTS stations, called *drop-in stations*, where they are loaded on PTS buses (or any other PTS transportation mean, like tram). They are then moved by buses to *drop-out stations*, where they are unloaded and picked-up by freighters (which can be bikers, porters, drones, droids...) that perform the last leg of the delivery, i.e., up to the customer's place. Figure 1 illustrates the described transport system. In this work, we focus on the strategic decision related to the design of the PTS network used in the delivery service, i.e., the choice of the bus lines and stations to be used in the delivery service.

Strategic decisions

FOT systems involve two kinds of stakeholders: public authorities managing the PTS and logistic companies performing last-mile delivery operations. FOT systems rely on agreements so that logistic companies make use of PTS for performing their deliveries. These agreements might result from a win-win perspective (public authorities are interested in reducing the societal and environmental impact of delivery operations while logistic companies are interested in reducing their costs) or from regulations imposed by public authorities (like limitations on commercial vehicles that are allowed to travel in urban areas).

As mentioned above, in this paper we focus on the strategic decision associated with the design of a FOT system. In particular, we focus on the public authority decision about the setting of the system, i.e.:

1. which part of the PTS network should be involved in the FOT system, i.e., the bus/metro/tram lines that will take part in the service and that, thus, have to be equipped accordingly;

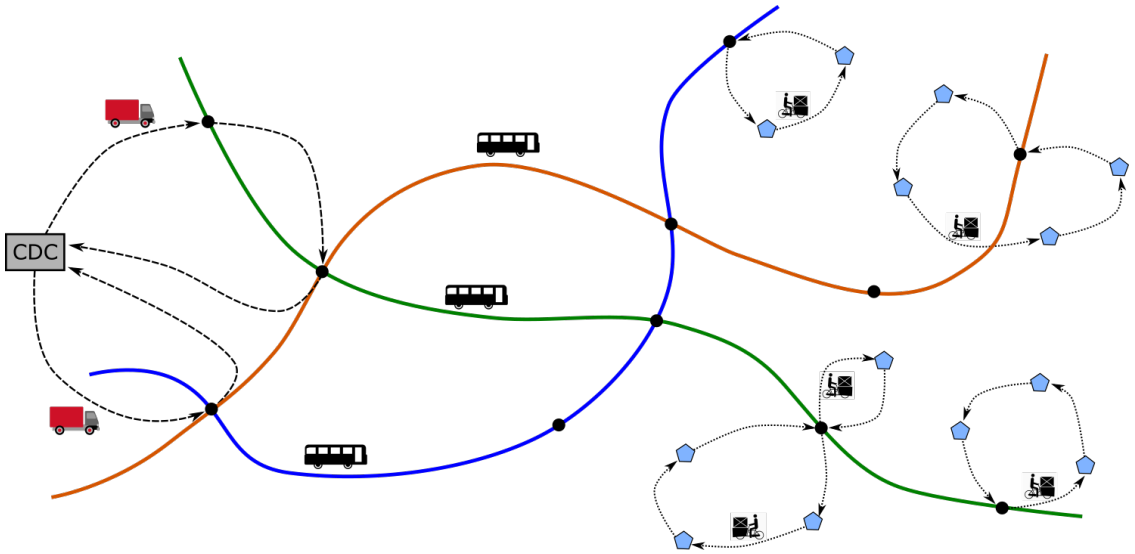


Figure 1: Illustrative example of a FOT system.

2. which stops have to be equipped for acting as drop-in or drop-out stations.

Both decisions involve a set-up cost in terms of equipment for accommodating freights that will be transported. There will be storage place for drop-in and drop-out stations and storage compartments in vehicles. Note that examples of such equipment already exist in practice, like lockers as storage space for stations or the storage equipment on public vehicles proposed by Amazon [2]. Many different factors have to be taken into account in the strategic decision. For the system to be as effective as possible in terms of resource utilization and congestion reduction, the main objective is to maximize the demand served by existing PTS resources. However, budget limitations related to the PTS equipment mentioned above have to be taken into account, since public authorities may not allow to use all bus lines or all stations. Also, some other important factors have to be considered, which are related to the inconvenience caused by such a service on passengers mobility. Indeed, storage equipment, both on vehicles and stations, would reduce the capacity formerly allocated to mobility. In addition, operations related to loading and unloading freights might interfere with passengers activities. Thus, the public authority could introduce restrictions related, for example, to a subset of existing public stops or public vehicles which cannot be involved in the service (for example, because they are typically too crowded), or time windows in which the FOT system cannot operate (peak hours).

In the following, we assume that the latter restrictions are known, meaning that the public authority has already deliberated on this kind of limitations. Thus, the problem is to decide the subset of lines and stops to be involved in FOT in such a way that the flow of goods transported is maximized. We assume that an estimated origin-destination (O-D) matrix is available, where for each pair of origin and destination the corresponding amount of freight to be delivered is known. Thus, the problem becomes designing the public transportation leg of , i.e., deciding which bus lines and stops should be included, with the aim of maximizing the flow of freight that can be moved through FOT while satisfying the budget constraints. We call the problem *Strategic Freight-on-Public-Transport* problem (SFOT).

Our contribution

Our contribution can be summarized as follows.

- We propose a new combinatorial optimization problem to help public authorities in analyzing, developing and implementing this novel logistic concept, in which freight and passenger transportation

are combined. The network design problem maximizes the covered demand, while respecting capacity restrictions reflecting the available budget and the PTS infrastructure. Major strategic decisions to be made are: which PTS lines and which stations should be used, and with which capacities.

- We propose three mixed integer programming (MIP) formulations for finding exact solutions of the problem. They are based on paths, flows and aggregated flows, respectively. We use instances of small and medium size to compare the computational performance of these models.
- We design an effective MIP heuristic based on a column generation approach associated with the path formulation, which is computationally the best-performing model among the three ones proposed. These MIP heuristic provides high-quality solutions for large instances of realistic size.
- We provide useful managerial insights. In particular, we analyze the impact of the available budget and the available stop- and line-capacity on the demand that can be covered. In addition, we simulate scenarios in which the demand is increased, or the number of available lines is enlarged, to understand their impacts and limitations for achieving the best resource exploitation and absorption of the existing demand.

The paper is organized as follows. In Section 2 we give an overview of the related literature. In Section 3.2 we provide a formal problem definition and we introduce the three MIP formulations. A MIP heuristic derived from the column generation procedure is presented in Section 4. Computational results are provided in Section 5 and final conclusions are drawn in Section 6. Additional computational details are provided in the Appendix.

2 Literature review

We first start this review of literature with general papers about integrating freight in PTS, outside the field of Operations Research and without any optimization model. In [10] first defines the term Freight on Transit (FOT) was introduced first. The authors conduct a survey from transportation experts to determine the greatest opportunities, challenges and difficulties of FOT. They depict very specific local solutions for the city of Toronto. One of the interesting ideas in this paper is to attach cargo trailers to buses. In [32] explore innovative solutions to integrate a global urban mobility strategy are explored, presenting new concepts for the TPS. In particular, they introduce two FOT projects, one in Germany and one in London, in which public buses carry small packages are presented. The paper shows some quantitative gains with the new systems. Different stages of analysis necessary to include freight into PTS are studied in [16]. The authors provide a case study based on the city of Bratislava, and propose a basic system. In [17], the authors depict crowdshipping systems based on public transport services, with individuals carrying parcels during their daily commute using the metro of Rome. In [5], the integration of passenger and freight transport is studied, and various key performance indicators (KPIs) of such integration are proposed. A case study using vessels of the local PTS in the Lagoon of Venice is presented. In [35], the authors study parcel delivery by subway. Their results show that using the subway to transport goods can efficiently reduce carbon emissions by 20–50 %. The interested reader will find a comprehensive list of papers in the recent survey of [14], reviewing both qualitative and quantitative research on the integration of freight in public transport. Note that, while in [32] no specific infrastructure is foreseen for the delivery service, the remaining contributions listed above all considers costs related to the equipment of the means of transportation and locations involved in the system.

We now review contributions which employ the methodology of Operations Research and Optimization. To the best of our knowledge, very few among the existing articles study the strategic network design questions, i.e., how to select a subset of public transport buses and stations for freight transportation. We found three papers that propose MIP formulations: two of them model hub location problems ([22] and [34]) and one models a location-routing problem ([19]). In [22], packages are transported by metro between the hubs, and by taxis or trucks between stores and hubs. In [34], same-day delivery during off-peak hours is performed

using subway, and the model aims again at locating the metro hubs with a p-median facility location model, with a case study in Shanghai. In the location-routing problem of [19], a subset of UDCs is to be selected around the city to be connected by a ring using public trains or shuttles, and electric vans perform pick-up and delivery inside the city. None of the above papers considers the selection of lines and maximum coverage of demand due to line capacities.

In a recent comprehensive survey dealing with freight on urban public transportation [14], 15 articles have been classified as quantitative. Among them, 11 deal with operational decisions, most of them involving vehicle routing. In the following, we provide a brief overview of the works that are most closely related to our contribution. In [8], packages are distributed by buses over a pre-designed network of bus lines and stops. Decisions concern the routing of packages pre-assigned to a bus line. Transshipment is allowed and a multi-commodity flow formulation minimizing the time-span of deliveries, and a heuristic, are provided. In [7], the authors consider a variant of the former problem in which a single path is given and transshipment is not allowed. In [25] a single Consolidation and Distribution Center is given together with a single bus line with multiple buses, where bus stops are fixed and time windows are associated with customers. The authors provide a model based on the classical arc-based 2-echelon VRP, minimizing the number of vehicles used, breaking ties concerning the distance traveled by the freighters. The authors also propose an Adaptive Large Neighborhood Search (ANLS) metaheuristic. In addition, they derive lower bounds by solving the linear relaxation of a set-partitioning formulation by column generation. In [3], the authors consider transporting freight using passenger trains to Paris, with a single rail line with fixed schedules and stations' capacities. The decisions are related to the assignment of packages to trains. A MIP formulation is proposed, however only heuristic methods are provided for computing solutions. The example of the French store Monoprix is given, transporting textiles, cosmetics, household goods and leisure products from warehouses in the outskirts of Paris to the Paris-Bercy station via RER D train line. In [28], an optimization model is proposed for city logistics sharing passenger trains and cargo trains. In [18], a MIP formulation is provided for a problem that fits our configuration with depot, drop-in and drop-out stations and destinations of demand requests. However, only operational decisions concerning the routing of vehicles carrying both parcels and passengers are optimized. Finally, the approach of [27] integrates autonomous pick-up and delivery robots sharing PTS capacity.

Note that the review of [14] classifies FOT integration systems in three categories, namely: *Shared track* where the public infrastructure like rails is used and the vehicles are generally light railways with only freight in the wagons but no passengers (such systems have been implemented in Dresden, Frankfurt and Zurich, see [24]); *Shared vehicle* where freight is transported inside a separate wagon attached to a light railway with passengers, or a trailer attached to a public bus, with dependent loading/unloading and transshipment operations (see [29] and [3]); and finally, *Shared Wagon*, where passengers and freight travel inside the same wagon or bus and peak periods of passenger flow have to be taken into account to share space in a proper and secure way (see [23]).

To summarize, to the best of our knowledge, this article is the first attempt to optimize strategic decisions in the context of designing a FOT system with selection of lines. Moreover, most of the above mentioned articles minimize (operational) cost for satisfying a given demand, without considering the maximization of the covered demand.

3 Problem statement and formulations

In this section we first provide a detailed problem statement, followed by three MIP formulations.

3.1 Problem statement

An instance of the *Strategic Freight-on-Public-Transport* problem (SFOT) is composed of several sets of elements and parameters. The *city* is represented by a set of *origins* O, a set of *destinations* D and a set

- O, D: Sets of origins and destinations
- L: Set of lines
- F: Set of fleets
- $L_f \subseteq L$: Set of lines of fleet f
- $\mathsf{f}(\ell)$: Fleet of line ℓ
- $S_\ell^{\mathsf{in}}, S_\ell^{\mathsf{out}}$: Sets of candidate drop-in and drop-out stops for line ℓ
- $S_\ell = S_\ell^{\mathsf{in}} \cup S_\ell^{\mathsf{out}}$: set of all stops for line ℓ
- $S := \bigcup_{\ell \in L} S_\ell$: set of all stops in the city
- $s_i^\ell \in S_\ell$: i-th stop in the path of line ℓ
- C_ℓ: daily capacity of packages which can be transported by line ℓ
- C_s: maximum number of packages which can be either dropped-in or dropped-out at stop s
- K: Set of commodities
- P_k: Number of packages of commodity k
- o_k, d_k: Origin and destination of commodity k
- $S_k^{\mathsf{in}}, S_k^{\mathsf{out}}$: Sets of drop-in stops nearby o_k and drop-out stops nearby d_k, respectively, that can be used to deliver commodity k
- $S_{\ell k}^{\mathsf{in}} = S_\ell^{\mathsf{in}} \cap S_k^{\mathsf{in}}$
- $S_{\ell k}^{\mathsf{out}} = S_\ell^{\mathsf{out}} \cap S_k^{\mathsf{out}}$
- $L_s^{\mathsf{in}}/L_s^{\mathsf{out}} \subseteq L$: Set of lines having stop s as a drop-in/drop-out stop
- $L_s := L_s^{\mathsf{in}} \cup L_s^{\mathsf{out}}$: Set of lines passing through stop s

Table 1: Notation summary

of (public transport) *lines* L, each of which is associated with a fleet serving the line (i.e., each fleet serves a subset of lines). We let F be the set of fleets. A line $\ell \in L$ is associated with a sequence of potential *drop-in stops* S_ℓ^{in} followed by a sequence of potential *drop-out stops* S_ℓ^{out}. We define $S_\ell = (S_\ell^{\mathsf{in}} \cup S_\ell^{\mathsf{out}})$ and $S = \bigcup_{\ell \in L} S_\ell$ as shortcuts. Note that we do not consider the case of 'circular lines', i.e., lines that make a go-and-back' path. This lines would not satisfy the assumption that drop-in stops precede drop-out stops. However, these lines can be easily modelled as two lines going in opposite directions. Each line $\ell \in L$ has a daily capacity $C_\ell \in \mathbb{N}$ of packages which can be transported. In turn, each stop $s \in S$ has a daily capacity of $C_s \in \mathbb{N}$ packages which can be stored at s (either as drops-in or drops-out). A set of commodities K is given, and each commodity $k \in K$ has a given origin $o_k \in O$, a given destination $d_k \in D$ and a number $P_k \in \mathbb{N}$ of packages to be delivered from o_k to d_k. In addition, only a subset S_k^{in} of the drop-in stops (those nearby o_k) and a subset S_k^{out} of the drop-out stops (those nearby d_k) can be used to deliver commodity k. We define $S_{\ell k}^{\mathsf{in}} = S_\ell^{\mathsf{in}} \cap S_k^{\mathsf{in}}$ and $S_{\ell k}^{\mathsf{out}} = S_\ell^{\mathsf{out}} \cap S_k^{\mathsf{out}}$ as shortcuts. Finally, strategic budgets $\mathsf{maxfleets}$, $\mathsf{maxlines}$ and $\mathsf{maxstops}$ (being non-negative integers) are given to limit the number of fleets, lines and stops, respectively, selected in the solution. We summarize in Table 1 the notation used in this paper.

In this work, we do not consider *transshipment* to be an option for delivering packages, this is, a package cannot be transported by more than one line. Therefore, the delivery of packages of a given commodity $k \in K$ with the FOT system consists on three *legs*:

1. First leg: From origin o_k to a drop-in stop $s \in S_k^{\mathsf{in}}$.
2. Public-transportation leg: From stop $s \in S_{\ell k}^{\mathsf{in}}$, through a line $\ell \in L_s^{\mathsf{in}}$ to a drop-out stop $t \in S_{\ell k}^{\mathsf{out}}$.
3. Last leg: From stop t to its final destination d_k.

Note that the public-transportation leg is mandatory in the FOT system, i.e., a package cannot be taken from o_k to a stop s and then from directly s to d_k, even if $s \in S_k^{\mathsf{in}} \cap S_k^{\mathsf{out}}$. We consider that in this case, the package could have been delivered directly, without using the FOT system.

As we are dealing with FOT at a strategical level, we consider each commodity to represent the (estimated) aggregated demand from one depot (origin) to a given neighborhood (destination). Hence, the set of packages corresponding to a commodity can be split and sent by different paths to their final destination.

Definition 1 (SFOT)**.** *With the above given notation, the SFOT asks to select a subset of fleets, lines and stops, satisfying budget constraints and to route packages through these elements, maximizing the number of delivered packages.*

3.2 MIP formulations

In the remainder of this section we propose three MIP formulations for the problem introduced in Section 3.1. The first is a path formulation where variables represent commodities' paths from origins to destinations. The remaining are two flow formulations. All presented formulations use the same set of binary variables to model discrete decisions concerning network design issues, i.e., the selection of fleets, lines and drop-in/drop-out stops to use:

- For each stop $s \in S$, a binary variable x_s is used to decide if s is selected as part of the solution or not.
- We use binary variable y_ℓ to decide if line $\ell \in L$ is selected in the solution.
- For each fleet $f \in F$, we use binary variable z_f to state if at least one line $\ell \in L_f$ is selected.

Each of the three formulations uses a different set of additional variables to model the flow of packages in the resulting network. We first present these formulations, and at the end of this section we introduce additional inequalities which can be used to strengthen their linear programming (LP) relaxations.

3.2.1 Path-based formulation

In addition to variables (x, y, z) introduced above, the *path-based formulation* uses a continuous variable $w^k_{\ell st} \in \mathbb{R}_+$, for each demand $k \in K$, each line $\ell \in L$ and each pair of stops $(s,t) \in S^{\mathsf{in}}_{\ell k} \times S^{\mathsf{out}}_{\ell k}$, to determine the flow of packages of commodity k using line ℓ, loaded at drop-in stop s and unloaded at drop-out stop t. Note that the choice of line ℓ along with the stops s and t determines a complete path from o_k to d_k and so each of the variables $w^k_{\ell st}$ is associated with a possible path from an origin to a destination. The path-based formulation for the SFOT can then be defined as follows:

$$\max \sum_{k\in K}\sum_{\ell\in L}\sum_{s\in S^{\mathsf{in}}_k}\sum_{t\in S^{\mathsf{out}}_k} w^k_{\ell st} \tag{1}$$

$$\sum_{\ell\in L}\sum_{s\in S^{\mathsf{in}}_{\ell k}}\sum_{t\in S^{\mathsf{out}}_{\ell k}} w^k_{\ell st} \le P_k \qquad \forall k \in K \tag{2}$$

$$\sum_{k\in K}\sum_{s\in S^{\mathsf{in}}_{\ell k}}\sum_{t\in S^{\mathsf{out}}_{\ell k}} w^k_{\ell st} \le C_\ell y_\ell \qquad \forall \ell \in L \tag{3}$$

$$\sum_{k\in K}\sum_{\ell\in L}\left(\sum_{t\in S^{\mathsf{out}}_{\ell k}} w^k_{\ell st} + \sum_{t\in S^{\mathsf{in}}_{\ell k}} w^k_{\ell ts}\right) \le C_s x_s \qquad \forall s \in S \tag{4}$$

$$\sum_{s\in S} x_s \le \mathsf{maxstops} \tag{5}$$

$$\sum_{\ell\in L} y_\ell \le \mathsf{maxlines} \tag{6}$$

$$\sum_{f\in F} z_f \le \mathsf{maxfleets} \tag{7}$$

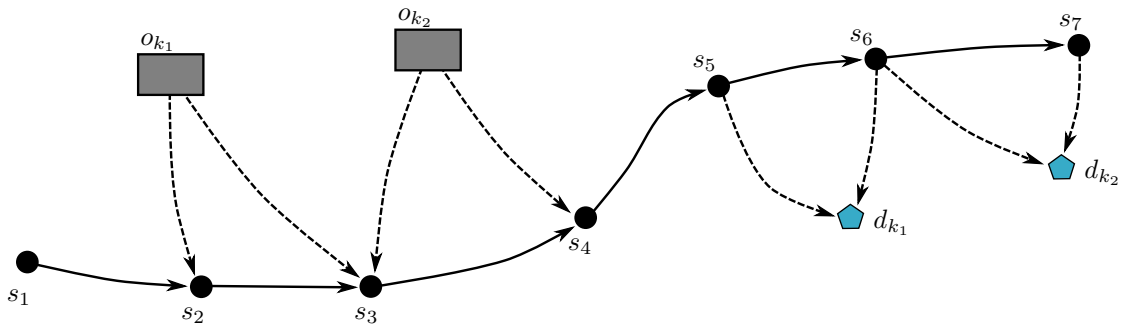


Figure 2: The graph G_ℓ for a line ℓ with 3 drop-in stops and 3 drop-out stops by which two commodities, namely k_1 and k_2, can be transported.

$$y_\ell \leq z_f \qquad \forall f \in F, \ell \in L_f \tag{8}$$

$$x_s, y_\ell, z_f \in \{0,1\} \qquad \forall s \in S, \ell \in L, f \in F \tag{9}$$

$$w^k_{\ell st} \in \mathbb{Z}_+ \qquad \forall k \in K, \ell \in L, s \in S^{\mathsf{in}}_{\ell k}, t \in S^{\mathsf{out}}_{\ell k} \tag{10}$$

Constraints (2) limit the aggregated flow of each demand along all possible paths. Constraints (3)-(4) ensure that lines and stops capacities are not violated. Constraints (5), (6) and (7) impose strategical restrictions on the number of stops, lines and fleets, while constraints (8) ensure the logical link between variables z and y, i.e., a line can be selected only if the corresponding fleet is selected too. We point out that constraints (3) are valid under the assumption that there are no drop-in stops after a drop-out stop. Nevertheless, if this assumption does not hold, our formulation can be easily adapted to deal with this case.

A potential disadvantage of this model is its size: even though the formulation is compact, the number of variables is in $O(|K||L||S|^2)$ and the number of constraints is in $O(|K| + |L| + |S|)$. To make the model computationally more tractable, we propose in Section 4 to solve the LP-relaxation of this model by dynamically pricing its columns. In the remainder of this section, we propose two alternative formulations, with significantly less decision variables.

3.2.2 Flow-based formulation

For the *flow-based formulation*, we start by focusing on the flow of packages for a given line $\ell \in L$. This can be done by considering each line separately, due to the fact that transshipment is not allowed. We construct a network, say $G_\ell = (V_\ell, A_\ell \cup A^{\mathsf{in}}_\ell \cup A^{\mathsf{out}}_\ell)$, representing a directed simple graph, in which V_ℓ is the set of stops visited by line ℓ along with all origins $o_k \in O$ and destinations $d_k \in D$ for each commodity k which can be routed through ℓ, i.e., $V_\ell = S_\ell \cup \{o_k \in O : k \in K \wedge S^{\mathsf{in}}_{\ell k} \neq \emptyset\} \cup \{d_k \in D : k \in K \wedge S^{\mathsf{out}}_{\ell k} \neq \emptyset\}$. We define three types of arcs:

- Set A_ℓ are the arcs (s_i, s_{i+1}), $i = 1, \ldots, |S_\ell| - 1$, connecting two consecutive stops of the line, where s_i is the i-th stop on the path of line ℓ. We slightly abuse the notation here and omit index ℓ for stations s_i, keeping in mind that we are referring to a single line ℓ.
- Set A^{in}_ℓ are the arcs (o_k, s), for each $k \in K$ and $s \in S^{\mathsf{in}}_{\ell k}$, and
- Set A^{out}_ℓ are the arcs (t, d_k), for each $k \in K$ and $t \in S^{\mathsf{out}}_{\ell k}$.

Figure 2 shows the graph G_ℓ for a line ℓ with 3 drop-in stops and 3 drop-out stops by which two commodities, namely k_1 and k_2, can be transported.

For each commodity $k \in K$ and each arc $(o_k, s) \in A^{\mathsf{in}}_\ell$ (resp. each arc $(t, d_k) \in A^{\mathsf{out}}_\ell$), variable $f^{\mathsf{in}}_{k\ell s} \in \mathbb{Z}_+$ (resp. $f^{\mathsf{out}}_{k\ell t} \in \mathbb{Z}_+$) represents the flow of packages of commodity k from origin o_k to stop s for line ℓ (resp. from stop t of line ℓ to destination d_k). To model multi-commodity flows on G_ℓ, we shall also use variables to indicate the amount of flow transported along each arc $(s_i, s_{i+1}) \in A_\ell$ and for each commodity $k \in K$.

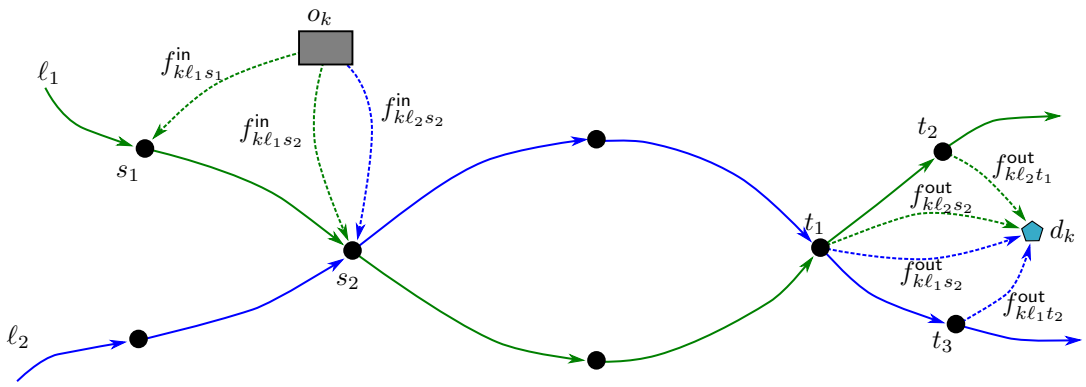


Figure 3: Example network illustrating the flow-based formulation variables.

However, due to the structure of our network (line ℓ is a simple path), these variables can be safely projected out, as we shall see below in Proposition 1. Figure 3 illustrates the proposed network and the role of each flow variable in the model. We note that the set of arcs on this network is the union of the arcs of the graphs G_ℓ, for every line $\ell \in L$ (characterized with different colors in the figure). Moreover, due to the fact that transhipment is not allowed, the flow on each graph G_ℓ is independent from the flow on the other graphs. The set of variables of this formulation is completed by using the binary variables x, y and z introduced above. Thus, the *flow-based formulation* for the SFOT can be written as:

$$\max \sum_{k\in K}\sum_{\ell\in L}\sum_{t\in S^{\text{out}}_{\ell k}} f^{\text{out}}_{k\ell t} \tag{11}$$

$$\sum_{s\in S^{\text{in}}_k}\sum_{\ell\in L^{\text{in}}_s} f^{\text{in}}_{k\ell s} \le P_k \qquad \forall k\in K \tag{12}$$

$$\sum_{k\in K}\sum_{s\in S^{\text{in}}_{\ell k}} f^{\text{in}}_{k\ell s} \le C_\ell y_\ell \qquad \forall \ell\in L \tag{13}$$

$$\sum_{\ell\in L^{\text{in}}_s}\sum_{\substack{k\in K\\ s\in S^{\text{in}}_{\ell k}}} f^{\text{in}}_{k\ell s} + \sum_{\ell\in L^{\text{out}}_s}\sum_{\substack{k\in K\\ s\in S^{\text{out}}_{\ell k}}} f^{\text{out}}_{k\ell s} \le C_s x_s \qquad \forall s\in S \tag{14}$$

$$\sum_{s\in S^{\text{in}}_{\ell k}} f^{\text{in}}_{k\ell s} = \sum_{t\in S^{\text{out}}_{\ell k}} f^{\text{out}}_{k\ell t} \qquad \forall k\in K, \ell\in L \tag{15}$$

$$f^{\text{in}}_{k\ell s} \in \mathbb{Z}_+ \qquad \forall k\in K, \ell\in L, s\in S^{\text{in}}_{\ell k} \tag{16}$$

$$f^{\text{out}}_{k\ell t} \in \mathbb{Z}_+ \qquad \forall k\in K, \ell\in L, t\in S^{\text{out}}_{\ell k} \tag{17}$$

$$x, y, z \text{ satisfy (5)-(9)}$$

Constraints (12)-(14) link flow-based and binary variables and limit the flow according to demands and capacities (in a similar way as constraints (2)-(4) for the path-based formulation). In addition, flow conservation constraints (15) ensure that incoming and outgoing flows are equal for each line ℓ and each commodity k.

Proposition 1 shows that our flow-based formulation is valid – it can be derived from the multi-commodity formulation imposed on graphs G_ℓ for each line $\ell \in L$, together with the capacity constraints (14). Indeed, the amount of flow for commodity k traversing the arc $(s_i, s_{i+1}) \in A_\ell$ (denoted as $\phi_{k\ell s_i}$ below) is implied by the incoming and outgoing flows (through arcs from A^{in}_ℓ and A^{out}_ℓ).

Proposition 1. *In any feasible solution for the flow-based formulation* (5)-(9), (12)-(17)*, the flow of a given commodity $k \in K$ along an arc $(s_i, s_{i+1}) \in A_\ell$, denoted by $\phi_{k\ell s_i}$, is determined by*

$$\phi_{k\ell s_i} = \sum_{j=1}^{i} (f^{\mathsf{in}}_{k\ell s_j} - f^{\mathsf{out}}_{k\ell s_j}) \tag{18}$$

where $f^{\mathsf{in}}_{k\ell s_j}$ and $f^{\mathsf{out}}_{k\ell s_j}$ are considered 0, whenever the corresponding arcs do not exist in the network G_ℓ.

Proof. The proof is given by induction on i and it follows from the flow conservation constraints imposed for the multi-commodity flows on G_ℓ. Clearly, $\phi_{k\ell s_1} = f^{\mathsf{in}}_{k\ell s_1} - f^{\mathsf{out}}_{k\ell s_1}$, due to the flow conservation for commodity k on line ℓ at this stop and the fact that s_1 has no entering arc from A_ℓ. For $i > 1$, the flow conservation for commodity k on line ℓ at stop s_i states that $\phi_{k\ell s_{i-1}} + f^{\mathsf{in}}_{k\ell s_i} = f^{\mathsf{out}}_{k\ell s_i} + \phi_{k\ell s_i}$. Therefore,

$$\begin{aligned}
\phi_{k\ell s_i} &= \phi_{k\ell s_{i-1}} + f^{\mathsf{in}}_{k\ell s_i} - f^{\mathsf{out}}_{k\ell s_i} \\
&= \sum_{j=1}^{i-1} (f^{\mathsf{in}}_{k\ell s_j} - f^{\mathsf{out}}_{k\ell s_j}) + f^{\mathsf{in}}_{k\ell s_i} - f^{\mathsf{out}}_{k\ell s_i} && \text{(by inductive hypothesis)} \\
&= \sum_{j=1}^{i} (f^{\mathsf{in}}_{k\ell s_j} - f^{\mathsf{out}}_{k\ell s_j}).
\end{aligned}$$

□

As in the path-based formulation, we point out that constraints (13) are valid under the assumption that there are no drop-in stops after a drop-out stop. Nevertheless, if this assumption does not hold, the formulation can be easily adapted to deal with this situation (e.g., by using Proposition 1 to determine the flow at each arc of the path of each line).

3.2.3 Aggregated flow-based formulation

The idea behind the *aggregated flow-based formulation* is to merge the flows of different lines entering (or leaving) the network from the same stop. To this end, for a commodity $k \in K$ and a potential drop-in stop $s \in S^{\mathsf{in}}_k$ (resp. drop-out stop $t \in S^{\mathsf{out}}_k$), variable $g^{\mathsf{in}}_{ks} \in \mathbb{Z}_+$ (resp. $g^{\mathsf{out}}_{kt} \in \mathbb{Z}_+$) represents the flow of packages from commodity k sent from origin o_k to stop s (resp. from stop t to destination d_k) to be delivered by using any of the lines passing through stop s. We note that variable g^{in}_{ks} represents the sum over all lines ℓ passing though stop s of variables $f^{\mathsf{in}}_{k\ell s}$ from the disaggregated flow-based formulation introduced in Section 3.2.2, i.e., $g^{\mathsf{in}}_{ks} = \sum_{\ell \in L^{in}_s} f^{\mathsf{in}}_{k\ell s}$. The same relation exists between variable g^{out}_{kt} and variables $f^{\mathsf{out}}_{k\ell t}$.

On the one hand, this aggregation reduces the number of variables in the formulation. On the other hand, the flow passing through arcs connecting two consecutive stops cannot be deduced anymore from variables f (as this was the case for the disaggregated formulation). We therefore need to introduce arc variables $\phi_{k\ell s}$ used in Proposition 1 representing the flow of packages of demand k transported through arc (s, s') for each line $\ell \in L$ and each stop $s \in S_\ell$, where s' is the stop following s on line ℓ. Figure 4 gives an example illustrating the role of each variable in the model. As in the previous formulations, we additionally use binary variables x, y and z. We further define $\mathsf{prev}(\ell, s)$ as the stop preceding s in line ℓ and L^{start}_s (resp. L^{end}_s) as the set of lines having s as the first (resp. the last) stop in their path. Finally, let $\tilde{S}_\ell$ be the set of stops associated with line ℓ except the last stop. The *aggregated flow-based formulation* for the SFOT can then be written as follows :

$$\max \sum_{k \in K} \sum_{t \in S^{\mathsf{out}}_k} g^{\mathsf{out}}_{kt} \tag{19}$$

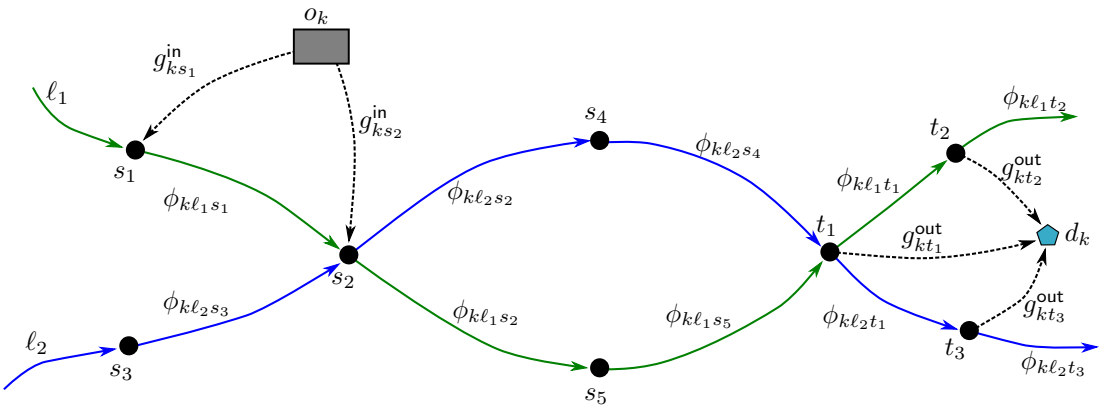


Figure 4: Example network illustrating the aggregated flow-based formulation variables.

$$\sum_{s \in S^{\text{in}}_k} g^{\text{in}}_{ks} \leq P_k \qquad \forall k \in K \tag{20}$$

$$\sum_{k \in K} \phi_{k\ell s} \leq C_\ell y_\ell \qquad \forall \ell \in L,\ s \in \tilde{S}_\ell \tag{21}$$

$$\sum_{k \in K} (g^{\text{in}}_{ks} + g^{\text{out}}_{ks}) \leq C_s x_s \qquad \forall s \in S \tag{22}$$

$$g^{\text{in}}_{ks} + \sum_{\ell \in L_s} \phi_{k\ell \text{prev}(\ell,s)} = \sum_{\ell \in L_s} \phi_{k\ell s} + g^{\text{out}}_{ks} \qquad \forall k \in K, s \in S \tag{23}$$

$$g^{\text{in}}_{ks} \leq \sum_{\ell \in L^{\text{in}}_s} \phi_{k\ell s} \qquad \forall k \in K, s \in S \tag{24}$$

$$g^{\text{out}}_{ks} \leq \sum_{\ell \in L^{\text{out}}_s} \phi_{k\ell \text{prev}(\ell,s)} \qquad \forall k \in K, s \in S \tag{25}$$

$$\sum_{\ell \in L'} \phi_{k\ell s} \leq g^{\text{in}}_{ks} + \sum_{\ell \in L'} \phi_{k\ell \text{prev}(\ell,s)} \qquad \forall k \in K, s \in S, L' \subseteq L_s \setminus L^{\text{end}}_s \tag{26}$$

$$\sum_{\ell \in L'} \phi_{k\ell \text{prev}(\ell,s)} \leq g^{\text{out}}_{ks} + \sum_{\ell \in L'} \phi_{k\ell s} \qquad \forall k \in K, s \in S, L' \subseteq L_s \setminus L^{\text{start}}_s \tag{27}$$

$$g^{\text{in}}_{ks} \in \mathbb{Z}_+ \qquad \forall k \in K, s \in S^{\text{in}}_k \tag{28}$$

$$\phi_{k\ell s} \in \mathbb{Z}_+ \qquad \forall k \in K, \ell \in L, s \in \tilde{S}_\ell \tag{29}$$

$$g^{\text{out}}_{kt} \in \mathbb{Z}_+ \qquad \forall k \in K, t \in S^{\text{out}}_k \tag{30}$$

$$x, y, z \text{ satisfy (5)-(9)}$$

Constraints (20)-(22) link aggregated flow-based and binary variables and limit the flow according to demands and capacities (in a similar way as constraints (2)-(4) for the path-based formulation). In addition, flow conservation constraints (23) ensure that the aggregated incoming and outgoing flows at each stop s are equal. Finally, constraints (24)-(27) ensure that the resulting flow satisfies the flow conservation constraints for each individual commodity:

- Constraint (24) forbids flow from an origin to a stop s unless it is picked up by lines having s as a potential drop-in stop. In turn, Constraint (25) states that the flow leaving stop s to its final destination must come from previous stops in the lines having s as a potential drop-out stop. In addition, the union of (24) and (25) avoids having flow not using any line (i.e., going from o_k to s and from s to d_k).

We note that this was not an issue in the former models, as the flow of packages was partitioned by lines (i.e., each variable is associated with a line ℓ), and a stop s cannot be in $S_\ell^{\mathsf{in}} \cap S_\ell^{\mathsf{out}}$, for any line ℓ. However, here we use aggregated flows over lines, thus we need constraints (24) and (25) to avoid such a case.

- Constraint (26) (resp. (27)) states that the flow leaving (resp. entering) s by a subset of lines L' cannot be greater than the flow entering (resp. leaving) s from this same set of lines plus the flow from the origin (resp. to the destination) of the corresponding demand k. Note that these constraints are not necessarily implied by (23), as the latter takes into account all flow entering (resp. leaving) s, even the flow of the lines ending (resp. starting) their paths at s (which are not counted in (26) nor in (27)). These constraints avoid the presence of transshipment at any feasible solution.

We note that *line capacity* constraints (21) do not rely on the assumption that there are no drop-in stops after drop-out stops. Furthermore, in order to consider different capacities for each arc of a path, it suffices to replace C_ℓ by the corresponding arc capacity in the right-hand-side of the constraint.

We remark that the aggregation conducted in this section comes at the cost of introducing an exponential number of constraints of type (26) and (27). However, for implementation purposes, these constraints can be omitted in an initial formulation, and dynamically added (in a branch-and-cut fashion) whenever an integer solution is found. Given an integer solution satisfying constraints (5)-(9), (20)-(30) except (26), the separation procedure for constraints (26) asks to find a commodity $k \in K$, a stop $s \in S$, and a subset of lines $L' \subseteq L_s \setminus L_s^{\mathsf{end}}$ such that

$$\sum_{\ell \in L'} (\phi_{k\ell s} - \phi_{k\ell \mathsf{prev}(\ell,s)}) > g_{ks}^{\mathsf{in}}. \tag{31}$$

The left-hand-side of (31) is obviously maximized for

$$L' = \{\in L_s \setminus L_s^{\mathsf{end}} : (\phi_{ks} - \phi_{k\mathsf{prev}(,s)}) \geq 0\}$$

Therefore, the separation procedure for constraints (26) can be solved in polynomial time for each k and s by calculating these values. The separation for constraints (27) is analogous.

In our implementation we start the solution process by adding, for each $k \in K$ and each $s \in S$, a constraint (26) (resp. a constraint (27)) for each individual line ℓ in $L_s \setminus L_s^{\mathsf{end}}$ (resp. in $L_s \setminus L_s^{\mathsf{start}}$), i.e., with $L' = \{\ell\}$. We add the rest of these inequalities in a dynamic fashion whenever they are violated.

3.3 Additional inequalities

The following inequalities can be added to all proposed formulation (as they involve only the common binary variables) with the goal of improving the continuous relaxation of the models.

$$x_s \leq \sum_{\ell \in L_s} y_\ell \qquad \forall s \in S \tag{32}$$

$$y_\ell \leq \sum_{s \in S_\ell^{\mathsf{in}}} x_s \qquad \forall \ell \in L \tag{33}$$

$$y_\ell \leq \sum_{s \in S_\ell^{\mathsf{out}}} x_s \qquad \forall \ell \in L \tag{34}$$

$$z_f \leq \sum_{\ell \in L_f} y_\ell \qquad \forall f \in F \tag{35}$$

Constraints (32) assert that a stop s is selected only if at least one of the lines passing through s is also selected (i.e., there are no isolated stops in the solution). Analogously, constraints (33) and (34) ensure that

a line ℓ is selected only if at least one of its drop-in stops and one of its drop-out stops is selected. Finally, constraints (35) forbid to select a fleet f if none of its associated lines L_f is also selected.

We remark that these constraints are not strictly speaking *valid inequalities* as they could be violated by feasible solutions. However, these constraints exclude solutions with redundant elements (e.g., stops for which no associated line is selected, etc.), hence we can safely add the above constraints to the formulations to improve their continuous relaxation. We observed in preliminary tests that adding these constraints actually improves the solution times of the proposed approaches.

4 A column generation based MIP heuristic for the path-based formulation

Although being polynomial, the number of variables of the path-based formulation may considerably grow as the size of the instance increases in the number of commodities, lines and stops. This is a strong drawback when attempting to solve real-size instances, in which even solving the continuous relaxation of the model may be challenging. In this section, we propose an alternative heuristic solution method based on the well-known *column generation* technique. In order to solve the linear relaxation of the model, we start with a *restricted master problem* (RMP) including only the binary variables, i.e., omitting the path variables w. In this case, a solution of value 0 is obtained, i.e., no package is delivered (note that this solution is feasible). After an optimal solution for this RMP is found, we solve the *pricing problem*, which asks to find a path variable $w^k_{\ell st}$ with a positive reduced cost. If such a variable is found, we add it to the RMP and iterate the process. If no such variable exists, we conclude that an optimal LP-solution is found. We refer the reader to [11] for more details on the column generation technique. After solving the linear relaxation of the formulation, our heuristic continues by solving the full path-based MIP formulation from scratch but restricting the set of path variables only to those generated during the column generation phase. The obtained solution is therefore a heuristic one, as not all the path variables are present in the MIP. However, the optimal value obtained from the linear relaxation may serve as an upper bound, thus giving a certificate of the optimality gap of the heuristic solution.

Throughout this section we will frequently refer to some elements of the dual problem of the linear relaxation of the path-based formulation, so we now provide the complete dual problem formulation. We use variables $\lambda_I^{(\star)}$ to represent the dual variable associated with the primal constraint $(\star)$ for the corresponding set of indices I. Additionally, we use σ^z, σ^y and σ^x to represent the dual variables associated with the upper-bounding constraints of the binary variables z, y and x (appearing in the continuous formulation when the integrality constraints are relaxed). In addition to Constraints (2)-(8), we consider the primal formulation including the additional valid inequalities (32)-(35) introduced in Section 3.3. Its dual is:

$$\min \quad \sum_{k\in K} P_k\lambda_k^{(2)} + \mathsf{maxstops}\ \lambda^{(5)} + \mathsf{maxlines}\ \lambda^{(6)} + \mathsf{maxfleets}\ \lambda^{(7)} + \sum_{f\in F}\sigma_f^z + \sum_{\ell\in L}\sigma_\ell^y + \sum_{s\in S}\sigma_s^x \tag{36}$$

$$\lambda_k^{(2)} + \lambda_\ell^{(3)} + \lambda_s^{(4)} + \lambda_t^{(4)} \geq 1 \qquad \forall k\in K, \ell\in L, s\in S^{\mathsf{in}}_{\ell k}, t\in S^{\mathsf{out}}_{\ell k} \tag{37}$$

$$\lambda^{(7)} - \sum_{\ell\in L_f}\lambda_{f\ell}^{(8)} + \lambda_f^{(35)} + \sigma_f^z \geq 0 \qquad \forall f\in F \tag{38}$$

$$-C_\ell\lambda_\ell^{(3)} + \lambda^{(6)} + \lambda^{(8)}_{\mathsf{f}(\ell),\ell} - \sum_{s\in S_\ell}\lambda_s^{(32)} + \lambda_\ell^{(33)} + \lambda_\ell^{(34)} - \lambda_{\mathsf{f}(\ell)}^{(35)} + \sigma_\ell^y \geq 0 \qquad \forall \ell\in L \tag{39}$$

$$-C_s\lambda_s^{(4)} + \lambda^{(5)} + \lambda_s^{(32)} - \sum_{\ell\in L_s^{\mathsf{in}}}\lambda_\ell^{(33)} - \sum_{\ell\in L_s^{\mathsf{out}}}\lambda_\ell^{(34)} + \sigma_s^x \geq 0 \qquad \forall s\in S \tag{40}$$

$$\lambda^{(*)}, \sigma^z, \sigma^y, \sigma^x \geq 0 \tag{41}$$

4.1 The pricing problem

Given a solution (λ, σ) of the dual of the RMP, the reduced cost of a variable $w^k_{\ell st}$ is given by

$$\bar{c}_{k\ell st} = 1 - \left(\lambda^{(2)}_k + \lambda^{(3)}_\ell + \lambda^{(4)}_s + \lambda^{(4)}_t\right). \tag{42}$$

Hence, the pricing problem asks to find values for $k \in K,\ \ell \in L,\ s \in S^{\mathsf{in}}_{\ell k}, t \in S^{\mathsf{out}}_{\ell k}$ such that $\bar{c}_{k\ell st}$ is positive and maximized, which turns to minimize the sum between brackets. Instead of performing a complete enumeration of all potential variables, we propose to tackle the minimization problem as a *shortest path problem* in a particular directed acyclic graph (DAG). We first develop the proposed solution approach and then further discuss its potential benefits in practice (versus a complete enumeration). Given an instance of the SFOT, we define the *pricing network* as the directed graph $G = (V, A)$, where $V = \{v_0\} \cup K \cup K_S \cup K_L \cup K_T \cup \{v_f\}$, with

- v_0 and v_f are the source and sink nodes of G
- $K_S = \{(k, s) \in K \times S : s \in S^{\mathsf{in}}_k\}$
- $K_L = \{(k, \ell) \in K \times L \ :\ S^{\mathsf{in}}_{\ell k} \neq \emptyset \wedge S^{\mathsf{out}}_{\ell k} \neq \emptyset\}$
- $K_T = \{(k, t) \in K \times S : t \in S^{\mathsf{out}}_k\}$

and $A = (\{v_0\} \times K) \cup X_1 \cup X_2 \cup X_3 \cup (K_T \times \{v_f\})$, with

- $X_1 = \{(k, (k, s)) \in K \times K_S\}$
- $X_2 = \{((k, s), (k, \ell)) \in K_S \times K_L :\ s \in S^{\mathsf{in}}_{\ell k}\}$
- $X_3 = \{((k, \ell), (k, t)) \in K_L \times K_T :\ t \in S^{\mathsf{out}}_{\ell k}\}$

Figure 5 shows an example of the pricing network described above, illustrating all parts of the structure. For a solution (λ, σ) of the dual of the RMP, we define the cost c_i of a node $i \in V$ as follows:

$$\begin{aligned}
c_{v_0} = c_{v_f} &= 0, && \\
c_k &= \lambda^{(2)}_k && \forall k \in K \\
c_{(k,s)} &= \lambda^{(4)}_s && \forall (k, s) \in K_S, \\
c_{(k,\ell)} &= \lambda^{(3)}_\ell && \forall (k, \ell) \in K_L, \\
c_{(k,t)} &= \lambda^{(4)}_t && \forall (k, t) \in K_T.
\end{aligned}$$

We observe that each variable $w^k_{\ell st}$ is associated with a path from v_0 to v_f in the given pricing network, namely $P^k_{\ell st} := \{v_0,\ k,\ (k, s),\ (k, \ell),\ (k, t),\ v_f\}$, and the sum of the costs on the arcs of this path is equal to the sum in brackets in the corresponding reduced cost (42). Therefore, the pricing problem can be solved by finding a shortest path in the pricing network, using arc lengths $\bar{c}_{ij} := c_j$, for each arc $(i, j) \in A$, where costs c_j are defined above.

Reducing the pricing problem to a shortest path problem instead of performing a full enumeration allows to exploit effective solution algorithms, e.g., Dijkstra's shortest path algorithm, to avoid analysing every possible path. In addition, the pricing network can be created once at the beginning (and stored in an efficient data structure) and then used for each pricing step of the algorithm. Moreover, the structure of the network is such that it can also be used implicitly (i.e., not stored in memory but traversed as needed during the search).

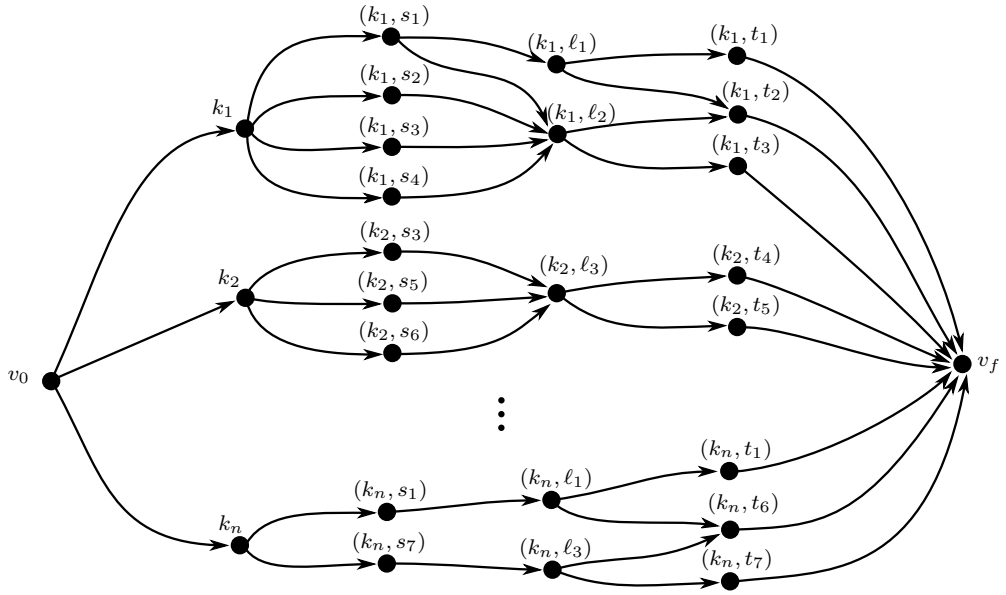


Figure 5: Example of a *pricing network* associated with the pricing problem.

4.2 Implementation details

During the solution of the RMP, several pricing problems are solved. Each of them consists in searching for a path variable with positive reduced cost to be added to the RMP. Even if the inclusion of a single variable is sufficient to continue with the process, it is widely known that adding several variables (with positive reduced costs) may significantly improve solution times. Specifically in our setting, adding variables for different commodities may be critical to finding optimal solutions fast. Next we describe some important features of our implementation, regarding mainly the search and inclusion of path variables into the RMP.

Number of columns. In each pricing step, we generate as many columns as possible to be added to the RMP. However, in order to control the size of the RMP, we define the following two parameters for the algorithm:

- MCK: maximum number of columns related to a single commodity that can be added per pricing step.
- MC: maximum number of columns (in total) that can be added per pricing step.

Commodities' pricing. The set of commodities is analysed in an arbitrary order. Also, with the goal of avoiding arbitrary asymmetries on the number of columns generated per commodity, at each pricing step we perform individual searches for each commodity separately, always limiting to MCK the number of generated columns for a given commodity. Furthermore, if at some time of one pricing step the global limit MC is reached (hence interrupting this pricing step), we store the current commodity k being analyzed and we use this commodity as a starting point for the next pricing step (shifting the order in a round-robin fashion).

Pricing strategies We implemented 3 different strategies to price each commodity. As each commodity is priced separately, all strategies are applied to the sub network associated with the corresponding commodity.

- DIJKSTRA: we use Dijkstra's t-shortest path algorithm with $t =$ MCK on the pricing network to find the best t columns with highest reduced cost.
- FIRST: we perform a straightforward enumeration of all potential columns and keep the first MCK columns with a positive reduced cost.

- MIX: we perform a straightforward enumeration and keep the first MCK columns the reduced cost of which is bigger than a given threshold μ. If less than MCK columns are added, we complete up to this number by keeping the best columns among the remaining ones with positive reduced cost. We remark that a variable with positive reduced cost is associated with a path on the pricing network with a cost smaller than 1. With this in mind, we fix the threshold $\mu = 0.5$, which means that the weight of the path should be smaller than 0.5.

MIP solution time. Our MIP heuristic consists in solving the linear relaxation of the formulation by means of column generation and then solving the MIP formulation restricting the set of path variables to those generated in the column generation phase. The whole algorithm is run within a given time limit and, especially on large instances, even solving the linear relaxation may consume the entire allocated time, causing the heuristic to end without a feasible solution. To avoid this situation, a percentage of the given time is saved as a *safe time* to run the final MIP. This means that if at some point of the column generation phase, the remaining time reaches the *safe time*, then the column generation is interrupted and the final MIP is run on this generated set of columns. We observe that when this happens, no upper bound can be deduced from the linear relaxation, as it is not solved to optimality. We further address this issue on Section 4.3.

4.3 Lagrangian bounds

We recall that the optimal value of the linear relaxation is an upper bound for the feasible solutions of the SFOT and can be used as a certificate of optimality gap of the obtained heuristic solution. However, if the column generation phase is interrupted before achieving optimality, no upper bound can be deduced. We address this issue proposing a method to derive proper upper bounds from the solutions of the intermediate RMPs. We note that for each $k \in K$, $\ell \in L$, $s \in S^{\mathsf{in}}_{\ell k}$ and $t \in S^{\mathsf{out}}_{\ell k}$, the dual constraint associated with variable $w^k_{\ell st}$ is constraint (37), i.e.,

$$\lambda^{(2)}_k + \lambda^{(3)}_\ell + \lambda^{(4)}_s + \lambda^{(4)}_t \geq 1.$$

Let $(\bar{\lambda}, \bar{\sigma})$ be a feasible (not necessarily optimal) solution of the dual of the RMP (as modelled in (36)–(41)) and its associated objective function value be $\bar{Z}$. This solution may not be feasible for the dual of the original formulation, as constraints (37) associated with the columns that are left out from the master may be violated by $(\bar{\lambda}, \bar{\sigma})$. For each $k \in K$, let $\bar{Z}_k$ be the maximum reduced cost associated with commodity k, i.e.,

$$\begin{aligned}\bar{Z}_k &= \max_{,s,t} \left\{1 - \left(\bar{\lambda}^{(2)}_k + \bar{\lambda}^{(3)}_\ell + \bar{\lambda}^{(4)}_s + \bar{\lambda}^{(4)}_t\right)\right\} \\ &= 1 - \min_{,s,t} \left\{\bar{\lambda}^{(2)}_k + \bar{\lambda}^{(3)}_\ell + \bar{\lambda}^{(4)}_s + \bar{\lambda}^{(4)}_t\right\}. \end{aligned} \tag{43}$$

Then, for each $\ell \in L$, $s \in S^{\mathsf{in}}_{\ell k}$ and $t \in S^{\mathsf{out}}_{\ell k}$, it holds by definition that

$$\bar{Z}_k \geq 1 - \left(\bar{\lambda}^{(2)}_k + \bar{\lambda}^{(3)}_\ell + \bar{\lambda}^{(4)}_s + \bar{\lambda}^{(4)}_t\right),$$

which may be stated as

$$(\bar{\lambda}^{(2)}_k + \bar{Z}_k) + \bar{\lambda}^{(3)}_\ell + \bar{\lambda}^{(4)}_s + \bar{\lambda}^{(4)}_t \geq 1. \tag{44}$$

With these definitions, we can obtain a feasible solution $(\tilde{\lambda}, \tilde{\sigma})$ for the dual of the (LP) master problem, based on $(\bar{\lambda}, \bar{\sigma})$ by setting $\tilde{\lambda}^{(2)}_k = max(\bar{\lambda}^{(2)}_k + \bar{Z}_k, 0)$, for each $k \in K$ (leaving all remaining variables unchanged from $(\bar{\lambda}, \bar{\sigma})$). We remark that $(\tilde{\lambda}, \tilde{\sigma})$ is feasible as $\lambda^{(2)}_k$ (i.e., the sole modified variable) appears only in the dual constraint associated to variables $w^k_{\ell st}$, i.e., in constraint (37), and the latter is satisfied by (44). Note

that by taking $\tilde{\lambda}_k^{(2)} = max(\bar{\lambda}_k^{(2)} + \bar{Z}_k, 0)$ we also prevent the non-negativity constraint of this variable being violated (in case of a negative $\bar{Z}_k$). As $(\tilde{\lambda}, \tilde{\sigma})$ is feasible for the dual of the (LP) master problem, then its objective function value

$$\tilde{Z} = \bar{Z} + \sum_{k \in K} (\tilde{\lambda}_k^{(2)} - \bar{\lambda}_k^{(2)}) P_k$$

is a valid upper bound for the master problem (we recall that P_k is the right-hand-side of (2), thus the coefficient of $\lambda_k^{(2)}$ in the dual objective function).

5 Computational experiments

In this section, we present numerical results on several computational experiments. We first compare the three formulations presented in Section 3.2. Secondly, we show the results obtained by the Column Generation (CG) heuristic of Section 4 when using different values for the parameters, in order to find the best configuration. We compare then the CG-heuristic approach with the three formulations of the SFOT on different criteria, such as running time and quality of the obtained solution. We close this section by performing a managerial sensitivity analysis, in order to gain further insight on the strategical issues related to FOT systems.

Our computational framework is a desktop PC running Microsoft Windows 10 on an Intel Xeon CPU at 3.7 GHz and 128 Gb of RAM. We use CPLEX version 20.1 to solve the MIP formulations. The column generation approaches are developed in JAVA by resorting to CPLEX's Java API. In all approaches, we use CPLEX default configuration, except for the number of parallel threads which is limited to one for a better control on the measuring of running times.

5.1 Instance generation

Gathering real data to generate realistic SFOT instances is a challenging issue, involving logistic real data (e.g., public transport lines and stops), geographical data (e.g., depots and destinations locations), realistic demand data (e.g., neighbourhood-dependent demands, etc.) and coherent infrastructure capacities (lines and stops capacities, strategical constraints limits). For this first work on the SFOT, we generate nearly-realistic instances from synthetic data. We developed an instance generator which generates instances of the SFOT based on the following input parameters: the number n_{lin} of lines to generate, the number n_{sto} of drop-in and drop-out stops on each line, the number n_{fle} of available fleets to which lines will be assigned, and the numbers n_{ori} and n_{des} of origins and destinations to be generated in the instance. For the sake of transparency, we describe next the key aspects of our instance generator.

Each location (i.e., stops, origins and destinations) is generated in a rectangular area composed of a subarea A_c of a fictional city and a subarea A_s of its surroundings. Our procedure starts by generating the n_{lin} lines, one at a time. Each line ℓ starts from an initial stop, which is randomly generated in A_s. Then, n_{sto} consecutive drop-in stops are generated for line ℓ, towards the central zone of the city area A_c (with potential random perturbations in the direction). The distance between each stop and the following stop is given by a predefined *step size* (with potential random perturbations). Finally, n_{sto} consecutive drop-out stops are generated following the same direction (and *step size*), towards the surroundings of the city on the opposite side of the initial point. If at any time, a stop is generated close enough to a previously generated stop, then the two stops are merged into a single one (thus, allowing different lines to share stops). For each line ℓ, we generate the reverse line by associating the stops in the inverse sequence (drop-in stops as drop-out and vice versa). In terms of the models, the reverse path would be represented as a different line ℓ', independent from ℓ (although associated with the same fleet). For a better representation of real bus paths, we fix a probability q for each stop of the reverse path to be placed at a different location than the one from ℓ (in which case a new stop may be generated). In our implementation we fix $q = 30\%$.

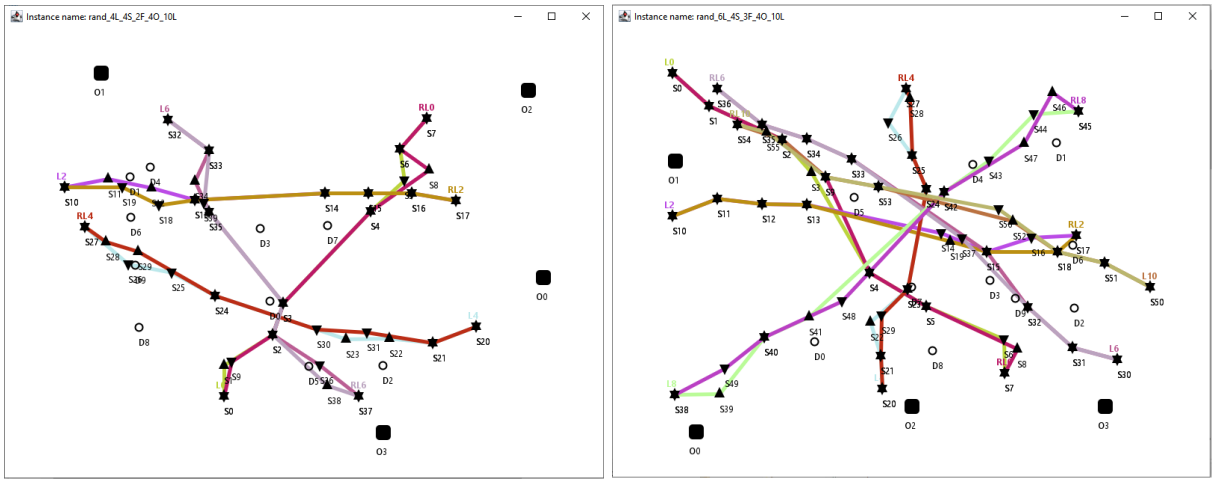


Figure 6: Two examples of instances created with our SFOT instance generator.

The location of the n_{ori} origins are randomly generated in A_s and the location of the n_{des} destinations are generated in A_c. As an example of the obtained geographical distribution, Figure 6 depicts two instances constructed by our generator; squares represent origins, circles represent destinations, each color line represents a public transport line and triangles represent stops (pointing up are drop-in stops and pointing down are drop-out stops; some stops may be both drop-in and drop-out stops, although for different lines).

As far as the demands of the generated instances, for each origin-destination pair (o, d), a probability p is defined to determine whether a demand k from o to d is generated or not. In our instances, we fix $p = 80\%$. If such demand k is generated, then the set of reachable drop-in stops S_k^{in} (resp. drop-out stops S_k^{out}) are those stops within a radius from o (resp. d) of a predefined number of steps (in terms of the mentioned *step size*). In our implementation we limit this to 5 steps from origin o_k for drop-in stops in S_k^{in}, and 3 steps from destination d_k for drop-out stops S_k^{out}. We remark that we do not generate origins or destinations with no reachable stops, however, even when both S_k^{in} and S_k^{out} are non-empty, demand k might be *unreachable* if there exists no line $\ell \in L$ with both a drop-in stop in S_k^{in} and a drop-out stop in S_k^{out}, i.e., if for each line $\ell \in L$, $S_{\ell k}^{\mathsf{in}} = \emptyset$ or $S_{\ell k}^{\mathsf{out}} = \emptyset$. We further explore this issue in Section 5.5.

All numerical elements generated for the instance (e.g., capacities, demanded packages, etc.) are randomly chosen within predefined ranges according to a uniform distribution. After some preliminary experiments, we set these ranges as follows:

- Demands: $P_k \in [1, 10]$.
- Capacities: $C_\ell \in [80, 400]$, $C_s \in [50, 200]$.
- Constraints:
  - $\mathsf{maxstops} \in [0.05 \times |S|, 0.2 \times |S|]$.
  - $\mathsf{maxlines} \in [0.1 \times |L|, 0.3 \times |L|]$.
  - $\mathsf{maxfleets} \in [0.3 \times |F|, 0.5 \times |F|]$.

For the purpose of this work we generated two sets of instances, with two different objectives. The first set is used to compare the computational performance of the various approaches to solve the SFOT, and it is composed of instances of different sizes: $n_{\mathsf{lin}} \in \{20, 40, 60\}$, $n_{\mathsf{sto}} \in \{4, 8\}$, $n_{\mathsf{ori}} \in \{5, 10\}$ and $n_{\mathsf{des}} \in \{50, 100, 200, 500\}$. Every possible combination of these values is considered except the ones having $n_{\mathsf{ori}} = 10$ and $n_{\mathsf{des}} \leq 200$, which we do not consider as realistic scenarios (recall that n_{ori} refers to the number of CDCs). For each combination of the above parameters, 3 random instances are generated, for a total of 90 instances. We call this set the *Group A*, which is used in the next three subsections. The second set of

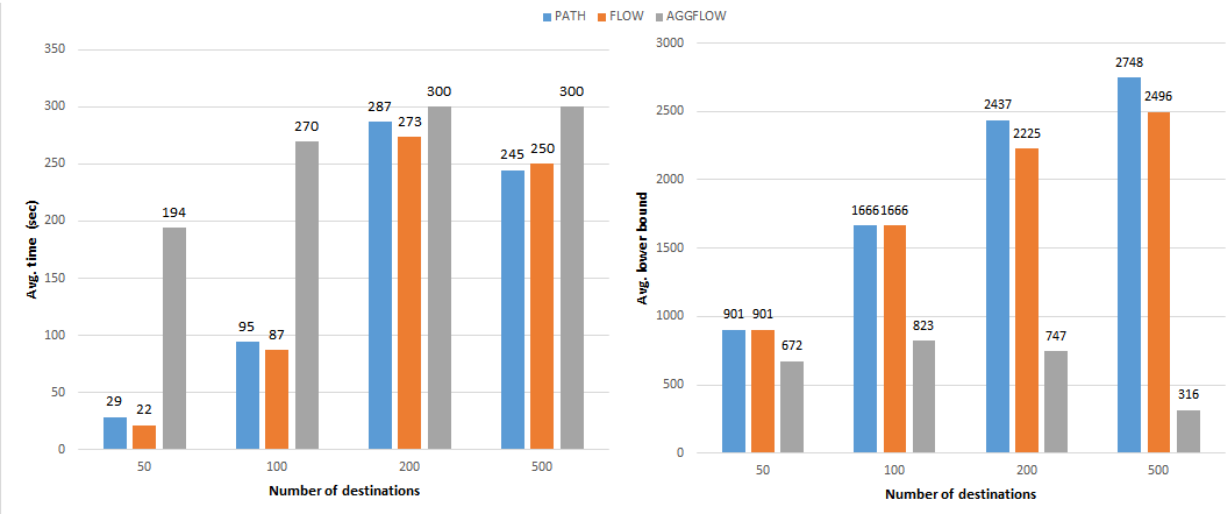


Figure 7: Average times (left) and lower bounds (right) obtained by the 3 MIP formulations on instances from Group A, grouping the instances by number of destinations.

instances, used in the managerial analysis, is described later in Section 5.5. All these instances generated for our computational experimentation will be available for public use after publication of this paper.

5.2 Empirical comparison of the three MIP models

The first set of experiments is made on the Group A instances and compares the performance of the three MIP models proposed in Section 3.2, with a time limit of 300 seconds. We call PATH, FLOW and AGGFLOW the path-based, flow-based, and aggregated flow-based formulations, respectively. Figure 7 shows the average running times and average lower bounds (i.e., best solution found) obtained by each formulation, grouping the instances according to the number of destinations.

Focusing on running times, AGGFLOW is clearly much more time-consuming than the other two formulations. It reaches the time limit for all instances with 200 and 500 destinations (even for 2 instances with 500 destinations which are solved in less than 10 seconds by PATH and FLOW). On the other hand, the running times of PATH and FLOW are tied; FLOW seems to be slightly faster on instances having up to 200 destinations, but it is outperformed by PATH on the group of largest instances.

When analyzing the obtained solutions (right chart on Figure 7), we can see again that AGGFLOW is considerably outperfomed by the other two formulations. In turn, this chart shows that PATH obtains better solutions than FLOW as the size of the instances grow. For a better understanding of the differences on the lower bounds obtained by the latter two formulations, we show in Figure 8 a cumulative chart of the dual gaps obtained for all the instances in Group A. The dual gap for an instance is calculated as $(ub - lb)/ub$, where lb and ub stand for the lower and upper bound, respectively, obtained by the model. In the cumulative chart, for each value t on the horizontal axis, we show the number of instances in which the obtained gap is not bigger than t. For a better visualization, we cut the horizontal axis to include only instances in which the obtained gaps are smaller than 100%. As can be seen, PATH shows to be more efficient than FLOW for closing dual gaps.

Based on the results presented above, we conclude that the PATH formulation is the most promising, especially when the size of the instances grows. In the following subsections we explore the limits of this formulation both as an exact method and as a heuristic one, by using column generation techniques described in Section 4.

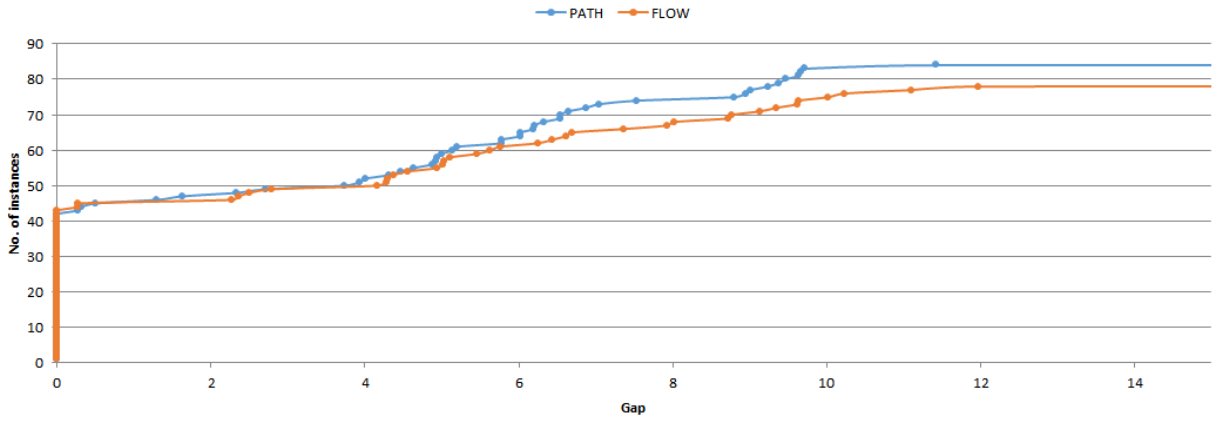


Figure 8: Number of instances with a dual gap smaller than a given value (on the horizontal axis).

5.3 Column generation strategies

In Section 4.2 we proposed three strategies to solve the pricing problem when applying column generation to solve the linear relaxation of the PATH formulation, namely: DIJKSTRA, FIRST and MIX. We recall that the pricing problem is run separately for each commodity and the routine is based on the following parameters:

- MCK: maximum number of columns associated with a single commodity that can be added per pricing step.
- MC: maximum total number of columns that can be added per pricing step.

Based on the results of exhaustive computational experiments (which are shown in Appendix A), it was possible to determine the best (most stable) configuration for each of the three strategies, which is MC = 500 and MCK = 10. In addition, based on the same experiments, we set the *safe time* parameter (see Section 4.2) to be 10% of the time limit.

5.4 Comparing the CG-heuristics with the PATH formulation

We now assess the performance of the GC-heuristics detailed in Section 4, i.e., solve the LP by means of CG, then solve a reduced MIP with the columns generated during the CG phase. We test the three strategies to solve the pricing problem with the best configuration settings for each of them (as described in Section 5.3). The experiments are made on instances in Group A, with a time limit of 300 seconds.

In Figure 9 we show the average running times (left) and average values (right) of the solutions obtained by PATH and the three CG-heuristics, over all instances in Group A. Instances are grouped by number of destinations. It can be seen that the average running times are considerably higher for PATH for all subgroups of instances. Even if the PATH formulation obtains better objective values on small instances (i.e., up to 200 destinations), the difference against solutions obtained by the CG-heuristics is not significant (PATH gets improvements of 3.4%, 2.8% and 4.2% for the three first subgroups of instances, respectively). Moreover, for the subgroup of biggest instances, the quality of the solutions obtained by PATH decreases and it is considerably outperformed by the CG-heuristics, whose solution values are around 30% better than PATH.

We further analyze the groups of biggest instances and we show in Figure 10 the same data as in Figure 9 but considering only instances with 200 and 500 destinations (grouped by the number of lines). We can see again a similar behaviour both on the running times and in the quality of the obtained solutions. In terms of solution quality, we note that the largest difference is obtained on the hardest instances (i.e., instances

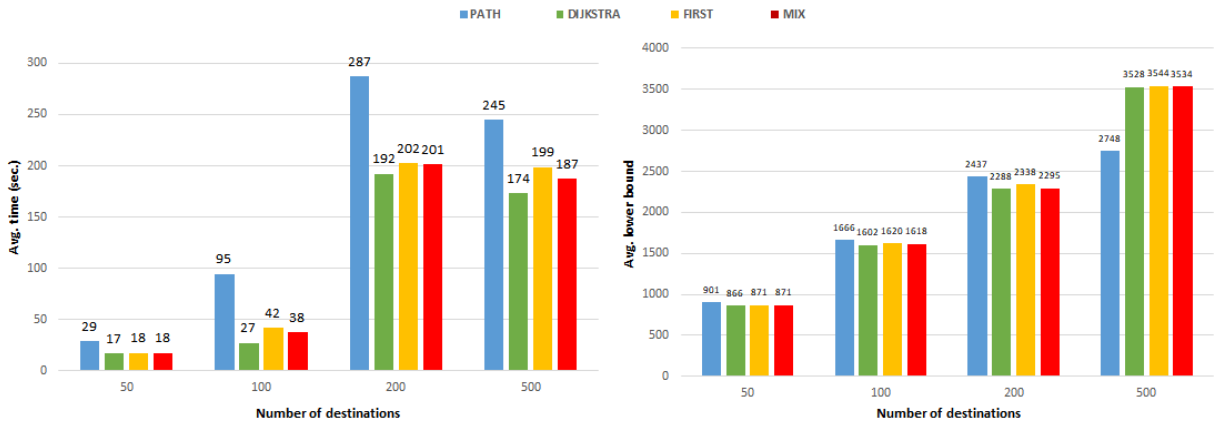


Figure 9: Average running times (left) and average values (right) of solutions obtained by PATH and the three CG-heuristics.

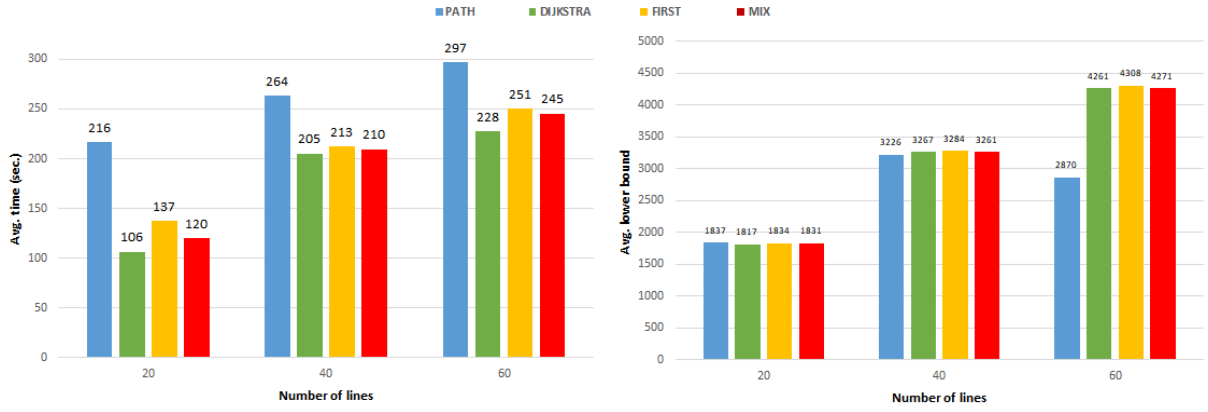


Figure 10: Average running times (left) and average values (right) of the solutions obtained by PATH and the three CG-heuristics (instances with 200 and 500 destinations only).

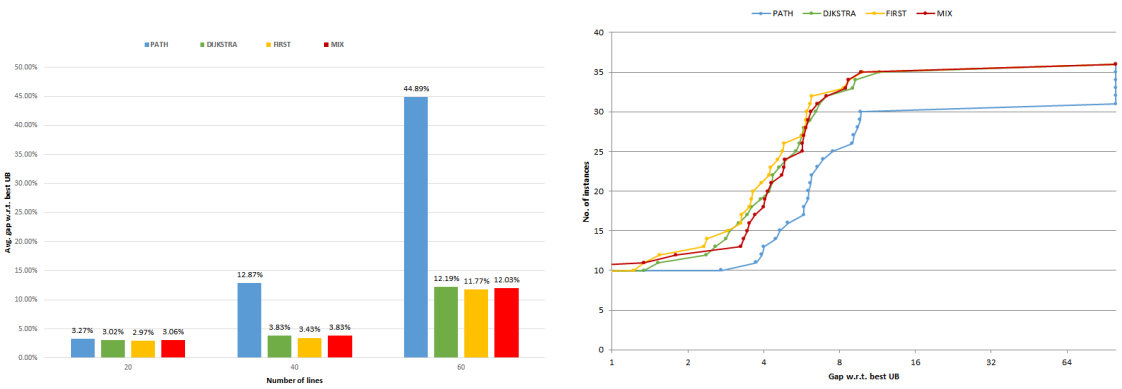


Figure 11: Dual gaps obtained by each approach on Group A instances with 500 destinations. Left chart shows the average dual gaps (grouped by number of public lines); right chart depicts the number of instances having a dual gap not greater than a given value.

with 60 lines), in which the average solution values found by the MIP heuristic achieve an improvement of around 50% with respect to the average values found by PATH.

To conclude this section, we analyze the dual gaps represented by the obtained solutions. To this end, for each individual instance, we consider the best (i.e., smallest) upper bound ub^* found within the given time limit by any of the three exact MIP formulations (i.e., the optimal solution if found, or the best achieved dual bound when time limit is reached), and we use this value to calculate the dual gap obtained by each of the methods analyzed in this section. We calculate this gap as $(ub^* - lb)/ub^*$, where lb is the solution value obtained by the corresponding method. We focus on the set of biggest instances, i.e., those with 500 destinations. The left chart of Figure 11 shows the average dual gap (as defined above) obtained by each method grouping instances by number of available lines. As evidenced by the chart, the CG-heuristics produces much better solutions. Only for the set of instances with 20 lines the obtained gaps are similar to the ones of PATH. However, when focusing on instances with 40 and 60 lines, we can see that the gap obtained by PATH is significantly larger than the one obtained by the CG-heuristics. Specifically, for these two groups of instances, the PATH formulation obtains average gaps of 12.87% and 44.89% while the CG-heuristics obtain gaps around 3.70% and 12.03% (average over the three strategies), respectively. With the goal of analyzing this data in a disaggregated fashion, the right chart of Figure 11 arranges this information in a cumulative chart. For each value t on the horizontal axis, we show the number of instances in which the obtained gap is at most t. For a better visualization, we display the horizontal axis in logarithmic scale. As can be seen, the gaps obtained by the CG-heuristics are usually smaller than the ones for PATH. In particular, we note that PATH obtains a gap of 100% for 6 of the 36 instances of this group, while the CG-heuristics achieve gaps smaller than 12% for all but one of these instances (and a gap of 100% for the remaining one), thus confirming their superiority.

Finally, we further restrict our attention on the subset of largest instances of this group, namely those with 40 and 60 lines, and we show in Figure 12 two box plot diagrams of the obtained gaps. The left diagram considers instances with 40 and 60 lines while the right diagram focuses on the instances with 60 lines only. This in order to show the trend of behavior of the approaches for an increasing difficulty of the instances. Center lines on each box show the medians and box limits indicate the 0.25 and 0.75 percentiles. Whiskers extend 1.5 times the interquartile range from the 0.25 and 0.75 percentiles and outliers are represented by dots. This figure evidences that the distribution of the individual results is considerably different when comparing PATH with the CG-heurisitics. In both diagrams we can see that the CG-heuristics concentrate almost all gaps in the range of 10% (with a single outlier of 100%), while the gaps obtained by PATH are often considerably higher and distributed along a wide range of gaps, up to almost 60% on the left diagram and up to 100% on the hardest set of instances (right diagram). Even the median gap obtained by PATH is greater than all gaps on the first three quartiles of the CG-heuristics results. These results show that the

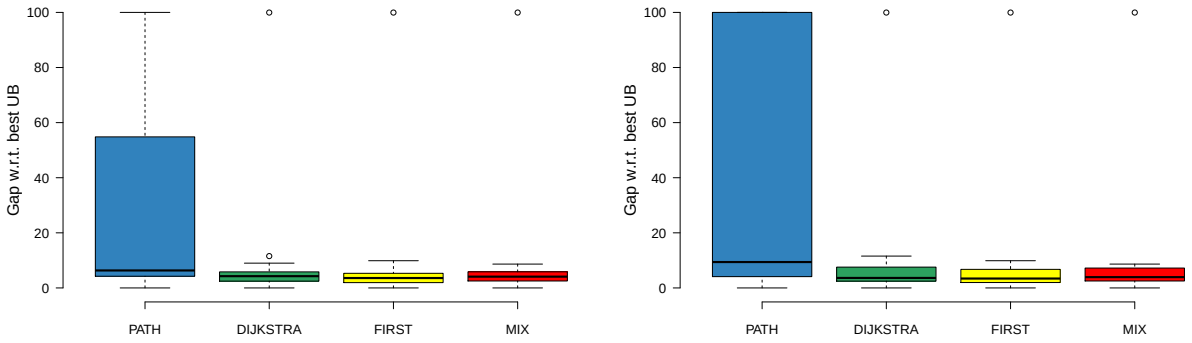


Figure 12: Box plot diagram of the dual gaps obtained by each method on the set of instances with 500 destinations. Left chart use instances with 40 and 60 lines; right chart only those with 60 lines.

CG-heuristics outperform the exact MIP approaches.

5.5 Sensitivity analysis and managerial insights

The instances of the SFOT involve many elements and parameters that affect the solutions in different aspects, such as lines and stops capacities, strategical constraints, number of available lines, number of commodities, etc.

In this section we perform a sensitivity analysis with the aim of finding some managerial insights on different elements of the SFOT. To this end, we designed a test set of *base instances*, from which several scenarios are constructed by relaxing/restricting constraints or increasing/reducing resources availability. Specifically, given a base instance I, we define the modified instance $I(f_{\mathsf{card}}, f_{\mathsf{cap}}, f_{\mathsf{lin}}, f_{\mathsf{dest}})$ as the instance obtained from I by doing the following modifications:

- strategical limits maxstops, maxlines and maxfleets are scaled by a factor f_{card},
- stops and line capacities C_s and C_ℓ are scaled by a factor f_{cap},
- the set of lines available for the instance is composed only by the first $\lceil f_{\mathsf{lin}}|L| \rceil$ lines in L (new lines may be generated if $f_{\mathsf{lin}} > 1$), and
- the set of destinations is composed only by the first $\lceil f_{\mathsf{dest}}|D| \rceil$ destinations in D with their corresponding demands (new destinations and demands may be generated if $f_{\mathsf{dest}} > 1$).

For example, given an instance I, we can create instances $I(f_{\mathsf{card}}, 1, 1, 1)$, for $f_{\mathsf{card}} \in \{1.2, 1.6, 1.8, 2.0\}$, to explore the improvement in the solution quality when the strategical limits (i.e., maxstops, maxlines and maxfleets) are increased by 20%, 60%, 80% and 100%, respectively.

Our analysis aims to explore the marginal contribution of varying the four key dimensions or parameters mentioned above, namely: increasing lines and stops capacities, relaxing cardinality constraints, expanding the set of available lines and expanding the set of destinations (and its associated demands). Our test set is constructed starting from a set $\mathcal{I}_{\mathsf{base}}$ of 10 *base instances* generated by our instance generator setting $n_{\mathsf{lin}} = 5$, $n_{\mathsf{sto}} = 10$, $n_{\mathsf{ori}} = 5$ and $n_{\mathsf{des}} = 100$ (see Section 5.1 for references on these parameters). From each of these instances we obtain 3 additional instances for each of the mentioned dimensions, in which the corresponding dimension is increased and/or relaxed by 20%, 60% and 100%. Specifically, we generate the following sets of instances (which, for completion, include also base instances):

- $\mathcal{I}_{\mathsf{card}} = \{I(f_{\mathsf{card}}, 1, 1, 1) : I \in \mathcal{I}_{\mathsf{base}} \text{ and } f_{\mathsf{card}} \in \{1, 1.2, 1.6, 2.0\}\}$
- $\mathcal{I}_{\mathsf{cap}} = \{I(1, f_{\mathsf{cap}}, 1, 1) : I \in \mathcal{I}_{\mathsf{base}} \text{ and } f_{\mathsf{cap}} \in \{1, 1.2, 1.6, 2.0\}\}$

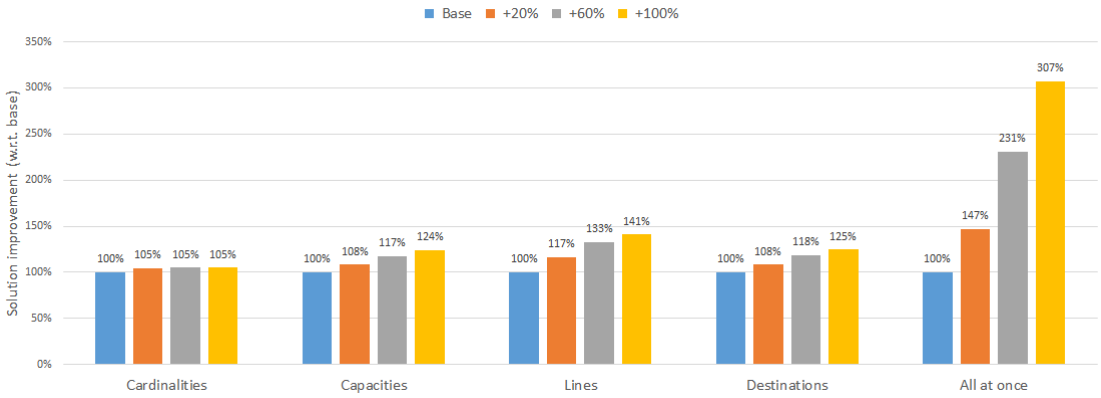


Figure 13: Average improvement on solution value when each key parameter is increased by 20%, 60%, 100%.

- $\mathcal{I}_{\mathsf{lin}} = \{I(1,1,f_{\mathsf{lin}},1) : I \in \mathcal{I}_{\mathsf{base}} \text{ and } f_{\mathsf{lin}} \in \{1, 1.2, 1.6, 2.0\}\}$
- $\mathcal{I}_{\mathsf{dest}} = \{I(1,1,1,f_{\mathsf{dest}}) : I \in \mathcal{I}_{\mathsf{base}} \text{ and } f_{\mathsf{dest}} \in \{1, 1.2, 1.6, 2.0\}\}$
- $\mathcal{I}_{\mathsf{all}} = \{I(f,f,f,f) : I \in \mathcal{I}_{\mathsf{base}} \text{ and } f \in \{1, 1.2, 1.6, 2.0\}\}$

We remark that the increments 20%, 60% and 100% are specifically chosen to yield integer values when expanding the discrete sets of available lines and destinations (i.e., $\mathcal{I}_{\mathsf{lines}}$ and $\mathcal{I}_{\mathsf{dest}}$), to be properly in line with the other two dimensions. Note that the first four set of instances are aimed at studying what happens by modifying one dimension at a time. The last set instead considers the modifications of all elements simultaneously.

5.5.1 Effects on the solution value

Figure 13 shows the solution values when each of the considered dimensions is increased by 20%, 60% and 100%. The values shown on the chart represent the average improvement of the solution value with respect to the base instances (the latter are included and labeled as Base in the chart for reference). Each dimension is analyzed individually on the first four groups of bars by resorting to the sets of instances $\mathcal{I}_{\mathsf{card}}$, $\mathcal{I}_{\mathsf{cap}}$, $\mathcal{I}_{\mathsf{lin}}$ and $\mathcal{I}_{\mathsf{dest}}$, respectivelly. In addition, we show in the fifth group of bars the results obtained when all four dimensions are increased at the same time in the set of instances $\mathcal{I}_{\mathsf{all}}$.

As can be seen in Figure 13, the percentage of value improvement is not always in line with the percentage of increase of the parameter w.r.t. the base instances, which is not surprising in sensitivity analysis. The smallest improvements are found when relaxing cardinality constraints (group Cardinality), which seems to have almost no impact on the solution quality, beyond the first increase of 20%, which in turn yields a marginal improvement in the solution value of just 5%. This suggests that cardinality constraints are slack after this point and the limitation is given by other dimensions of the problem. The best improvements are obtained when adding more lines (group Lines), however, even when doubling the number of lines (+100%) the average improvement is just 41%. We remark that adding lines to an instance not only adds delivery capacity, but may also increase the set of *reachable demands*, thus increasing the potential of the objective function.

Even though individual increase of each dimension do not result in the same increase o the solution value, a remarkable fact is that when all four dimensions are relaxed at the same time (group All at once), significant improvements are obtained. In particular, doubling all dimensions at once (+100%) allows to increase the objective values by a factor 3 compared to those of the base instances.

5.5.2 Effects on the resource saturation levels

For a better comprehension of the limitations imposed by each of the four considered dimensions, we focus now on the *level of saturation* of different element of the SFOT. Specifically, we analyze the following aspects:

- *Covered demand:* the percentage of the reachable demand covered by the solution. Recall that for a given set of lines, not necessarily all demand is reachable (see Section 5.1). We distinguish between the *covered reachable demand* (referred as covered demand in what follows) and *covered total demand* (which may change, when e.g., additional lines are introduced).
- *Lines/stops capacity:* the percentage of the full capacity of the lines/stops selected by the solution which is actually used.
- *Fleets/lines/stops cardinality:* the percentage of the maximum number of fleets/lines/stops imposed by the strategic constraints which has been selected as part of the solution. For example, if $\mathsf{maxlines} = 4$ and only 3 lines are employed in the solution, then the line-cardinality saturation level is 75%.

Figure 14 depicts the average saturation levels of the elements described above, achieved when relaxing each of the four dimensions considered in the former section. We show one chart for each dimension and finally a chart on which all dimensions are relaxed at the same time.

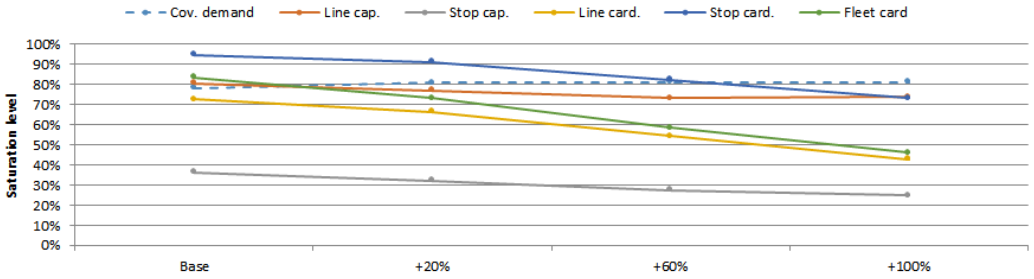
We observe from Figure 14a that when the maximum number of stops, lines and fleets is increased, the capacity of the network cannot be fully exploited. For example, even though we allow to double the number of fleets, the number of actually used ones only marginally increases. Indeed, after doubling the allowed cardinality, we realize that there is only a negligible increase in the covered demand (namely from 79% for the base setting, to 81% for the setting where the cardinality is doubled). This indicates that the cardinality constraints are less constraining than the other elements of the model like capacity constraints, geographical dispersion of the demand and distribution of the lines.

Figure 14b depicts the features of solutions obtained when stop- and line-capacities are increased. The effect on the covered demand is much more pronounced in this case: we observe an increase from 79% of the total covered demand (base setting) to 92% (in the setting in which the allowed capacities are doubled). There is also a positive effect on reducing the number of used stops, fleets and lines. The latter reductions are only marginal, which is again due to the geographical features of the input graph, i.e., dispersion of the demand and distribution of the lines.

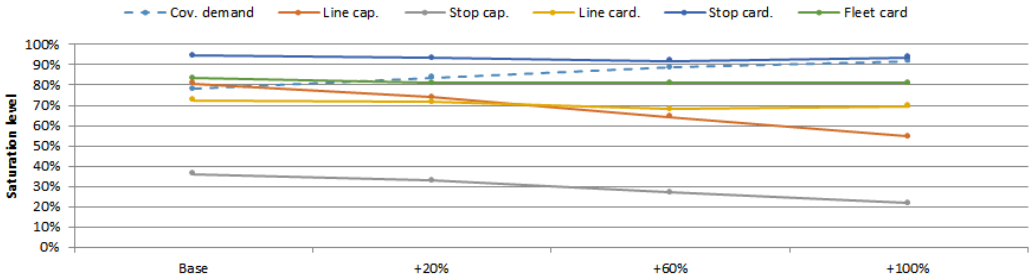
For decision makers concerned with maximizing the covered *total demand*, another important factor is the number of available lines and their geographical distribution. Indeed, as Figure 14c indicates, if the number of lines in the city is doubled, even with the same capacity and cardinality constraints, we are able to increase the covered total demand from 60% to 80%. A potentially even larger increase could be achieved, if these lines are ideally distributed in the network. However, such ideal topology is less likely to happen in practice. Concerning the covered reachable demand, we also notice an increase from 79% (base setting) to 89% (when the number of available lines is doubled).

In Figure 14d we analyze the effects caused by increasing the number of potential customers (i.e., the number of destinations). We assume that the available capacities and cardinalities are not changed, and we look at the impact of doubling the number of destinations. Not surprisingly, the covered reachable demand drops from 79% to 50%, if the number of destinations gets doubled. On the other hand, there is almost no effect on the size of the fleet and the number of lines used, which can be explained by the aggregation of demand. Indeed, more demand is getting served by the existing stops and existing lines, which can be seen from the increase in the utilization of the stop- and line-capacities.

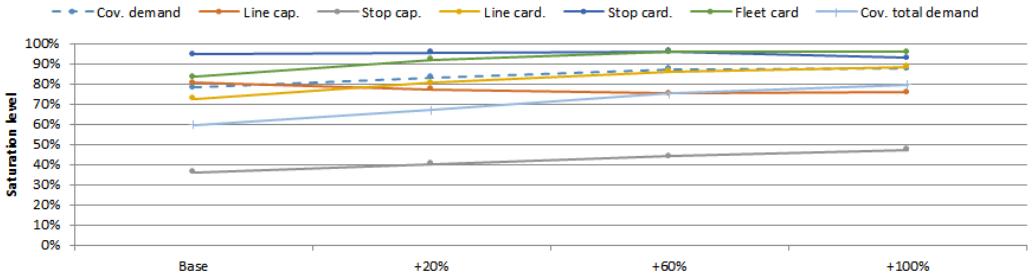
To summarize, we observe that major effects on increasing the covered (total or reachable) demand are due to improving the available infrastructure , i.e., adding additional lines and spreading them closer to the demand points. This goes in line with the analysis provided in Section 5.5.1. Additional but much smaller increase of the covered demand can be achieved by increasing the stop and line capacities. Finally, a marginal increase is also possible by allowing to select a larger number of lines, stops or fleets.



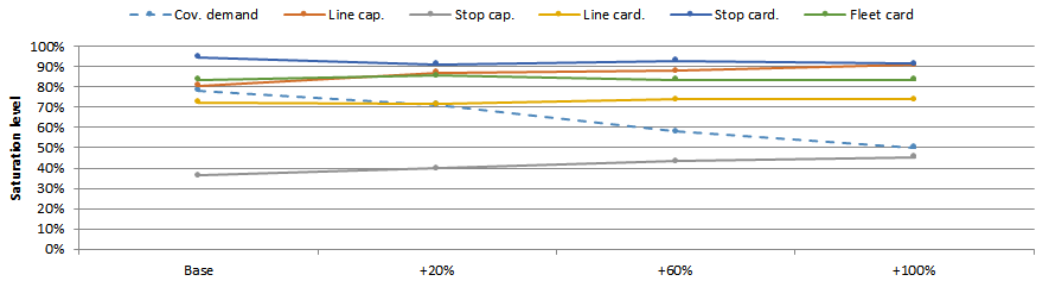
(a) Relaxation of cardinality restrictions



(b) Increase of capacities



(c) Increase of number of lines available



(d) Increase of number of destinations

Figure 14: Changes on the covered demands and resource utilisation when different elements of the problem are modified.

6 Conclusions

In this paper we studied a new problem related to a recent logistic concept in last-mile delivery, in which freight and passenger transportation are combined. We proposed three mathematical formulations for supporting strategic decisions, i.e., deciding which public lines to use and where to locate drop-out and drop-in stops. The three models are computationally assessed on a set of instances of medium size. For the best-performing of the three models, we then developed column generation-based MIP heuristics, capable of solving instances of realistic size. These heuristics provide high-quality solutions in acceptable times. A detailed analysis of managerial insights indicates that the topology of the public transportation network, in combination with the distribution and density of the demand, plays the most important role in maximizing the demand coverage.

We emphasize that the accuracy of the strategic model could be improved by considering the uncertainty and variability of customer demand using stochastic optimization. Our results indicate that the path-based formulation is a promising candidate for developing a two-stage stochastic model to tackle this problem.

Many open questions related to tactical and operational decisions still remain to be answered. In the strategic approach, we do not model the last leg of transportation. However, once the public transportation infrastructure is made available, one has to decide on the modes of transportation for the last leg and the associated fleet. In addition, daily paths for the package delivery have to be determined. To this end, new mathematical models and approaches need to be developed to solve these challenging combinatorial optimization problems in real time.

Acknowledgements

This work was funded by CY Initiative of Excellence (grant "Investissements d'Avenir" ANR-16-IDEX-0008"). This support is greatly appreciated.

References

[1] L. Alfandari, I. Ljubić, and M. D. M. da Silva. "A tailored Benders decomposition approach for last-mile delivery with autonomous robots". In: *European Journal of Operational Research* (2021).

[2] "Amazon packages to ride public transport". In: *Post & Parcel* (Feb. 2019). URL: `https://postandparcel.info/100304/news/amazon-packages-to-ride-public-transport/` (visited on 09/02/2021).

[3] W. Behiri, S. Belmokhtar-Berraf, and C. Chu. "Urban freight transport using passenger rail network: Scientific issues and quantitative analysis". In: *Transportation Research Part E: Logistics and Transportation Review* 115 (2018), pp. 227–245.

[4] A. Boudet. "Monoprix choisit Fret SNCF pour approvisionner ses magasins parisiens". In: *Les Echos* (July 2007). URL: `https://www.lesechos.fr/2007/07/monoprix-choisit-fret-sncf-pour-approvisionner-ses-magasins-parisiens-534433` (visited on 09/02/2021).

[5] F. Bruzzone, F. Cavallaro, and S. Nocera. "The integration of passenger and freight transport for first-last mile operations". In: *Transport Policy* 100 (2021), pp. 31–48.

[6] R. Chapuis, K. Tadjeddine, D. Chinn, R. Holmes, A. Knol, L. Speksnijder, K. Wolfs, C. Lotz, and S. Stern. "Restoring public transit amid COVID-19: What European cities can learn from one another". In: *McKinsey & Company* (June 2020). URL: `https://www.mckinsey.com/industries/travel-logistics-and-infrastructure/our-insights/restoring-public-transit-amid-covid-19-what-european-cities-can-learn-from-one-another` (visited on 09/02/2021).

[7] G. Cheng, D. Guo, J. Shi, and Y. Qin. "Planning city-wide package distribution schemes using crowd-sourced public transportation systems". In: *IEEE Access* 7 (2018), pp. 1234–1246.

[8] G. Cheng, D. Guo, J. Shi, and Y. Qin. "When packages ride a bus: Towards efficient city-wide package distribution". In: *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE. 2018, pp. 259–266.

[9] "Climate-Friendly Package Delivery with Cargo Bikes". In: (2021). URL: https://smartcity.wien.gv.at/en/remihub/ (visited on 09/21/2021).

[10] K. Cochrane, S. Saxe, M. J. Roorda, and A. Shalaby. "Moving freight on public transit: Best practices, challenges, and opportunities". In: *International Journal of Sustainable Transportation* 11.2 (2017), pp. 120–132.

[11] J. Desrosiers and M. E. Lübbecke. "A primer in column generation". In: *Column Generation*. Springer, 2005, pp. 1–32.

[12] S. Edrington, Z. Elgart, K. Miller, S. Tan, and J. Warner. "Guidebook: Using Public Transportation to Facilitate Last Mile Package Delivery". In: *Texas A&M Transportation Institute* (Mar. 2017). URL: https://tti.tamu.edu/tti-publication/guidebook-using-public-transportation-to-facilitate-last-mile-package-delivery/.

[13] "El reparto de última milla se sube a la bici en el área metropolitana". In: *La Vanguardia* (2021). URL: https://www.lavanguardia.com/local/baix-llobregat/20210721/7614052/reparto-ultima-milla-sube-bici-area-metropolitana.html/ (visited on 09/21/2021).

[14] R. Elbert and J. Rentschler. "Freight on urban public transportation: A systematic literature review". In: *Research in Transportation Business & Management* (2021), p. 100679.

[15] EU Commission and others. "Green paper, towards a new culture for urban mobility". In: *European Union, Brussels* (2007).

[16] A. Galkina, T. Schlosserb, O. Galkinaa, D. Hodákováb, and S. Cápayováb. "Investigating using Urban Public Transport For Freight Deliveries". In: *Transportation Research Procedia* 39 (2019), pp. 64–73.

[17] V. Gatta, E. Marcucci, M. Nigro, and S. Serafini. "Sustainable urban freight transport adopting public transport-based crowdshipping for B2C deliveries". In: *European Transport Research Review* 11.1 (2019), pp. 1–14.

[18] V. Ghilas, E. Demir, and T. Van Woensel. "Integrating passenger and freight transportation: Model formulation and insights". In: *Proceedings of the 2013 Beta Working Papers; Technische Universiteit Eindhoven: Eindhoven, The Netherlands* 441 (2013), pp. 1–23.

[19] P. Gianessi, L. Alfandari, L. Létocart, and R. Wolfler Calvo. "The multicommodity-ring location routing problem". In: *Transportation Science* 50.2 (2016), pp. 541–558.

[20] "Handbook on External Costs of Transport (European Commission)". In: *CE Delft.* (2019). URL: https://www.cedelft.eu/en/publications/2311/handbook-on-the-external-costs-of-transport-version-2019/ (visited on 09/21/2021).

[21] IPC. "Online shopping frequency according to online shoppers worldwide as of October 2018". In: *Statista* (Jan. 2019). URL: https://www.statista.com/statistics/664770/online-shopping-frequency-worldwide/ (visited on 09/02/2021).

[22] Y. Ji, Y. Zheng, J. Zhao, Y. Shen, and Y. Du. "A Multimodal Passenger-and-Package Sharing Network for Urban Logistics". In: *Journal of Advanced Transportation* 2020 (2020).

[23] J. Kelly and M. Marinov. "Innovative interior designs for urban freight distribution using light rail systems". In: *Urban Rail Transit* 3.4 (2017), pp. 238–254.

[24] M. Marinov, F. Giubilei, M. Gerhardt, T. Özkan, E. Stergiou, M. Papadopol, and L. Cabecinha. "Urban freight movement by rail". In: *Journal of Transport Literature* 7 (2013), pp. 87–116.

[25] R. Masson, A. Trentini, F. Lehuédé, N. Malhéné, O. Péton, and H. Tlahig. "Optimization of a city logistics transportation system with mixed passengers and goods". In: *EURO Journal on Transportation and Logistics* 6.1 (2017), pp. 81–109.

[26] MHL. "Delivery Time Top Priority for Online Shoppers". In: *Material Handling and Logistics* (Sept. 2016). URL: https://www.mhlnews.com/transportation-distribution/article/22051729/delivery-time-top-priority-for-online-shoppers (visited on 09/02/2021).

[27] A. Mourad, J. Puchinger, and T. Van Woensel. "Integrating autonomous delivery service into a passenger transportation system". In: *International Journal of Production Research* 59.7 (2021), pp. 2116–2139.

[28] O. Ozturk and J. Patrick. "An optimization model for freight transport using urban rail transit". In: *European Journal of Operational Research* 267.3 (2018), pp. 1110–1121.

[29] J. Shen, F. Qiu, W. Li, and P. Feng. "A new urban logistics transport system based on a public transit service". In: *CICTP 2015*. 2015, pp. 650–661.

[30] Statista. "Projected market size of the autonomous last mile delivery worldwide from 2021 to 2030 (in billion U.S. dollars)". In: *Statista* (Oct. 2020). URL: https://www.statista.com/statistics/1103574/autonomous-last-mile-delivery-market-size-worldwide/ (visited on 09/02/2021).

[31] N. B. of Statistics of China. *National data*. 2020. URL: http://www.stats.gov.cn/.

[32] A. Trentini and N. Mahléné. "Toward a shared urban transport system ensuring passengers & goods cohabitation". In: *TeMA-Journal of Land Use, Mobility and Environment* 3.2 (2010).

[33] Vision Monday. "E-commerce share of total global retail sales from 2015 to 2024". In: *Statista* (Jan. 2021). URL: https://www.statista.com/statistics/534123/e-commerce-share-of-retail-sales-worldwide/ (visited on 09/02/2021).

[34] L. Zhao, H. Li, M. Li, Y. Sun, Q. Hu, S. Mao, J. Li, and J. Xue. "Location selection of intra-city distribution hubs in the metro-integrated logistics system". In: *Tunnelling and Underground Space Technology* 80 (2018), pp. 246–256.

[35] F. Zhou and J. Zhang. "Freight Transport Mode Based on Public Transport: Taking Parcel Delivery by Subway as an Example". In: *ICTE 2019*. American Society of Civil Engineers Reston, VA, 2020, pp. 745–754.

A Column generation strategies

In Section 4.2 we proposed 3 strategies to solve the pricing problem when applying column generation to solve the linear relaxation of PATH, namely, DIJKSTRA, FIRST and MIX. In this appendix we show the results of an exhaustive computational experimentation comparing their performance under different configuration values for parameters MCK and MC. We tested the 3 presented strategies on all instances in Group A (with a time limit of 300 seconds) with different configurations, setting MC $\in \{10, 100, 200, 500\}$, and MCK $\in \{1, 5, 10\}$. Figure 15 reports the results obtained by each configuration for each strategy. Specifically, we show the average running time to solve the linear relaxation of PATH (top chart), the average number of added columns (middle chart) and the average number of pricing steps (bottom chart). These averages are taken among all instances in Group A.

The results suggest that DIJKSTRA seems to be the fastest strategy, also reducing both the number of columns and iterations needed. The FIRST strategy obtains fairly similar running times but it usually demand more columns and iterations to finish. We note that the running times reported for the MIX strategy are significantly higher than the other two strategies. We recall that when there are not enough columns with a reduced cost greater than 0.5, the MIX strategy needs to check every possible column to select the best columns within the remaining ones. This explains why significantly more time is needed for the column generation of the MIX strategy, even though its number of added columns is comparable to the one of FIRST (in which no threshold is needed to use a positive reduce column). As we will see later, even though this property may sound disadvantageous, it brings a diversity of the columns that helps us to find feasible solutions of the higher quality.

Figure 15: Results obtained by the 3 proposed pricing strategies with different configurations.

As far as the setting of MC and MCK is concerned, it is clear that using MC = 10 does not yield the best results, but there seems to be just slight differences for values greater than 10. Regarding MCK, all strategies seem to benefit from setting MCK = 10, although the differences are not significant. Based on these results, we decide to choose the best (most stable) configuration for each of the three strategies, which seems to be to use MC = 500 and MCK = 10.