Multi-objective integration of timetables, vehicle schedules and user routings in a transit network

Gilbert Laporte [a], Francisco A. Ortega [b], Miguel A. Pozo [c, *], Justo Puerto [d]

[a] HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montreal, H3T 2A7, Canada
[b] Department of Applied Mathematics I, University of Seville, Spain
[c] Department of Statistics and Operational Research, University of Seville, Spain
[d] Department of Statistics and Operational Research, University of Seville, Spain

Keywords:
Timetabling
Vehicle scheduling Schedule delay
Location-allocation
Multi-objective optimization

A B S T R A C T

The Transit Network Timetabling and Scheduling Problem (TNTSP) aims at determining an optimal timetable for each line of a transit network by establishing departure and arrival times at each station and allocating a vehicle to each timetable. The current models for the planning of timetables and vehicle schedules use the *a priori* knowledge of users' routings. However, the actual route choice of a user depends on the timetable. This paper solves the TNTSP in a public transit network by integrating users' routings in the model. The proposed formulation guarantees that each user is allocated to the best possible timetable, while satisfying capacity constraints. In addition, we perform a trade-off analysis by means of a multi-objective formulation which jointly optimizes the operator's and the users' criteria.

1. Introduction

The Transit Network Timetabling and Scheduling Problem (TNTSP) aims at determining an optimal timetable for each line of a transit network by establishing departure and arrival times at each station and allocating a vehicle to each timetable (see e.g. Guihaire and Hao, 2008a). The input data consist of a public transportation network (PTN) made up of a set of stations and links between them, a set of lines, a fleet of capacitated vehicles, a fixed budget for line runs and an origin/destination (O/D) matrix. The output data consist of a set of arrival and departure times at the stations for the vehicles. Several criteria can be taken into consideration in these problems, for example waiting times, short transfers, fleet size, travel time, load factor, users utilization and, in general terms, users' and operator's costs. Traditionally, the Transit Network Timetabling Problem (TNTP, see e.g. Cacchiani and Toth, 2012) has been studied as a preliminary step for the Vehicle Scheduling Problem (VSP, see e.g. Bunte and Kliewer, 2009), i.e. the TNTP output is an input for the VSP. Unfortunately, this approach leads to suboptimal solutions for the TNTSP. In this paper we solve the TNTSP by integrating the TNTP and VSP.

In this paper we assume that users are interested in minimizing their *scheduled delay* (see, e.g., Small, 1982; de Palma and Lindsey, 2001; Mesa et al., 2014) that can be defined as the deviation between desired departure and arrival times and actual ones. When the travel time is constant, the actions of boarding late or arriving early do not incur any inconvenience themselves, but for many users boarding early involves some wasted time and arriving late typically has even more severe

* Corresponding author.
E-mail addresses: gilbert.laporte@cirrelt.ca (G. Laporte), riejos@us.es (F.A. Ortega), miguelpozo@us.es (M.A. Pozo), puerto@us.es (J. Puerto).

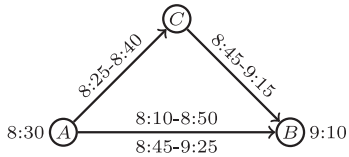

Fig. 1. Three strategies for traveling from A at 08:30 to B at 9:10.

$$\begin{array}{ccccc} \text{TNTP}^S & \subseteq & \text{TNTP}^O & \subseteq & \text{TNTP}^U \\ \cup\!| & & \cup\!| & & \cup\!| \\ \text{TNTSP}^S & \subseteq & \text{TNTSP}^O & \subseteq & \text{TNTSP}^U \end{array}$$

Fig. 2. Inclusion relationships between the solution spaces of the TNTSP subproblems when projecting the solution spaces of these problems over the space of the variables that describe the TNTP^U.

consequences. We define a *strategy* as a combination of an itinerary and a potential timetable that a user can choose to travel in a PTN. If we first assume uncapacitated vehicles, each user can freely choose the strategy that minimizes his scheduled delay. This means that a suboptimal shortest path in the PTN with a convenient timetable may lead to a user's scheduled delay lower than that corresponding to a shortest path with an inconvenient timetable. If we now include vehicle capacity constraints (CC), the PTN operator may have to assign specific itineraries to users in order to avoid overloading the vehicles. If this option is not available (as it happens in many bus or metro networks where users freely choose their routes) the PTN operator may have to design the timetables and allocate the vehicles in such a way that capacities are respected in the event where all users choose their optimal strategy. This gives rise to a leader-follower behavior: demand allocation depends on the timetables, and vice versa. The constraints we impose to guarantee these conditions are called optimal assignment constraints.

Example 1. Consider a user who aims to travel from station A to station B (Fig. 1) departing from A at 8:30 and reaching B at 9:10. There are two itineraries $A \to B$ and $A \to C \to B$, but if we consider the timetable this user may choose between three strategies: (1) depart from A at 8:10 to arrive at B at 8:50, (2) depart from A at 8:45 to arrive at B at 9:25, and (3) depart from A at 8:25 to arrive at C at 8:40 and from C at 8:45 to arrive at B at 9:15. In option 1, 20 min are wasted for boarding in advance while option 2 leads to 15 min of lateness. Option 3 generates five wasted minutes for boarding in advance and 5 min of lateness. If all these strategies are inconvenient for the user, he can choose an alternative mode of transportation.

We define several subproblems of the TNTSP depending on the types of constraints included. First, the vehicle scheduling constraints (VSC) are those that force the timetables to be operated with a fixed fleet size. The TNTSP without VSC reduces to the Transit Network Timetabling Problem (TNTP) which assumes unlimited fleet size and involves different problems according to how the CCs are included. If there are no CCs, the users will travel in the PTN by following their optimal strategy. This problem is called the User TNTP (TNTP^U). If there are vehicle capacities and the transit operator is able to route users in the network, the problem is called Operator TNTP (TNTP^O). However, it is also possible to include CCs and let users follow their optimal strategies. We call this problem the System TNTP (TNTP^S). Including the VSC in the previous subproblems, we obtain respectively the User TNTSP (TNTSP^U), the Operator TNTSP (TNTSP^O) and the System TNTSP (TNTSP^S). If we project the solution spaces of these problems over the space of variables that describe the TNTP^U, the sets of feasible solutions will be related as illustrated in Fig. 2.

In this paper, we study the TNTSP and its variants as described above. After reviewing our main contributions to the TNTSP, we define the different subproblems and, in particular, we describe how to compute the available strategies for a transportation request in a PTN. Timetables and vehicle schedules are usually computed assuming the knowledge of users' routings from the results of a previous phase. However, the actual route a user will take depends on the timetable which is not yet known a priori. In this paper we integrate user route choices within the TNTSP which, as far as we are aware, is a new scientific contribution. This new solution framework is flexible and yields an optimal allocation of transportation requests under vehicle capacity constraints. We compare the six TNTSP subproblems previously defined and their solution spaces. We generate a testbed of randomly generated instances over several network configurations and we report computational results. In addition, we perform a trade-off analysis by means of a multi-objective formulation that jointly optimizes the users' and operator's criteria.

The remainder of this paper is structured as follows. Section 2 reviews the most relevant contributions related to this paper. Section 3, provides the description of the problem and the information required to compute itineraries and strategies in a transit network made up of several lines. Section 4 presents the mathematical programming model for the integrated

TNTSP and for the subproblems just described. Computational results are presented in Section 6, followed by conclusions in Section 7.

2. Literature review

Our literature review focuses on contributions that integrate the timetabling and the vehicle scheduling problems. For reviews on the TNTP we refer the reader to Cacchiani and Toth (2012); Caprara et al. (2007, 2011), and Lusby et al. (2011). For reviews on the VSP see Törnquist (2007) and Bunte and Kliewer (2009).

Recent research in transportation planning has focused on the benefits that can be derived through the integration of different stages belonging to the transit planning process, known as network design, line planning, frequency setting, timetabling, vehicle scheduling and crew rostering (Guihaire and Hao, 2008a). However, not much research has been developed on the integration of timetabling and vehicle scheduling (Ibarra-Rojas et al., 2014). Since the VSP is easy to solve, it is straightforward to implement an iterative procedure that modifies the current timetable and then solves the VSP. If a complete integration is desired, the departure times of trips become decision variables. In this case, network flow formulations and algorithms for the VSP are difficult to implement since the model lacks a fixed network. Instead, it deals with a set of potential networks that depend on timetabling decisions.

Solution approaches for integrating the two subproblems of the planning process can be divided into two types: sequential integration and complete integration. Sequential integration considers the characteristics of a subproblem while the other one is optimized, which may lead to a suboptimal solution. We refer to Ceder (2001), van den Heuvel et al. (2008), Guihaire and Hao (2008b), Ceder (2011) and Petersen et al. (2013), for partial integration methodologies for the TNTSP. In contrast, a complete integration defines a formulation or a solution methodology to determine the decisions of two or more subproblems simultaneously.

Complete integration schemes for the TNTP with the VSP have been considered in only a few papers. As far as we know, the first study on a timetabling problem that includes the number of required vehicles in the objective function is that of Chakroborty et al. (2001). Here, the transit network is made up of multiple lines with a single transfer node. Castelli et al. (2004) deal with non-periodic timetables assuming that routes, means of transport and quality of service are fixed in advance. The operator's main objective is to minimize cost while serving as many users as possible. These authors integrate constraints on the number of available vehicles in the transit network timetabling problem. Liu and Shen (2007) have proposed a bilevel optimization problem consisting of a hierarchical formulation in which the upper level solves the VSP with the objective of minimizing the fleet size and the deadheading time, while the lower level optimizes the TNTP in order to minimize the total transfer time for users. Fleurent and Lessard (2009) established a measure function for the TNTP that incorporates key elements of synchronization, such as the number of users transferring from one line to another and the related waiting time. These authors include vehicle scheduling decisions and other measures, such as vehicle usage costs. Guihaire and Hao (2010) integrate the TNTP and VSP through an optimization model with a weighted objective function that considers the quantity and quality of transfers, the evenness of headway times, the fleet size, and the length of the deadheads. They assume the existence of an initial timetable and feasible time intervals for departures and arrivals. This information serves to design line and trip shift movements that are used to modify the feasible timetable and then find the optimal vehicle schedule.

More recently, Ibarra-Rojas et al. (2014) proposed a bi-objective optimization problem to jointly solve the single depot VSP and the bus TNTP by considering time windows for departure times and assuming constant demand. The objectives are the maximization of the number of users who benefit from well-timed transfers, and the minimization of the fleet size. The authors implemented an ϵ-constraint algorithm to obtain Pareto-optimal solutions. Numerical results show that in some instances using one more vehicle leads to significant reductions in the number of user transfers.

User-oriented optimization of public transport requires data about the users in order to develop realistic models. Current models take user data into account by using the following two-phase approach. First, user routes are determined. Second, the actual planning of timetables takes place using the knowledge of which routes users wish to travel given the results of the first phase. However, the actual route a user will take strongly depends on the timetable, which is not yet known in the first phase (Borndörfer et al., 2015). Hence, the two-phase approach often yields suboptimal solutions. Almost all available models assume that user routes are fixed before the design of a timetable starts but this topic has recently received more attention (see Siebert and Goerigk, 2013; Schmidt and Schöbel, 2015b; 2015a; Goerigk and Schmidt, 2016 and references therein).

Combining information on timetables, vehicle scheduling and user choices, Mesa et al. (2014) presented an integrated approach for jointly planning timetables and vehicle schedules along a single transit line while emphasizing the perspective of potential users. Their setting assumes a model of fully disaggregated demand for a scenario that includes capacity constraints and demand behavior according to different criteria. These authors propose a p-median-based formulation which includes specific constraints for the VSP. Demand behavior is handled through the inclusion of closest assignment type constraints. The authors developed a clustering based algorithm in order to provide an alternative methodology for solving instances of the problem when computational time is limited. Their computational experiments show the difficulty of including closest assignment constraints in a transportation problem and the advantages of deriving a clustering algorithm that allows an appropriate preprocessing and handling of the data.

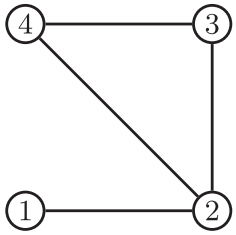

Fig. 3. A PTN with four nodes (stations) and eight arcs.

Our paper differs significantly from previous research contributions. First, we develop a framework for integrating the TNTP, the VSP and user routings. As far as we know, the integration of these three problems has never previously been studied. Our approach not only pursues transfer coordination but also users' preferences in terms of departure and arrival times for a fully disaggregated demand. Moreover, each transportation request is treated individually, considering hard time windows constraints for trip duration, departure and arrival times, as well as inconvenience costs related to deviations from these times. Second, we formulate the TNTSP starting from the TNTP and adding constraints regarding capacities, optimal user assignment and fleet size. We test and compare these formulations on a testbed of random instances and on different networks as similarly proposed in previous studies. In addition, we study the multi-objective version of the problem in order to analyze the trade-off between user and operator criteria. We use an ϵ-constraint method to obtain the exact Pareto front of solutions between users' and operator's costs.

3. Problem description and formulation

We now formally define and formulate the problem. The reader is referred to the Appendix for a full list of the notation used.

3.1. Infrastructure

We first distinguish between the infrastructure network and the network containing all lines and walking corridors for transferring between different lines. A *public transportation network* (PTN) is a graph $G = (S, A)$ with a set of nodes S representing stations and a set of arcs A, where each arc represents a direct connection between two stations of S. Given a PTN, $G = (S, A)$, a *public transportation line* (PTL) is a connected directed graph $G_l = (S_l, A_l)$ where index l belongs to the set of lines $\mathcal{L}$. The set of nodes $S_l \subseteq S$ represents stations and a set of arcs $A_l \subseteq A$ contains the direct connections between two stations using line l. We distinguish between two types of lines. The set of directed path lines $\overline{\mathcal{L}} \subseteq \mathcal{L}$ and a set of directed cycle lines $\mathring{\mathcal{L}} \subseteq \mathcal{L}$. Moreover, the set of lines $\mathcal{L}$ can be split into the set of lines going forward, $\overrightarrow{\mathcal{L}} \subseteq \mathcal{L}$ and the set of lines going backward, $\overleftarrow{\mathcal{L}} \subseteq \mathcal{L}$. Therefore, it is obvious that $\mathcal{L} = \overline{\mathcal{L}} \cup \mathring{\mathcal{L}} = \overrightarrow{\mathcal{L}} \cup \overleftarrow{\mathcal{L}}$. In addition, each line $l \in \overrightarrow{\mathcal{L}}$ has a corresponding line $l' \in \overleftarrow{\mathcal{L}}$ running in opposite direction and is identified by means of $l' = l + |\overrightarrow{\mathcal{L}}|$, where $|\cdot|$ stands for the cardinal of a set. A *terminal station* is a station at which line runs can start and finish, so it can be stated that path lines have two terminal stations (at both ends of the line) and cyclic lines have only one terminal station.

Example 2. Fig. 3 shows a PTN of 4 nodes $S = \{1, 2, 3, 4\}$ and 8 arcs $A = \{(1, 2), (2, 1), (2, 4), (2, 3), (3, 2), (3, 4), (4, 2), (4, 3)\}$.

In the graph of Fig. 3 we define lines $\mathcal{L} = \{1, 2, 3, 4\}$ where $A_1 = \{(1, 2), (2, 3)\}, A_2 = \{(2, 3), (3, 4), (4, 2)\}, A_3 = \{(3, 2), (2, 1)\}, A_4 = \{(2, 4), (4, 3), (3, 2)\}$. In addition we can define sets $\overline{\mathcal{L}} = \{1, 3\}$, $\mathring{\mathcal{L}} = \{2, 4\}$, $\overrightarrow{\mathcal{L}} = \{1, 2\}$, $\overleftarrow{\mathcal{L}} = \{3, 4\}$. □

Given a PTN $G = (S, A)$, the associated *Change&Go Network* (CGN) is a graph $\mathcal{G}$ defined in order to include transfer activities between lines (see e.g. Schöbel and Scholl, 2006 Goerigk and Schmidt, 2016). We define here a slightly different version of this concept adapted to our problem. It can be denoted as $\mathcal{G} = (\mathcal{N} \cup \ddot{\mathcal{N}}, \mathcal{A} \cup \mathcal{A}^{(tra)})$, where $\mathcal{N} = \{(l, i) : l \in \mathcal{L}, i \in S_l\}$ is the set of nodes of each line, $\ddot{\mathcal{N}} = \{|S| + 1, \ldots, |S| + |\ddot{\mathcal{N}}|\}$ is the set of transfer nodes between lines, $\mathcal{A} = \{(l, i, j) : l \in \mathcal{L}, (i, j) \in A_l\}$ is the set of arcs of all lines and $\mathcal{A}^{(tra)} = \{(l, i, n) : l \in \mathcal{L}, i \in S_l, n \in \ddot{\mathcal{N}}\}$ is the set of transfer edges between lines. Basically, the CGN replicates each node of the PTN once for each line and analogously with the arcs. In addition, transfer nodes are added to the CGN as well as transfer edges for each line.

Example 3. Given the PTN and the linepool defined in Example 2, the associated CGN is depicted in Fig. 4 with corresponding line nodes $\mathcal{N} = \{(1, 1), (3, 1), (1, 2), (2, 2), (3, 2), (4, 2), (1, 3), (2, 3), (3, 3), (4, 3), (2, 4), (4, 4)\}$, transfer nodes $\ddot{\mathcal{N}} = \{5, 6\}$, line arcs $\mathcal{A} = \{(1, 1, 2), (1, 2, 3), (2, 2, 3), (2, 3, 4), (2, 4, 2), (3, 3, 2), (3, 2, 1), (4, 2, 4), (4, 4, 3), (4, 3, 2)\}$ and transfer arcs $\mathcal{A}^{(tra)} = \{(1, 2, 5), (1, 5, 2), (2, 2, 5), (2, 5, 2), \ldots, (4, 3, 6), (4, 6, 3)\}$.

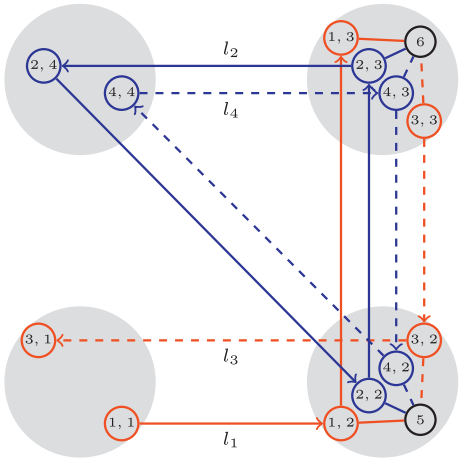

Fig. 4. Change&Go Network associated to the network of Fig. 3.

3.2. Timetables and vehicle schedules

Given a transit line $l \in \mathcal{L}$, a *line run* is the journey of a vehicle making stops for boardings and alightings at every station along the line. We assume a discretization T of the continuous time interval in order to assign departures of the κ available vehicles to the set of time slots of T. We assume that all vehicles will have the same speed and stopping time at every (non-terminal) station along l and, therefore, a fixed travel time τ_l will be required in order to complete a line run on l (including intermediate stops). We denote by c_l the cost of implementing a line run in l and we allow a maximum cost ρ to implement all line runs. We also assume that vehicles have the same capacity Q.

The reader may note that there are two main assumptions throughout the paper that refer to identical vehicles, fixed running times and fixed stopping times at stations. These assumptions can be relaxed in order to model more realistic timetables (see e.g. Mesa et al., 2013) in exchange of increasing considerably the modeling size. However, since we are integrating several problems in this paper it makes sense to simplify these decisions.

A vehicle allocated to $l \in \overrightarrow{\mathcal{L}}$ can start a line run at any time slot $t \in T$. Once the line run is completed, the associated vehicle becomes part of the fleet size of the line $l + |\overrightarrow{\mathcal{L}}|$ ($l - |\overrightarrow{\mathcal{L}}|$ if $l \in \overleftarrow{\mathcal{L}}$), and a new line run with this vehicle can be started in any time slot $t' \in T$ such that $t' \geq t + \tau_l$. Circular lines, are similar to path lines except that they have only one terminal station. Therefore, any itinerary that involves traversing the terminal station will require a transfer at that station.

Example 4 (cont.)**.** In the graph of Fig. 4, once a vehicle of line 1 reaches station 3, it changes its direction and becomes a vehicle of line 3. The same occurs for lines 2 and 4 once a terminal station is fixed (for example at station 2). This means that once a vehicle of line 2 reaches station 2, it becomes a vehicle of line 4 (it cannot be used in a line run of line 2 at that moment). Thus, if a user travels from station 4 to station 3 using line 2, a transfer will be required at station 2 toward line 2 (again) or line 1.

3.3. Demand

Each user has fixed upper and lower bounds associated to the departure and arrival times. Additionally other inconveniences related to in-vehicle times, line-change penalties and deviation between desired departure and arrival times will be taken into account. The concept of *schedule delay*, introduced by Small (1982), arises with the fact that arriving early is likely to involve some time wasted while for most users, arriving late has more severe repercussions. Let I be the set of user transportation requests. In what follows, we use the terms user and request indiscriminately. Each request $i \in I$ involves an origin and a destination station and preferred departure times t_i and $t_{i+|I|}$ to start and finish a trip between the origin and destination of i. Furthermore, t_i^- denotes the earliest time at which user i can start the trip and $t_{i+|I|}^+$ denotes the latest time at which i can reach his destination. Note that these parameters are dependent on the travel time estimated by the user considering the structure of the PTN (see Example 5). In addition we can impose a penalty of one unit if a request is not served. Next, we will introduce in the following section an inconvenience parameter to measure the service quality perceived by each user.

3.4. User strategies

Given a CGN, each itinerary offers different travel *options* according to each combination of the potential timetables from the different lines that can be used for completing a trip. Let Π be the set of all itineraries and $\Pi_i \subseteq \Pi$ the subset of

itineraries that can be used by request i. Note that each itinerary $\pi \in \Pi$ is related with a set of lines $\mathcal{L}_\pi$ and transfer nodes used to complete a trip. Once his path is defined, a user can consider different options of departure times, depending on the combinations of timetables that can be implemented on each line of the path. By $R_{i\pi}$ we denote the set of options that can be used to serve request i by means of itinerary π. Given this notation, we can define an inconvenience cost function parameter $\varphi_{i\pi r}$ which computes the cost of allocating request i to itinerary $\pi \in \Pi$ and option $r \in \mathcal{R}_{i\pi}$. Additionally, in order to keep track of the capacity usage, we define a binary parameter $m_{\pi a}$ equal to one if and only if arc $a \in A$ is used along itinerary π. In addition, we denote by $t_{i\pi rl}$ the departure time slot used for a vehicle serving request i on line l when itinerary π and option r are used. Note that, in order to simplify the notation, we will consider strategies involving at most one transfer. This makes sense given the number of transfers that users are usually willing to perform in practice (Stern, 1996). However, the proposed setting allows the implementation of several transfers in a general CGN.

Example 5. Users can travel inside the network of Fig. 3 following different strategies. As an example, a user ($i = 1$) going from $s = 4$ to $s = 3$ can travel choosing one of the following itineraries:

- itinerary 1, involving a trip from node (4, 4) to node (4, 3) (denoted by (4, 4) → (4, 3) in the following),
- itinerary 2, involving two trips (4, 4) → (4, 2), (4, 2) → (4, 3) and a intermediate transfer (at station 2),
- itinerary 3, involving two trips (4, 4) → (4, 2), (1, 2) → (1, 3) and a intermediate transfer (at station 2).

For simplicity, we assume that the distances between adjacent stations of the PTN are all equal to 1. This way a user ($i = 2$) traveling from station 1 to station 3 can only choose the itinerary (1, 1) → (1, 2) → (1, 3) since in this case it does not make sense to transfer at station 2 to another line. The different options for user $i = 2$ and his unique available itinerary are (assuming $t_2 = 9$, $t_{2+|I|} = 11$, $t_2^- = 5$ and $t_{2+|I|}^+ = 15$) starting his trip at any of the time slots {5, 6, 7, 8, 9, 10, 11, 12, 13}. The different options for user $i = 1$ and itinerary 2, are (assuming $t_1 = 9$, $t_{1+|I|} = 10$, $t_1^- = 5$ and $t_{1+|I|}^+ = 14$) the following:

- starting his trip at time slot 5 and transferring at any of the time slots {6, 7, 8, 9, 10, 11, 12, 13},
- starting his trip at time slot 6 and transferring at any of the time slots {7, 8, 9, 10, 11, 12, 13},
- starting his trip at time slot 7 and transferring at any of the time slots {8, 9, 10, 11, 12, 13},
- starting his trip at time slot 8 and transferring at any of the time slots {9, 10, 11, 12, 13},
- starting his trip at time slot 9 and transferring at any of the time slots {10, 11, 12, 13},
- starting his trip at time slot 10 and transferring at any of the time slots {11, 12, 13},
- starting his trip at time slot 11 and transferring at any of the time slots {12, 13},
- starting his trip at time slot 12 and transferring at time slot 13.

The scheduled delay costs will depend on the selected combination itinerary-option. See Section 6 for an example of the inconvenience cost function $\varphi_{i\pi r}$. □

4. Mathematical programming model

We now define the six models described in Section 2. We make decisions concerning the number of line runs located on each line (ρ_l) and on the number of vehicles initially available on each line (κ_l). In addition we make binary decisions by means of variable x_{lt} equal to one if and only if a line run is located on line l at time slot t, and by means of variable $y_{i\pi r}$ equal to one if and only if user i is allocated to itinerary $\pi \in \Pi_i$ and option $r \in \mathcal{R}_{i\pi}$.

We first consider the problem of establishing the departure times from terminal stations of the lines. Each departure performs a line run from line l that has an associated cost c_l. The formulation is as follows:

$$\text{TNTP}^U : \text{minimize} \sum_{i \in I} [\sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} \varphi_{i\pi r} y_{i\pi r} + (1 - \sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} y_{i\pi r})] \tag{1a}$$

$$\text{subject to} \sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} y_{i\pi r} \leq 1 \qquad i \in I \tag{1b}$$

$$\sum_{t \in T} x_{lt} \leq \rho_l \qquad l \in \mathcal{L} \tag{1c}$$

$$|\mathcal{L}_\pi| y_{i\pi r} \leq \sum_{l \in \mathcal{L}_\pi} x_{lt_{i\pi rl}} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi} \tag{1d}$$

$$\sum_{l \in \mathcal{L}} c_l \rho_l \leq \rho \tag{1e}$$

$$x_{lt} \in \{0, 1\} \qquad l \in \mathcal{L}, t \in T \tag{1f}$$

$$y_{i\pi r} \in \{0,1\} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi} \tag{1g}$$

$$\rho_l \in \mathbb{Z}^+ \qquad l \in \mathcal{L}. \tag{1h}$$

The objective function (1a) minimizes the total user inconvenience. The first term computes the inconvenience of the users who use the system whereas the second term computes the penalty cost of those users not assigned to any service of the system. Constraints (1b) ensure that no more than one strategy is selected for user i, thus avoiding negative terms in the second part of the objective function. Constraints (1c) ensure that no more than ρ_l line runs are located on each line $l \in \mathcal{L}$. Constraints (1d) ensure that no request will be allocated to a strategy that cannot be carried out with the available line runs. Constraint (1e) ensures that the total cost incurred by the line runs is not exceeded.

We recall that TNTP^U does not take into account the fleet size or the capacity of the vehicles. Constraints (2) ensure that the number of requests that use an arc of the CGN in a time slot do not exceed Q.

$$\sum_{i\in I} \sum_{\pi\in\Pi_i : l\in\mathcal{L}_\pi} \sum_{r\in\mathcal{R}_{i\pi} : t_{i\pi rl}=t} y_{i\pi r} m_{\pi a} \le Q x_{lt} \quad l \in \mathcal{L}, a \in A_l, t \in T. \tag{2}$$

The TNTP^U together with constraints (2) defines the TNTP^O.

Assuming that users know in advance all information related to the itineraries and timetables, it is reasonable to assume that they will choose a strategy that enables them to reach their destination at minimum inconvenience cost. In addition, as previously mentioned, the network operator may not be able to route transportation requests within the CGN. Therefore, in order to model the TNTP^S, we add two additional constraints. First, we impose that if there exists a given strategy (an option within an itinerary) for a given user, then that user must be allocated to one strategy:

$$\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} \le (|\mathcal{L}_\pi| - 1) + \sum_{\pi'\in\Pi_i} \sum_{r'\in\mathcal{R}_{i\pi}} y_{i\pi' r'} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi}. \tag{3}$$

Second, constraints (4) ensure that if there exists a given strategy for a given user, then that user cannot be allocated to another strategy with higher inconvenience cost:

$$\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} + \sum_{\pi'\in\Pi_i} \sum_{\substack{r'\in\mathcal{R}_{i\pi}: \\ \varphi_{i\pi' r'} > \varphi_{i\pi r}}} y_{i\pi' r'} \le |\mathcal{L}_\pi| \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi}. \tag{4}$$

Constraints (3) are activated (a user is allocated to any strategy) when $\sum_{\pi'\in\Pi_i} \sum_{r'\in\mathcal{R}_{i\pi}} y_{i\pi' r'} = 1$. Therefore, (3) are activated according to the values of the number of lines in the chosen itinerary ($|\mathcal{L}_\pi|$) and the number of line runs that are active for the chosen option r of itinerary π, namely $\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}}$, that is one at most if $|\mathcal{L}_\pi| = 1$ and 2 at most if $|\mathcal{L}_\pi| = 2$.

- If $|\mathcal{L}_\pi| = 2$, (3) is activated if and only if $\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} = 2$ (both required line runs are active).
- If $|\mathcal{L}_\pi| = 1$, (3) is activated if and only if $\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} = 1$ (the required line run is active).

Similarly, constraints (4) are activated (a user is not allocated to a strategy with a higher inconvenience cost) when $(\sum_{\pi'\in\Pi_i} \sum_{\substack{r'\in\mathcal{R}_{i\pi}: \\ \varphi_{i\pi' r'} > \varphi_{i\pi r}}} y_{i\pi' r'} = 0)$.

- If $|\mathcal{L}_\pi| = 2$, (4) is activated if and only if $\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} = 2$ (both required line runs are active).
- If $|\mathcal{L}_\pi| = 1$, (4) is activated if and only if $\sum_{l\in\mathcal{L}_\pi} x_{lt_{i\pi rl}} = 1$ (the required line run is active).

Constraints (4) are adapted from Wagner and Falkson (1975) and belong to the so-called class of closest assignment constraints. According to Espejo et al. (2012), such constraints can be modeled in many different ways, giving rise to better or worse linear programming relaxations. As shown in that paper, the constraints provided by Wagner and Falkson (1975) can be strengthened by using the fixed numbers of facilities (line runs) to locate. Even with such improvements, the inclusion of constraints (3) and (4) considerably increases the computational complexity of the proposed model.

In addition to the previous considerations we can assume that the timetables are carried out by a limited number κ of vehicles. On each line l, vehicles change the direction of the line once the terminal station is reached and then become a vehicle available for line $l + |\overrightarrow{\mathcal{L}}|$ ($l - |\overrightarrow{\mathcal{L}}|$ if $l \in \overleftarrow{\mathcal{L}}$). We describe the vehicle scheduling problem through the following sets of constraints:

$$\sum_{t'=1}^{\tau_l} x_{lt'} \le \kappa_l \qquad l \in \mathcal{L} \tag{5a}$$

$$\sum_{t'=1}^{t} x_{lt'} - \sum_{t'=1}^{t-\tau_l} x_{l+|\overrightarrow{\mathcal{L}}|,t'} \le \kappa_l \qquad l \in \overrightarrow{\mathcal{L}}, t \in T : t > \tau_l \wedge t < |T| - \tau_l \tag{5b}$$

$$\sum_{t'=1}^{t} x_{lt'} - \sum_{t'=1}^{t-\tau_l} x_{l-|\overrightarrow{\mathcal{L}}|,t'} \leq \kappa_l \qquad l \in \overleftarrow{\mathcal{L}}, t \in T : t > \tau_l \wedge t < |T| - \tau_l \tag{5c}$$

$$\sum_{l \in \mathcal{L}} \kappa_l \leq \kappa \tag{5d}$$

$$\kappa_l \in \mathbb{Z}^+ \qquad l \in \mathcal{L}. \tag{5e}$$

Constraints (5a) ensure that no more than κ_l vehicles (the fleet size of line l) depart from line l within the time interval $[1, \tau_l]$ (that is, before any vehicle from the opposite direction could have arrived). We recall that τ_l is the time required to complete a line run along line l. After time τ_l, constraints (5b) ensure that the difference between the number of vehicles that have departed from line l (going forward) and the number of vehicles that have arrived at l (coming from the corresponding line in opposite direction) never exceeds the fleet size of l. Constraints (5c) impose, for each line going backwards, the same as (5b). Constraint (5d) ensures that the total fleet size is not exceeded.

The TNTP^U, TNTP^O, TNTP^S together with constraints (5a)–(5d) are defined as the TNTSP^U, TNTSP^O, TNTSP^S respectively.

4.1. The multi-objective TNTSP

There are two criteria considered by the previous problems, the level of service (captured by the timetabling) and the operating costs (in terms of vehicle usage). A high level of service guarantees users' benefit, but cost savings are necessary to sustain a profitable system. As mentioned in Ibarra-Rojas et al. (2014), there is a lack of cost-benefit analyses for the daily operation of a transport system. Since the level of service and the operating costs are naturally in conflict, our motivation is to propose a methodology that will take into account the trade-off between these criteria by jointly optimizing the timetabling and vehicle scheduling decisions. To this end, we propose a multi-objective optimization model and algorithm for the TNTSP and its variants described in Section 1. This will allow us to answer questions such as: How to improve the service at minimum cost? How many users are favoured by introducing one more line-run or one more vehicle to the fleet? That is, we develop a tool to quantify how much the level of service increases (or decreases) per line-run or vehicle.

The multi-objective function (6a) minimizes simultaneously the users' inconvenience, the line runs costs and the fleet size costs. The domain defined by (6b)–(6n) includes the TNTSP domain defined by (1c)–(1g), (2)–(4) and (5a)–(5d). The multi-objective problem is then:

$$\text{minimize } (z_1, z_2, z_3) \equiv \text{minimize} \left(\sum_{i \in I} \left[\sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} \varphi_{i\pi r} y_{i\pi r} + \left(1 - \sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} y_{i\pi r} \right) \right], \sum_{l \in \mathcal{L}} \sum_{t \in T} x_{lt} c_l, \sum_{l \in \mathcal{L}} \kappa_l \right) \tag{6a}$$

$$\text{subject to} \sum_{\pi \in \Pi_i} \sum_{r \in \mathcal{R}_{i\pi}} y_{i\pi r} \leq 1 \qquad i \in I \tag{6b}$$

$$\sum_{t \in T} x_{lt} \leq \rho_l \qquad l \in \mathcal{L} \tag{6c}$$

$$|\mathcal{L}_\pi| y_{i\pi r} \leq \sum_{l \in \mathcal{L}_\pi} x_{lt_{i\pi rl}} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi} \tag{6d}$$

$$\sum_{l \in \mathcal{L}} c_l \rho_l \leq \rho \tag{6e}$$

$$\sum_{i \in I} \sum_{\pi \in \Pi_i : l \in \mathcal{L}_\pi} \sum_{r \in \mathcal{R}_{i\pi} : t_{i\pi rl} = t} y_{i\pi r} m_{\pi a} \leq Q x_{lt} \qquad l \in \mathcal{L}, a \in A_l, t \in T \tag{6f}$$

$$\sum_{l \in \mathcal{L}_\pi} x_{lt_{i\pi rl}} \leq (|\mathcal{L}_\pi| - 1) + \sum_{\pi' \in \Pi_i} \sum_{r' \in \mathcal{R}_{i\pi}} y_{i\pi' r'} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi} \tag{6g}$$

$$\sum_{l \in \mathcal{L}_\pi} x_{lt_{i\pi rl}} + \sum_{\pi' \in \Pi_i} \sum_{\substack{r' \in \mathcal{R}_{i\pi} : \\ \varphi_{i\pi' r'} > \varphi_{i\pi r}}} y_{i\pi' r'} \leq |\mathcal{L}_\pi| \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi} \tag{6h}$$

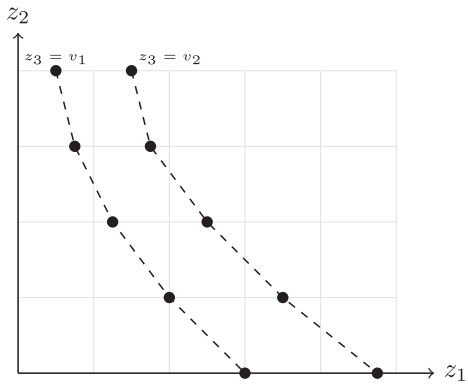

Fig. 5. Level curves for a set of non-dominated points ($\mathcal{P}$), objectives z_1, z_2, z_3, and values for z_3 equal to v_1 and v_2.

$$\sum_{t'=1}^{\tau_l} x_{lt'} \leq \kappa_l \qquad l \in \mathcal{L} \tag{6i}$$

$$\sum_{t'=1}^{t} x_{lt'} - \sum_{t'=1}^{t-\tau_l} x_{l+|\overrightarrow{\mathcal{L}}|,t'} \leq \kappa_l \qquad l \in \overrightarrow{\mathcal{L}}, t \in T : t > \tau_l \wedge t < |T| - \tau_l \tag{6j}$$

$$\sum_{t'=1}^{t} x_{lt'} - \sum_{t'=1}^{t-\tau_l} x_{l-|\overrightarrow{\mathcal{L}}|,t'} \leq \kappa_l \qquad l \in \overleftarrow{\mathcal{L}}, t \in T : t > \tau_l \wedge t < |T| - \tau_l \tag{6k}$$

$$\sum_{l \in \mathcal{L}} \kappa_l \leq \kappa \tag{6l}$$

$$x_{lt} \in \{0, 1\} \qquad l \in \mathcal{L}, t \in T \tag{6m}$$

$$y_{i\pi r} \in \{0, 1\} \qquad i \in I, \pi \in \Pi_i, r \in \mathcal{R}_{i\pi}. \tag{6n}$$

$$\rho_l \in \mathbb{Z}^+ \qquad l \in \mathcal{L} \tag{6o}$$

$$\kappa_l \in \mathbb{Z}^+ \qquad l \in \mathcal{L}. \tag{6p}$$

The multi-objective methodology presented here is not only useful to determine incomparable solutions based on the users' and the operator's costs, but also to avoid solving several problems sequentially in order to guarantee optimal solutions. Our multi-objective optimization problem is solved by an ϵ-constraint method (Chankong and Haimes, 1983) which yields exact trade-off solutions between users' inconvenience and operator's costs. In this manner, we are able to study the compromises that can be made between the three criteria used in the planning process by computing exact Pareto optimal fronts.

5. An ϵ-constraint solution approach for the multi-objective TNTSP

Our multi-objective algorithm for the TNTSP aims at jointly minimizing users' inconvenience (in terms of coverage and schedule delays) as well as the operator's costs (in terms of number of vehicles and budget for line-runs). We are interested in computing Pareto optimal solutions, i.e., an undominated set of solutions with respect to the values of the different objective functions. Since we are minimizing three different objectives, a special solution representation is required to avoid plotting of a 3-dimensional non-dominated set of solutions ($\mathcal{P}$), which would be difficult to analyze by the decision maker. To this end, we project the solutions onto the $z_1 \times z_2$ plane, where z_1 is the users' inconvenience and z_2 is the line runs cost (Fig. 5). In this plane the level curves plot points with the same z_3 value (fleet size). We identify these curves by their

Algorithm 1: Non-dominated points set $\mathcal{P}$.

```
  Select values ρ^min, ρ^max, κ^min, κ^max, ε_1, ε_2;                    // select input parameters
1 ρ := ρ^max; κ := κ^max; P = {∅};                                        // reset values
2 while κ ≥ κ^min do                                                      // while budget κ is above its lower bound...
3     while ρ ≥ ρ^min do                                                  // while budget ρ is above its lower bound...
4         solve: (z*_1, z*_2, z*_3) := arg lex min(z_1, z_2, z_3) s.t. (6p)-(6p);   // compute a new non-dominated point
5         P := P ∪ {(z*_1, z*_2, z*_3)};                                  // Add the new point to the set on non-dominated points
6         ρ := z*_2 − ε_1;                                                // decrease ρ
7     κ := κ − ε_2;                                                       // set κ for obtaining a new level curve
8     ρ := ρ^max;                                                         // reset ρ
```

position (e.g. first, second,...) from left to right, or by their z_3 values. This kind of graphical representation is very informative since it shows the demand improvement obtained by increasing either the number of vehicles or the budget for line-runs.

As stated by Kim and de Weck (2005), solving optimization problems with weighted objective functions often produces poorly distributed solutions along a Pareto front and does not yield Pareto optimal solutions in non-convex regions. To overcome these inconveniences, one may resort to adaptative weighted sum methods and additional inequality constraints. Here we propose an ϵ-constraint method as in Ibarra-Rojas et al. (2014), extended to the case of three objectives and with other features as described in the following.

The steps of our ϵ-constraint method for the multi-objective TNTSP are given in Algorithm 1 which presents the basic idea for obtaining the different Pareto fronts (level curves). We first provide lower and upper bounds for parameters ρ and κ (denoted by ρ^{min}, ρ^{max}, κ^{min} and κ^{max} respectively). The Pareto set $\mathcal{P}$ is initialized as empty. We start by computing non-dominated points from the highest value of κ to its lowest value, and for each value of κ we compute non-dominated points from the highest value of ρ to its lowest value. Step 5 yields a non-dominated point at each iteration and Step 6 adds it to the Pareto front. Step 7 changes the values of ρ, to generate a new non-dominated point in the level curve, and Steps 8 and 9 change the values of κ and ρ to obtain a new non-dominated point in the next level curve.

6. Computational experiments

Next, we report on the results of some computational experiments we have run in order to empirically compare the proposed TNTSP formulation and its variants. We have first studied the multi-objective TNTSP and we have chosen the TNTSP^U for presenting the level curves of the Pareto fronts for different networks. Second, we analyze the computational performance of the TNTSP and its variants for specific values of ρ and κ.

6.1. Data generation

In order to illustrate the applicability of the previous models, we have considered six different networks (Fig. 6) inspired from Laporte et al. (1994) and Laporte et al. (1997). For the sake of reproducibility, we assume topological structures, so no geographical configurations underlie and travel times between nodes are all equal to 1. All networks contain 13 nodes and have similar numbers of edges (ranging from 12 to 16). Each configuration can be obtained by setting out three lines, obtaining from one to five intersections. Note that each line, represented by continuous dashed and dotted lines, contains two path lines, except for the cartwheel configuration where the dotted line contains two circular lines. First, for each O/D pair we have precomputed the different itineraries by using a k-shortest path algorithm (Shier, 1979). We have then generated 10 random instances of transportation requests for $|I| = 100$ with an origin, a destination and a desired arrival time $t_{i+|I|} \in T$, with $|T| = 60$ (all random values following a discrete uniform distribution). Next, for each configuration, we have computed the desired departure time t_i of each request by means of the corresponding shortest path through the PTN. Each desired departure time t_i lies within a time window $[\max\{0, t_i - 4\}, \min\{|T|, t_i + 4\}]$, or equivalently within a set of feasible time slots $T_i = \{\max\{0, t_i - 4\}, \ldots, \max\{0, t_i - 1\}, t_i, \min\{|T|, t_i + 1\}, \ldots, \min\{|T|, t_i + 4\}\}$ (analogously for the desired arrival time $t_{i+|I|}$). The different travel options were calculated for each user, according to the available itineraries, time windows, and travel times in the network. Each of these options gives rise to a pair $(t_i^*(r, \pi), t_{i+|I|}^*(r, \pi))$ representing actual departure an arrival times for user i using itinerary π and option r. Finally, the inconvenience cost function $(\varphi_{i\pi r})$ was computed using the function

$$\varphi_{i\pi r} = \min\left\{1, \tilde{\varphi}(-|t_i - t_i^*(\pi, r)|^+) + \tilde{\varphi}(|t_{i+|I|}^*(\pi, r) - t_{i+|I|}|^+)\right\}, \tag{7}$$

where $|z|^+ = \max\{0, z\}$ and $\tilde{\varphi}$ is a discrete function defined as in Fig. 7. Note that $\tilde{\varphi}$ can be defined in many different ways according to the assumptions or requirements of the demand. For example, $\tilde{\varphi}$ defined as in Fig. 7 implies that an early deviation from the desired departure time is as inconvenient as a late deviation from the desired arrival time. However, $\tilde{\varphi}$ can be easily adapted in order to emphasize early departures or late arrivals.

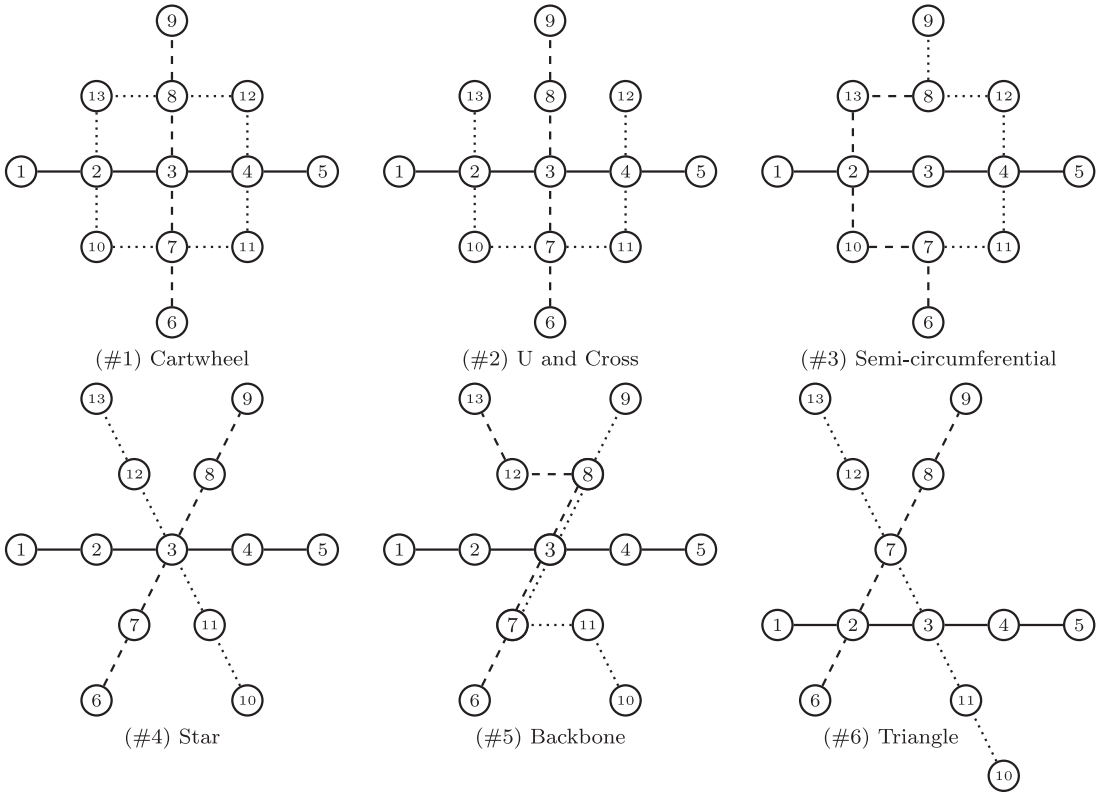

Fig. 6. Basic configurations obtained from 3 lines.

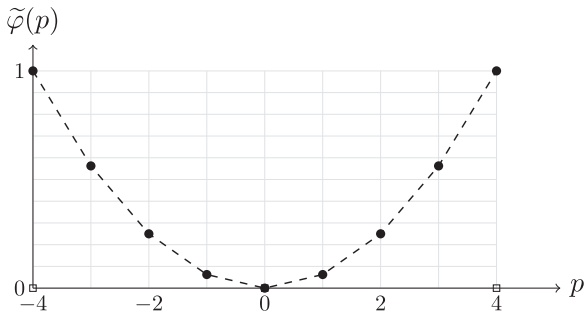

Fig. 7. Discrete inconvenience costs under consideration.

All instances were solved to optimality with the MIP Xpress 7.5 optimizer, under a Windows 7 environment in an Intel(R) Core(TM)i7 CPU 2.93 GHz processor and 8 GB RAM. Default values were initially used for all parameters of Xpress solver.

6.2. Non-dominated solutions

Fig. 8 presents the level curves of the Pareto fronts for the different graph configurations. For each configuration five level curves were obtained for values of $\kappa \in \{6, 9, 12, 15, 18\}$. We have generated random instances for $|I| = 100$ and $|T| = 60$ and travel times depending on $c_l = 3|A_l|$. For a given fleet size (κ) the upper-left point represents the minimum inconvenience cost value achievable by the highest number of line runs that can be performed, and the lower-right point represents the minimum inconvenience cost value that can be achieved by the smallest number of line runs. This explains why for $\kappa = 18$ the level curve in configuration #1 does not appear for values of $\rho \leq 300$ while in configurations #2 and #3 this curve contains only one or two points. We have bounded $\rho \in [100, 300]$ and $\kappa \in [6, 18]$, but more flexibility in this bounding should normally give rise to more trade-off solutions.

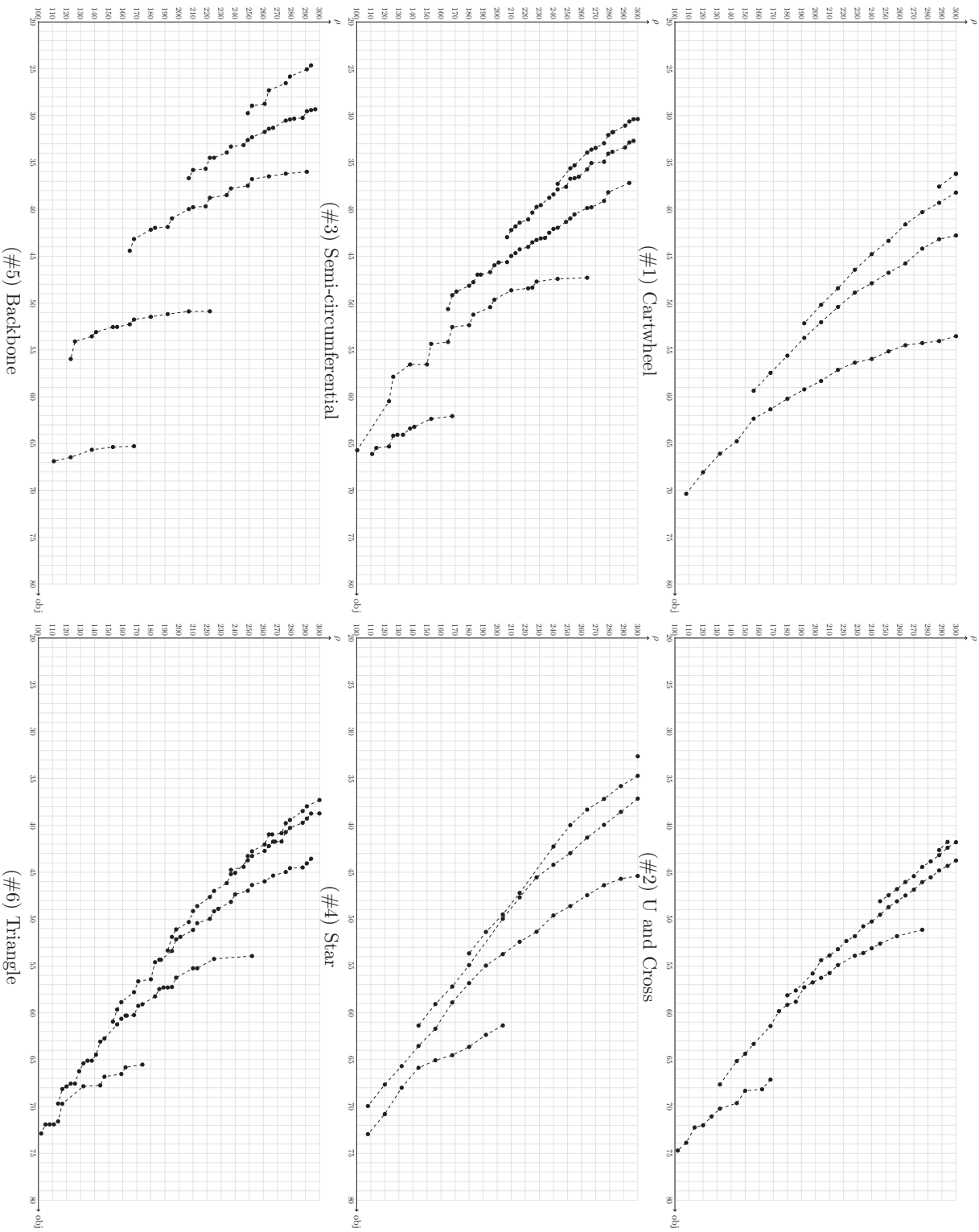

Fig. 8. Level curves ($\kappa \in \{18, 15, 12, 9, 6\}$) of the Pareto fronts for the different graph configurations.

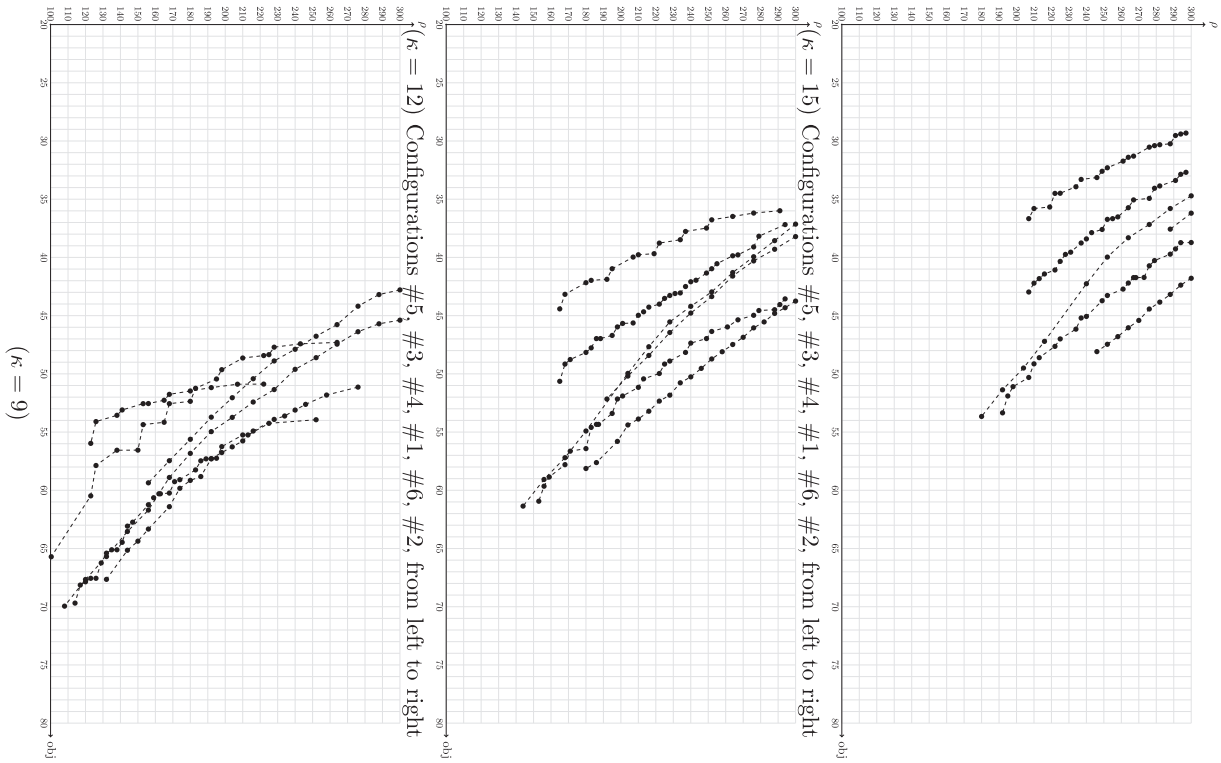

Fig. 9. Level curves of the Pareto fronts comparing the different graph configurations for values of $\kappa \in \{9, 12, 15\}$.

For decision makers, the kind of graphical solutions of Fig. 8 can be very informative. For example, for each configuration one can easily see the maximum (or minimum) number of line-runs that can be efficiently covered by a given fleet size, and one can also see the demand inconvenience improvement resulting from adding more vehicles. Note that in some cases, for a given value of ρ adding one more vehicle leads to a demand inconvenience percentage reduction of almost 15 points. However, when $\kappa \geq 15$ adding an extra vehicle yields small benefits in most cases. In any case, these graphs show that the Pareto frontier in the TNTSP is not necessarily convex and the ϵ-constraint algorithm yields the complete set of non-dominated points.

Complementary, we can add some additional remarks concerning the configurations comparison. Fig. 9 presents the level curves of the Pareto fronts comparing the different graph configurations for values of $\kappa \in \{9, 12, 15\}$. We note that now the level curves may not only overlap, but also intersect (see e.g. $\kappa = 9$ and $\rho \in [190, 230]$). This means that for a given value of κ, the best configuration in a comparison may depend on the value of ρ. In any case, for values of $\kappa \in \{12, 15\}$ the best configurations are #5, followed by #3, and the least efficient is configuration #2, followed by #6. The effect of the common lines in #5, seems to be very convenient for travelers in the common segments since they have two travel possibilities. However, for those transportation requests out of that sector, the common lines only imply a lack of supply. We have validated this comparisons in a preliminary study comparing results for 10 different instances. However, for the sake of simplicity, we just plot here the level curves corresponding to one of them.

6.3. Parametric analysis of solutions

We now analyze the computational performance of the TNTSP and its variants for the different configurations and specific values of ρ and κ. Therefore we now focus on the minimization of a single objective (the users' inconvenience cost), we vary $\rho \in \{170, 240, 300\}$, and fleet sizes are defined as $\kappa \in \{6, 12, 18\}$.

Each table reports the following information. Each row corresponds to a group of 10 instances with the same characteristics $(\#G, \rho, Q)$ or $(\#G, \rho, \kappa, Q)$ indicated in the first columns, where $\#G$ stands for configuration $\#i$, with $i \in \{1, 2, 3, 4, 5, 6\}$. Column *gapLR* reports the relative gap computed with the optimal solution and the optimal value of the linear relaxation at the root node. Column t reports the average running time in seconds for the 10 instances corresponding to each row. The reader may note that all instances were solved to optimality within the time limit. Column *nodes* indicates the average number of nodes explored in the branch-and-bound tree. Finally, column *obj* reports the mean objectives values that we recall represent the total user inconvenience. All tables report analogous information for the different formulations

Table 1
Computational results for the TNTP^U, TNTP^O and TNTP^S.

			TNTP^U				TNTP^O				TNTP^S			
#G	ρ	*Q*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*
1	170	1	60.39	34.2	1392	46.6	**0.33**	**0.4**	**1**	75	2.09	8.7	9	77.9
1	170	2	60.39	34.2	1392	46.6	**3.68**	**11**	**104**	61.3	9.22	996.8	14,287	65.7
1	170	3	60.39	**34.1**	1392	46.6	**11.6**	56.5	**1306**	54	14.8	1259.1	11,026	56.2
1	240	1	66.66	71.6	3393	32.9	**0**	**0.4**	**1**	65.9	4.37	49.6	1352	70.6
1	240	2	66.66	71.6	3393	32.9	**6.92**	**29.5**	**1090**	48.4	14.34	2987.3	33,326	53
1	240	3	66.66	**71.7**	**3393**	32.9	**23.75**	1236.9	37,124	40	26.13	1668.7	11,751	41.5
1	300	1	69.46	48.8	2701	24.1	**0.84**	**0.5**	**1**	59	8.47	141	10,080	65.9
1	300	2	69.46	48.8	2701	24.1	**12.1**	**31.3**	**850**	38.9	20.57	2115.6	39,186	43.4
1	300	3	69.46	**48.6**	**2701**	24.1	**36.46**	890.4	30,420	30.1	39.68	3257.1	30,149	31.9
2	170	1	39.1	4.3	647	54	**0.34**	**0.2**	**1**	75.3	2.2	1.5	1090	77.5
2	170	2	39.1	4.3	647	54	**3.42**	**2.2**	**28**	63.4	5.72	11.5	889	65.2
2	170	3	39.1	**4.3**	647	54	**9.94**	6.4	**440**	57.6	11.44	25	1533	58.6
2	240	1	44.94	5	259	41.3	**0.08**	**0.2**	**1**	66.4	3.06	2.1	253	69.6
2	240	2	44.94	5	**259**	41.3	**4.87**	**2.9**	568	51.3	8.42	40.5	5402	53.5
2	240	3	44.94	**5**	**259**	41.3	**17.08**	8.8	913	44.6	18.2	40.4	2995	45.3
2	300	1	49.33	6.4	2899	32.8	**0.05**	**0.2**	**1**	59.4	4.66	3.6	1710	63.8
2	300	2	49.33	6.5	2899	32.8	**7.57**	**4**	**606**	42.3	11.44	64.8	8131	44.3
2	300	3	49.33	**6.5**	2899	32.8	**25.21**	15.3	**2422**	35.5	26.08	34.8	2444	35.9
3	170	1	41.47	6.7	477	47.9	**0.1**	**0.2**	**1**	73.6	3.96	1.6	80	78.1
3	170	2	41.47	6.7	477	47.9	**2.28**	**3.1**	**280**	60.7	7.45	768.4	49,515	64.5
3	170	3	41.47	6.7	477	47.9	**7.21**	**5**	**63**	52.8	10.77	1215.8	39,547	55.1
3	240	1	44.9	7.2	**335**	34.4	**0.25**	**1.5**	2756	65.4	6.41	2.8	1511	71.3
3	240	2	44.9	**7.2**	**335**	34.4	**4.5**	16.9	4896	48.4	11.46	1235.1	52,726	52.7
3	240	3	44.9	**7.2**	**335**	34.4	**16.38**	39.3	6506	39.3	20.45	1294.7	37,958	41.5
3	300	1	46.57	8.8	1665	26	**0.1**	**0.3**	**3**	58.9	8.15	3.1	516	65.5
3	300	2	46.57	8.8	1665	26	**7.69**	**7.3**	**1405**	38.9	16.47	1411.1	86,987	43.4
3	300	3	46.57	**8.9**	**1665**	26	**25.8**	19.7	2090	29.6	29.85	1309.8	47,232	31.5
4	170	1	51.16	1.7	6	47.2	**0.33**	**0.1**	**1**	73.9	2.58	0.6	**1**	76.1
4	170	2	51.16	1.7	6	47.2	**4.03**	**1.1**	**4**	60.6	6.93	14.6	2916	62.7
4	170	3	51.16	**1.7**	**6**	47.2	**12.68**	3.4	67	53.7	15.43	24.5	3157	55.6
4	240	1	56.96	0.9	**1**	32.7	**0**	**0.1**	**1**	64.9	2.99	0.9	38	67.4
4	240	2	56.96	**0.9**	**1**	32.7	**7.22**	1.8	258	47.7	11.7	34.6	7368	50.5
4	240	3	56.96	**0.9**	**1**	32.7	**22.41**	5.5	832	39.4	25.52	113.6	20,297	41.1
4	300	1	59.99	0.8	**1**	24.2	**0.85**	**0.2**	**1**	57.9	5.01	1.6	70	61.3
4	300	2	59.99	**0.8**	**1**	24.2	**12.61**	2.3	238	38.8	17.9	132.4	29,688	41.7
4	300	3	59.99	**0.8**	**1**	24.2	**35.3**	7.8	1982	29.8	37.3	53.3	6373	30.9
5	170	1	1.29	**0**	**1**	69.6	**0.17**	**0**	**1**	77.6	1.11	**0**	136	78.6
5	170	2	1.29	**0**	**1**	69.6	**0.59**	0.1	150	70.4	0.86	0.1	161	70.6
5	170	3	1.29	**0**	**1**	69.6	**0.67**	0.1	228	69.6	**0.67**	0.2	1472	69.6
5	240	1	1.54	**0**	**1**	62.4	**0.18**	**0**	**1**	70.6	1.22	**0**	12	71.6
5	240	2	1.54	**0**	**1**	62.4	0.51	0.1	98	63	**0.47**	0.1	103	63
5	240	3	1.54	**0**	**1**	62.4	**0.7**	0.2	1904	62.4	**0.7**	0.3	1388	62.4
5	300	1	1.34	**0**	**1**	56.9	**0.2**	**0**	**1**	64.6	1.33	0.1	60	65.6
5	300	2	1.34	**0**	**1**	56.9	0.34	**0**	7	57.2	**0.32**	**0**	**1**	57.3
5	300	3	1.34	**0**	**1**	56.9	**0.46**	**0**	7	56.9	**0.46**	**0**	**1**	56.9
6	170	1	42.94	9.9	4917	49.4	**0.32**	**0.3**	**95**	73.4	3.35	0.7	266	76.9
6	170	2	42.94	10	4917	49.4	**2.68**	**1.8**	**475**	61.2	7.21	51.3	14,902	64.6
6	170	3	42.94	9.8	4917	49.4	**9.33**	**5.8**	**1776**	54	12.18	85.9	12,160	55.9
6	240	1	47.62	5.2	686	35.4	**0.24**	6.6	91,574	64.7	5.59	**1.1**	**149**	69.9
6	240	2	47.62	5.2	**686**	35.4	**5.3**	**4.3**	2482	49.1	11.18	247.7	64,006	52.8
6	240	3	47.62	**5.2**	**686**	35.4	**17.08**	10.1	2510	40.8	19.76	111.9	15,287	42.2
6	300	1	49.79	6.1	3248	26.9	**0.04**	**0.2**	**1**	58	8.13	1.5	144	64.8
6	300	2	49.79	6.1	3248	26.9	**8.19**	**4.9**	**2689**	39.9	15.41	352.1	63,831	43.7
6	300	3	49.79	**6.1**	3248	26.9	**25.04**	9.4	**1576**	31.4	27.18	65.8	8034	32.4

described in the paper. In order to facilitate the comparison among all tables, we have marked in bold the best result among all in the same group.

There exists a dependence between the network configuration and the results presented in the tables (Table 1 for the TNTP and Tables 2 and 3 for the TNTSP). First, regarding the running times, the instance that was more difficult to solve was the Cartwheel (#1) with a significant difference with respect to the other configurations. The main reason for this is because it is the network that offers the largest number of possible itineraries and hence strategies. In contrast, configuration

Table 2
Computational results for the TNTSP^U, TNTSP^O and TNTSP^S.

				TNTSP^U				TNTSP^O				TNTSP^S			
#G	ρ	κ	Q	GapLR	t	Nodes	Obj	GapLR	t	Nodes	Obj	GapLR	t	Nodes	Obj
1	170	6	1	60.7	190.1	7426	50	**0.45**	**3.2**	**8**	76.2	4.5	52.2	909	80.7
1	170	6	2	60.7	191.3	7426	50	**4.79**	**27.7**	**214**	63.9	10.78	1795.3	14,497	68.6
1	170	6	3	60.7	**192.7**	7426	50	**13.98**	265.2	**4743**	57.3	18.74	2933	22,425	60.9
1	170	12	1	60.39	35.3	1165	46.6	**0.35**	**0.6**	**1**	75.1	2.09	14	116	77.9
1	170	12	2	60.39	35.1	1165	46.6	**3.68**	**12.8**	**120**	61.3	9.23	1327.6	20,677	65.7
1	170	12	3	60.39	**35.4**	**1165**	46.6	**11.6**	70.3	1910	54	14.94	2254.9	18,775	56.3
1	170	18	1	60.39	46.5	2166	46.6	**0.33**	**0.5**	**1**	75	2.09	10	24	77.9
1	170	18	2	60.39	46.8	2166	46.6	**3.68**	**11.7**	**78**	61.3	9.22	999.4	15,943	65.7
1	170	18	3	60.39	**46.3**	2166	46.6	**11.6**	60.4	**1327**	54	14.8	1725.2	12,831	56.2
1	240	6	1	63.88	360.6	12,539	43.4	**0.31**	**3.6**	**1**	70.9	8.65	314.9	8257	78.5
1	240	6	2	63.88	361.1	12,539	43.4	**7.55**	**67.3**	**1056**	56.9	18.03	3004.1	19,511	64.5
1	240	6	3	63.88	**360.9**	12,539	43.4	**21.84**	740.8	**9728**	50.2	27.4	2876.9	16,903	54.1
1	240	12	1	66.94	126.9	6953	33.2	**0.01**	**0.8**	**1**	66	5.27	124.7	5547	71.3
1	240	12	2	66.94	125.5	6953	33.2	**7.22**	**61.1**	**2521**	48.6	16.15	3056.1	32,869	54.2
1	240	12	3	66.94	**126**	**6953**	33.2	**23.94**	921.3	27,740	40.1	26.4	2134.1	19,077	41.7
1	240	18	1	66.66	69.4	3108	32.9	**0**	**0.5**	**1**	65.9	4.37	60.1	2357	70.6
1	240	18	2	66.66	69.3	3108	32.9	**6.92**	**35.2**	**1384**	48.4	14.27	3168.8	34,816	52.9
1	240	18	3	66.66	**69**	**3108**	32.9	**23.75**	853.2	27,198	40	25.78	2150.4	15,994	41.3
1	300	6	1	63.58	297.7	11,609	42.9	**0.52**	**6**	**10**	69.7	10.18	347.2	9426	78.5
1	300	6	2	63.58	297.7	11,609	42.9	**7.65**	**58.7**	**667**	55.7	18.79	2949.7	19,425	63.8
1	300	6	3	63.58	**297.6**	11,609	42.9	**21.65**	349.8	**4478**	48.9	29.3	2804.3	16,830	54.4
1	300	12	1	69.77	139.8	9021	24.9	**0.91**	**1.5**	**1**	59.5	9.98	269	15,482	67.5
1	300	12	2	69.77	**140.3**	9021	24.9	**12.75**	153	**5481**	39.8	25.41	2643.1	30,208	46.9
1	300	12	3	69.77	**141.3**	**9021**	24.9	**37.44**	1779.3	40,717	30.9	41.6	3125.1	24,323	33.3
1	300	18	1	69.46	70.1	3819	24.1	**0.84**	**0.8**	**1**	59	8.6	179.3	12,586	66
1	300	18	2	69.46	69.9	3819	24.1	**12.1**	**35.2**	**1002**	38.9	20.64	2796.6	36,600	43.5
1	300	18	3	69.46	**70.1**	**3819**	24.1	**36.46**	1031.1	40,814	30.1	39.69	3343.3	31,074	31.9
2	170	6	1	38.02	13.5	**680**	57.6	**0.5**	**1.9**	1135	76.9	3.48	7.7	800	80.1
2	170	6	2	38.02	13.5	680	57.6	**3.9**	**5.9**	**196**	66.7	6.72	71.8	6418	68.9
2	170	6	3	38.02	13.5	680	57.6	**10.25**	**11.4**	**266**	61.5	12.86	96.3	3916	63.4
2	170	12	1	39.1	5.9	257	54	**0.34**	5.3	40,666	75.3	2.22	**2.8**	**156**	77.5
2	170	12	2	39.1	5.9	257	54	**3.41**	**2.6**	**55**	63.4	5.91	24.7	1979	65.3
2	170	12	3	39.1	**5.9**	**257**	54	**9.94**	6.9	668	57.6	11.44	24.2	1163	58.6
2	170	18	1	39.1	5.2	336	54	**0.34**	**0.4**	**1**	75.3	2.2	2.6	1127	77.5
2	170	18	2	39.1	5.2	336	54	**3.42**	**2.7**	**65**	63.4	5.72	14.8	1111	65.2
2	170	18	3	39.1	**5.3**	**336**	54	**9.94**	7.8	1071	57.6	11.44	28.9	2010	58.6
2	240	6	1	40.78	16.6	451	53.1	**0.11**	**0.8**	**4**	72.5	5.94	13	2274	78.1
2	240	6	2	40.78	16.7	451	53.1	**5.23**	**8.5**	**68**	62	9.22	125	4698	64.9
2	240	6	3	40.78	**16.7**	**451**	53.1	**15.12**	27.4	715	57.6	17.54	193.4	5286	59.4
2	240	12	1	44.85	7.6	280	41.4	**0.11**	**1.1**	**1**	66.8	3.62	6.9	2001	70.4
2	240	12	2	44.85	7.5	280	41.4	**4.92**	**5.4**	**74**	51.7	10	320.4	34,073	54.8
2	240	12	3	44.85	**7.5**	**280**	41.4	**17.68**	26.5	2890	45	19.01	82.4	5867	45.8
2	240	18	1	44.94	6.3	440	41.3	**0.08**	**0.3**	**1**	66.4	3.06	3.2	204	69.6
2	240	18	2	44.94	**6.3**	**440**	41.3	**4.87**	**6.3**	1750	51.3	8.42	50.8	5787	53.5
2	240	18	3	44.94	**6.3**	**440**	41.3	**17.08**	9.8	628	44.6	18.2	46.8	3044	45.3
2	300	6	1	40.68	17.5	461	53	**0.03**	**0.3**	**1**	72.4	5.98	11.5	1996	78.1
2	300	6	2	40.68	17.5	461	53	**5.26**	**11.3**	**56**	62	9.25	106.7	5190	64.9
2	300	6	3	40.68	**17.5**	**461**	53	**15.11**	24.3	477	57.6	17.55	214.6	5789	59.4
2	300	12	1	48.11	6.6	111	33.1	**0.15**	**0.9**	**76**	60.7	6.17	18.3	6119	65.9
2	300	12	2	48.11	**6.5**	**111**	33.1	**7.67**	14.4	2444	43.5	42.61	1625.7	40,798	44.8
2	300	12	3	48.11	**6.6**	**111**	33.1	**25.32**	49.5	5419	36.2	26.89	119.3	6866	37.1
2	300	18	1	49.33	11	4128	32.8	**0.02**	**0.3**	**1**	59.5	5.1	8.9	3575	64.1
2	300	18	2	49.33	11	4128	32.8	**7.57**	**6.3**	**312**	42.3	11.44	84.9	9582	44.4
2	300	18	3	49.33	**11**	4128	32.8	**25.21**	14.5	1933	35.5	26.08	30	**1199**	35.9
3	170	6	1	42.05	129.1	8770	56.2	**0.41**	**1.6**	**376**	76.8	4.74	9	1358	81.7
3	170	6	2	42.05	128.9	8770	56.2	**3.49**	**8.5**	**460**	66.2	9.52	1012.5	46,765	71
3	170	6	3	42.05	129.2	8770	56.2	**11.13**	**28.3**	**949**	60.5	13.53	168.4	3944	62.3
3	170	12	1	41.65	13	2465	48.1	**0.09**	**0.4**	**46**	73.6	4.14	3	364	78.3
3	170	12	2	41.65	13	2465	48.1	**2.32**	**7.1**	**1246**	60.8	7.54	1220.4	45,343	64.6
3	170	12	3	41.65	13	2465	48.1	**7.56**	**10.4**	**565**	53	11.04	1211.8	28,587	55.2
3	170	18	1	41.47	11.5	3135	47.9	**0.1**	**0.5**	306	73.6	3.96	2.9	**54**	78.1
3	170	18	2	41.47	11.4	3135	47.9	**2.28**	**4.2**	**186**	60.7	7.45	1213.3	68,758	64.5
3	170	18	3	41.47	11.4	3135	47.9	**7.21**	**10.3**	**428**	52.8	10.75	1213.6	46,455	55.1
3	240	6	1	42.5	125	6327	55	**0.08**	**0.5**	**1**	75.6	5.54	11.5	1336	81.4
3	240	6	2	42.5	125.4	6327	55	**4.43**	**15.3**	**154**	64.9	10.87	1283.6	32,296	69.9
3	240	6	3	42.5	124.8	6327	55	**13.18**	**34.7**	**750**	59.5	15.35	285.1	3947	61.1
3	240	12	1	45.53	18.7	1473	35.1	**0.18**	**0.8**	**363**	65.9	7.02	7.6	2135	72.3

(*continued on next page*)

Table 2 (*continued*)

				TNTSP^U				TNTSP^O				TNTSP^S			
#G	ρ	κ	*Q*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*
3	240	12	2	45.53	**18.6**	**1473**	35.1	**5.23**	20.1	7235	49.5	13.1	1370.6	53,813	54.3
3	240	12	3	45.53	**18.7**	**1473**	35.1	**17.1**	41.2	9433	40.2	23.47	1660.8	69,918	43.8
3	240	18	1	44.9	8.2	**372**	34.4	**0.25**	**2.6**	5533	65.4	6.44	4.6	1426	71.3
3	240	18	2	44.9	**8.2**	**372**	34.4	**4.5**	33.1	18,875	48.4	11.61	1247.7	48,034	52.7
3	240	18	3	44.9	**8.2**	**372**	34.4	**16.38**	69.9	22,311	39.3	20.91	1307.3	45,680	41.8
3	300	6	1	42.5	83.3	3700	55	**0.08**	**0.3**	**1**	75.6	5.54	11.3	1290	81.4
3	300	6	2	42.5	83.4	3700	55	**4.43**	**11.7**	**118**	64.9	10.87	1274.4	37,486	69.9
3	300	6	3	42.5	83.6	3700	55	**13.18**	**36.9**	**754**	59.5	15.35	294	4176	61.1
3	300	12	1	45.8	30.1	2654	27.6	**0.2**	**1.7**	**40**	60.3	10.48	33.2	7894	68.7
3	300	12	2	45.8	**30.2**	**2654**	27.6	**9.37**	51.8	5872	42.1	19.89	2360.5	112,090	48
3	300	12	3	45.8	**30.1**	**2654**	27.6	**25.82**	50.4	3088	32.3	32.44	1540.4	43,657	35.7
3	300	18	1	46.57	14.1	3136	26	**0.12**	**0.8**	**1**	59	8.59	8.4	1506	65.9
3	300	18	2	46.57	14.1	**3136**	26	**7.72**	**12.5**	4307	39	17.42	1859.3	120,764	43.9
3	300	18	3	46.57	**14.3**	3136	26	**25.8**	28.1	**2065**	29.6	30.24	1420.2	74,767	31.6

Backbone (#5) was the fastest to solve. Surprisingly, the Backbone (#5) was faster to solve than the Star configuration (#4). In spite of the fact that #5 offers more possibilities for completing the trips than #4, these trips are along common lines over edges (3,7) and (3,8), and this seems to make the allocation of users easier. Regarding U and Cross (#2) and Triangle (#6), we observe a similar performance except for the TNTSP^U problem where #2 seems to be more difficult. With respect to column *gapLR* we observe a direct relationship with column *t*, that is, those configurations which left a larger integrality gap at the root node were the most difficult to solve. Column *nodes* shows that, in spite of the big *gapLR* values, many of the TNTP^U and TNTSP^U instances could be solved at the root node of the branch-and-bound tree search. The last column *obj* shows that the lowest level of inconvenience depends of the configuration, as discussed in Section 6.2.

Comparing the discussed results among the different problems we found that some of the instances corresponding to TNTP^O and TNTSP^O were more easily solvable than TNTP^U and TNTSP^U, respectively, in terms of *t, gapLR* and *nodes*. This makes sense considering that the capacity constraints significantly reduce the solution space. Those transportation requests that cannot fit in the available vehicles are simply neglected (that is not served and, therefore, included in the objective function with the maximum inconvenience/penalty), which is not a major difficulty in the allocation problem. This is not the case of the TNTP^S or TNTSP^S for which the assignment is more difficult since it must be optimal for each transportation request once that a timetable is fixed (in the x variables). In general terms, adding the vehicle scheduling constraints makes it more difficult to solve each timetabling variant.

7. Conclusions

We have presented a new methodology for solving the integration of the Transit Network Timetabling and Scheduling Problem simultaneously with the users' routing problem. Traditionally, these problems have been studied sequentially but this approach leads to suboptimal solutions. We have presented a flexible framework that allows the allocation of transportation requests to their optimal strategies under capacity constraints. This approach not only pursues transfer coordination but also users' preferences in terms of preferred departure and arrival times for a fully disaggregated demand. Moreover, each transportation request is handled individually, under hard time windows constraints for departure and arrival times, as well as inconvenience costs related to trip duration and time deviations for desired departure and arrival times. We have generated a testbed of random instances for different network configurations and we have reported and analyzed extensive computational results. In addition, we have studied the multi-objective version of the problem in order to establish trade-offs between users' and operator's criteria. We have applied an ϵ-constraint method to obtain the exact Pareto front of solutions.

Acknowledgments

This research was partly supported by the Canadian Natural Sciences and Engineering Research Council under grant 2015–06189. The authors were also partially supported by the projects FQM-5849 (Junta de Andalucía\FEDER), MTM2013-46962-C02-01 (MICINN, Spain) and MTM2015-67706-P (MINECO/FEDER, UE). Thanks are due to the reviewers for their valuable comments.

Table 3
Computational results for the TNTSP^U, TNTSP^O and TNTSP^S.

				TNTSP^U				TNTSP^O				TNTSP^S			
#G	ρ	κ	*Q*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*
4	170	6	1	49.5	7.5	16	53	**0.25**	**0.3**	**1**	75	4.39	6.2	597	79.2
4	170	6	2	49.5	7.4	**16**	53	**4.57**	**3.9**	252	64.2	8.51	242.5	16,984	67.2
4	170	6	3	49.5	7.4	**16**	53	**13.54**	**6.5**	52	58.2	16.87	187.3	9640	60.6
4	170	12	1	51.25	3	**1**	47.3	**0.33**	**0.3**	**1**	73.9	2.52	1.1	7	76.1
4	170	12	2	51.25	**3**	**1**	47.3	**4.33**	3.2	22	60.8	7.18	33	5346	63
4	170	12	3	51.25	**3**	**1**	47.3	**12.73**	4.9	276	53.8	15.89	60.5	8678	55.9
4	170	18	1	51.16	2.3	5	47.2	**0.33**	**0.3**	**1**	73.9	2.58	1	**1**	76.1
4	170	18	2	51.16	2.3	5	47.2	**4.03**	**1.8**	**1**	60.6	6.93	14.1	1796	62.7
4	170	18	3	51.16	**2.3**	**5**	47.2	**12.68**	5	186	53.7	15.43	28	3212	55.6
4	240	6	1	45.54	2.5	**1**	44.9	**0.06**	**0.6**	**1**	72.3	7.37	19	5362	78.9
4	240	6	2	45.54	**2.5**	**1**	44.9	**4.86**	5.6	83	59.3	12.44	205.7	10,547	64.7
4	240	6	3	45.54	**2.5**	**1**	44.9	**12.96**	5.4	18	51.7	20.4	764	33,283	56.5
4	240	12	1	56.69	3	8	32.9	**0.04**	**0.3**	**1**	64.9	3.91	3.2	332	68.3
4	240	12	2	56.69	**3**	**8**	32.9	**7.85**	4.3	315	48.9	12.6	249.3	28,900	51.8
4	240	12	3	56.69	**3**	**8**	32.9	**22.7**	17.9	2480	40.1	26.4	742	78,345	42.3
4	240	18	1	56.96	1.2	**1**	32.7	**0**	**0.2**	**1**	64.9	3.06	1.7	77	67.5
4	240	18	2	56.96	**1.2**	**1**	32.7	**7.24**	2.1	133	47.8	11.96	173	21,388	50.7
4	240	18	3	56.96	**1.2**	**1**	32.7	**22.41**	5.8	491	39.4	25.52	139.3	17,382	41.1
4	300	6	1	45.54	2.4	**1**	44.9	**0.06**	**0.5**	**1**	72.3	7.37	19.7	4960	78.9
4	300	6	2	45.54	**2.3**	**1**	44.9	**4.86**	6	187	59.3	12.44	465.4	26,716	64.7
4	300	6	3	45.54	**2.3**	**1**	44.9	**12.86**	5	12	51.6	20.4	372.7	14,964	56.5
4	300	12	1	57.67	1.7	**1**	24.5	**0.69**	**0.9**	**1**	58.8	7.95	22	6338	64.5
4	300	12	2	57.67	**1.7**	**1**	24.5	**12**	8.1	216	40.7	20.06	1936.1	153,327	45.1
4	300	12	3	57.67	**1.7**	**1**	24.5	**34.08**	36.2	3003	31.6	37.37	1391.6	78,921	33.3
4	300	18	1	59.99	2	**1**	24.2	**0.89**	**0.3**	**1**	57.9	5.49	3.2	170	61.7
4	300	18	2	59.99	**2**	**1**	24.2	**12.76**	5.3	828	38.9	18.44	463.6	70,128	42
4	300	18	3	59.99	**2**	**1**	24.2	**35.3**	11.9	1580	29.8	37.48	156.2	18,699	31
5	170	6	1	0.88	5.5	48,332	70.4	**0.67**	75.7	729,220	78.3	1.63	**1.1**	**7033**	79.4
5	170	6	2	0.88	**5.5**	**48332**	70.4	**0.53**	10.7	84,442	71.7	0.82	35	214,672	72
5	170	6	3	0.88	**5.5**	**48332**	70.4	**0.38**	11.3	97,090	70.5	**0.38**	8.4	86,260	70.5
5	170	12	1	1.29	**0.3**	**1560**	69.6	**0.17**	38.1	1,296,782	77.6	1.11	37.7	662,709	78.6
5	170	12	2	1.29	**0.3**	1560	69.6	**0.59**	0.4	**1104**	70.4	0.86	3.5	39,160	70.6
5	170	12	3	1.29	**0.3**	**1560**	69.6	**0.67**	1.1	9175	69.6	**0.67**	1.4	12,795	69.6
5	170	18	1	1.29	**0.3**	**895**	69.6	**0.17**	36.2	598	77.6	1.11	6.4	116,376	78.6
5	170	18	2	1.29	**0.3**	**895**	69.6	**0.59**	0.8	5912	70.4	0.86	2.1	21,780	70.6
5	170	18	3	1.29	**0.3**	**895**	69.6	**0.67**	3.4	36,812	69.6	**0.67**	1.7	15,733	69.6
5	240	6	1	0.78	**0.1**	**1**	64.3	**0.22**	0.2	294	71.7	2.92	0.9	221	74.3
5	240	6	2	0.78	**0.1**	**1**	64.3	**0.27**	0.3	10	65.5	1	0.7	1156	66
5	240	6	3	0.78	**0.1**	**1**	64.3	**0.27**	**0.1**	**1**	64.4	**0.27**	**0.1**	**1**	64.4
5	240	12	1	1.54	0.2	1287	62.4	**0.18**	**0.1**	**1**	70.6	1.22	12.1	168,020	71.6
5	240	12	2	1.54	**0.2**	**1287**	62.4	0.51	30.4	283,976	63	**0.47**	59.3	538,854	63
5	240	12	3	1.54	**0.2**	**1287**	62.4	**0.7**	72.5	789,900	62.4	**0.7**	40	416,043	62.4
5	240	18	1	1.54	**3.4**	**47183**	62.4	**0.18**	19.1	933,441	70.6	1.22	21.2	272,756	71.6
5	240	18	2	1.54	**3.3**	**47183**	62.4	0.51	11	180,486	63	**0.47**	529	5,158,983	63
5	240	18	3	1.54	3.3	47,183	62.4	**0.7**	**1**	**10042**	62.4	**0.7**	43.6	416,546	62.4
5	300	6	1	0.64	**0.1**	**1**	63.1	**0.15**	0.3	55	69.5	4.36	0.2	**1**	73.8
5	300	6	2	0.64	0.1	**1**	63.1	**0**	**0**	**1**	64	0.87	0.2	4	64.6
5	300	6	3	0.64	0.1	**1**	63.1	**0.15**	**0**	**1**	63.2	**0.15**	0.1	**1**	63.2
5	300	12	1	1.31	**0.2**	**58**	56.9	**0.2**	581.6	7,609,794	64.6	2.22	1.4	11,173	66.3
5	300	12	2	1.31	**0.2**	**58**	56.9	0.34	0.3	239	57.4	**0.32**	0.3	196	57.4
5	300	12	3	1.31	**0.2**	**58**	56.9	**0.44**	0.5	1132	57	**0.44**	0.3	349	57
5	300	18	1	1.35	**0.1**	**1**	56.9	**0.2**	**0.1**	4	64.6	1.33	0.2	**1**	65.6
5	300	18	2	1.35	**0.1**	**1**	56.9	0.34	0.3	20	57.2	**0.32**	**0.1**	**1**	57.3
5	300	18	3	1.35	**0.1**	**1**	56.9	**0.46**	0.2	2	56.9	**0.46**	0.2	**1**	56.9
6	170	6	1	42.65	13.9	668	55.6	**0.23**	**1.6**	**158**	75.7	5.07	3.8	1140	80.6
6	170	6	2	42.65	13.9	668	55.6	**3.75**	**6.8**	**240**	65.3	9.28	80.5	7154	69.7
6	170	6	3	42.65	13.9	668	55.6	**11.02**	**12.3**	**346**	59.8	13.12	46.2	2176	61.3
6	170	12	1	42.79	48.4	55,714	49.4	**0.35**	1200.3	5,522,963	73.5	3.42	**1.4**	**45**	76.9
6	170	12	2	42.79	47.5	55,714	49.4	**2.7**	**3.1**	**1082**	61.2	7.3	39.6	10,221	64.7
6	170	12	3	42.79	47.2	55,714	49.4	**9.36**	**11.4**	**5724**	54	12.54	580.3	70,596	56.1
6	170	18	1	42.94	43.8	54,006	49.4	**0.32**	1200	5,584,532	73.4	3.35	**1.4**	**409**	76.9
6	170	18	2	42.94	44.6	54,006	49.4	**2.68**	**3.5**	**1765**	61.2	7.21	37.6	8987	64.6
6	170	18	3	42.94	43.6	54,006	49.4	**9.33**	**11.9**	**6701**	54	12.18	147.2	20,555	55.9
6	240	6	1	42.68	11.1	152	53.4	**0.2**	**1.1**	**4**	73.8	6.29	4.2	422	79.9
6	240	6	2	42.68	11.1	152	53.4	**4.69**	**9.4**	**76**	63.3	10.68	63.6	4269	68
6	240	6	3	42.68	**11.1**	**152**	53.4	**12.77**	14	372	57.8	14.79	37.4	516	59.2
6	240	12	1	47.48	14.4	6108	36	**0.4**	3.9	9176	65.1	6.94	**3.7**	**2032**	71.2

(*continued on next page*)

Table 3 (*continued*)

				TNTSP^U				TNTSP^O				TNTSP^S			
#G	ρ	κ	*Q*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*	*GapLR*	*t*	*Nodes*	*Obj*
6	240	12	2	47.48	14.4	6108	36	**5.96**	**9.5**	**3136**	50	12.49	274.4	36,802	54.2
6	240	12	3	47.48	**14.6**	6108	36	**17.19**	24.5	**5284**	41.5	21.22	1285.7	81,063	43.7
6	240	18	1	47.62	7.4	2339	35.4	**0.24**	4.8	26,133	64.7	5.64	**2.3**	**260**	69.9
6	240	18	2	47.62	7.4	2339	35.4	**5.3**	**4.8**	**886**	49.1	11.19	308.2	77,276	52.9
6	240	18	3	47.62	**7.3**	**2339**	35.4	**17.08**	18.4	6464	40.8	19.76	366.1	38,674	42.2
6	300	6	1	42.68	10.9	149	53.4	**0.2**	**1**	**2**	73.8	6.29	4.1	250	79.9
6	300	6	2	42.68	10.9	149	53.4	**4.69**	**7.8**	**143**	63.3	10.68	72.6	4181	68
6	300	6	3	42.68	**11**	**149**	53.4	**12.77**	13.9	340	57.8	14.79	43.9	1061	59.2
6	300	12	1	48.06	18.1	3647	28.4	**0.39**	**2.7**	**434**	59.6	10.55	5.3	1426	67.9
6	300	12	2	48.06	18.1	3647	28.4	**8.23**	**9.4**	**844**	42.5	18.12	1826.7	153,738	48
6	300	12	3	48.06	**18.2**	3647	28.4	**24.41**	30	**2326**	33.8	28.05	1376.6	58,543	35.6
6	300	18	1	49.79	16.6	14,680	26.9	**0.04**	**0.3**	**34**	58.1	8.55	3.3	1316	65.1
6	300	18	2	49.79	16.8	14,680	26.9	**8.14**	**6.8**	**1743**	39.9	15.61	334.6	53,534	43.8
6	300	18	3	49.79	16.7	14,680	26.9	**25.04**	**10.1**	**1182**	31.4	27.41	147.4	16,557	32.5

Appendix A. Notation

Infrastructure:

G	Graph corresponding to the PTN
S	Node set (stations)
A	Set of arcs (indexed by a)
$\mathcal{L}$	Set of lines (indexed by l)
$S_l \subseteq S$	Subset of stations used by line l
$A_l \subset A$	Subset that contains all edges used by line l
$\overline{\mathcal{L}} \subseteq \mathcal{L}$	Set of directed path lines
$\mathring{\mathcal{L}} \subseteq \mathcal{L}$	Set of directed cycle lines
$\overrightarrow{\mathcal{L}} \subseteq \mathcal{L}$	Set of path lines going forward
$\overleftarrow{\mathcal{L}} \subseteq \mathcal{L}$	Set of path lines going backwards (a line $l \in \overrightarrow{\mathcal{L}}$ is associated to its respective (opposite) line $l' \in \overleftarrow{\mathcal{L}}$ by means of $l' = l + \lvert\overrightarrow{\mathcal{L}}\rvert$
$\mathcal{N}$	Set of nodes of all lines ($\mathcal{N} = \{(l, i) : l \in \mathcal{L}, i \in S_l\}$)
$\mathcal{A}$	Set of arcs of all lines ($\mathcal{A} = \{(l, i, j) : l \in \mathcal{L}, (i, j) \in A_l\}$)
$ddot\mathcal{N}$	Set of transfer nodes
$\mathcal{A}^{(tra)}$	Set of transfer edges ($\mathcal{A}^{(tra)} = \{(l, i, n) : l \in \mathcal{L}, i \in S_l, n \in \ddot{\mathcal{N}}\}$)

Timetables and vehicle scheduling:

T	Set of time slots ($T = \{1, \ldots, \lvert T\rvert\}$)
τ_l	Fixed travel time required to complete a line run in line l
Q	Vehicle capacity
c_l	Cost associated to locate a line run in line l
ρ	Total available budget to locate line runs
κ	Fleet size

Demand:

$i \in I$	Set of transportation requests
t_i, $t_{i+\lvert I\rvert}$	Preferred departure and arrival times for request i
t_i^-, t_i^+	Earliest and latest times that are admissible for serving request i

Strategies:

$\pi \in \Pi$	Set of possible itineraries within the PTN
$\Pi_i \subset \Pi$	Subset of itineraries that are valid for user i
$\mathcal{L}_{i\pi} \subseteq \mathcal{L}$	Set of lines used by request i when itinerary $\pi \in \Pi_i$ is selected
$r \in \mathcal{R}_{i\pi}$	Set of options available for request i when using itinerary $\pi \in \Pi_i$
$\varphi_{i\pi r}$	Cost of allocating request i to itinerary π and option option r
$m_{\pi a}$	Binary parameter equal to one if itinerary $\pi \in \Pi_i$ occupies arc $a \in A$
$t_{i\pi rl}$	Time slot that is used for a vehicle departure in line l when the itinerary π and the option r are used

Decision variables:

ρ_l	Number of line runs to locate in line l
κ_l	Fleet size to assign in line l

$x_{lt} \in \{0, 1\}$ Binary variable equal to 1 if and only if a vehicle starts a line run in line l at time t
$y_{i\pi r} \in \{0, 1\}$ Binary variable equal to 1 if and only if request i is allocated to itinerary $\pi \in \Pi_i$ and option $r \in \mathcal{R}_{i\pi}$

References

Borndörfer, R., Hoppmann, H., Karbstein, M., 2015. Timetabling and passenger routing in public transport. In: Proceedings of Conference on Advanced Systems in Public Transport 2015 (CASPT2015).

Bunte, S., Kliewer, N., 2009. An overview on vehicle scheduling models. Public Transp. 1, 299–317.

Cacchiani, V., Toth, P., 2012. Nominal and robust train timetabling problems. Eur. J. Oper. Res. 219(3), 727–737.

Caprara, A., Kroon, L., Monaci, M., Peeters, M., Toth, P., 2007. Passenger railway optimization. In: Barnhart, C., Laporte, G. (Eds.), Handbooks in Operations Research and Management Science, volume 14, chapter 3. Elsevier, Amsterdam, pp. 129–187.

Caprara, A., Kroon, L.G., Toth, P., Smith, J.C., 2011. Optimization problems in passenger railway systems. In: Cochran, J., Cox, L., Keskinocak, P., Kharoufeh, J. (Eds.), Wiley Encyclopedia of Operations Research and Management Science, volume 6. John Wiley & Sons, Inc., pp. 3896–3905.

Castelli, L., Pesenti, R., Ukovich, W., 2004. Scheduling multimodal transportation systems. Eur. J. Oper. Res. 155(3), 603–615.

Ceder, A., 2001. Efficient timetabling and vehicle scheduling for public transport. In: Voß, S., Daduna, J. (Eds.), Lecture Notes in Economics and Mathematical Systems, volume 505, chapter Computer-Aided Scheduling of Public Transport. Springer, Berlin Heidelberg, pp. 37–52.

Ceder, A., 2011. Optimal multi-vehicle type transit timetabling and vehicle scheduling. Procedia Soc. Behav. Sci. 20, 19–30.

Chakroborty, P., Deb, K., Sharma, R.K., 2001. Optimal fleet size distribution and scheduling of transit systems using genetic algorithms. Transp. Plann. Technol. 24(3), 209–226.

Chankong, V., Haimes, Y.Y., 1983. Multiobjective Decision Making: Theory and Methodology. North-Holland, Amsterdam.

Espejo, I., Marín, A., Rodríguez-Chía, A.M., 2012. Closest assignment constraints in discrete location problems. Eur. J. Oper. Res. 219(1), 49–58.

Fleurent, C., Lessard, R., 2009. Integrated Timetabling and Vehicle Scheduling in Practice. Technical report. GIRO Inc. Montreal, Canada.

Goerigk, M., Schmidt, M., 2016. Line Planning With User-Optimal Route Choice. Lancaster University Press, Lancaster, United Kingdom. Working Paper

Guihaire, V., Hao, J.K., 2008a. Transit network design and scheduling: a global review. Transp. Res. Part A 42(10), 1251–1273.

Guihaire, V., Hao, J.K., 2008b. Transit network re-timetabling and vehicle scheduling. In: Thi, H.L., Bouvry, P., Pham Dinh, T. (Eds.), Modelling, Computation and Optimization in Information Systems and Management Sciences, Communications in Computer and Information Science, volume 14. Springer, Berlin Heidelberg, pp. 135–144.

Guihaire, V., Hao, J.K., 2010. Transit network timetabling and vehicle assignment for regulating authorities. Comput. Ind. Eng. 59(1), 16–23.

Van den Heuvel, A., Van den Akker, J., Van Kooten, M., 2008. Integrating Timetabling and Vehicle Scheduling in Public Bus Transportation. Technical report, uu-cs-2008-003. Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands.

Ibarra-Rojas, O.J., Giesen, R., Rios-Solis, Y.A., 2014. An integrated approach for timetabling and vehicle scheduling problems to analyze the trade-off between level of service and operating costs of transit networks. Transp. Res. Part B 70(0), 35–46.

Kim, I.Y., de Weck, O.L., 2005. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. Struct. Multidiscip. Optim. 29(2), 149–158.

Laporte, G., Mesa, J.A., Ortega, F.A., 1994. Assessing topological configurations for rapid transit networks. Stud. Locational Anal. 7, 105–121.

Laporte, G., Mesa, J.A., Ortega, F.A., 1997. Assessing the efficiency of rapid transit configurations. TOP 5(1), 95–104.

Liu, Z.G., Shen, J.S., 2007. Regional bus operation bi-level programming model integrating timetabling and vehicle scheduling. Syst. Eng.-Theory Pract. 27(11), 135–141.

Lusby, R., Larsen, J., Ehrgott, M., Ryan, D., 2011. Railway track allocation: models and methods. OR Spectr. 33 (4), 843–883.

Mesa, J.A., Ortega, F.A., Pozo, M.A., 2013. A geometric model for an effective rescheduling after reducing service in public transportation systems. Comput. Oper. Res. 40(3), 737–746.

Mesa, J.A., Ortega, F.A., Pozo, M.A., 2014. Locating optimal timetables and vehicle schedules in a transit line. Ann. Oper. Res. 222, 439–455.

de Palma, A., Lindsey, R., 2001. Optimal timetables for public transportation. Transp. Res. Part B 35(8), 789–813.

Petersen, H., Larsen, A., Madsen, O.B.G., Petersen, B., Ropke, S., 2013. The simultaneous vehicle scheduling and passenger service problem. Transp. Sci. 47, 603–616.

Schmidt, M., Schöbel, A., 2015a. The complexity of integrating passenger routing decisions in public transportation models. Networks 65(3), 228–243.

Schmidt, M., Schöbel, A., 2015b. Timetabling with passenger routing. OR Spectr. 37(1), 75–97.

Schöbel, A., Scholl, S., 2006. Line planning with minimal traveling time. In: Kroon, L., Möhring, R. (Eds.), 5th Workshop on Algorithmic Methods and Models for Optimization of Railways (ATMOS'05), OpenAccess Series in Informatics (OASIcs), Vol. 2. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany.

Shier, D.R., 1979. On algorithms for finding the k-shortest paths in a network. Networks 9, 195–214.

Siebert, M., Goerigk, M., 2013. An experimental comparison of periodic timetabling models. Comput. Oper. Res. 40(10), 2251–2259.

Small, K.A., 1982. The scheduling of consumer activities: work trips. Am. Econ. Rev. 72(3), 467–479.

Stern, R., 1996. Passenger transfer system review. Synthesis of Transit Practice 19. Transportation Research Board, National Research Council, National Academy Press, Washington D.C.

Törnquist, J., 2007. Railway traffic disturbance management–an experimental analysis of disturbance complexity, management objectives and limitations in planning horizon. Transp. Res. Part A 41(3), 249–266.

Wagner, J.L., Falkson, L.M., 1975. The optimal nodal location of public facilities with price-sensitive demand. Geogr. Anal. 7(1), 69–83.