A cost-minimization model for bus fleet allocation featuring the tactical generation of short-turning and interlining options

Gkiotsalitis, K.; Wu, Z.; Cats, O.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

A cost-minimization model for bus fleet allocation featuring the tactical generation of short-turning and interlining options

K. Gkiotsalitis[a], Z. Wu[b,*], O. Cats[c]

[a] *University of Twente, De Horst 2, 7522LW Enschede, the Netherlands*
[b] *Imperial College London, South Kensington Campus, London SW7 2AZ, UK*
[c] *Delft University of Technology, Postbus 5, 2600 AA Delft, the Netherlands*

A B S T R A C T

Urban public transport operations in peak periods are characterized by highly uneven demand distributions and scarcity of resources. In this work, we propose a rule-based method for systematically generating and integrating alternative lining options, such as short-turning and interlining lines, into the frequency and resource allocation problem by considering the dual objective of (a) reducing passenger waiting times at stops and (b) reducing operational costs. The bus allocation problem for existing and short-turning/interlining lines is modeled as a combinatorial, constrained and multi-objective optimization problem that has an exponential computational complexity and a large set of decision variables due to the additional set of short-turning/interlining options. This constrained optimization problem is approximated with an unconstrained one with the use of exterior point penalties and is solved with a Genetic Algorithm (GA) meta-heuristic. The modeling approach is applied to the bus network of The Hague with the use of General Transit Feed Specification (GTFS) data and Automated Fare Collection (AFC) data from 24 weekdays. Sensitivity analysis results demonstrate a significant reduction potential in passenger waiting time and operational costs with the addition of only a few short-turning and interlining options.

1. Introduction

Ideally, public transport supply will perfectly correspond and scale to passenger demand. However, this is impossible in real-world operations due to the uneven distribution of demand over time and space. This results in inefficiencies for both passengers and operators and creates the need to re-dimension the fleet and circulate vehicles between demand areas.

Planning decisions regarding public transport services in general, and bus networks in particular, are typically made at the strategic, tactical and operational planning level (Ibarra-Rojas et al., 2015). At the strategic level, the network and route-design problem is addressed where the alignment of the bus lines and the location of the bus stops are determined (Mandl, 1980; Ceder and Wilson, 1986; Pattnaik et al., 1998; Szeto and Wu, 2011; Borndörfer et al., 2007). Subsequently, at the tactical planning level, the sub-problems of bus frequency settings (Gkiotsalitis and Cats, 2017), timetable design (Ceder et al., 2001; Gkiotsalitis and Maslekar, 2018a; Gkiotsalitis and Maslekar, 2018b), vehicle scheduling (Ming et al., 2013), driver scheduling (Wren and Rousseau, 1995) and driver rostering (Moz et al., 2009) are typically addressed in a sequential order.

Apart from the strategic and tactical planning, bus operators can take decisions over the course of the daily operations. In the
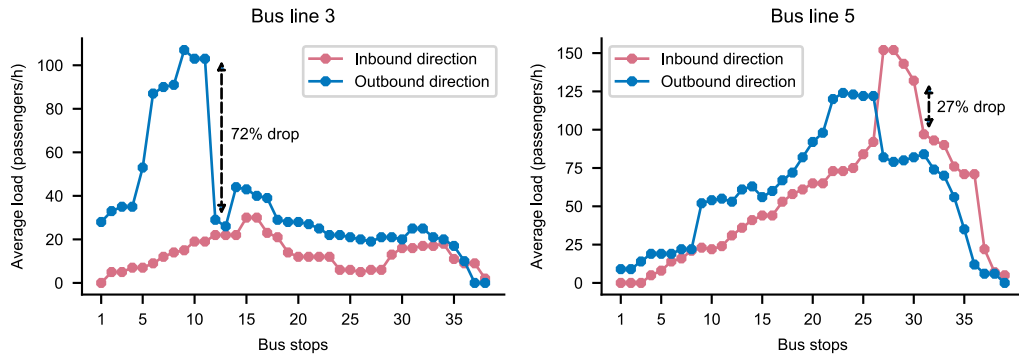

Fig. 1. Average bus-load per bus stop for bus line 3 and bus line 5 in The Hague from 4 pm to 5 pm.

operational planning phase, near real-time control measures such as stop-skipping (Sun and Hickman, 2005; Yu et al., 2015; Chen et al., 2015), dispatching time changes (Gkiotsalitis and Stathopoulos, 2016) or bus holdings at specific stops (Newell, 1974; Hernández et al., 2015; Wu et al., 2017; Gavriilidou and Cats, 2018) can be deployed. Notwithstanding, bus holding tends to increase the inconvenience of on-board passengers who are held at stops (Fu and Yang, 2002) and stop-skipping increases the inconvenience of passengers who cannot board the bus that skips their stop (Liu et al., 2013).

Typically, the strategic, tactical and operational planning problems are addressed at different levels with the exception of a number of works that solve together the strategic-level problem of route design and the tactical-level problems of frequency settings and timetable design (Yan et al., 2006; Zhao and Zeng, 2008). Especially, the simultaneous solution of the route design and the frequency settings problem has the potential of improving the efficiency of the operations by modifying the bus routes and the corresponding frequencies to better cater for the passenger demand imbalances.

The frequency settings problem has been studied by several works in literature (Yu et al., 2009; Shireman, 2011; Gkiotsalitis and Cats, 2018). Unlike frequencies, modifying bus routes on a regular basis for improving the demand matching (i.e. operating different routes on different times of the day) and reducing the operational costs is not practical because passengers rely heavily on the pre-defined routes of the bus network. Therefore, frequent route changes increase significantly the passenger inconvenience even if they are properly communicated (Kepaptsoglou and Karlaftis, 2009; Daganzo, 2010). Given the above, bus operators tend to modify the frequencies of bus lines, but they are reluctant to modify the bus routes that cover specific segments of bus lines which exhibit significant demand imbalances (examples of which are illustrated in Fig. 1).

In Fig. 1, one can observe that the average bus-load can be significantly higher at specific segments of a bus line. As a result, if the bus frequencies are set according to the well-known maximum loading point rule (Ceder, 2016) which ensures that the frequency is such that the bus load at the most heavily-used point along the route does not exceed the bus capacity, then buses will be significantly underutilized for the remaining parts of their routes.

Fig. 1 presents the average bus occupancy levels of two bus lines (line 3 and line 5 in The Hague) from 4 pm until 5 pm and indicates the problem of vehicle underutilization when modifying frequencies is the only option. For instance, the buses of the outbound direction of line 3 serve more than 20 passengers between stops 7 and 11 but are significantly underutilized at stops 1–6 and 12–38 which account for $\simeq 87\%$ of the route. Instead, the generation of new routes for serving only this specific segment can resolve this problem in a more efficient way than a mere modification of bus frequencies.

It should be noted here that the passenger utilization of route segments presented in Fig. 1 can be inferred from smartcard data (Munizaga and Palma, 2012). Moreover, the recurrence of travel patterns and related user profiles and user preferences can be inferred using clustering and choice modelling techniques (Ma et al., 2013; Gkiotsalitis and Stathopoulos, 2015; Goulet-Langlois et al., 2016; Yap et al., 2018). This information can then be instrumental in identifying systematic patterns in relation to the correspondence between supply and demand.

Given the practical and public acceptance issues associated with bus route variants, other flexible approaches which consider the deployment of short-turning and interlining can be considered. The works of Verbas and Mahmassani (2013) and Verbas et al. (2015) provide a first step in this direction since they do not allocate bus frequencies at a line level, but at a segment level considering a pre-defined set of short-turning options.

This work leverages on the potential flexibility embodied in short-turning and interlining lines in catering more efficiently to the prevailing passenger demand variations. First, observed passenger demand variations are used for generating a set of potential switch points along existing bus service lines where short-turning and interlining operations are allowed. The switch points are a subset of the bus stops of the network. Short-turning and interlining options are permitted at each switch point; thus, there is an additional set of (sub-) lines which can serve a set of targeted line segments. We denote the generated candidate short-turning and interlining lines as "virtual lines" for which vehicles can be allocated if deemed desirable. With this approach, we introduce an additional flexibility in allocating buses to lines because apart from the originally planned lines, buses can also be allocated to the set of virtual lines in order to match the passenger demand variation at different segments of bus lines without serving unnecessarily all the stops of the originally planned lines.

The generation of virtual short-turning and interlining lines enables the allocation of vehicles at specific line segments with significant passenger demand, but at the same time increases dramatically the number of lines where buses may be allocated. Given the combinatorial nature of the vehicle allocation problem and the vast number of potential bus allocation combinations to originally planned and virtual lines, the combinatorial solution space cannot be exhaustively explored for obtaining a globally optimal solution. To this end, this work contributes by (a) modeling the above-mentioned problem for the first time and introducing an automated, rule-based scheme for generating switch point stops for short-turning and interlining "virtual lines", (b) introducing an exterior point penalization scheme for penalizing the violation of constraints and approximating the constrained optimization problem with an unconstrained one, (c) developing a problem-specific genetic algorithm that returns improved solutions without performing an exhaustive exploration of the combinatorial solution space and (d) investigating the potential gains in operational costs and passenger waiting times by applying the set of the above-described methods at the bus network of The Hague, the Netherlands.

2. Related studies

The frequency settings problem has been extensively studied by several works in the literature (Farahani et al., 2013; Ceder, 2007; Barra et al., 2007; Cipriani et al., 2012; Fan and Machemehl, 2008). Most works on setting the optimal bus frequencies address the problem as an exercise of balancing the passenger demand with the available supply of buses (Furth and Wilson, 1981; Cipriani et al., 2012) or utilize the passenger load profile/maximum loading point rule-based techniques (Ceder, 1984; Ceder, 2007; Hadas and Shnaiderman, 2012)).

Examining in more detail the works on bus frequency settings, Yu et al. (2009) determined the optimal bus frequencies subject to the fleet size constraints using a bi-level model, which consisted of a genetic algorithm and a label-marking method. Hadas and Shnaiderman (2012) used AVL and automatic passenger counting (APC) data to construct the statistical distributions of passenger demand and travel time by time of day and used them for determining the bus frequencies based on the minimization of empty-seats and the avoidance of passenger overload. Bellei and Gkoumas (2010) and Li et al. (2013) also considered stochastic demand and travel times when optimizing the bus frequencies.

dellOlio et al. (2012) developed a bi-level optimization model for determining the bus sizes and the frequency settings. In their work, the upper-level model allowed buses of different sizes to be assigned to public transport lines and the lower-level optimized the frequency of each line according to the passenger demand using the Hooke-Jeeves algorithm. Huang et al. (2013) developed a bi-level programming model for optimizing bus frequencies while considering uncertainties in bus passenger demand. They used a genetic algorithm (GA) to solve the model for an example network in the city of Liupanshui, China resulting in a 6% reduction in the total cost of the transit system.

Another line of works has jointly addressed the route design and the frequency settings problem. Arbex and da Cunha (2015) solved both the route design and the frequency setting problem with the use of a genetic algorithm aiming at minimizing the sum of passengers' and operators' costs. Similarly, Szeto et al. (2011) addressed the same problem by using a genetic algorithm for optimizing the route design problem and a neighborhood search heuristic for optimizing the frequency setting problem for a suburban bus network in Hong Kong. Both Arbex and da Cunha (2015) and Szeto et al. (2011) design the routes and bus frequencies at the strategic planning level and do not permit any route modification (such as the inclusion of short-turning or interlining lines) at the tactical planning stage.

This lack of consideration of short-turning and interlining lines poses a substantial limitation since allocating the optimal amount of resources (i.e., buses) to originally planned service lines does not guarantee the optimal utilization of vehicles. This is supported by several studies such as Furth and Wilson (1981) and Hadas and Shnaiderman (2012) that explore the issue of bus underutilization (empty-seats) when setting frequencies according to load profile-based techniques or techniques that try to match the passenger demand with the available bus supply without allowing route alterations.

The introduction of a flexible route design and vehicle allocation scheme in the tactical planning phase (where service frequencies are not set per line, but per line segment based on the automatically generated short-turning and interlining lines that serve those segments) is a key feature of the approach adopted in this study. The works of Delle Site and Filippi (1998), Cortés et al. (2011), Verbas and Mahmassani (2013) and Verbas et al. (2015) focus on generating short-turning lines for serving the demand variation at specific line segments and are therefore the most relevant studies to our work. Cortés et al. (2011) showed that short turning lines can yield large savings of operational costs even if they require more deadheading for performing the short-turning routes. In Verbas and Mahmassani (2013) and Verbas et al. (2015) the frequencies of buses were not allocated at the line level, but at the segment level using also short-turning lines. Previous studies considered only pre-defined short-turning lines that can cover the spatiotemporal demand variations at different segments of the service lines based on historical passenger demand data. In contrast, in this work sub-lines and inter-lines are generated automatically by introducing a framework that allows not only for short-turning lines but interlining options as well as detailed in the following section.

3. Methodology

3.1. Overall framework

Before presenting the overall framework, we first clarify the use of the terms short-turning lines (also referred to as sub-lines) and interlining lines (also referred to as interlines). In the context of this work, a short-turning line is a line that serves all stops of a segment of an originally planned line (in both directions). The bus stops in that segment are served in the same order as they would
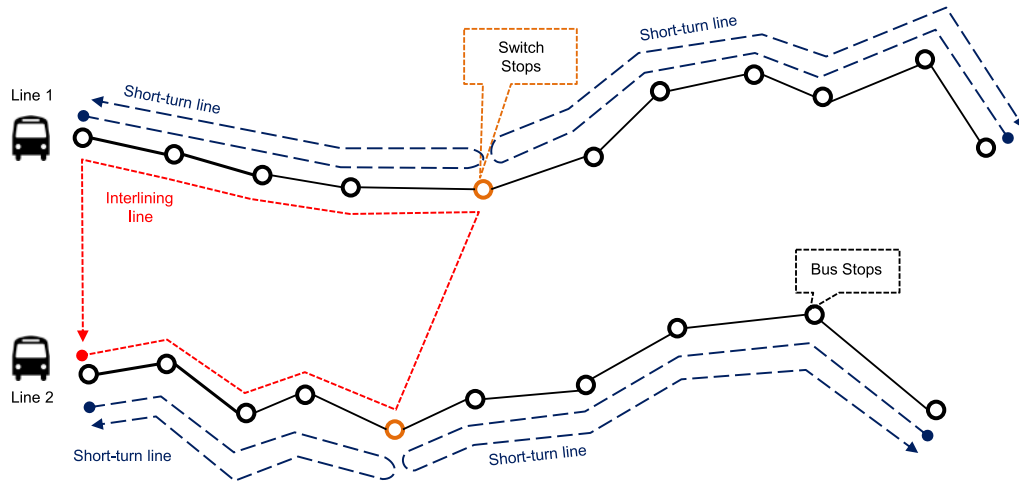
Fig. 2. Originally planned lines (black) and a potential generation of short-turning lines (blue dashed) or interlining lines (red) at specific switch stops (orange). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

have been served by the originally planned line service. In contrast, an interlining line serves one direction of a segment of an originally planned line and another segment from another originally planned line (see Fig. 2). The interlining line serves those two segments uni-directionally resulting in a loop form.

Given the above conventions, we can have an initial indication of whether a short-turning or interlining line fits a particular scenario of passenger demand. First, a short-turning line must always serve all stops in both directions of a segment of an originally planned line. Hence, a short-turning line is more suitable for accommodating segments of an originally planned line which exhibit significantly higher ridership levels in both directions. In contrast, an interlining line is beneficial for line segments with significant bus loads at one direction only since they will serve only that direction and then serve a series of stops of another originally planned line segment.

For the generation of potential short-turning and interlining lines from the existing bus lines, one needs to establish first a set of switch point stops. Theoretically, the number of switch points for an originally planned bus line can be equal to the number of its bus stops. Nevertheless, generating all possible sub-lines and inter-lines considering each bus stop as a potential switch point is a computationally complex task and may result in a service that is difficult to operate and communicate to passengers. For this reason, works such as Verbas and Mahmassani (2013) and Verbas et al. (2015) propose to pre-define a limited set of switch stops at bus stops where a significant demand variation is observed while others, such as Cortés et al. (2011) and Ghaemi et al. (2017), consider the selection of switch points as a decision variable of the short-turning problem.

In this work, we follow the approach of Verbas and Mahmassani (2013, 2015) in determining the switch stops based on the observed variations in passenger demand. Notwithstanding, since our work focuses on generating also inter-lines (and not only sub-lines) we examine transfer stops as well because such stops can be used for interlining without inducing additional deadheading times. An illustration of potentially generated sub-lines and inter-lines based on the switch points is presented in Fig. 2.

Given the fact that some transfer stops might be very close to bus stops where a significant variation of passenger demand is observed, for each bus line $l \in L$ with stops sequentially numbered as $S_l = \{1, 2, ..., s, ..., |S_l|\}$ if $s \in S_l$ is a switch stop and other bus stops in close vicinity of stop s are also potential switch stops, then the bus operator is inclined to merge them into one representative switch point stop for simplifying the practical implementation of short-turning and interlining lines. This "close vicinity" can be defined on a case-by-case basis based on the specific settings and the preferences of the bus operator. For instance, if a bus operator is willing to exclude a_1 preceding stops ($s - a_1$, $s - a_1 + 1$, ..., $s - 1$) and a_2 following stops ($s + 1$, ..., $s + a_2 - 1$, ..., $s + a_2$) of a switch stop s from the set of switch stop candidates because they are too close to stop s, then a set $A_s = (s - a_1, ..., s - 1, s, s + 1, ..., s + a_2)$ can be used for excluding such bus stops from further consideration. In the boundary case where the switch point stop is $s = 1$, then there is no stop preceding stop $s = 1$ and the set of excluded switch point stop candidates is $A_s = (s, s + 1, s + 2, ..., s + a_2)$. Note that stop s is excluded because it is already a switch point stop. The other boundary case where the switch point stop is the last stop at the end of the line is solved following a similar approach.

To generalize, we include boundary conditions in set A_s by defining the following dummy variables:

$$a_1' = \begin{cases} a_1 & \text{if } s - a_1 \geqslant 1 \\ s - 1 & \text{otherwise} \end{cases} \tag{1}$$

and

$$a_2' = \begin{cases} a_2 & \text{if } s + a_2 \leqslant |S_l| \\ |S_l| - s & \text{otherwise} \end{cases} \tag{2}$$
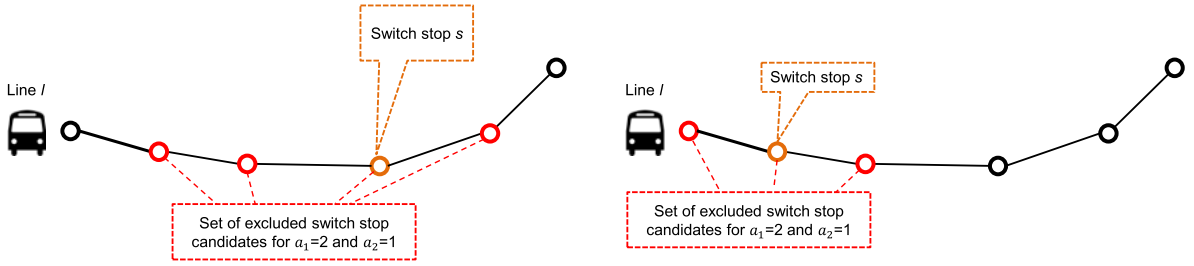
Fig. 3. [Left] The set of excluded switch point stop candidates around switch point stop s when $a_1' = 2$ and $a_2' = 1$ is $A_s = \{s - 2, s - 1, s, s + 1\}$; [Right] The set of excluded switch point stop candidates around switch stop s when $a_1' = s - 1 = 2 - 1 = 1$ and $a_2' = 1$ is $A_s = \{s - 1, s, s + 1\}$.

To incorporate the boundary conditions, set A_s becomes $A_s = (s - a_1', ..., s + a_2')$. An illustrative example of this is provided in Fig. 3.

This ad hoc rule helps to reduce the number of switch points without affecting significantly the final outcome (i.e., short-turning lines that perform short-turns at neighboring stops are not expected to perform much differently).

In addition to the above, we establish the following assumptions for (a) the determination of the switch points and (b) the generation of potential sub/inter-lines:

(1) All transfer stops are considered as potential switch points. Bus stops where a significant ridership change is observed (i.e., bus stops at which the on-board passenger change is greater than a pre-defined percentage of $z\%$) are also considered as potential switch points.
(2) Neighboring bus stops, A_s, of a switch stop s that belong to the same line cannot be considered as switch points.
(3) Interlining connections are required to return to the origin station after completing their trip (as illustrated in Fig. 2).
(4) Interlining lines can serve segments of at most two originally planned bus lines.
(5) Any interlining line which serves segments of two originally planned lines cannot have a total trip travel time which exceeds a pre-defined limit of y minutes (which may be defined by the transit agency and prevents the generation of excessively long interlining lines).
(6) Lengthy deadheading times may not be allowed by transit agencies; thus, an upper limit of k minutes for total deadheading times is applied for each of the virtual lines.

Furthermore, this work is situated at the tactical planning stage where the round-trip travel times of bus trips which are used for allocating buses to originally planned and short-turning/interlining lines are based on historical values. Such values contain implicit information on congestion. In future work, our methodology can be expanded to online resource reallocation (i.e., in short-term horizons), by integrating information from the road traffic.

Before proceeding further into the analysis of the problem, the notation presented in Tables 1 and 2 is introduced.

3.2. Generating the set of switch stops

Using the above notation and the rules described in assumptions (1)–(2), an exhaustive, rule-based graph search is devised for determining the switch points of the bus network. The rule-based graph search for determining the switch points is presented in Algorithm 1.

Algorithm 1. Rule-based graph search for determining the switch points

```
 1: function Rule-based graph search
 2:   Initialize an empty set of switch point stops S* ← ∅;
 3:   for each originally planned line l ∈ L_o do  4:     for each bus stop s ∈ {2, ..., |S_l| − 1} do
 5:       if bus stop s is a transfer stop and s ∉ S* ∧ s ∉ S' then
 5:         Set S* ← S* ∪ {s};
 7:         for each neighboring stop s' ∈ A_s do
 8:           if s' ∈ S* then
 9:             S* ← S*\{s};
10:           end if
11:         end for
12:       end if
13:       if the on-board occupancy r_{l,s} varies by more than z% from r_{l,s−1} then
14:         if s is not an excluded switch point candidate and s ∉ S* then
15:           Set S* ← S* ∪ {s}
16:         end if
17:       end if
18:     end for
19:   end for
20: end function
```

Table 1
Nomenclature (1/2).

$\{L, S\}$	is a bus network with $L = \{1, 2, ..., |L|\}$ bus lines including original and virtual lines. Virtual lines represent sub-lines and inter-lines of the originally planned ones;
$L_o = \{1, 2, ..., |L_o|\}$	is the set of the originally planned lines;
$S = \{1, 2, ..., |S|\}$	is the set of stops of the bus network;
$S_l = \{1, 2, ..., |S_l|\}$	a set denoting the bus stops of line $l \in L$ in a sequential order starting from the first stop;
$S' \subset S$	set of stops that cannot be used as switch points due to regulatory or operational constraints;
$\mathbf{T} \in \mathbb{R}_+^{|S| \times |S|}$	a $|S| \times |S|$ dimensional matrix where each $t_{i,j} \in \mathbf{T}$ denotes the planned travel time between the bus stop pair i, j including the dwell time component (boarding and alighting times) at stop j;
$\mathbf{U} \in \mathbb{R}_+^{|S| \times |S|}$	a $|S| \times |S|$ dimensional matrix where each $u_{i,j} \in \mathbf{U}$ denotes the planned travel time between the bus stop pair i, j excluding the dwell times for boarding/alighting (utilized for estimating the deadheading times);
$\mathbf{r} \in \mathbb{R}_+^{|L|}$	vector where each $r_l \in \mathbf{r}$ denotes the total round-trip time required for completing the round-trip of line $l \in L$ in hours;
$\mathbf{n} \in \mathbb{R}_+^{|L|}$	vector where each $n_l \in \mathbf{n}$ denotes the number of buses required for operating line $l \in L$ for a given frequency f_l;
$\mathbf{f} \in \mathbb{R}_+^{|L|}$	vector where each $f_l \in \mathbf{f}$ denotes the frequency of bus line $l \in L$ in vehicles per hour (note: $f_l = \frac{n_l}{r_l}, \forall l \in L$);
$\mathbf{h} \in \mathbb{R}_+^{|L|}$	vector where each $h_l \in \mathbf{h}$ denotes the dispatching headway of bus line $l \in L$ (note: $h_l = \frac{60 min / h}{f_l}, \forall l \in L$);
$\mathbf{B} \in \mathbb{N}^{|L_o| \times |S| \times |S|}$	a matrix where each $b_{l_o,i,j} \in \mathbf{B}$ denotes the passenger demand between each pair of bus stops i, j for each originally planned line $l_o \in L_o$;
$\mathbf{D} \in \mathbb{N}^{|L_o| \times |S|}$	a matrix where each $d_{l_o,s} \in \mathbf{D}$ denotes the average on-board occupancy for the segment starting at stop s for an originally planned line $l_o \in L_o$;
$\delta_{l,l_o,i,j}$	a dummy variable where $\delta_{l,l_o,i,j} = 1$ if line $l \in L$ is able to serve the passenger demand $b_{l_o,i,j}$ and $\delta_{l,l_o,i,j} = 0$ if not;
γ	a constant denoting the total number of available buses (note: $\sum_{l \in L} n_l \leqslant \gamma$ for ensuring that the total number of buses utilized from all lines $l \in L$ is within the allowable number of buses);
$\mathbf{O} \in \mathbb{R}_+^{|L_o| \times |S| \times |S|}$	a matrix where each $O_{l_o,i,j} \in \mathbf{O}$ denotes the passenger-related waiting cost for every Origin–Destination (OD) pair of the originally planned line l_o;
$\mathbf{e}$	an $|L|$-valued vector of dummy variables where $e_l = 1$ denotes that at least one vehicle has been assigned to bus line $l \in L$ and $e_l = 0$ denotes that no vehicles are assigned to that line (in such case, $n_l = 0$);

The 5-th line in Algorithm 1 states that if a stop s is a transfer stop, it does not belong already to the set of switch points and does not belong to the set of stops that cannot be used as switch points due to regulatory constraints; then, it can be added to the set of switch points. After this, it is checked whether there are any neighboring stops of the examined bus stop, s, that are already allotted to the switch points' set and, if this is the case, bus stop s is excluded from the set of switch stops (lines 7–11 of Algorithm 1).

A bus stop s can also be a switch point even if it is not a transfer stop as described in lines 13–17 of Algorithm 1. In more detail, if bus stop s is not yet a switch point and the ridership change between stop s and $s + 1$ is more than $z\%$; then, this bus stop can be added to the switch points' set. Before adding bus stop s to the switch points' set, the algorithm checks whether (a) bus stop s is not already in the set S^* and (b) bus stop s is not an excluded switch point candidate (these requirements are expressed in the 14th line of Algorithm 1).

One should note that the number of switch points that are generated through this process is not fixed a priori and it can vary based on the value of $z\%$ that determines the threshold value of ridership change upon which a bus stop can be considered as a candidate for the switch points' set. This flexible formulation allows transit agencies to control the generation of sub-lines and inter-lines by reducing or increasing the number of potential switch point stops according to their preferences. Once the value of $z\%$ is determined, the deterministic rule-based graph search of Algorithm 1 will be executed. The proposed algorithm always returns a unique solution (the computed set of switch stops is unique and the rule-based search of Algorithm 1 prioritizes always the same solution based on the above-mentioned rules even if multiple solutions with different switch stop sets are equally good).

Table 2
Nomenclature (2/2).

|	
ψ	a percentage denoting the lowest bound for the number of buses that should be allocated to the originally planned lines;
η	a constant denoting the total number of virtual lines that can be operational (i.e., operated by at least one bus);
k	maximum allowed limit of deadheading times for each virtual line (min);
y	maximum total trip travel time for inter-lining lines (min);
Q	discrete set of values from which one can select the number of buses allocated to an originally planned line;
Q'	discrete set of values from which one can select the number of buses allocated to a virtual line;
z	a percentage beyond which a change in passenger ridership (i.e., on-board occupancy) between two consecutive bus stops can justify the generation of sub/inter-lines;
β_1	unit time value associated with the passenger-related waiting time cost (€/h);
β_2	unit time value associated with the total vehicle travel time for serving all lines (€/h);
β_3	unit time value associated with the depreciation cost of using an extra bus (€/bus);
S^*	the set of the generated switch points (note: $S^* \subset S \wedge S^* \cap S' = \varnothing$);
τ	the planning period, a constant.
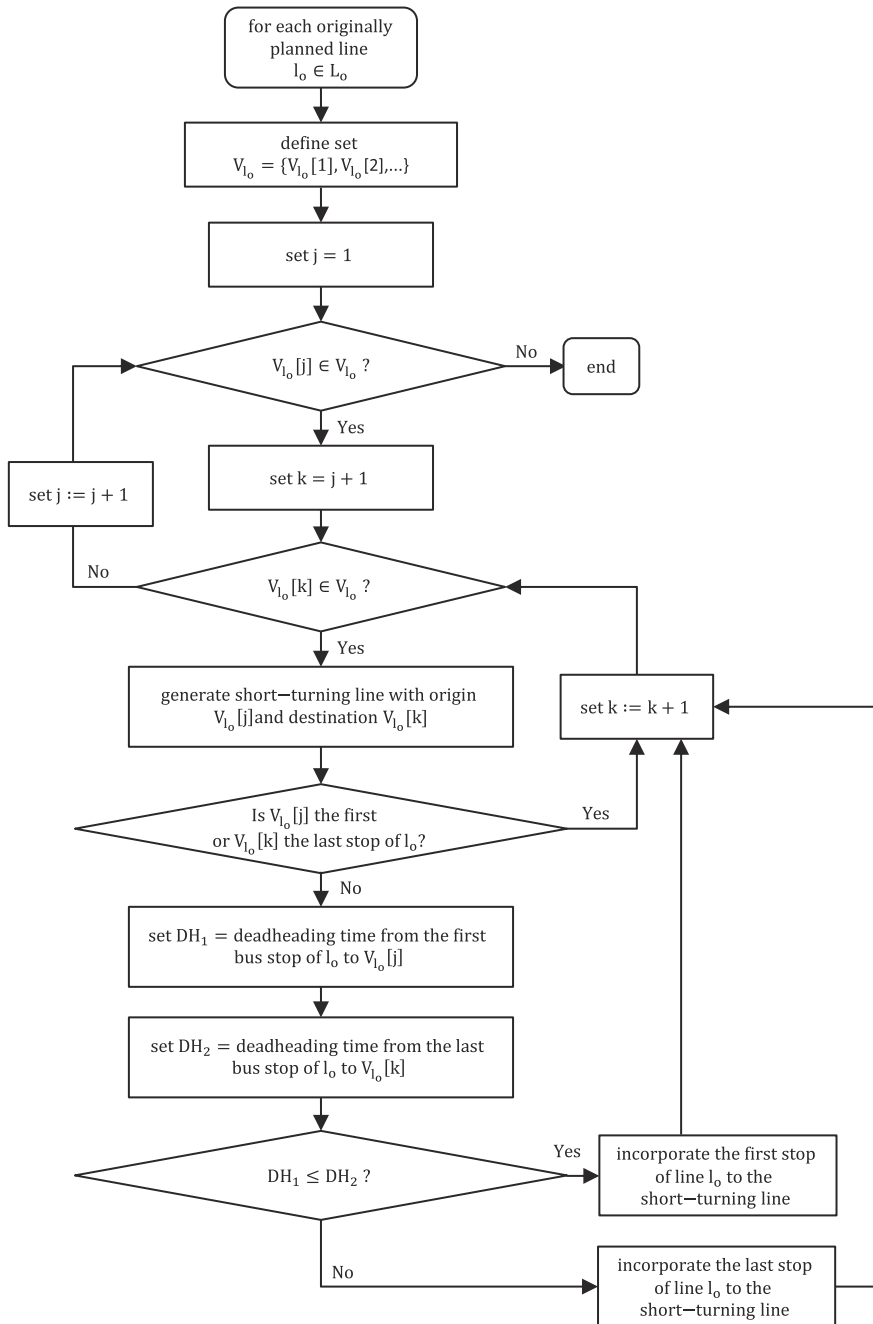
Fig. 4. Process of generating short-turning lines at specific switch points.

3.3. Generating candidate short-turning and interlining lines

Given the switch points determined by Algorithm 1, short-turning and interlining lines are generated using an exhaustive graph search. For generating short-turning lines, for each originally planned line, $l_o \in L_o$, we define as set of V_{l_o} the set that contains the first and last stop of line l_o and all switch point stops that are served by line l_o. Each short-turning line is generated by considering a pair of stops that belong to the set V_{l_o} as the origin and destination of that short-turning line. In case that the origin and destination bus stops of a short-turning line are neither the first nor the last stop of the corresponding originally planned line, then a deadhead is needed after the completion of each trip to allow bus drivers to rest at one of the two terminals of the originally planned line before starting their next trip. The automated procedure for generating short-turning lines based on the switch point stops is detailed in the flow diagram of Fig. 4.
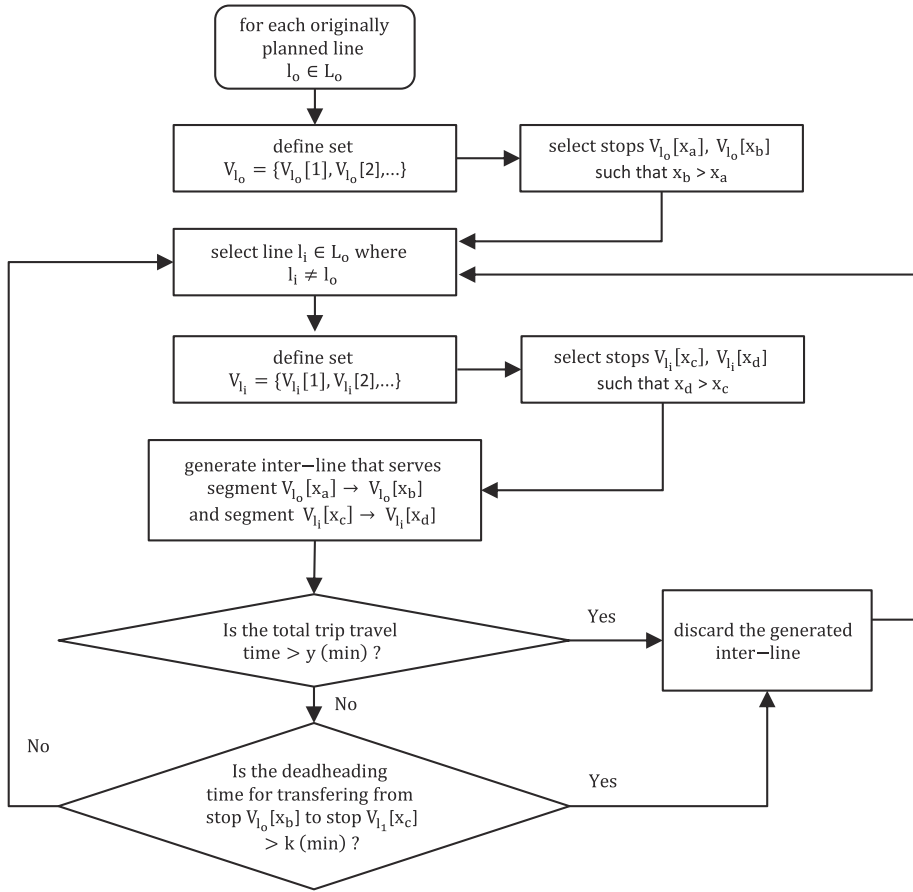
Fig. 5. Process of generating inter-lining lines at specific switch points.

From the flow diagram of Fig. 4, one can note that the process starts from the first stop of each originally planned line and new short-turning lines are generated by using as destination stop each switch point stop which belongs to that originally planned line. The procedure continues until all stops that belong to the set V_{l_o} are used as destination stops for generating new short-turning lines. After that, a new stop from the set V_{l_o} is used as a first stop from which we generate short-turning lines and the procedure continues until exhausting the set of stops that belong to V_{l_o}.

The process of generating inter-lining lines involves further steps for finding routes that serve segments of two originally planned lines. If an inter-line serves segments of two originally-planned lines and the transfer occurs at a transfer stop between those lines, then the inter-line does not incur any deadheading costs. In any other case, an inter-line induces a deadheading cost for transferring from one originally planned line to another. Following assumption (4) which states that an inter-lining line should serve segments of two originally planned lines, assumption (5) which states that the total trip travel time of an inter-line should not exceed a maximum time limit of y minutes and assumption (6) which states that the incurred deadheading time of a generated virtual line should not be greater than k minutes, the potential inter-lines of a bus network are generated via a rule-based enumeration as presented in the flow diagram of Fig. 5.

3.4. Vehicle allocation and frequency determination

The vehicle allocation problem to originally planned and virtual lines is formulated considering the inherently contradictory objectives of reducing the waiting cost of passengers at bus stops and reducing the operational costs. The operational costs are expressed in the form of (a) vehicle running times and (b) depreciation costs for each extra vehicle allocated to the bus network. In this work, we formulate a single, compensatory objective function by introducing the weight factors, $\beta_1, \beta_2, \beta_3$ that convert the passengers' waiting costs and the operational costs into monetary values.

Given that the dummy variable $\delta_{l,l_o,i,j}$ denotes whether a bus line $l \in L$ serves the passenger demand $b_{l_o,i,j}$ or not, the joint headway of all lines serving the i, j demand pair of the originally planned line $l_o \in L_o$ is:

$$\left[\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right]^{-1}$$

(3)

In addition, if each $O_{l_o,i,j} \in \mathbf{O}$ denotes the passenger-related waiting cost for each OD pair of the originally planned line l_o and passenger arrivals at stops are random (an assumption that is commonly used for high-frequency services Osuna and Newell (1972)); then,

$$O_{l_o,i,j} = \frac{b_{l_o,i,j}}{2} \left[\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right]^{-1} \tag{4}$$

The decision variables of the optimization problem are the number of buses $\mathbf{n} = (n_1, n_2, ..., n_L)$ that can be allocated to each line $l \in L$. In addition, bus operators have to conform to a set of constraints. First, the total number of allocated buses to all lines, $\sum_{l \in L} n_l$, should not exceed the number of available buses γ:

$$\sum_{l \in L} n_l \leqslant \gamma \tag{5}$$

Furthermore, a minimum percentage $\psi\%$ of the total number of available buses should be allocated to the originally planned lines to ensure a minimum level of service for the originally planned lines. This constraint is introduced because in many cases the bus operators have a contractual commitment for operating at least a number of buses at the original lines:

$$\sum_{l \in L_o} n_l \geqslant \psi\gamma \tag{6}$$

In addition, in this study the average waiting of passengers is constrained by an upper threshold value Θ to ensure that the bus operator does not reduce the operational costs to such an extent that the quality of service for passengers is significantly compromised:

$$\sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} \frac{b_{l_o,i,j}}{2} \left(\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right)^{-1} \bigg/ \sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} (b_{l_o,i,j}) \leqslant \Theta \tag{7}$$

Finally, it is possible to set the lowest and highest bounds for the number of buses that can be allocated to the original and virtual lines. The number of buses n_l that are allocated to each original line L_o can take values from an admissible set Q and the buses that are allocated to virtual lines $L - L_o$ can take values from another set Q' since the original and virtual lines can have different distinct core requirements. For instance, all originally planned lines should be operational and a minimum number of buses should be allocated to them. In contrast, virtual lines that do not improve the service might not be used; thus, the set Q' permit refraining from assigning any vehicles to a virtual line.

The sets Q and Q' can be defined by the bus operator according to the lowest and highest frequency that is permitted for each virtual and original line. For instance, some virtual lines might be set to have a frequency value equal to zero (inactive virtual lines) whereas all originally planned lines might need to have a frequency of at least three vehicles per hour to satisfy service requirements.

The resulting optimization program considering the passengers' waiting times and the operational costs is:

$$\underset{\mathbf{n}}{\operatorname{argmin}} f(\mathbf{n}) := \beta_1 \left(\sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} \frac{b_{l_o,i,j}}{2} \left(\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right)^{-1} \right) + \beta_2 \left(\sum_{l \in L} n_l \eta \lfloor \frac{\tau}{\eta} \rfloor \right) + \beta_3 \left(\sum_{l \in L} n_l \right) \tag{8}$$

$$\text{subject to:} \quad c_1(\mathbf{n}) := \left(\sum_{l=1}^{L} n_l \right) - \gamma \leqslant 0 \tag{9}$$

$$c_2(\mathbf{n}) := \psi\gamma - \sum_{l \in L_o} n_l \leqslant 0 \tag{10}$$

$$c_3(\mathbf{n}) := \frac{\sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} \frac{b_{l_o,i,j}}{2} \left(\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right)^{-1}}{\sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} (b_{l_o,i,j})} - \Theta \leqslant 0 \tag{11}$$

$$n_l \in Q, \forall\, l \in L_o \tag{12}$$

$$n_l \in Q', \forall\, l \in L - L_o \tag{13}$$

$$\eta \geqslant \sum_{l \in L - L_o} e_l \tag{14}$$

The first term of the objective function computes the waiting times of passengers at all stops for a given allocation of $\mathbf{n}$ vehicles to originally planned and virtual lines. The second term computes the total vehicle running times for serving all bus lines within a planning period τ where the round-trip travel time η of any line $l \in L$ contains the required layover times (i.e., deadheading and resting times of drivers). Finally, the third term corresponds to the depreciation costs when using $\sum_{l \in L} n_l$ vehicles.

The inequality constraint of Eq. (9) ensures that the total number of allocated vehicles to originally planned and virtual lines,

$\sum_{l \in L} n_l$, should not exceed the number of available buses, γ. The inequality constraint of Eq. (10) denotes that at least a percentage $\psi\%$ of the total number of available vehicles, γ, should be allocated to the originally planned lines $l \in L_o$.

The inequality constraint of Eq. (11) introduces an upper limit, Θ, to the average waiting time per passenger ensuring that solutions which yield significantly longer passengers' waiting times are not considered even if they reduce the operational costs. Eq. (12) and (13) ensure that the number of buses allocated to each line is selected from a discrete set of values determined by the transit agency. Finally, the inequality constraint of Eq. (14) ensures that the number of operational virtual lines, $\sum_{l \in L - L_o} e_l$, does not surpass the maximum allowed number of operational virtual lines, η.

The above constrained optimization problem of allocating buses to originally planned and virtual lines has a fractional, nonlinear objective function and one fractional constraint together with other linear constraints. In addition, the problem of allocating buses to lines is an integer programming problem since the number of buses that can be allocated at each originally planned or virtual line is a discrete variable.

Lemma 3.1. *The exploration of the entire solution space for finding a globally optimal solution for the vehicle allocation to originally planned and short-turning/interlining lines has an exponential computational complexity.*

Proof. To each bus line $l \in L$ we can allocate any number of vehicles that belongs to the set Q if l is an originally planned line or Q' if it is a virtual one. If we have two bus lines (i.e., two originally planned lines) the number of potential combinations for the allocation of buses is $|Q|^2$. Let $|Q^*|$ be the minimum of $|Q|$ and $|Q'|$. Then, evaluating the performance of all potential combinations of allocated buses to L lines requires at least $|Q^*|^L$ computations. Therefore, the solution space increases exponentially with the number of lines (regardless whether they are originally planned or virtual lines) prohibiting an exhaustive search of a globally optimal solution even for small-scale scenarios.

Given that we cannot explore the solution space exhaustively, other exact optimization methods can be considered. Because of the fractional, nonlinear objective function, our problem cannot be solved with linear or quadratic programming methods. An alternative is the use of sequential quadratic programming which starts from an initial solution guess and can find a local optimum of the mathematical program by solving its continuous relaxation. Then, the results from the sequential quadratic programming can be combined with a branch-and-bound method for converging to a discrete solution based on the lower and upper bounds derived from the sequential quadratic programming method. This approach though has two disadvantages. First, the enumeration tree of the branch-and-bound method can grow in an unsustainable manner if the decision variables are too many (which is the case when we allow the allocation of buses to a vast number of virtual lines) resulting in a computationally intractable problem. Second, there is no guarantee that the local optimum computed at each iteration by the sequential quadratic programming method is a globally optimal solution because this depends on the convexity of the objective function. We therefore develop an approximation of the combinatorial, constrained optimization problem as detailed in the following section.

4. Solution method

Given the computational intractability of the proposed bus allocation optimization problem, a solution method is introduced based on the approximation of the constrained bus allocation optimization problem by an unconstrained one which can be solved with the use of evolutionary optimization for obtaining an improved solution.

4.1. Approximating the constrained vehicle allocation problem using exterior point penalties

The constrained bus allocation optimization problem of Eq. (8)–(14) can be simplified by using a penalty method which yields an unconstrained formulation. This approximation is structured such that its minimization favors the satisfaction of the constraints through prescribing a high cost for any constraint violation Bertsekas (1990). Given the highly constrained environment within which service providers operate, we introduce exterior penalties so that the satisfaction of constraints is prioritized.

By introducing a penalty function, $\wp(\mathbf{n})$, which approximates the constrained optimization problem of Eq. (8)–(14), the following unconstrained one is obtained:

$$
\begin{aligned}
\underset{\mathbf{n}}{\text{argmin}}\ \wp(\mathbf{n}) &:= \quad f(\mathbf{n}) + w_1(\min[-c_1(\mathbf{n}), 0])^2 + w_2(\min[-c_2(\mathbf{n}), 0])^2 + w_3(\min[-c_3(\mathbf{n}), 0])^2 \\
\text{subject to:} &\quad n_l \in Q, \forall\, l \in L_o \\
&\quad n_l \in Q', \forall\, l \in L - L_o \\
&\quad \eta \geqslant \sum_{l \in L - L_o} e_l
\end{aligned}
\tag{15}
$$

where w_1, w_2 and w_3 are used to penalize the violation of constraints and are positive real numbers with sufficiently high values to ensure that priority is given to the satisfaction of constraints. The penalty function $\wp(\mathbf{n})$ is equal to the score of the objective function $f(\mathbf{n})$ if at some point we reach a solution $\mathbf{n}$ for which $w_1(\min[-c_1(\mathbf{n}), 0])^2 + w_2(\min[-c_2(\mathbf{n}), 0])^2 + w_3(\min[-c_3(\mathbf{n}), 0])^2 = 0$, indicating that all constraints are satisfied for such solution. The penalty terms are added to the objective function of the constrained optimization problem and dictate that if a constraint $c_i(\mathbf{n})$ has a negative score, then $\min[-c_i(\mathbf{n}), 0] = -c_i(\mathbf{n})$ and the constraint is violated for the current set of variables $\mathbf{n}$. In that case, the objective function $f(\mathbf{n})$ is penalized by the term $w_i(-c_i(\mathbf{n}))^2$ where the weight factor w_i expresses the violation importance of this constraint in relation to all others.

Formulating the penalty function $\wp(\mathbf{n})$ ensures that violating constraints $c_i(\mathbf{n}) < 0$ penalize progressively the penalty function by adding their squared value $c_i(\mathbf{n})^2$ to its score. Therefore, the penalty function is over-penalized if some violating constraints $c_i(\mathbf{n}) < 0$ are significantly greater than zero.

In addition, adding different weights, w_1, w_2, w_3, to the constraints is useful in the case of problem infeasibility because in such case all constraints cannot be satisfied simultaneously; therefore, with the use of different weight factor values, the bus operator can prioritize the most important constraints at the expense of others.

4.2. Solving the unconstrained problem with a problem-specific Genetic Algorithm

To solve the unconstrained optimization problem of Eq. (15) one needs to explore a vast, discrete solution space resulting in a significant computational burden. For instance, as discussed in Section 3, applying a classical exact optimization method for discrete optimization problems such as the brute-force algorithm requires an exponential number of problem evaluations in order to find a globally optimal solution.

As an alternative to classical exact optimization methods, metaheuristics from the area of evolutionary optimization can be employed. In contrast to the classical exact optimization methods, evolutionary algorithms perform fewer calculations for finding a generally good (but inexact) solution to a combinatorial optimization problem (Simon, 2013).

For combinatorial optimization problems several evolutionary optimization algorithms can be applied such as simulated annealing (Kirkpatrick et al., 1983) or tabu search (Glover, 1986). In this work, we employ a problem-specific genetic algorithm (GA) which considers a pool of solutions rather than a single solution at each iteration although other heuristic optimization methods may also be used for solving this problem.

One of the first works on GAs was the book of Holland (1975) that detailed the principal stages of a GA as: (1) encoding the initial population; (2) evaluating the fitness of each population member; (3) parent selection for offspring generation; (4) crossover; and (5) mutation. In the following sub-sections we detail the stages of the problem-specific GA that yields an (inexact) solution of the optimization problem of Eq. (15).

4.2.1. Encoding

A typical GA contains a number of strings which form the population at each of the iterations. Each string is a population member (individual) and represents a potential solution to the optimization problem. The first decision that needs to be made at the initialization stage of the GA is the population size. This parameter can be determined based on the trade-off between solution space exploration and computational cost since a GA with a larger population size is expected to conduct a more comprehensive exploration of the solution space but requires also more time for evaluating all possible solutions and performing the corresponding crossover/mutation operations.

For solving the unconstrained optimization problem of Eq. (15), an initial population P with $\{1, 2, ..., |P|\}$ members is introduced. Each population member, $\mathbf{m} \in P$, is a vector $\mathbf{m} = (m_1, ..., m_l, ..., m_{|L|})$ with $|L|$ elements (known as genes) where each element $m_l \in \mathbf{m}$ represents the number of buses allocated to the corresponding line $l \in L$ in case this solution is adopted. Each gene $m_l \in \mathbf{m}$ of an individual $\mathbf{m}$ is allowed to take an integer value from the set Q (when line l is an originally planned line) or set Q' (when line l is a subline or inter-line).

Therefore, a random initial population P can be generated as follows:

```
For m = 1 to |P|
   Introduce the mᵗʰ population member m = (m₁, ..., mₗ, ..., m|L|)
   For l = 1 to |L|
      If l ∈ L₀: mₗ ← random. choice(Q)
      If l ∈ L − L₀: mₗ ← random. choice(Q′)
   Next l
Next i
```

where $m_l \leftarrow random.\ choice(Q)$ denotes that m_l can take any value from the discrete set Q and $m_l \leftarrow random.\ choice(Q')$ denotes that m_l can take any value from the set Q'.

4.2.2. Evaluating the fitness of individuals and selecting individuals for reproduction

A GA requires only the existence of a fitness function which can be evaluated and does not consider the properties of the function such as convexity, smoothness or existence of derivatives (Bakirtzis et al., 2002). GAs are typically designed to maximize fitness. Notwithstanding, given the fact that our problem of Eq. (15) is casted as a minimization problem, in our study a population member $\mathbf{m}$ is considered more fit when its fitness function value, $\wp(\mathbf{m})$, is lower.

In the parent selection stage the fittest population members (individuals) are selected for reproduction and they pass their genes to the next generation. At each parent selection, two individuals from the population are selected where individuals with better fitness values have a higher probability of being selected for producing an offspring. This can be achieved by using the well-known roulette-wheel selection method (Goldberg and Deb, 1991). In the roulette-wheel selection method, each individual $\mathbf{m}$ has a probability of being selected which is proportional to its fitness value divided by the fitness values of all other population members.

After selecting one parent using the roulette-wheel selection method, another parent is selected with the same method and the two parents cross over to produce two offsprings. The same process is repeated until the number of parents which are selected for reproduction is the same as the population size $|P|$.

4.2.3. Crossover and mutation

At the crossover stage, two parents exchange their genes at a randomly selected crossover point selected from the set $\{1, 2, ..., |L|\}$ for generating two offsprings. For instance, if the crossover point of two parents $\mathbf{m} = (m_1, ..., m_l, m_{l+1}, ..., m_{|L|})$ and $\mathbf{m}' = (m_1', ..., m_l', m_{l+1}', ..., m_{|L|}')$ which are selected for reproduction is $l \in L$; then, the two generated offsprings will have the set of genes $(m_1, ..., m_l, m_{l+1}', ..., m_{|L|}')$ and $(m_1', ..., m_l', m_{l+1}, ..., m_{|L|})$.

After the crossover stage follows the mutation stage. The mutation can be potentially applied to any generated offspring after the crossover stage to facilitate the exploration of new information that is not contained in the pair of parents that were used at the crossover stage. In our case, we specify a small probability, p_c, for replacing each gene of the generated offspring with a random value from the set Q if that gene corresponds to an originally planned line and set Q' if it corresponds to a virtual one.

The procedure described above continues iteratively until a pre-determined number of population generations, μ^{max}, is reached. The population member with the best performance is then selected as the final solution and its genes represent the number of buses that should be allocated to each original or virtual line, where, for many virtual lines, this number can be equal to zero (resulting in inactive virtual lines). This procedure is summarized in Algorithm 2.

In Algorithm 2, lines 10–11 denote the parent selection step according to the roulette-wheel approach, line 12 is the crossover step that produces two new offsprings and lines 13–22 express the mutation step for each newly generated offspring. In lines 13–22 one can note that a mutation occurs if a randomly selected number from the continuous set $[0, 1]$ is lower than the mutation probability p_c.

Algorithm 2.

1: **function** Genetic Algorithm search
2:   Initialize a random population $P = \{1, 2, ..., |P|\}$ where each population member $\mathbf{m} \in P$ has $|L|$ genes;
3:   **for** each population member $\mathbf{m} \in P$ **do**
4:     Calculate its fitness: $- \wp(\mathbf{m})$;
5:   **end for**
6:   Initialize the counter of generation evolutions as generation$\leftarrow$ 1;
7:   **while** generation $\leqslant \mu^{max}$ **do**
8:     Initialize the population of the next generation $P' = \varnothing$
9:     **while** $|P'| < |P|$**do**
10:       Select one parent $\mathbf{m} \in P$ using the roulette-wheel method;
11:       Select another parent $\mathbf{m}' \in P$ where $\mathbf{m}' \in P\backslash\{\mathbf{m}\}$ using the roulette-wheel method;
12:       Exchange the genes of parent $\mathbf{m}$ and $\mathbf{m}'$ at a randomly selected crossover point $l \in L$ and generate two offsprings $(m_1, ..., m_l, m_{l+1}', ..., m_{|L|}')$ and
         $(m_1', ..., m_l', m_{l+1}, ..., m_{|L|})$;
13:       **for** each one of the two offsprings **do**
14:         **for** each $l \in L$
15:           If $l \in Lo$ and $random. choice([0, 1]) < p_c$ **then**
16:             Replace the value of the lth gene of the offspring with $random. choice(Q)$;
17:           **end if**
18:           if $l \in L - Lo$ and $random. choice([0, 1]) < p_c$**then**
19:             Replace the value of the lth gene of the offspring with $random. choice(Q')$;
20:           **end i**
21:         **end for**
22:       **end for**
23:       Expand set P' by adding the two generated offsprings to it;
24:     **end while**
25:     Replace the previous generation with the new one: $P \leftarrow P'$;
26:     Update the number of generation evolutions: generation $\leftarrow$ generation + 1;
27:   **end while**
28:   **return** the fittest population member;
29: **end function**

The number of population members $|P|$, the mutation rate, p_c, and the maximum number of population generations, μ^{max}, are parameters of the GA which should be externally defined and can affect the performance of the computed solution. For this reason, several scenarios with different parameter options can be conducted for increasing the probability of finding a solution which is more close to a globally optimal one.

5. Numerical experiments

5.1. Case study description

The proposed methodology for the allocation of buses to originally planned and virtual lines is tested for the bus network of The Hague. The Hague is a mid-sized European city and its bus network consists of $|L_o| = 8$ originally planned urban bus lines,
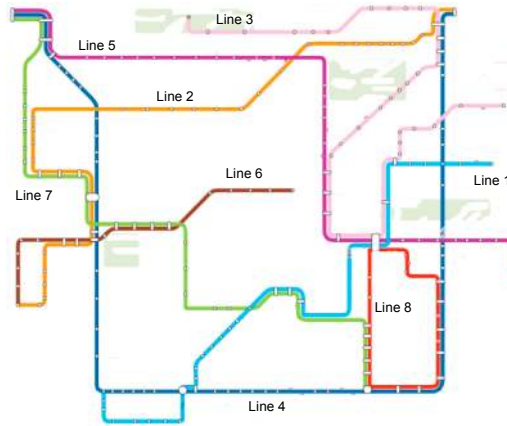
Fig. 6. Illustration of the bus network in The Hague.

complementing and interfacing with the tram network. The originally planned bus lines cover a compact geographical area that enables the generation of several interlining lines without requiring long deadheading times. As presented in Fig. 6, seven of the bus lines are bi-directional and one is circular (bus line 8). The circular line serves two of the main train stations in the city and all bus lines operate under high frequencies since they serve the central city. For ease of reference, the eight bus lines in the Hague are named 1, 2, …, 8. The actual identification numbers of the eight bus lines can be found at: https://www.htm.nl/media/498240/17066htm_ a4haltekrttrambus_va01juli17_web.pdf.

In our case study, we analyze a 6-h period of the day that was empirically found to exhibit a relatively stable ridership pattern (from 07:00 to 13:00). The total number of available buses for operating the service trips from 07:00 to 13:00 is $\gamma = 220$. For the optimal allocation of buses to the eight originally planned bus lines, we use the parameter values summarized in Table 3.

For such parameter values, the optimal allocation of buses to originally planned lines is the one that minimizes the value of the penalty function $\wp(\mathbf{n})$ in Eq. (15). Because the mathematical program in Eq. (15) considers also virtual lines and we want to allocate buses to originally planned lines only, we exclude all virtual lines by enforcing the total number of virtual lines that can be operational, η, to be equal to zero. In such case, a solution to the mathematical program of Eq. (15) represents an optimal allocation of buses to originally planned lines only.

After applying the GA presented in Algorithm 2 for finding an optimal bus allocation to the 8 originally planned lines, the solution with the lowest total cost is presented in Table 4. This bus allocation: (a) requires a total bus travel time of 21,616 min (360.26 h); (b) results in an average waiting time of $\simeq 1.78$ minutes per passenger; and (c) requires the use of $111 + 88 = 199$ buses out of the 220 available ones.

It is important to indicate that because of the parameter values of Table 3 the optimal allocation of buses to originally planned lines uses only $111 + 88 = 199$ out of the 220 available buses. The reasons behind this are the high vehicle running time and depreciation costs that favor the use of less resources.

5.2. Allocating buses to short-turning and interlining lines

In this study, we used detailed smartcard data logs from 24 weekdays in order to analyze the spatio-temporal passenger demand variation from 07:00 until 13:00. The smartcard logs contain information about the origin and destination station of each passenger that used one of the eight originally planned lines in The Hague during the analysis period (2nd of March 2015–2nd of April 2015). The smartcard logs are used for constructing passenger OD matrices per bus line and were instrumental in (a) understanding the temporal variation of demand within the day for each bus line by splitting the day into 6-h periods; (b) investigating the spatial demand variations at the line level; and (c) identifying potential switch stops for generating short-turning and interlining lines based on variations in the cumulative ridership at each stop.

For instance, from the average hourly passenger load of bus line 3 from 07:00 to 13:00 which is presented in Fig. 7, one can observe that bus stops 6 and 12 are potential switch stops because of a passenger load change of more than $z = 20\%$ occurring at these

Table 3
Parameter Values for the allocation of the available buses to the eight originally planned lines of the bus network in The Hague.

γ (total number of available buses)	220
z (minimum percentage of passenger ridership change to justify the generation of a switch stop)	20%
β_1 (unit time value associated with the passenger-related waiting time cost)	4 (€/h)
β_2 (unit time value associated with the total vehicle travel time for serving all lines)	60 (€/h)
β_3 (unit time value associated with the depreciation cost of using an extra bus)	20 (€/bus)
Q (number of buses that can be allocated to an original line from 07:00 to 13:00)	{6, 7, 8, …,41}

Table 4
Round-trip times and initial bus allocation to the originally planned lines from 07:00 to 13:00.

	r_l: Round-trip time in minutes	Allocated buses		r_l: Round-trip time in minutes	Allocated buses
Line 1	108	29	Line 5	110	31
Line 2	107	22	Line 6	50	22
Line 3	112	21	Line 7	79	25
Line 4	172	39	Line 8	138	10
Total:		111			88
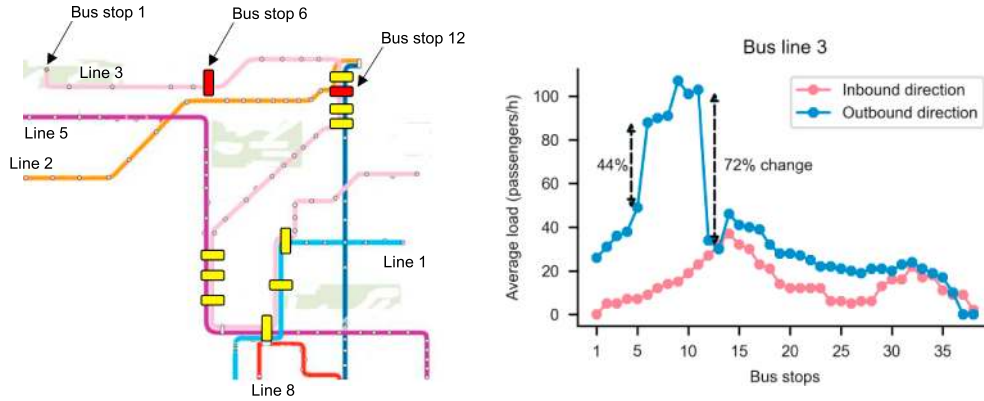
Fig. 7. Enumeration of all possible switch stops for bus line 3.

stops. In addition, the bus stops 11, 13, 14, 23, 24, 25, 27, 28 and 29 of bus line 3 which are marked with yellow are transfer stops; thus, they are also switch stop candidates. Following the steps detailed in the deterministic Algorithm 1 for $a_1 = 2$ and $a_2 = 2$, only 5 out of the 11 switch stop candidates are selected as switch stops for bus line 3 (namely, the bus stops 6, 11, 14, 23 and 27). Bus stop 11 is selected instead of bus stop 12 where a significant passenger load change is observed - bus stop 11 is a transfer stop and has a higher priority than other neighboring switch stop candidates.

The deployment of Algorithm 1 for generating the switch stops for all bus lines and the algorithms presented in Figs. 4 and 5 for generating the short-turning and interlining lines yielded 29 short-turning lines and 323 interlining lines out of 4344 possible combinations.

By allocating buses to originally planned and short-turning/interlining lines, this study investigates the potential of improving the weighted sum of Eq. (8) which consists of the (a) passenger waiting times, (b) total vehicle running times and (c) depreciation costs from the use of additional vehicles. The allocation of buses to short-turning and interlining lines is performed by using the GA presented in Algorithm 2.

When performing an optimal vehicle allocation to originally planned and virtual lines, the bus operator can determine several parameter values. In particular, the minimum percentage of buses that should be allocated to originally planned lines, ψ, and the total trip travel time limit for interlining lines, y, among others. This provides an extra flexibility to the bus operator that can tailor the use of the interlining and short-turning lines to its operational needs by adjusting the problem parameters accordingly.

Initially, we allocate buses to originally planned and short-turning lines following the scenario of Table 5 which depicts the values of the problem parameters.

Table 5
Parameter Values

γ (total number of available buses)	220
ψ (minimum percentage of buses that should be allocated to the originally planned lines)	60%
η (total number of virtual lines that can be operational)	20
k (maximum allowed limit of deadheading times for each virtual line)	20 min
y (maximum total trip travel time for inter-lining lines)	1 h 30 min
z (percentage of passenger ridership change that justifies the generation of a switch stop)	20%
Θ (upper limit of the average waiting time of passengers)	3 min
β_1 (unit time value associated with the passenger-related waiting time cost)	4 (€/h)
β_2 (unit time value associated with the total vehicle travel time for serving all lines)	60 (€/h)
β_3 (unit time value associated with the depreciation cost of using an extra bus)	20 (€/bus)
Q (number of buses that can be allocated to an original line from 07:00 to 13:00)	{6, 7, 8, …,41}
Q' (number of buses that can be allocated to a virtual line from 07:00 to 13:00)	{0, 3, 4, …,15}
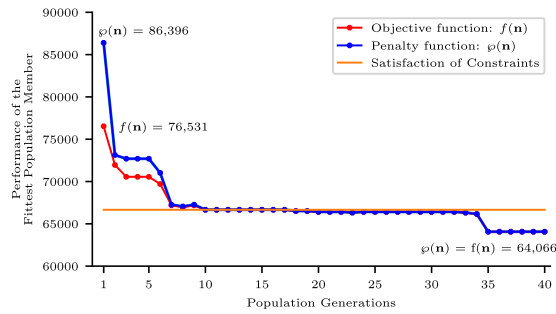
Fig. 8. Improvement of the exterior point penalty function score after a number of μ^{max} population generations. The horizontal line represents the area below which all constraints are satisfied (feasible solution space).

Using the existing service provision as the starting point, we allow the re-allocation of buses to the 8 original, L_o, and $(29 + 323) = 352$ virtual lines, $L - L_o$. Given the large number of decision variables and the combinatorial nature of the bus allocation problem, we employ the GA proposed in this study. For the implementation of the GA, we use the *Distributed Evolutionary Algorithms in Python* (Deap) package (Fortin et al., 2012). From this package, we use the eaSimple() algorithm with the hyperparameter values of $|P| = 200$ population members; $p_c = 0.2$ mutation probability; and $\mu_{max} = 40$ maximum population generations. For the evaluation of the fitness of each population member, the penalty function of Eq. (15) is programmed in Python 2.7 and the tests are implemented in a general-purpose computer with 2.40 GHz CPU and 16 GB RAM.

The GA algorithm is applied for the re-allocation of buses to originally planned and virtual lines and the convergence results are presented in Fig. 8. The goal of the convergence is the minimization of the penalty function score of Eq. (15) which is the weighted sum of the objective function and the constraint violation penalties.

The fittest population member (solution) in the initial population has a penalty function value of 86, 396€ and an objective function value of 76, 531€. The initial 9, 865€ gap between the objective and the penalty function values indicates that the solution of the fittest population member of the initial population violates some of the constraints of the bus allocation problem.

After six iterations, we reach a point where all constraints are satisfied (at this point, the penalty function value is equal to the objective function value). At this stage, the first feasible solution is obtained. Then, the iterations continue until we reach the predefined maximum number, $\mu^{max} = 40$, of allowed population generations. The fittest solution at the 40th population generation has a penalty function value of 64, 066€ and satisfies all constraints.

As discussed in the solution method section, the use of a GA for solving the problem of allocating buses to both originally planned and short-turning/interlining lines is one option. Other heuristic solution methods for discrete problems (which are appropriate for solving a problem more quickly when classic methods are too slow) can be employed. For our specific scenario, we provide results regarding the performance of other heuristic solution methods, such as simulated annealing and hill-climbing, in Table 6. In Table 6 we report the results of the best performing vehicle allocation solutions computed by different heuristic algorithms.

Notably, the hill-climbing method terminates its search significantly faster than the other methods. However, this fast convergence occurs because the algorithm is trapped in a local optimum without being able to move to another "hill" for continuing its exploration.

Using the vehicle allocation of the GA, Table 7 summarizes the potential benefits of the bus allocation optimization when buses are allocated not only to originally planned lines, but also to short-turning/interlining lines. The optimal bus allocation to both originally planned and virtual (short-turn and interlining) lines to the bus network of The Hague demonstrated a potential reduction of 13.85% in operational costs and 4.85% in the average waiting time per passenger.

The resulting bus allocation to originally planned and virtual lines using the GA is presented in Fig. 9. As expected, the lion share of the 352 virtual lines remain inactive in the solution attained as GA solution filtered out 346 out of the 352 virtual lines. This solution involves 3 interlining and 3 short-turning operations. The interlining involves a relatively small number of buses and is used to circulate buses between busy lines that have an asymmetric passenger demand. Short-turning is deployed for lines that have to be partitioned due to a noticeably uneven demand pattern.

To provide more details on the performance improvement after the introduction of short-turning and interlining lines, Fig. 10 presents the overall waiting time costs, the vehicle running costs and depreciation costs when (a) only originally planned lines are considered; and (b) when short-turning/interlining lines are also considered. In the latter case, the overall waiting time costs are reduced from 43, 436€ to 41, 445€ and the vehicle running costs from 21, 616€ to 18, 621€. Because only the depreciation costs
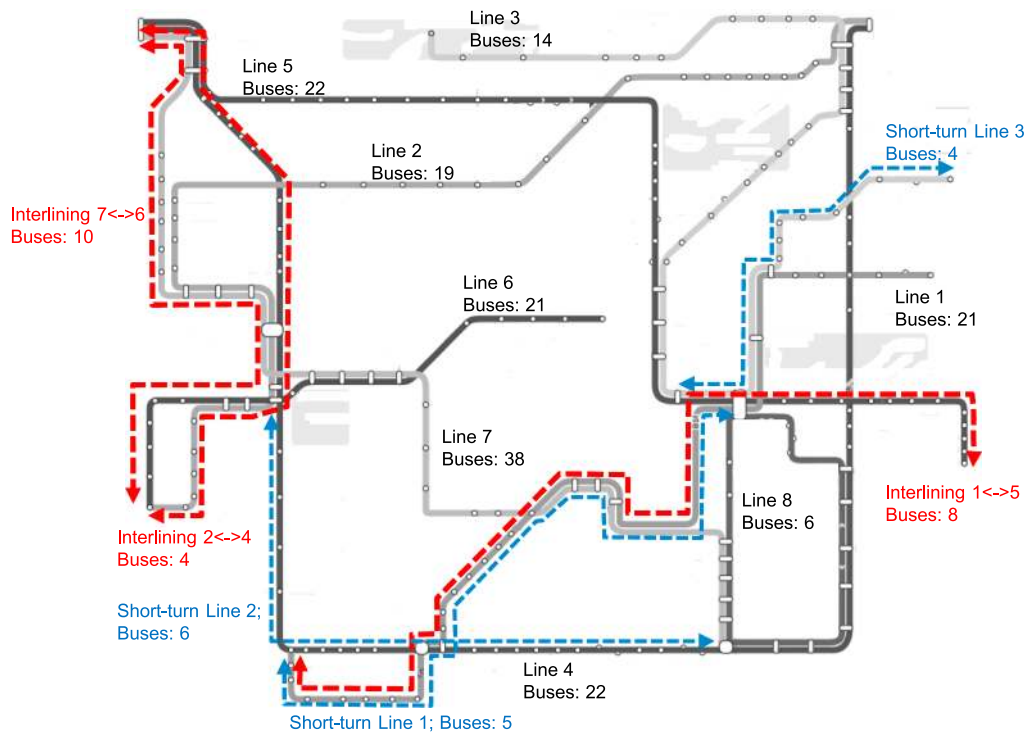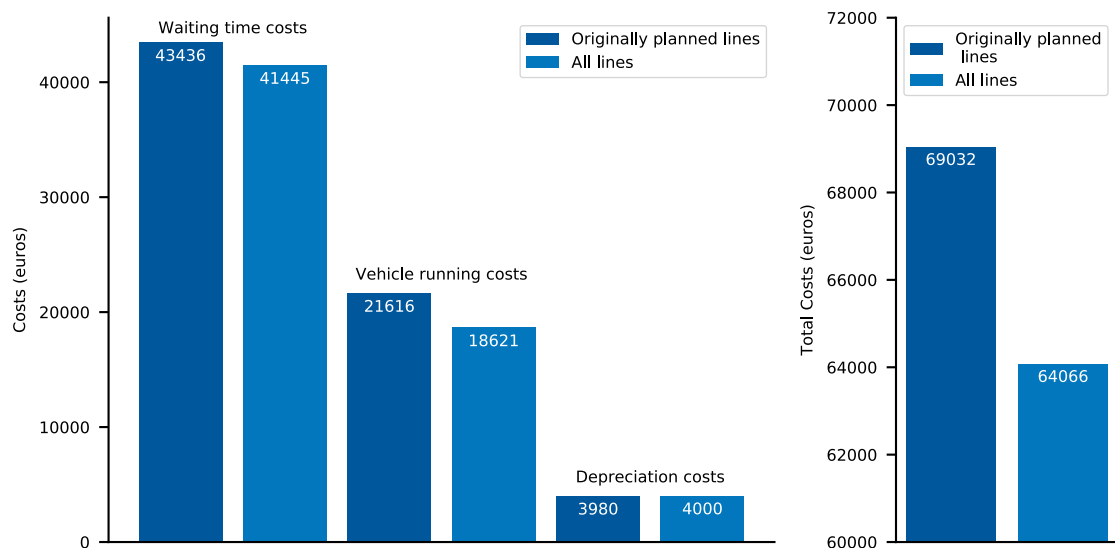
Table 6
Genetic Algorithm, Stochastic Annealing and Hill Climbing: solution approximation results

	Best-performing total cost of the computed solution	Computational time
Genetic Algorithm	64,066€	21.17 min
Hill-climbing	74,531€	3.52 min
Simulated Annealing	64,438€	24.42 min

Table 7

Performance improvement when using short-turning and interlining lines for $\beta_1 = 4$€/h, $\beta_2 = 60$€/h and $\beta_3 = 20$€/bus.

	Originally planned lines only	Considering short-turning and interlining lines	Improvement
Average waiting time per passenger	1.785 min	1.703 min	4.58%
Total running time of buses	360.26 h	310.35 h	13.85%
Number of buses	199	200	−0.50%
Total costs $f(\mathbf{n})$	69,032€	64,066€	7.19%

Fig. 9. Bus allocation to originally planned and active virtual lines.

Fig. 10. Costs when using (a) originally planned lines only and (b) originally planned lines along with interlining and short-turning lines.

Table 8

Performance sensitivity to a 10% and 25% passenger demand increase.

	Nominal passenger demand	10% passenger demand increase	25% passenger demand increase
Average Waiting Time per passenger	1.703 min	1.698	1.694 min
Total running time of buses	310.35 h	324	339.10 h
Number of used buses	200	209	220
Waiting time costs	41,445€	45,846€	50,820€
Vehicle running costs	18,621€	19,440€	20,340€
Depreciation costs	3980€	4180€	4400€
Total costs $f(\mathbf{n})$	64,066€	69,466€	75,560€

increase slightly from 3, 980€ to 4, 080€ the reduction of the total costs is 7.19% (from 69,032€ to 64,066€).

For performing a sensitivity analysis of the total costs to the changes in passenger demand, we compare the performance of the optimal bus allocation at the scenario with the nominal passenger demand against the performance of the optimal bus allocation at scenarios with 10% and 25% passenger demand increase. The results are summarized in Table 8 demonstrating an increase in the vehicle running costs and the depreciation costs due to the use of more vehicles. Note that because of the 25% increase in passenger demand, the respective bus allocation uses all 220 available buses for limiting the increase of the passenger waiting time costs.

5.3. Sensitivity analysis of cost parameters

As previously discussed, some stakeholders might place more emphasis on reducing the waiting times of passengers while others might focus on reducing the operational costs. For this reason, we investigate the trade-off between the passenger-related and the operational-related costs by modifying the values of the weight factors β_1, β_2, β_3.

First, we increase the weight factor value of β_1 which is multiplied by the passenger-related waiting time costs from 4(€/h) to 12(€/h) and 20(€/h) while the weight factor values of β_2 and β_3 remain the same (β_2 remains always 60(€/h) and $\beta_3 = 20$(€/bus)). Then, for each of the three generated scenarios we allocate buses to the originally planned and short-turning/interlining lines in order to investigate the impact of changes in the passenger-related costs and the operational-related costs when the bus operator places more emphasis on reducing the overall waiting time of passengers.

The results of this analysis are summarized in Fig. 11 which presents the changes in the average waiting time per passenger, the number of running buses and the total running times of the buses.

The main observations from Fig. 11 are:

- to obtain a slight reduction of 0.41% on the average waiting time per passenger (from 1.703 min to 1.696 min), 4 additional buses are needed and the total running times of buses increase significantly (by 6.98%);
- reducing further the average waiting time per passenger from 1.696 min to 1.694 min increases the total running times of buses by 2.11% and requires 16 extra buses.

The above analysis can facilitate the decision-making process of the bus operators by providing them information regarding the required sacrifices in terms of vehicle running costs for a slight reduction of the passenger waiting times.

Second, we analyze the performance of the operations when more importance is attached to reducing bus running times. For this reason, we generate three scenarios where the weight factor β_2, which is the unit time value associated with the total vehicle travel times, increases from $\beta_2 = 60$(€/h) to $\beta_2 = 90$(€/h) and $\beta_2 = 120$(€/h) respectively. In all three scenarios the weight factors β_1 and β_3 remain the same (β_1 is always 4 (€/h) and $\beta_3 = 20$(€/bus)). The sensitivity of the passenger waiting times, the number of deployed buses and bus running times to the changes in the weight factor β_2 is presented in Fig. 12.
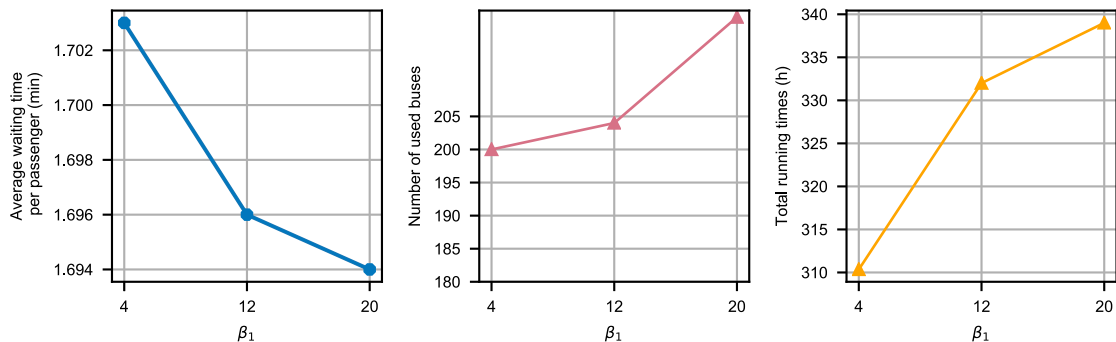
Fig. 11. Sensitivity to changes of weight factor β_1 which is related to the passenger waiting time costs.
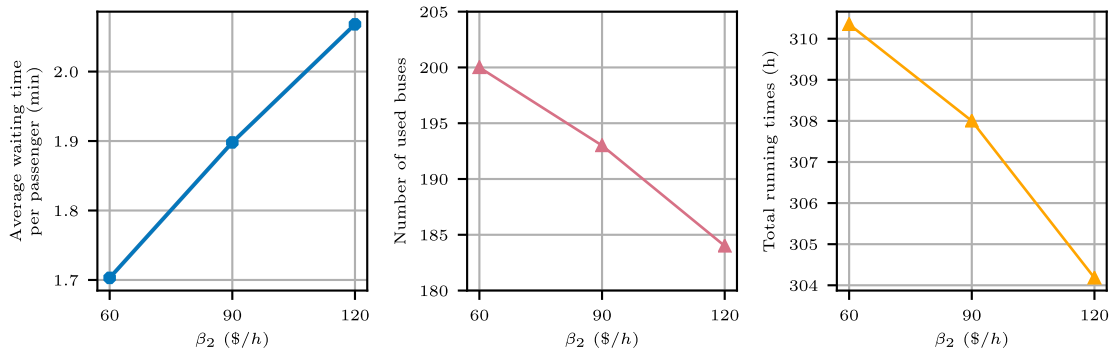
Fig. 12. Sensitivity to changes of weight factor β_2 which is related to the vehicle running time costs.

The main observations from Fig. 12 are:

- the improvement of the vehicle running costs after doubling the value of β_2 is 6.17 h in total (or 1.99%) whereas the average waiting time per passenger increases by 0.365 min or $\simeq 20\%$;
- the total number of allocated buses is reduced by 8% (16 buses);
- higher weight to the total running costs, β_2, reduces the number of used buses, but the average waiting time per passenger increases faster.

The findings from Fig. 12 dictate that bus operators should act with caution when trying to obtain solutions that reduce significantly the vehicle running costs because this can lead to increased headways and disproportionally longer passenger waiting times.

Finally, we examine the effect of the weight factor that determines the relative importance of the bus depreciation costs. In this case, we generate three scenarios with β_3 values of 20, 110 and 200(€/bus) for which the weight factors β_1 and β_2 remain the same (β_1 is always 4(€/h) and β_2 is 60(€/h)). After allocating the buses to the originally and short-turn/interlining lines for each one of the three scenarios using the GA of Algorithm 2, the results are presented in Fig. 13.

From Fig. 13 one can observe that:

- interestingly, the total running time of buses increases from 310.35 h to 315.6 h even if the number of deployed buses is reduced from 200 to 187;
- a 2% reduction of the number of used buses (from 200 to 196 buses) increases the average waiting time per passenger by 5.7 s.

The above sensitivity analysis underlines the importance of selecting adequate weight factors which do not substantially compromise the passenger-related costs for a slight reduction of the operational costs or vice versa.

5.4. Pareto Frontiers

In the previous sub-section we evaluated the sensitivity of the average waiting time per passenger, the number of used buses and the total running times to the changes of the weight factor value β_1, β_2, β_3. Following the work of Chow and Pavlides (2018), we relax such specification and we construct a Pareto Frontier of two objective functions which can be represented by a curve where each point of the curve indicates an efficient solution when those objective functions are being optimized simultaneously. First, the objective function $f(\mathbf{n})$ is reformulated as:
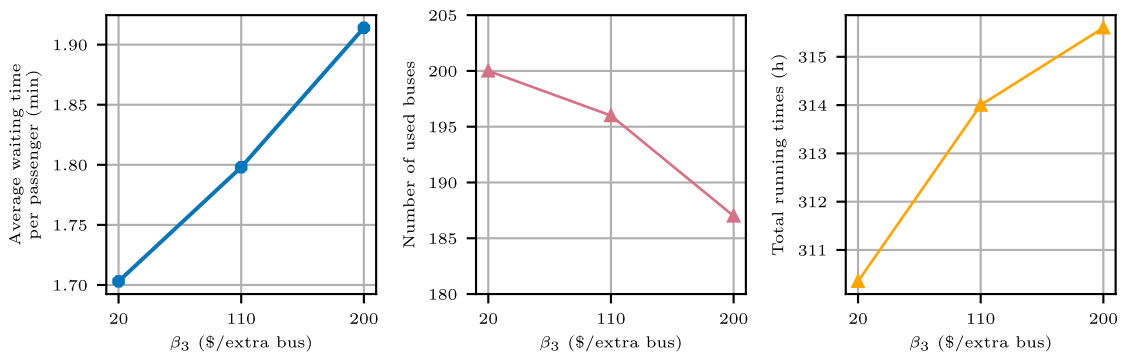

Fig. 13. Sensitivity to changes of weight factor β_3 which is related to the cost of operating each extra bus.

Table 9
Cost coefficients for deriving the Pareto frontier of passenger waiting times against depreciation costs.

Scenario	ω_1	ω_3
1	0.09	0.91
2	0.15	0.85
3	0.50	0.50
4	0.75	0.25
5	0.83	0.17

$$\operatorname*{argmin}_{\mathbf{n}} f'(\mathbf{n}) = \omega_1 \beta_1 \left(\sum_{l_o \in L_o} \sum_{i \in S} \sum_{j \in S} \frac{b_{l_o,i,j}}{2} \left(\sum_{l \in L} \delta_{l,l_o,i,j} \frac{n_\rho}{r_\rho} \right)^{-1} \right) + \omega_2 \beta_2 \left(\sum_{l \in L} n_l \eta_l \lfloor \frac{\tau}{\eta} \rfloor \right) + \omega_3 \beta_3 \left(\sum_{l \in L} n_l \right)$$

where

$$\sum_{i=1}^{3} \omega_i = 1 \tag{16}$$

In this way, we calculate the corresponding solutions with different combinations of ω_i in Eq. (16) and derive the Pareto frontier accordingly. Following this approach, the Pareto frontiers illustrate the competition between pairs of objective functions during the optimization process. Table 9 shows the set of values of cost coefficients when considering the two objective functions of passenger waiting costs and depreciation costs.

Based upon the setting in Table 9, Fig. 14 shows the corresponding Pareto frontiers for the 'passenger waiting times' (Objective 1) against the 'number of used buses' (Objective 3). Each point in Fig. 14 represents the corresponding cost values of 'average waiting time per passenger' and 'number of used buses' for each combination of ω_i in Table 9.

Fig. 14 reveals the conflicting nature between the two objectives: the value of the 'passenger waiting time' cost decreases as that of the 'depreciation cost' increases. The results reveal the non-linearity between such trade-off relationship and we can notice a stabilization in 'passenger waiting time' when the depreciation cost reaches $\simeq$€4 k. This indicates that additional buses result in a very small benefit in terms of passenger waiting times.

Considering now the Parento frontiers for the passenger waiting times against the running costs, we use the set of values in Table 10 to generate the Pareto frontiers in Fig. 15.

Again, the trade-off between the passenger waiting times and the vehicle running costs has a non-linear nature and the passenger waiting times stabilize for a vehicle running cost of around €18.6 k. After that point, more vehicle-kilometers-traveled will not reduce the passenger waiting times significantly.

5.5. Sensitivity analysis of the parameters related to the generation of virtual lines

Apart from the weight factors β_1, β_2, β_3 that control the relative importance of the waiting time, running time and depreciation costs, the parameters that control the generation of short-turning and interlining lines can affect the performance of the bus allocation solution.

For this reason, we investigate the performance changes for different values of the parameters which control the generation of short-turning and interlining lines. For example, the parameter ψ determines the minimum percentage of buses that should be allocated to the originally planned lines and its value was initially set to 60% (in the scenario of Table 5). Some bus operators might, however, be more conservative wish to ensure that at least 80 or 90% of the deployed buses are allocated to originally planned lines.

Similar to the above, some bus operators might not be willing to generate switch stop candidates at bus stops with slight ridership
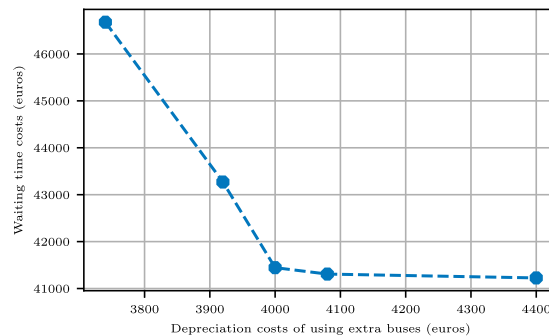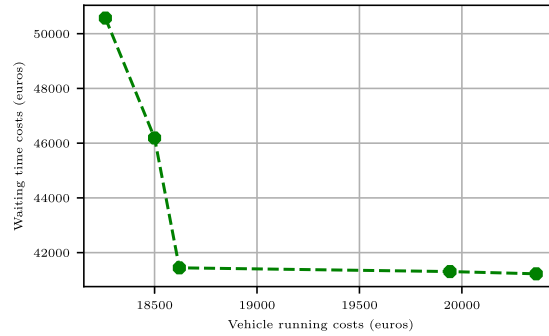

Fig. 14. Pareto frontiers: waiting time costs against depreciation costs of using extra buses.

Table 10
Cost coefficients for deriving the Pareto frontier of passenger waiting times against running costs.

Scenario	ω_1	ω_2
1	0.33	0.66
2	0.40	0.60
3	0.50	0.50
4	0.75	0.25
5	0.80	0.20

**Fig. 15.** Pareto frontiers: waiting time costs against vehicle running costs.

changes. Instead, they might consider a bus stop as switch stop candidate only when a significant ridership change is observed (i.e., $z > 50\%$). In principle, more conservative bus allocations that allocate a smaller portion of the fleet to short-turning and interlining lines are expected to have an inferior performance compared to bus allocations that adopt more flexible bus allocation schemes.

The results from this analysis are presented in Fig. 16 where the performances of the optimal bus allocation solutions for different values of ψ and z are presented. It should be noted here that apart from the values of ψ and z, all other parameters maintain their values from Table 5.

In Fig. 16 the total cost of the operations for $\psi = 60\%$ and $z = 20\%$ is 64,066€ (as reported in Table 7). The total cost of the operations is the lowest (64,019€) for the most flexible scenario where the minimum number of buses that must be allocated to originally planned lines is $\psi = 40\%$ of the total number of deployed buses and $z = 10\%$.

From Fig. 16 one can observe that there is a broad range of values, i.e. $\psi = 60–80\%$ and $z = 10–30\%$, for which the total cost of the optimal bus allocations is relatively stable and hovers around 64,100€. This is an important finding because a more conservative (and practical) bus allocation where at least 80% of the deployed buses are allocated to the originally planned lines can be adopted without significantly increasing the total cost of operations.

Another important finding is that the solution is more sensitive to changes in ψ than in z. For instance, when $\psi = 90\%$ and $z = 20\%$ the total cost of the bus allocation is 67,923€ which is very close to the total cost of the optimal bus allocation when considering only
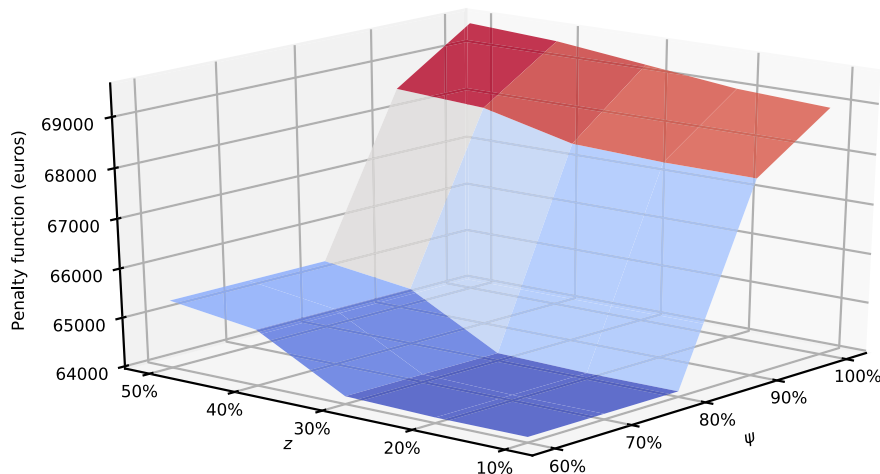
Fig. 16. Total cost of the optimal bus allocation for different values of the parameter ψ which controls the minimum percentage of buses that should be allocated to originally planned lines and z which affects the set of switch stop candidates.
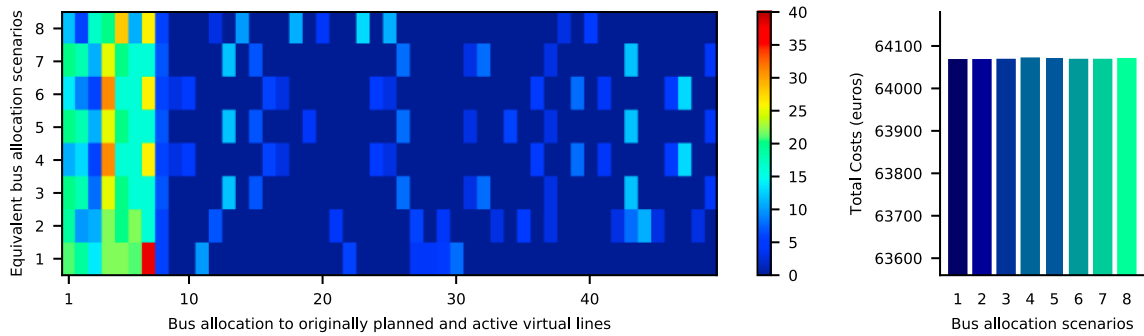
Fig. 17. Bus allocations to originally planned and active virtual lines for each one of the eight local optima and their corresponding total costs.

originally planned lines (this cost was 69,032€). Notwithstanding, a comparable performance was observed when at least 60%, 70% or 80% of the buses are allocated to originally planned lines. This provides a strong advantage to the bus operator that can have a maximum benefit by allocating the vast majority of its buses to originally planned lines and still benefit from a significant improvement of passenger/operational-related costs. Hence, by allocating the minority of the buses (20–40%) to short-turning and interlining lines can have the same effect of potentially impractical bus allocations that require the allocation of many buses to virtual lines.

5.6. Exploitation of the local optima

When allocating buses to both originally planned and short-turning/interlining lines there can be more than one solutions that result in quite similar total costs. For example, when allocating buses to originally planned and short-turning/interlining lines using the parameters of Table 5, we might have different bus allocations than the one presented in Fig. 9 that exhibit quite similar performances. All these are different locally optimal solutions that exhibit an almost equally good performance.

Each local optimum can be computed by solving the bus allocation problem using the GA proposed in this study and employing a different set of initial solution guesses each time the problem is solved. Fig. 17 illustrates it by visualizing some of the local optima (eight different solutions with almost equal performance) where each one of them represents a different bus allocation with total costs in the range of 64,066–64,071€.

In Fig. 17, we present the detailed allocation of buses to originally planned (eight first lines) and active virtual lines (lines 9–49) for each of the eight bus allocations that exhibit a similar performance. Fig. 17 presents also the performance of each bus allocation in terms of total costs.

6. Concluding remarks

This work develops a framework for allocating buses to originally planned and short-turning/interlining bus lines in order to reduce the passenger-related and the operational-related costs while satisfying a set of operational constraints. Following the problem formulation, a meta-heuristic solution approach is developed and applied to a case study network. It became apparent after the formulation of the problem that its combinatorial nature and its exponential complexity does not allow for the computation of optimal bus allocations to a large set of virtual lines. For this reason, this work approximated the nonlinear constrained optimization problem with the use of exterior point penalties and introduced a GA for allocating buses to originally planned and virtual lines.

Model application demonstrates that the partial replacement of current services with virtual lines can significantly reduce (i.e. 13.85% for the real-world case study network) the vehicle running times while reducing also the average waiting time per passenger by $\simeq 5\%$. In the proposed approach, the operational short-turning and interlining lines are endogenously generated (in contrast to the works of Delle Site and Filippi (1998), Verbas and Mahmassani (2013), Verbas et al. (2015)), by considering a pool of virtual lines as part of the optimization process. The results indicate that the plurality of bus allocation options when considering a broader set of virtual lines can return a range of bus allocation combinations that offer almost equally large benefits. This provides a strong decision-support tool to bus service planners and operators who might have latent preferences or requirements (e.g. familiarity of bus drivers with certain lines, preference towards serving originally planned lines).

The sensitivity analysis of the model application demonstrated that re-allocating even a small share of vehicles to virtual lines can have a significant impact on the total costs of the operations (i.e., significant improvements are observed even if 80% of the deployed buses are allocated to originally planned lines). This finding demonstrates that bus operators do not need to change significantly the deployment of their buses for reducing the passenger/operational-related costs.

Concerning the limitations of our approach:

- our work allocates buses to originally planned and short-turning/interlining lines at the tactical planning stage using historical distributions of the round-trip travel times. For near real-time re-allocation of buses to lines, our approach should be modified for incorporating short-term traffic predictions;

- buses need to be equipped with electronic destination signs (e.g., LED-type destination signs) which can modify the displayed location of the final destination if they operate a short-turning or interlining service;
- either observed or forecasted passenger demand between each pair of stops of each originally planned line needs to be available as input.

Future research direction may consider the demand elasticity to changes in service frequency. Moreover, the development of tactical planning tools that incorporate transit assignment models will potentially allow capturing the impacts of such interactions on passenger flow re-distribution.

Acknowledgement

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.trc.2018.11.007.

References

Arbex, R.O., da Cunha, C.B., 2015. Efficient transit network design and frequencies setting multi-objective optimization by alternating objective genetic algorithm. Transport. Res. Part B: Methodol. 81, 355–376.

Bakirtzis, A.G., Biskas, P.N., Zoumas, C.E., Petridis, V., 2002. Optimal power flow by enhanced genetic algorithm. IEEE Trans. Power Syst. 17 (2), 229–236.

Barra, A., Carvalho, L., Teypaz, N., Cung, V.-D., Balassiano, R., 2007. Solving the transit network design problem with constraint programming. In: 11th World Conference in Transport Research-WCTR 2007.

Bellei, G., Gkoumas, K., 2010. Transit vehicles headway distribution and service irregularity. Public Transport 2 (4), 269–289.

Bertsekas, D.P., 1990. Nonlinear Programming, second ed. Athena Scientific, Belmont, Massachusetts.

Borndörfer, R., Grötschel, M., Pfetsch, M.E., 2007. A column-generation approach to line planning in public transport. Transport. Sci. 41 (1), 123–132.

Ceder, A., 1984. Bus frequency determination using passenger count data. Transport. Res. Part A: Gen. 18 (5–6), 439–453.

Ceder, A., 2007. Public Transit Planning and Operation: Theory, Modeling and Practice. Elsevier, Oxford.

Ceder, A., 2016. Public Transit Planning and Operation: Modeling, Practice and Behavior. CRC Press.

Ceder, A., Golany, B., Tal, O., 2001. Creating bus timetables with maximal synchronization. Transport. Res. Part A: Policy Pract. 35 (10), 913–928.

Ceder, A., Wilson, N.H., 1986. Bus network design. Transport. Res. Part B: Methodol. 20 (4), 331–344.

Chen, X., Hellinga, B., Chang, C., Fu, L., 2015. Optimization of headways with stop-skipping control: a case study of bus rapid transit system. J. Adv. Transport. 49 (3), 385–401.

Chow, A.H., Pavlides, A., 2018. Cost functions and multi-objective timetabling of mixed train services. Transport. Res. Part A: Policy Pract. 113, 335–356.

Cipriani, E., Gori, S., Petrelli, M., 2012. Transit network design: a procedure and an application to a large urban area. Transport. Res. Part C: Emerg. Technol. 20 (1), 3–14.

Cortés, C.E., Jara-Díaz, S., Tirachini, A., 2011. Integrating short turning and deadheading in the optimization of transit services. Transport. Res. Part A: Policy Pract. 45 (5), 419–434.

Daganzo, C.F., 2010. Structure of competitive transit networks. Transport. Res. Part B: Methodol. 44 (4), 434–446.

Delle Site, P., Filippi, F., 1998. Service optimization for bus corridors with short-turn strategies and variable vehicle size. Transport. Res. Part A: Policy Pract. 32 (1), 19–38.

dellOlio, L., Ibeas, A., Ruisánchez, F., 2012. Optimizing bus-size and headway in transit networks. Transportation 39 (2), 449–464.

Fan, W., Machemehl, R.B., 2008. Tabu search strategies for the public transportation network optimizations with variable transit demand. Comput.-Aid. Civil Infrastruct. Eng. 23 (7), 502–520.

Farahani, R.Z., Miandoabchi, E., Szeto, W.Y., Rashidi, H., 2013. A review of urban transportation network design problems. Euro. J. Oper. Res. 229 (2), 281–302.

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., Gagné, C., 2012. DEAP: evolutionary algorithms made easy. J. Mach. Learn. Res. 13, 2171–2175.

Fu, L., Yang, X., 2002. Design and implementation of bus-holding control strategies with real-time information. Transport. Res. Rec.: J. Transport. Res. Board (1791), 6–12.

Furth, P.G., Wilson, N.H., 1981. Setting frequencies on bus routes: theory and practice. Transport. Res. Rec. 818 (1981), 1–7.

Gavriilidou, A., Cats, O., 2018. Reconciling transfer synchronization and service regularity: real-time control strategies using passenger data. Transport. A: Transport Sci. 1–29.

Ghaemi, N., Cats, O., Goverde, R.M., 2017. A microscopic model for optimal train short-turnings during complete blockages. Transport. Res. Part B: Methodol. 105, 423–437.

Gkiotsalitis, K., Cats, O., 2017. Exact optimization of bus frequency settings considering demand and trip time variations. In: Transportation Research Board, 96th Annual Meetings, No: 17-01871.

Gkiotsalitis, K., Cats, O., 2018. Reliable frequency determination: incorporating information on service uncertainty when setting dispatching headways. Transport. Res. Part C: Emerg. Technol. 88, 187–207.

Gkiotsalitis, K., Maslekar, N., 2018a. Multiconstrained timetable optimization and performance evaluation in the presence of travel time noise. J. Transport. Eng., Part A: Syst. 144 (9), 04018058.

Gkiotsalitis, K., Maslekar, N., 2018b. Towards transfer synchronization of regularity-based bus operations with sequential hill-climbing. Public Transport 10 (2), 335–361.

Gkiotsalitis, K., Stathopoulos, A., 2015. A utility-maximization model for retrieving users willingness to travel for participating in activities from big-data. Transport. Res. Part C: Emerg. Technol. 58, 265–277.

Gkiotsalitis, K., Stathopoulos, A., 2016. Demand-responsive public transportation re-scheduling for adjusting to the joint leisure activity demand. Int. J. Transport. Sci. Technol. 5 (2), 68–82.

Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res. 13 (5), 533–549.

Goldberg, D.E., Deb, K., 1991. A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. In: Elsevier, pp. 69–93.

Goulet-Langlois, G., Koutsopoulos, H.N., Zhao, J., 2016. Inferring patterns in the multi-week activity sequences of public transport users. Transport. Res. Part C: Emerg. Technol. 64, 1–16.

Hadas, Y., Shnaiderman, M., 2012. Public-transit frequency setting using minimum-cost approach with stochastic demand and travel time. Transport. Res. Part B: Methodol. 46 (8), 1068–1084.

Hernández, D., Muñoz, J.C., Giesen, R., Delgado, F., 2015. Analysis of real-time control strategies in a corridor with multiple bus services. Transport. Res. Part B: Methodol. 78, 83–105.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, Michigan.

Huang, Z., Ren, G., Liu, H., 2013. Optimizing bus frequencies under uncertain demand: case study of the transit network in a developing city. Math. Probl. Eng.

Ibarra-Rojas, O., Delgado, F., Giesen, R., Muñoz, J., 2015. Planning, operation, and control of bus transport systems: a literature review. Transport. Res. Part B: Methodol. 77, 38–75.

Kepaptsoglou, K., Karlaftis, M., 2009. Transit route network design problem: review. J. Transport. Eng. 135 (8), 491–505.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. Science 220 (4598), 671–680.

Li, Y., Xu, W., He, S., 2013. Expected value model for optimizing the multiple bus headways. Appl. Math. Comput. 219 (11), 5849–5861.

Liu, Z., Yan, Y., Qu, X., Zhang, Y., 2013. Bus stop-skipping scheme with random travel time. Transport. Res. Part C: Emerg. Technol. 35, 46–56.

Ma, X., Wu, Y.-J., Wang, Y., Chen, F., Liu, J., 2013. Mining smart card data for transit riders travel patterns. Transport. Res. Part C: Emerg. Technol. 36, 1–12.

Mandl, C.E., 1980. Evaluation and optimization of urban public transportation networks. Euro. J. Oper. Res. 5 (6), 396–404.

Ming, W., Bo, S., Wenzhou, J., 2013. A bi-level programming model for uncertain regional bus scheduling problems. J. Transport. Syst. Eng. Inform. Technol. 13 (4), 106–112.

Moz, M., Respício, A., Pato, M.V., 2009. Bi-objective evolutionary heuristics for bus driver rostering. Public Transport 1 (3), 189.

Munizaga, M.A., Palma, C., 2012. Estimation of a disaggregate multimodal public transport origin-destination matrix from passive smartcard data from santiago, chile. Transport. Res. Part C: Emerg. Technol. 24, 9–18.

Newell, G.F., 1974. Control of pairing of vehicles on a public transportation route, two vehicles, one control point. Transport. Sci. 8 (3), 248–264.

Osuna, E., Newell, G., 1972. Control strategies for an idealized public transportation system. Transport. Sci. 6 (1), 52–72.

Pattnaik, S., Mohan, S., Tom, V., 1998. Urban bus transit route network design using genetic algorithm. J. Transport. Eng. 124 (4), 368–375.

Shireman, M.T., 2011. Using automatically collected data for bus service and operations planning. Ph.D. thesis, Massachusetts Institute of Technology.

Simon, D., 2013. Evolutionary Optimization Algorithms. John Wiley & Sons.

Sun, A., Hickman, M., 2005. The real–time stop–skipping problem. J. Intell. Transport. Syst. 9 (2), 91–109.

Szeto, W.Y., Solayappan, M., Jiang, Y., 2011. Reliability-based transit assignment for congested stochastic transit networks. Comput.-Aid. Civil Infrastruct. Eng. 26 (4), 311–326.

Szeto, W.Y., Wu, Y., 2011. A simultaneous bus route design and frequency setting problem for tin shui wai, hong kong. Euro. J. Oper. Res. 209 (2), 141–155.

Verbas, I., Mahmassani, H., 2013. Optimal allocation of service frequencies over transit network routes and time periods: formulation, solution, and implementation using bus route patterns. Transport. Res. Rec.: J. Transport. Res. Board (2334), 50–59.

Verbas, İ.Ö., Frei, C., Mahmassani, H.S., Chan, R., 2015. Stretching resources: sensitivity of optimal bus frequency allocation to stop-level demand elasticities. Public Transport 7 (1), 1–20.

Wren, A., Rousseau, J.-M., 1995. Bus driver schedulingan overview. In: Computer-Aided Transit Scheduling. Springer, pp. 173–187.

Wu, W., Liu, R., Jin, W., 2017. Modelling bus bunching and holding control with vehicle overtaking and distributed passenger boarding behaviour. Transport. Res. Part B: Methodol. 104, 175–197.

Yan, S., Chi, C.-J., Tang, C.-H., 2006. Inter-city bus routing and timetable setting under stochastic demands. Transport. Res. Part A: Policy Pract. 40 (7), 572–586.

Yap, M., Cats, O., van Arem, B., 2018. Crowding valuation in urban tram and bus transportation based on smart card data. Transport. A: Transport Sci. 1–20 (in press).

Yu, B., Yang, Z., Yao, J., 2009. Genetic algorithm for bus frequency optimization. J. Transport. Eng. 136 (6), 576–583.

Yu, Y., Ye, Z., Wang, C., 2015. Study of bus stop skipping scheme based on modified cellular genetic algorithm. In: CICTP 2015. pp. 2397–2409.

Zhao, F., Zeng, X., 2008. Optimization of transit route network, vehicle headways and timetables for large-scale transit networks. Euro. J. Oper. Res. 186 (2), 841–855.