Design of mixed fixed-flexible bus public transport networks by tracking the paths of on-demand vehicles

Fielbaum, Andres; Alonso-Mora, Javier

DOI
10.1016/j.trc.2024.104580

Publication date
2024

Document Version
Final published version

Published in
Transportation Research Part C: Emerging Technologies

Transportation Research Part C 168 (2024) 104580

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Design of mixed fixed-flexible bus public transport networks by tracking the paths of on-demand vehicles☆

Andres Fielbaum [a,*], Javier Alonso-Mora [b]

[a] School of Civil Engineering, The University of Sydney, Darlington 2008 NSW, Australia
[b] Department of Cognitive Robotics, TU Delft, Mekelweg 2, 2628 CD Delft, The Netherlands

ARTICLE INFO

Keywords:
Public transport design
On-demand mobility
Ridepooling
Walking legs
Fixed routes

ABSTRACT

On-demand ridepooling (ODRP) vehicles follow routes that are fully flexible. However, when the system does not provide door-to-door service and users can be asked to walk, their paths tend to concentrate, particularly along main streets that connect highly demanded areas of the city. These frequently travelled segments are hence useful to multiple passengers, which can be used as an indicator that it would be efficient to allocate a fixed public transport line there.

In this paper, we formalise this idea and propose a novel method to design a public transport network, where bus fixed lines are combined with ODRP. Given a network and a transport demand, we first simulate how to serve it using only ODRP with walking. For this, we employ a state-of-the-art assignment algorithm, and take as output the resulting users' paths. These paths are then processed by a tailored algorithm to create fixed lines where the paths accumulate the most. Users who do not have an available fixed line (i.e., those whose paths were barely shared) are served by ODRP in the mixed system. Simulations using real-life data from Utrecht, The Netherlands, and the Sunshine Coast, Australia, reveal the merits of our method compared to several benchmarks. Crucially, our method builds a small number of fixed lines while still serving the majority of the demand through them.

This study contributes not only to the design of public transport networks, but also to the understanding of the patterns that naturally appear in intrinsically flexible mobility systems.

1. Introduction

The problem of designing public transport networks, namely which lines should be offered to serve a given transport demand, is one of the most traditional and important problems in transport theory. It has been studied for decades, with seminal papers that are over 40 years old (as the one by Dubois et al., 1979), and it still captures attention from researchers (Mauttone et al., 2021; Iliopoulou et al., 2019; Durán-Micco and Vansteenwegen, 2022). The emergence of massive *on-demand ridepooling* (ODRP) implies a radical update on this problem, as it is now possible to consider the chance of complementing traditional fixed-line services with some flexible ones. Some first studies to design such a *mixed* network have been proposed by Pinto et al. (2020) and Gurumurthy et al. (2020). As these are quite complex problems, they have been solved through tailored heuristics, typically combinatorial approaches that explicitly build the routes based on the Origin–Destination Matrix (e.g. Barabino, 2009, which we use as a benchmark later in

☆ This paper has been accepted for presentation at the 25th International Symposium on Transportation and Traffic Theory (ISTTT25), July 15–17, 2024, Ann Arbor, Michigan, USA.

* Corresponding author.
E-mail address: andres.fielbaum@sydney.edu.au (A. Fielbaum).

https://doi.org/10.1016/j.trc.2024.104580
Received 27 December 2023; Accepted 18 March 2024
Available online 28 March 2024

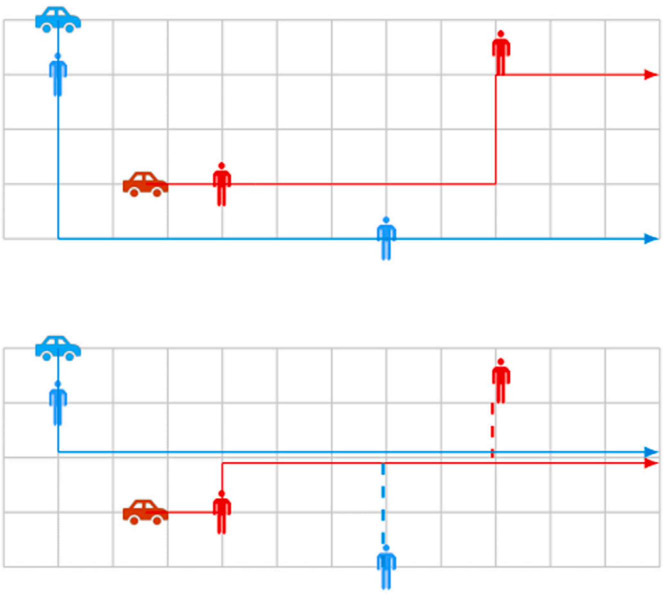


Fig. 1. An example of how including walks push ODRP vehicles to concentrate on some routes. In both rows each vehicle is serving a group of two users, where only their pickup locations are shown. In the top row the service is door-to-door so the vehicles routes do not intersect with each other; in the bottom, two of the passengers walk towards the central street, making the vehicles paths to coincide there.

this paper), or optimisation techniques that require defining some initial set of feasible routes (e.g. Borndörfer et al., 2007) to then decide which of them should be actually offered.

In this paper, we propose a completely new approach to designing both the fixed routes and how to route the on-demand vehicles. To decide which fixed routes to offer, we formulate and make intense use of the following hypothesis: If ODRP is not door-to-door, but users can be told to walk some legs in order to avoid long detours, ODRP vehicles tend to concentrate on some streets and routes. This hypothesis is illustrated in Fig. 1, where in the bottom row the two vehicles' routes coincide in the central street, suggesting that there is potential for fixed-route buses to run there.

The main reason why vehicles are expected to concentrate is because street networks have consistently been found to be *hierarchical* in most cities worldwide. That is, there is a small number of large streets (the avenues) that play a central role to connect the whole network. This has been determined by finding that such avenues repeatedly appear in shortest paths between distant nodes within a city, concentrate the largest number of intersections (through the so-called dual graph of a street network), and can be used to generate a simplified representation of a city (Masucci et al., 2014; Sanders and Schultes, 2005; Porta et al., 2006; Fielbaum et al., 2017; Figueiredo and Amorim, 2007). This implies that the hypothesis mentioned above is expected to hold in most real-life cities, but might not be valid in particular cities lacking this hierarchical structure,[1] such as cities with multiple disconnected centres.

The role played by the ODRP vehicles in our model is also different than most previous studies. As detailed in Section 2, such studies have usually assumed that the on-demand services are to solve the *first mile/last mile* problem, i.e., as a feeder that brings passengers towards (or from) the main fixed network. However, it is not clear that this is the best possible strategy, and other alternatives need to be considered. It is especially relevant to avoid increasing the number of transfers, as they are known to be very uncomfortable to users and could make them opt for a different transport mode (Garcia-Martinez et al., 2018). In our method, we divide the users into two categories: some users perform their whole trips via ODRP, and the rest in traditional fixed-route lines. Our method endogenously makes this division according to the following rationale: some users have a very large degree of *shareability*, i.e., there are plenty of other requests with whom they could share the vehicle without deviating too much for the shortest path between their origin and destination[2]: these users can usually travel by bus, which are better suited to fixed routes. But some users make trips that are not so shareable, either because they represent a non-popular origin–destination pair, or because they are located in areas that are not easily reachable, so these users can be better served through ODRP.

[1] According to Figueiredo and Amorim (2007), some planned cities exhibit a structure much closer to a perfect grid — thus not hierarchical. Some examples are Gama and Ceilândia-Taguatinga in Brazil. The street networks of these cities present topological indices that are much closer to a grid than those of non-planned cities. Interestingly, some cities (such as Barcelona, Spain) present a combination of planned neighbourhoods that are close to grids (like the *Eixample* in Barcelona) and others that naturally developed with a more hierarchical structure.

[2] A formal definition of this notion of shareability, as well as calculations for different cities, have been studied by Santi et al. (2014) and Tachet et al. (2017).

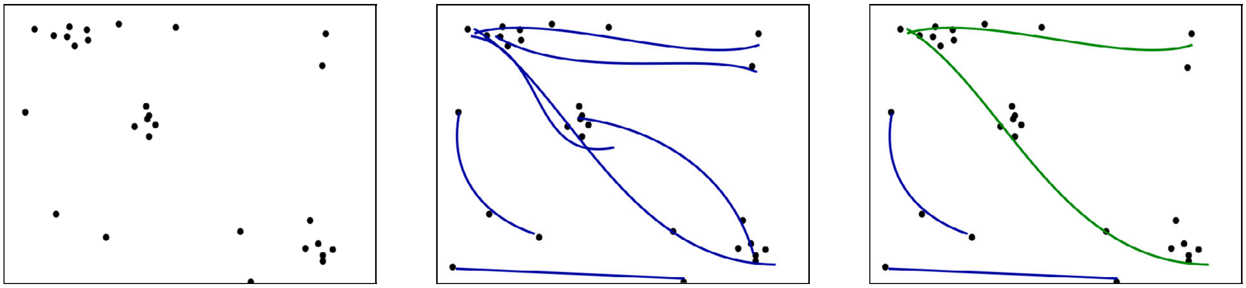


Fig. 2. A representation of how paths' concentration can be used to design fixed public transport networks. Black dots represent origins and destinations (left); they are connected by 7 ODRP vehicles, whose trajectories are represented in blue (centre); some of those trajectories are covered by two fixed public transport lines represented in green, while other users remain served by ODRP (right).

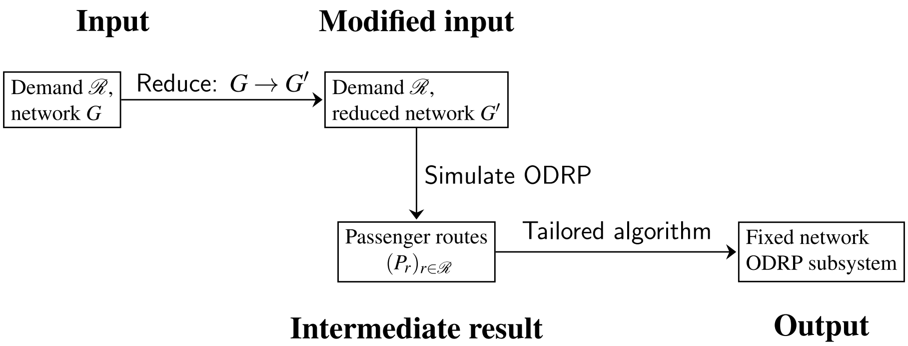


Fig. 3. Schematic representation of our method.

Our method can be roughly described as follows: we first simulate how to serve the demand using (non door-to-door) ODRP only, to output the paths followed by each user[3]; we then track which routes were followed frequently, and allocate fixed lines there through a tailored greedy algorithm. Those users whose routes were barely shared are the ones that will use ODRP in the mixed system. This idea is represented in Fig. 2, where the black dots represents origins and destinations, the blue curves are cars' trajectories to join them, and the green curves are the fixed lines that could be built.

The hypothesis that vehicles concentrate is more likely to be fulfilled if vehicles run in a subset of the original network, which is coherent with the fact that traditional public transport vehicles do not enter every street: Our algorithms could work over any exogenous subnetwork, and we also propose an endogenous method to *reduce the graph*. Once we have the subnetwork, we then *simulate the ODRP operation*, and finally *build the mixed public transport network* based on the passengers' routes. The whole method is represented in Fig. 3.

We highlight three relevant merits of this method: First, it automatically produces a mixed network, endogenously identifying which users have the potential to share a fixed line and which require a smaller flexible vehicle; second, the construction of the fixed sub-network does not decide *a priori* the bus stop for each trip (as usual in the heuristics in the literature), but can decide between all the walking-reachable nodes; third, the aim to translate flexible vehicles' routes into fixed networks is somewhat intuitive and has been suggested previously in the literature (e.g. Kucharski and Cats, 2022), and up to our knowledge, this is the first paper that formalises and executes this idea.

The paper is structured as follows. Section 2 summarises the most relevant related literature. Section 3 summarises the notation and setup of the problem. Section 4 explains the method to reduce the graph. The core of the method is explained in Section 5 where we explain how to design the mixed network; particularly, Section 5.1 explains how to obtain the passengers routes (using existing approaches), and Section 5.2 explains the novel algorithm we propose to transform those passengers routes into fixed bus lines. Section 6 shows numerical results, and Section 7 concludes and proposes directions for further research.

2. Literature review

The problem of deciding the optimal set of routes for a given network and transport demand has been thoroughly studied in the last decades. Older papers usually focus solely on fixed routes (as the ability to connect users and vehicles online turned

[3] Note that the vehicles' paths would be a more direct expression of the idea that vehicles tend to concentrate. However, users' paths are of course closely correlated, and are the true expression of what needs to be moved.

really efficient less than a decade ago), where one can find roughly two types of approaches: strategic ones that work on predefined simplified city representations, such as Badia et al. (2014, 2016), Daganzo (2010), Fielbaum et al. (2016), Jara-Díaz and Gschwender (2003) and Chen et al. (2018), or heuristics that take the graph and the demand as inputs, such as Jha et al. (2019), Borndörfer et al. (2007) and Cancela et al. (2015). The latter methods have been surveyed by Guihaire and Hao (2008), Durán-Micco and Vansteenwegen (2022), and some of them were compared by Fielbaum et al. (2018). Public transport systems with some degree of flexibility (usually denoted as *demand responsive transit*), designed before the ride-hailing era were surveyed by Ronald et al. (2015). A recent survey was done by Vansteenwegen et al. (2022).

Although the problem with fixed routes is still open, in the last years researchers have paid more attention to the design of mixed networks, where fixed routes can be complemented by vehicles with a certain degree of flexibility, including ODRP. Most of this research has assumed that on-demand vehicles are to feed the massive vehicles that follow fixed routes (in other words, to solve the first mile/last mile problem).

Let us describe some of the studies that have analysed different aspects of a mixed public transport system where the flexible component serves as a feeder. Bürstlein et al. (2021) and Shen et al. (2018) study the effect of replacing car by ODRP for people travelling towards large public transport stations; Calabrò et al. (2021) and Fielbaum (2020) use continuous-approximation models to analyse mixed systems where ODRP is a feeder and the trunks are large buses; Maheo et al. (2019) and Pinto et al. (2020) optimise the whole network together, i.e. the fixed lines together with the flexible vehicles that serve as feeders or in low-demand areas; Grahn et al. (2023) focuses on the environmental impact of the on-demand feeder; He et al. (2023) considers that the on-demand component does not have dedicated drivers but commuters (such as BlaBlaCar).

All the examples in the previous paragraph reveal that assuming the on-demand vehicles to act as feeders has been extensively studied by researchers, whereas alternative ways to utilise them have been barely proposed. Some few examples are as follows: Kim et al. (2023), who consider three sub-systems with different degrees of flexibility, assigning each passenger to one of the three; Tang et al. (2023) and Ma et al. (2019), where the flexible vehicles can serve either as feeders or to provide door-to-door service, but the fixed-lines' routes and frequencies are exogenous; Banerjee et al. (2021), who focus on the commuter choice part of the problem by including pricing in the model; and Basciftci and Van Hentenryck (2023), whose emphasis is on the ability of the system to attract car drivers. The question about the convenience of utilising on-demand mobility to act as feeder was explicitly studied by Fielbaum et al. (2022a) in a simplified city representation, who found that this is an optimal strategy only under very specific demand configurations, or if transfer discomfort is disregarded. This conclusion is reinforced by the analyses done by Zuniga-Garcia et al. (2022), Thorhauge et al. (2022) and Perera et al. (2020), who study real-life implementations and find that they have either failed to increase public transport ridership, or are often used to fulfil local trips rather than (or in addition to) connectors to the trunk services. As such, utilising the on-demand vehicles beyond being feeders, and jointly designing the whole public transport network, is still an open challenge.

2.1. Contribution

The main contribution of this paper is designing a mixed fixed-flexible public transport network following whole new methods. While recent literature has provided several new methods to design public transport networks in which on-demand mobility is involved, this paper presents two crucial novelties that have not been studied before:

1. We build the fixed subnetwork by tracking the paths of the on-demand vehicles. Up to our knowledge, this has not been done before, and can have implications beyond the ones developed in this paper, as this shows that even flexible vehicles do concentrate in some arcs and streets. For instance, this could be leveraged to decide on infrastructure-related questions, such as stops or prioritised lanes, or to determine congestion pricing strategies.
2. Propose a completely different logic in terms of the role played by the on-demand vehicles. Namely, we endogenously split the demand into two subsets, one of them travelling by fixed-routes and the other via ODRP. This idea, closely related to the different degrees of shareability encountered within a city, can be exploited to propose other alternative designs. As described above, to the best of our knowledge, similar ideas have only been studied in papers where the fixed-lines subnetwork is exogenous.

Both contributions are tested numerically through experiments in Utrecht, The Netherlands, and The Sunshine Coast, Australia. We develop tailored benchmark methods so that these contributions are tested jointly and separately.

3. Method preliminaries

Our methods take place over a directed graph $G = (V, E)$, where vehicles move and users walk. We assume the graph to be strongly connected (i.e., there is at least one directed path connecting every $u_1, u_2 \in V$). Further, we assume there are no leaves, i.e., every node u is connected (regardless of the direction) to at least two distinct nodes. Note that having no leaves is a harmless assumption, as it only implies that the routing method must avoid the dead ends (cul-de-sacs).[4] Every arc $a \in E$ is characterised by the times required to cross it either by vehicle $t_V(a)$ or walking $t_W(a)$, where the latter can be done in both directions. We denote by $t_V(u_1, u_2), t_W(u_1, u_2)$ the time-length of the shortest paths between u_1 and u_2 by vehicle or walking, respectively. We assume no

[4] Moreover, one could replace any leaf by a triangle.

passenger is willing to walk more than an exogenous parameter Ω_a, and we say that two nodes u_1, u_2 are at *walking distance* if $t_W(u_1, u_2) \leq \Omega_a$.

The other input of the problem is the transport demand.[5] It is described as a set of requests $\mathscr{R}$, with every request $r \in \mathscr{R}$ a triplette $r = (o_r, d_r, t_r)$, representing the origin, destination, and time of request, respectively. Origins and destinations are assumed to be nodes in the graph.

For any node $u \in V$, we denote by $\delta^+(u)$ its out-degree: $\delta^+(u) = |\{u' \in V : (u, u') \in E\}|$, and by $\delta^-(u)$ its in-degree $\delta^-(u) = |\{u' \in V : (u', u) \in E\}|$. We denote by $P(u)$ and $S(u)$ the set of "parents" and "sons" respectively: $P(u) = \{u' : (u', u) \in E\}, S(u) = \{u' : (u, u') \in E\}$, so $\delta^+(u) = |S(u)|, \delta^-(u) = |P(u)|$. We also use the terms parents and son for the arcs: e_2 is a son of e_1 if e_2 can be toured right after e_1.

Our purpose is to output two different objects:

1. A set of fixed-route lines, each of them represented by a sequence of stops (nodes in the graph), a fleet, and a period-dependent frequency. A request $r \in \mathscr{R}$ will use the fixed-route system if there exists at least one line ℓ, and two stops s_1, s_2 in ℓ, fulfilling two conditions:

   (i) s_1, s_2 are at walking distance from her origin and destination, i.e., $t_W(o_r, s_1) \leq \Omega_a, t_W(d_r, s_2) \leq \Omega_a$.
   (ii) The detour is acceptable, i.e., the time required by ℓ to tour all its stops between s_1 and s_2 is lower than $t_V(o_r, d_r) + \Omega_v$, where Ω_v denotes the maximum tolerable detour.

   When determining the frequencies and bus capacities, a necessary constraint is that they need to be able to carry all the passengers using the line in every segment.
2. A fleet of flexible-route vehicles, and an itinerary for each of them, so that all the users that do not have an available fixed-line are served here. Hard constraints on the capacity of the vehicle, and on users' delay, and waiting and walking times, must be fulfilled.

We aim at minimising the total costs of the system, defined as the sum of users' plus operators' costs. User r faces costs:

$$C_U(r) = \alpha_w t_w(r) + \alpha_v t_v(r) + \alpha_a t_a(r) \tag{1}$$

where t_w, t_v, t_a represent the time spent waiting, in-vehicle, and walking, respectively, and the $\alpha_h (h = w, v, a)$ are the corresponding weights. We assume that these costs are faced regardless of the type of vehicle used by r.[6] Regarding operators, following (Tirachini and Hensher, 2011), we assume that a vehicle ν of capacity K_ν, that moves during a time E_ν, costs

$$C_O(\nu) = \left[c_{BC} + c_{KC} K_\nu\right] + \left[E_\nu \cdot (c_{BO} + c_{KO} K_\nu)\right] \tag{2}$$

where the first square parenthesis represents capital costs and the second one operational costs. The parameters $c_{BC}, c_{KC}, c_{cBO}, c_{KO}$ represent the different components of operators' costs.

4. Reducing the graph

Here we explain how we select the set of streets where the vehicles will run. We remark that, in the context of this paper, this is relevant because we are assuming that these are the same streets available for buses. However, we also regard it as a contribution by itself, because restricting the streets in order to concentrate the vehicles in the most relevant ones can be a powerful way to enhance ODRP, regardless of the role of public transport.

The selection of the streets is done by adapting the method by Sanders and Schultes (2005). The original method is meant to leverage the hierarchical structure of street networks in order to hasten the search for shortest paths, so it outputs an integer number for each arc, representing its hierarchy. Their main idea is that the shortest paths typically reach high-hierarchy streets promptly, and only leave them when close to the destination. In our case, there are two crucial differences:

1. Our decision for each arc is binary, i.e., either we keep it or not.
2. We need our resulting graph to remain (i) strongly connected, and (ii) at walking distance from any node.

In order to do so, we prune arcs in an iterative way. During each iteration, we identify the lowest-hierarchy remaining arcs and prune them. We stop iterating when pruning would imply leaving at least one original node at a distance greater than Ω_a from the remaining network.

Each iteration consists of four steps: (1) Contract, (2) Select, (3) Connect, and (4) Unfold. The first step is a straightforward adaptation of the method by Sanders and Schultes (2005) to a digraph, where all the *lines* in the network are contracted to a single edge. The second one depends on a parameter H and is the core of the original method: it selects those arcs that appear in at least one shortest path not involving its H closest neighbours. The third step deals with the second issue listed above, ensuring that after each iteration we have a strongly connected digraph. The fourth step basically does the opposite to step 1. Let us now explain each step in more detail. We denote by $G' = (V', E')$ the resulting reduced graph.

[5] As usual in the literature, we assume the public transport demand as fixed, i.e., we focus on the question of which is the optimal way to serve a given demand. Our methods could be coupled with mode choice models if competition with other modes was considered as well.

[6] We opt for this neutral assumption, instead of favouring the small vehicles as they are more comfortable (Mohamed et al., 2020), or public fixed-routes vehicles as their routes are more reliable (Fielbaum and Alonso-Mora, 2020).

4.1. Step 1: Contract

In this step, we contract the lines within the graph, meaning that we replace them with a single arc. Formally, a line is a path $\Gamma = (u_1, \dots, u_k)$ in the graph (being a path means that $(u_i, u_{i+1}) \in E\ \forall i$), fulfilling one of the following two properties:

1. It is a *unidirectional* path if $\forall i = 2, \dots, k-1, \delta^-(u_i) = \delta^+(u_i) = 1$.
2. It is a *bidirectional* path if $\forall i = 2, \dots, k-1, P(u_i) = S(u_i) = \{u_{i-1}, u_{i+1}\}$.

Each line Γ is replaced by a connection between its first node u_1 and its last node u_k. Formally speaking, we remove all the nodes $u_2, \dots, u_{k-1}$, and add the *representative* arc (u_1, u_k). If the line was bidirectional, we also add the representative arc (u_k, u_1). The length of the said arc is set to be equal to the length of the original path (note that in the bidirectional case, the length of the two representative arcs does not need to coincide). A technical detail is that, for each line, we need to compute the maximum distance between its interior points and its arcs

$$\Omega(\Gamma) = \max_{u_i, i=1,\dots,k} \min \left[t_V(u_i, u_1), t_V(u_i, u_k) \right] \tag{3}$$

If $\Omega(\Gamma) \geq \Omega_a$, removing Γ in a subsequent step would imply that at least one of its nodes would not be at walking distance to the remaining network. Therefore, we mark all the lines Γ s.t. $\Omega(\Gamma) \geq \Omega_a$ as "unremovable".

Crucially, when a representative arc is kept after the **Select** step, it will be unfolded in the last step to reconstruct the original lines. If two nodes were connected by more than one line, only the shortest one (per direction) will be unfolded, plus any unremovable one.

The purpose of this step is that when applying the **Select** step, lines do not count to define which are the H closest neighbours.

4.2. Step 2: Select

This is the core of the contraction method, and it depends on a parameter H. For each node $u \in V$, we define:

- $N_u^+ = \{u' \in V : u' \neq u, |\{w \in V : t_V(u, w) \leq t_V(u, u')\}| \leq H\}$. That is to say, N_u^+ is the set of the H closest nodes from the origin u by vehicle.
- $N_u^- = \{u' \in V : u' \neq u, |\{w \in V : t_V(w, u) \leq t_V(u', u)\}| \leq H\}$. That is to say, N_u^- is the set of the H closest nodes towards the destination u by vehicle.

Then, an arc $e = (u, w)$ will be kept if and only if $\exists u', w' \in V$, such that $u \notin N_{u'}^+, w \notin N_{w'}^-$, and e is part of the shortest path going from u' to w'. Keeping e also implies that we keep u and w in the set of nodes. Additionally, we add all the arcs representing unremovable lines.

The condition of belonging to a shortest path effectively combines the speed of a street (one would like to have the fastest streets in the resulting network) with their topological relevance (e.g., one would like to keep the bridges, regardless of their speed). Requiring that this shortest path does not involve a neighbour ensures that the selected arcs play a global connecting role.

4.3. Step 3: Connect

The graph resulting from the previous step might not be strongly connected. To fix this, we need to add some arcs: for each u, w in the resulting graph, we analyse the shortest path $P = (u, v_1, \dots, v_k, w)$ in the original graph. Within P, we take the first node v_i (possibly w) that remains in the graph. We add the arc uv_i, with a cost equal to the original shortest distance between u and w_i.

4.4. Step 4: Unfold

In this step, each of the representative arcs built during the Contract step that remained in the graph is replaced by its original line (unidirectional or bidirectional). Note that this includes the unremovable lines.

5. Design of the mixed network

In this section, we describe the method that takes the graph (already reduced) and the demand, and outputs the mixed network. Before explaining the details, let us provide an overall description. First, we simulate as if we were to serve the whole of the demand via ODRP, through a state-of-the-art assignment-and-routing algorithm described in Section 5.1. The only relevant outputs of this simulation are the in-vehicle paths followed by each user to get from their pickup point to their drop-off point, which are used as the inputs for the next step. In Section 5.2, we describe how these paths are processed through a novel algorithm specifically designed for this purpose, which identifies the paths that were followed by a large number of users, and builds fixed lines along those paths, as long as they have sufficient demand (Section 5.2.1); the frequency and fleet of each line are optimised in Section 5.2.2. As not all the users will have an available fixed line, we serve the remaining ones through ODRP (Section 5.3). The other subsections discuss the limits of our approach and describe the methods we use as benchmarks.

5.1. Tracking passengers' paths in ODRP

To identify those paths that are useful for a large number of users, we first simulate the whole operational period *as if everyone was to travel through ODRP*.[7] To simulate the operation of ODRP, i.e., to decide how to route the vehicles and which users assign to each vehicle, we combine the following two extensions to the well-known method by Alonso-Mora et al. (2017):

- We consider that users can walk to optimised pick-up and drop-off (PUDO) nodes, following the method proposed by Fielbaum et al. (2021).
- We compute the fleet size endogenously, following the method proposed by Fielbaum et al. (2023a).

The procedure takes as input the network and the set of requests; the network is reduced as explained in Section 4, meaning that vehicles can only drive in the remaining arcs, but users can walk in all of them. It is a *rolling horizon* method, i.e., it accumulates the requests that appear every δ_t, and assigns them all together in consecutive iterations. The system begins with no vehicles, and each iteration is tackled as follows:

- The fleet of vehicles Veh is the one inherited from the previous iteration, plus a *potential* vehicle placed at the origin of each request waiting to be assigned (if the origin is not kept in the reduced graph, the vehicle is placed at the closest kept node). Each vehicle has capacity κ. The set of such requests is denoted by R.
- We compute all the feasible *trips*, where a trip T is defined by a vehicle v, a group g of requests waiting to be assigned, PUDO points for each user in g, and a route to serve the users in g and those that were previously assigned to v. A feasible route must respect hard constraints on walking, waiting, and total delay for each user. The route π minimises the cost $c(T,\pi)$:

$$c(T,\pi) = \sum_{r\in g} C_U(r,\pi) + \sum_{r\in v} \Delta C_U(r,\pi) + (c_{BO} + c_{KO}\kappa)\Delta Length(\pi,v) + (c_{BC} + c_{KC}\kappa)Act(v) \tag{4}$$

  The first two terms in Eq. (4) correspond to the user costs (following Eq. (1)) of the users in g and the extra user costs to the users previously assigned to v, respectively. In this step, to concentrate the vehicles further, we consider $\alpha_a = 0$. The third term corresponds to the extra operational costs imposed on the vehicle, which are proportional to the added time-length of the vehicle's itinerary, following the operational component of Eq. (2); the term $\Delta Length(\pi,v)$ is the difference between the length of π and the route currently being followed by v. The fourth term corresponds to the capital costs of Eq. (2), which is only paid once, i.e. it is an *activation cost*: The function $Act(v)$ takes the value 1 if the vehicle has not been activated yet, and 0 otherwise. When the activation cost has been paid, the vehicle is no longer considered potential.[8]
  For each trip we only consider the minimum cost $c(T)$ among all the feasible routes π. As the number of trips can be very large, a problem that becomes much worse when PUDO points are also optimised, a number of heuristics are introduced to hasten the calculations (see (Fielbaum et al., 2021) for details).
- The following ILP is solved to determine which of the feasible trips are selected.

$$\min_{x_T\in\{0,1\}} \sum_T x_T c(T) \tag{5}$$

$$\text{s.t.} \sum_{T:r\in T} x_T = 1 \qquad \forall r\in R \tag{6}$$

$$\sum_{T:v\in T} x_T \le 1 \qquad \forall v\in Veh \tag{7}$$

  Where the binary variable $x_T = 1$ iff the trip T is selected. Eq. (6) implies that each request is assigned to exactly one trip, and Eq. (7) ensures that each vehicle is assigned to at most one trip. Note that this problem is always feasible, as one could serve every user with its corresponding potential vehicle.
- Vehicles that were not activated and that were not assigned to any trip are deleted from the system. The other vehicles update their positions till the next iteration takes place.
- If there are some idle vehicles after the assignment, they get rebalanced exactly as done by Alonso-Mora et al. (2017): roughly speaking, they are sent towards the origins of the requests that required activating a new vehicle.

The only relevant output from these simulations is the routes followed by each user. Formally, for each user r served throughout the operational period, we output the sequence of nodes P_r that were visited by r while onboard the vehicle.

[7] Note that in a real-life implementation of our method, these simulations would not need to take place in reality, but could remain as virtual simulations. Their sole purpose would be to generate the inputs required to execute the algorithms described in Section 5.2. As such, the aim of our methods is the direct design of novel mixed fixed-flexible public transport networks (either for newly developed areas or when revisiting an existing system), rather than a sequential implementation of first ODRP and then fixed lines.

[8] Note that an actual operation of the ODRP system would need to have a fixed fleet, as the number of vehicles cannot be decided on the run. This method provides a heuristic to determine which fleet to use to serve a given demand

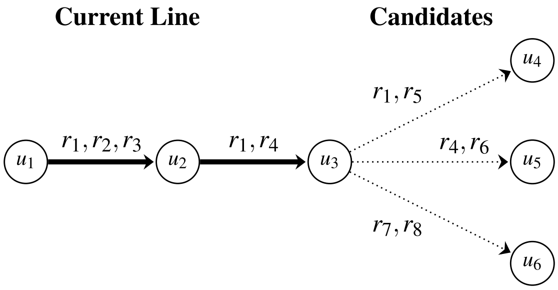


Fig. 4. The current line is u_1, u_2, u_3. r_1 and r_4 could use this line if extended in a direction useful for them, i.e., that coincides with their paths P_{r_1}, P_{r_4}. There are three candidate arcs to extend the line, i.e., three arcs that can be toured after u_3. All of them have a flow of two requests. If the line is extended towards u_4, it is still useful for r_1 but no longer useful for r_4; if the line is extended towards u_5, it is still useful for r_4 but no longer for r_1; if it is extended towards u_6, the line is no longer useful to either r_1 or r_4. r_5, r_6, r_7, and r_8 are requests that commenced their paths in u_3, so the line could be useful for them if extended in the right direction. According to our algorithm, the line will be extended towards u_4 to prioritise keeping r_1 because her path is already covered by two arcs.

5.2. Building the fixed part of the mixed public transport network

This part of the method decides on the routes of the fixed lines, and also on their fleets and frequencies. It takes as input the paths $(P_r)_{r \in \mathscr{R}}$ computed in the previous step. To build the fixed subsystem, we first decide on the lines' routes, through a tailored greedy algorithm that identifies the paths that were most frequently visited by the users; as such, **this is the main part of our method**. Once the lines' routes have been computed, we determine the fleet and the period-dependent frequency of each of them.

5.2.1. Computing the fixed routes

We implement a tailored greedy algorithm for this purpose.

Informal description of the algorithm. Let us explain how we build a new line each time, assuming that a number of lines (possibly zero) have already been built, each of them associated with some users that can board it (as detailed below). To build the new line, we first compute the total amount of remaining passenger flow in every arc e, i.e., the number of not-yet-assigned passengers r such that $e \in P_r$, meaning that those users traversed that arc. The arc with the greatest flow is selected to begin the new line, which is then extended first forward and then backward. Let us explain those extensions when going forward:

To decide where to extend the line, the candidate arcs are the sons of the current arc. For each candidate, we identify those users who either can start their trip there, or whose path is already partially covered by the line and their next arc is the candidate under scrutiny. The longer the partial coverage of the path, the more we weigh the passenger, and we extend the line by adding the candidate with the maximum weight. This step is exemplified in Fig. 4. This process is continued until one of the following two conditions is met: (i) The difference between the most and the least loaded arcs becomes too large (which would imply a large idle capacity), or (ii) The line returns to a node that is already part of it. When this happens, we extend backward from the starting arc following an analogous procedure. When this is done (due to the same reasons i or ii), we begin building a new line. We stop adding lines when the maximum load of the new one is too low (so we only use vehicles reasonably large), or when it has too few passengers.

This informal description is formalised in Algorithm 1, where we use the notation $L + e$ to represent that we add e at the end of L. As we are reserving the word "line" to refer to the fixed components of the public system, we will use the word "row" to refer to the different steps in Algorithm 1. Rows 1–3 initialise the problem. The WHILE cycle between rows 4–33 is the main component of the algorithm, with each repetition of the cycle creating a new line. Rows 5–8 initialise the new line, which begins with the arc with the largest remaining flow (row 8), as explained in the informal description of the algorithm. The WHILE cycle in rows 9–21 extends the line forward: the main step is row 11, where the weight of each candidate arc is calculated, so that the line is extended towards the candidate with the greatest weight (row 12). The line is then extended backwards following the same procedure (row 22). Rows 23–32 determine whether the new line is indeed added to the network, or if it is discarded because its load is too low, which also means that the algorithm is finished.

We remark that every time we finish building the route of a line, we identify all the users that can board it (line 25 of Algorithm 1), rather than only those whose paths are fully contained in the line. By doing so, we are able to gather passengers whose ODRP-routes were close but did not overlap.

5.2.2. Computing frequencies and fleets

We assume the day is divided into consecutive periods that last $E_1, \dots, E_n$. For the sake of simplicity, we do not model the transition between consecutive periods. For each line, we express users' and operators' costs as a function of the vector of frequencies,

Algorithm 1 Algorithm to compute the fixed routes.

Input: $(P_r)_{r\in\mathscr{R}}, \Delta_1, \Delta_2, \Delta_3$ // $\Delta_1, \Delta_2, \Delta_3$ define the conditions to break the steps.
$Q \leftarrow \mathscr{R}$ // The passengers yet to be assigned.
$\mathscr{L} \leftarrow \emptyset$ // The set of lines.
while TRUE **do** // This condition will be broken when the new added line does not have enough load.
  $Dif_L \leftarrow 0$
  Define $\forall e \in E, \rho_e = |\{r \in Q : e \in P_r\}|$ // ρ_e is the remaining flow in arc e.
  $e_0 \leftarrow argmax_{e\in E}\rho_e$ // The arc with the largest flow
  $L \leftarrow e_0, Pax_L \leftarrow \{r \in Q : P_r \cap L \neq \emptyset\}$ // The line begins in arc e_0, with the passengers that use that arc.
  while $Dif_L < \Delta_2$ **do** // When this condition is violated, we terminate extending forward the current line.
    $C = \{e : e \text{ is son of } e_0\}$ // Candidates to extend the line forward.
    Define $\forall e \in C : s(e) = \sum_{r\in Q} \begin{cases} (1+|L\cap P_r|)^2 & \text{if } r \in Pax_L \text{ and } e_0 \in P_r \\ 1 & \text{if } r \notin Pax_L \text{ and the first arc of } P_r \text{ is } e_0 \\ 0 & \text{otherwise} \end{cases}$ // To determine which is the next arc, we weigh more those users whose path has been already more covered by L. Here we use a quadratic weight but other increasing functions could be used.
    $e_1 = argmax_{e\in E} s(e)$
    $Dif_L = \frac{\text{Load in } e_1}{\text{Max load in } L+e_1}$
    if $Dif_L < \Delta_2$ and $L \cap e_1 = \emptyset$ **then**
      $L \leftarrow L + e_1$
      Update Pax_L // Add passengers beginning in e_1, remove passengers that needed an arc different than e_1 or whose path gets fully covered when adding e_1
      $e_0 \leftarrow e_1$
    else
      Stop extending the line forward
    end if
  end while
  Replicate the analogous version of the previous WHILE cycle to extend backwards
  F = Max load of L
  if $F > \Delta_1$ **then**
    $P_L = \{r : o_r, d_r$ are at walking distance from L and resulting detour is $< \Omega_v\}$ // All the passengers of this line.
    if $|P_L| \geq \Delta_3$ **then**
      $\mathscr{L} \leftarrow \mathscr{L} \cup \{L\}$
      $Q \leftarrow Q \setminus P_L$
    end if
  else
    Break While
  end if
end while

as we now explain, where we denote by $f_{\ell,p}$ the frequency of line ℓ at period p, and by $line(r), per(r)$ the line and period that corresponds to user r. When we have the total costs of line ℓ written as a function of the vector f_ℓ, we find the optimal frequency vector for each line ℓ.

Operators costs. For a given period p and line ℓ, we divide the period into the $E_p/f_{\ell,p}$ intervals between consecutive buses. It is then straightforward to identify which users are boarding and alighting the vehicle at every stop/interval, so we can compute what is the largest load K_ℓ a bus will carry, and we assume the buses of that line to have a capacity equal to K_ℓ. Further, we assume each user takes time t to board and alight (here we use $t = 10$ seconds), so the cycle time of each bus $\tau_{p,\ell}$ is calculated as the time needed to tour all the arcs involved plus the time waiting for boarding and alighting, and then the number of buses utilised at period p is $B_p = \tau_{p,\ell} \cdot f_{p,\ell}$. Following Eq. (2), operators' costs are then

$$C_{O\ell} = \left(\max_p B_p\right)\left[c_{BC} + c_{KC}K_\ell\right] + \sum_p E_p B_p \left[c_{BO} + c_{KO}K_\ell\right] \tag{8}$$

Users costs. Recall that users costs depend on walking, waiting, and in-vehicle time (Eq. (1)). Walking times do not depend on the frequency and are constant at this stage of the method (where the lines have already been designed so every fixed-line user knows which line to utilise). The in-vehicle time of user r is computed directly, as we know the stops visited by her while on-board: it is calculated by summing the travelling time between each pair of consecutive nodes, plus the time spent at each stop waiting for others to board and alight. Finally, we assume a constant headway between consecutive buses of the same period, so

$$t_w(r) = \frac{1}{2f_{line(r),per(r)}} \tag{9}$$

A remark on computing frequencies after deciding the lines' routes. In the literature, there is a distinction between TNDP (Transit Network Design Problem, e.g. Nnene et al., 2023) and TNDFSP (Transit Network Design and Frequency Setting Problem, e.g. Bertsimas et al., 2020). These two problems have been thoroughly surveyed by Durán-Micco and Vansteenwegen (2022). As they argue, the TNDFSP is, in principle, more complete, because "frequency setting is required for a proper evaluation of a candidate transit network". On the other hand, the complexity of TNDFSP is considerably harder than TNDP so it requires more simplifying assumptions, which is why many new studies still solve the problem sequentially, by first deciding on the routes without considering the frequencies (i.e., solving the TNDP), and only optimise the frequencies for the selected network. Our paper lies in this latter category, which we deem as reasonable given that this is the first method that creates the network based on the observed routes of ODRP. Integrating our ideas with frequency setting to develop a method that tackles the TNDFSP is a promising avenue for future research.

5.3. Re-simulating ODRP for the remaining passengers

We take the set of requests $\mathscr{R}_0$ that cannot reach any fixed line. We use exactly the same method described in Section 5.1, but with the real value of α_a (i.e., $\neq 0$), to serve them through ODRP. The resulting users' costs are calculated directly by computing their waiting, in-vehicle and walking times, while operators' costs result from computing the final fleet, and the amount of time each of those vehicles was moving.

5.4. Limits

Up to our knowledge, this is the first method that formalises the idea that flexible routes can indicate where to allocate public transport lines. As such, there are some relevant simplifications and limits that are regarded as future research. The most important ones are:

1. We do not account for transfers within the fixed routes. In other words, Algorithm 1 could be further enhanced if we would not only consider covering passengers' full paths, but also part of them so that the remainder would be fulfilled after a transfer.
2. We do not consider that some users could have more than one possible line, which could be tackled by coupling this method with some state-of-the-art route choice model (e.g. Arriagada et al., 2022).

5.5. Benchmarks

In order to test the merits of our method, we develop two benchmarks to compare with. These two benchmarks are based on the contributions stated in Section 2.1: first, we assess the quality of our network of fixed routes by comparing to a different way to generate them; second, we analyse the decision of transporting the remaining passengers directly by ODRP, by comparing to a feeder-trunk scheme.[9]

5.5.1. The G-benchmark

The first benchmark is to evaluate the idea of building the fixed-routes via following the ODRP paths. That is, in this benchmark, we use an alternative method to compute the routes. To do so, we adapt the method proposed by Barabino (2009), which we denote the G-Benchmark (G stands for "greedy") because its rationale is to iteratively connect the most-demanded origin–destination pairs. The pseudo-code is described in Algorithm 2, where we use $M = OD(Q)$ to represent that M is the origin–destination matrix associated to the set Q, i.e. for every pair of nodes $i, j \in V'$

$$M_{i,j} = |\{r \in Q : o_r = i, d_r = j\}| \tag{10}$$

As we are assuming that buses can only drive in the reduced graph, in Algorithm 2 we do not use the exact origins and destinations, but the closest node out of those that are in the reduced graph. To make a fair comparison, the rest of the method is kept the same, i.e., all the users that have no available line at the end will perform their whole trips via ODRP.

[9] There exist some more sophisticated algorithms (as discussed in Section 2). However, most of them have a different set of assumptions — for example, regarding demand elasticity, such as Basciftci and Van Hentenryck (2023), or the possibility of using fares to induce certain behaviours, as explored by Banerjee et al. (2021). Therefore, direct comparisons are barely possible. We opt for these two benchmarks because they allow for a direct qualitative analysis of the two main contributions of this paper as pinpointed in Section 2.1.

Algorithm 2 Benchmark: Greedy algorithm to compute the fixed routes.

Input: $(o_r, d_r)_{r\in\mathscr{R}}, \Delta_4$ `// Δ4 defines when we stop iterating.`
$Q \leftarrow \mathscr{R}$ `// The passengers yet to be assigned.`
$\mathscr{L} \leftarrow \emptyset$ `// The set of lines.`
while TRUE **do** `// This condition will be broken when the new added line does not have enough load.`
  $M \leftarrow OD(Q)$ `// The updated origin--destination matrix`
  $(i^*, j^*) = \text{argmax}_{i,j\in V'} M_{i,j}$ `// We select the most demanded OD pair`
  $L \leftarrow$ Shortest path from i^* to j^*
  $Pax_L \leftarrow \{r \in Q : o_r, d_r$ are at walking distance from L and resulting detour is $< \Omega_v\}$
  if $|Pax_L| < \Delta_4$ **then**
    Break While
  else
    $\mathscr{L} \leftarrow \mathscr{L} \cup \{L\}$
    $Q \leftarrow Q \setminus Pax_L$
  end if
end while

5.5.2. The F-benchmark

The second benchmark is to evaluate the merits of fully transporting the remaining passengers via ODRP, instead of a first-last-mile-connector. In this case, the alternative is a feeder-trunk scheme, where the ODRP vehicles are *feeders* that bring the passengers towards the *trunk* fixed lines. The feeder-trunk scheme has the potential to (i) reduce the number of ODRP vehicles because trips become shorter, and (ii) improve the quality of service of the fixed-lines thanks to the well-known sources of scale economies in public transport (Fielbaum et al., 2020). On the other hand, they induce up to two extra transfers for each passenger transported this way, which is uncomfortable for the users. As usual in the public transport design literature, we model this discomfort through a *transfer penalty* P_T, i.e., every time a transfer occurs we add a value P_T to the total costs. Jara-Diaz et al. (2022) find that P_T is in a range of 13–18 equivalent in-vehicle minutes; to make a robust comparison, we assume the minimum value (i.e., the most favourable for the feeder-trunk scheme):

$$P_T = 13\alpha_v \tag{11}$$

In Eq. (11) it is assumed that α_v is in units of [$/minute].

This benchmark, denoted as the F-Benchmark, works as follows. The set of fixed routes is the same as in the original method, i.e., the output of Algorithm 1. All the passengers that were not assigned to any fixed line (i.e., those that are in Q when Algorithm 1 finishes) will perform the trip in a feeder-trunk fashion. To do this, we find for each of them the line and its origin and destination stops that minimise the total travelling time. After that, we do the following:

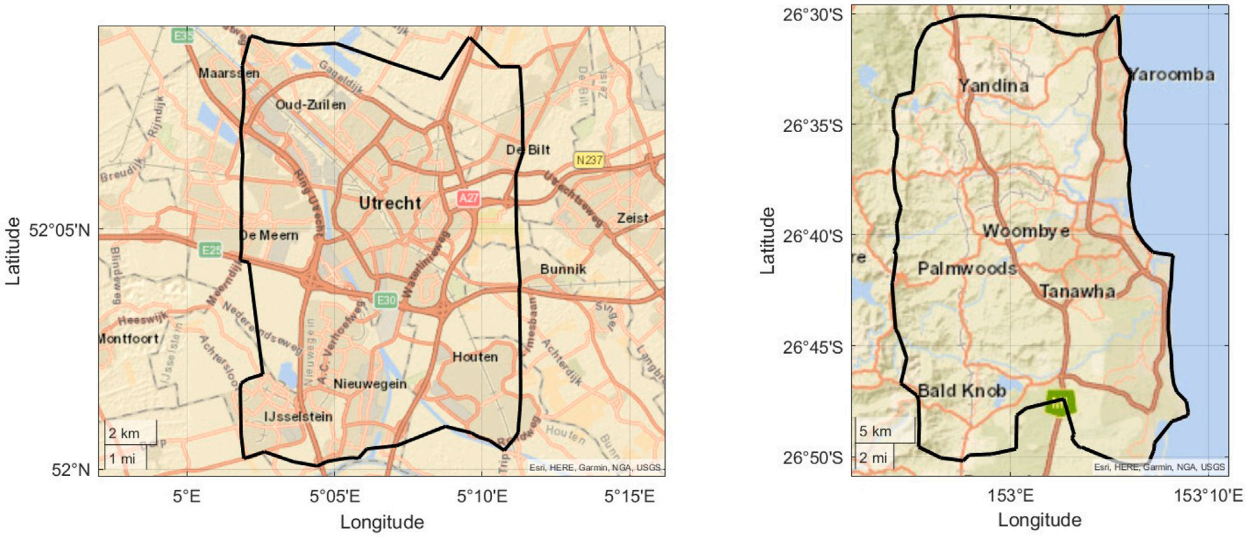
1. Compute the optimal frequencies, following the same procedure explained in Section 5.2.1, namely minimising the sum of Eqs. (8)–(9). Note that the set of users per line is now greater than in the original method, because we have added those users that utilise the fixed routes as trunks.
2. We operate ODRP to bring users from their origins (perhaps with a walk) to the corresponding origin bus stop, and from the destination bus stop to the final destination. The only exception is that sometimes either the origin or the destination can be at walking distance from the bus line, case in which ODRP serves only one leg and the corresponding user faces only one transfer. Note that it is not possible that both ends are at walking distance from the fixed line, as in said case that passenger would have been assigned to that line by Algorithm 1.

It is worth commenting that the feeder-trunk structure could be further exploited to increase ODRP's efficiency, such as aiming for a coordination between bus and flexible vehicles arrivals (Kim and Schonfeld, 2014), or developing some tailored anticipatory techniques that leverage the fact that requests will be concentrated in some nodes in the graph — the bus stops (Fielbaum et al., 2022b). These specific enhancements are beyond the scope of this paper and regarded as future research.

6. Experiments and results

6.1. Description of the scenarios

We run simulations utilising the real-life networks and bus demands from Utrecht, The Netherlands, and a section of the Sunshine Coast, Queensland, Australia, specifically the area covered by Lines 600, 602, 603, 606, 607, 609, 611, 614, 615, 616, 617, 618, and 619 . Both networks were obtained via Open Street Maps. The demand dataset was generated by QBuzz (a local public transport company) in the case of Utrecht, and is publicly available in Queensland. In both cases, the demand is known stop-to-stop, so we randomly distribute it to the nearby nodes following a simple procedure proposed by Fielbaum et al. (2023b). In Utrecht, a subsample of the total demand was created to match the demand size reported by Durán-Micco et al. (2022), who study the Utrecht



(a) Utrecht (b) Sunshine Coast

Fig. 5. The areas where we simulate our methods.

Table 1
Number of nodes and edges in the original networks and in the ones obtained after the reduction.

Scenario	Nodes		Edges	
	Original	Reduced	Original	Reduced
Utrecht	9,616	8,553	22,179	18,955
Sunshine Coast	12,455	7,262	18,056	14,742

case in depth. In the Sunshine Coast, the demand is aggregated at a monthly level and by periods, so we subsampled assuming a uniform distribution among weekdays. Both areas are depicted in Fig. 5.

Utrecht presents a much higher demand than the Sunshine Coast, so we only consider the morning peak period 6–9 AM, whereas in the Sunshine Coast we consider a daily demand. The total demand sets are formed by 26,897 requests in Utrecht and 2876 requests in the Sunshine Coast.

Reduction of the graphs. We follow the method discussed in Section 4 to reduce the graphs and maintain only a subset of the nodes and arcs, namely the most important ones. In both cases, using $H = 100$, in just one iteration we arrive at the resulting network. The sizes of the original and the reduced networks are shown in Table 1. It is worth mentioning that the actual lines operating within the same Sunshine Coast zone utilise 5264 edges. This shows that our method achieves a reasonable balance in effectively reducing the size of the original network, while not being confined to streets currently in use.

6.2. Results

We execute our method as explained in the previous sections, and considering two different scenarios regarding technology: human-driven or automated vehicles. The main change is that c_{BC} becomes much lower when vehicles are automated, due to the absence of the driver's salary; as a response, c_{KC} becomes greater, but the total cost of a vehicle with any realistic capacity is still lower thanks to automation. The corresponding parameters are obtained from Fielbaum et al. (2023a), which follows Tirachini and Antoniou (2020). The numeric values of all of the parameters are summarised in Table 3 in Appendix A. The results for the main method, the two mixed benchmarks, and a simulation containing only ODRP, are shown in Table 2.

The most important conclusions are as follows:

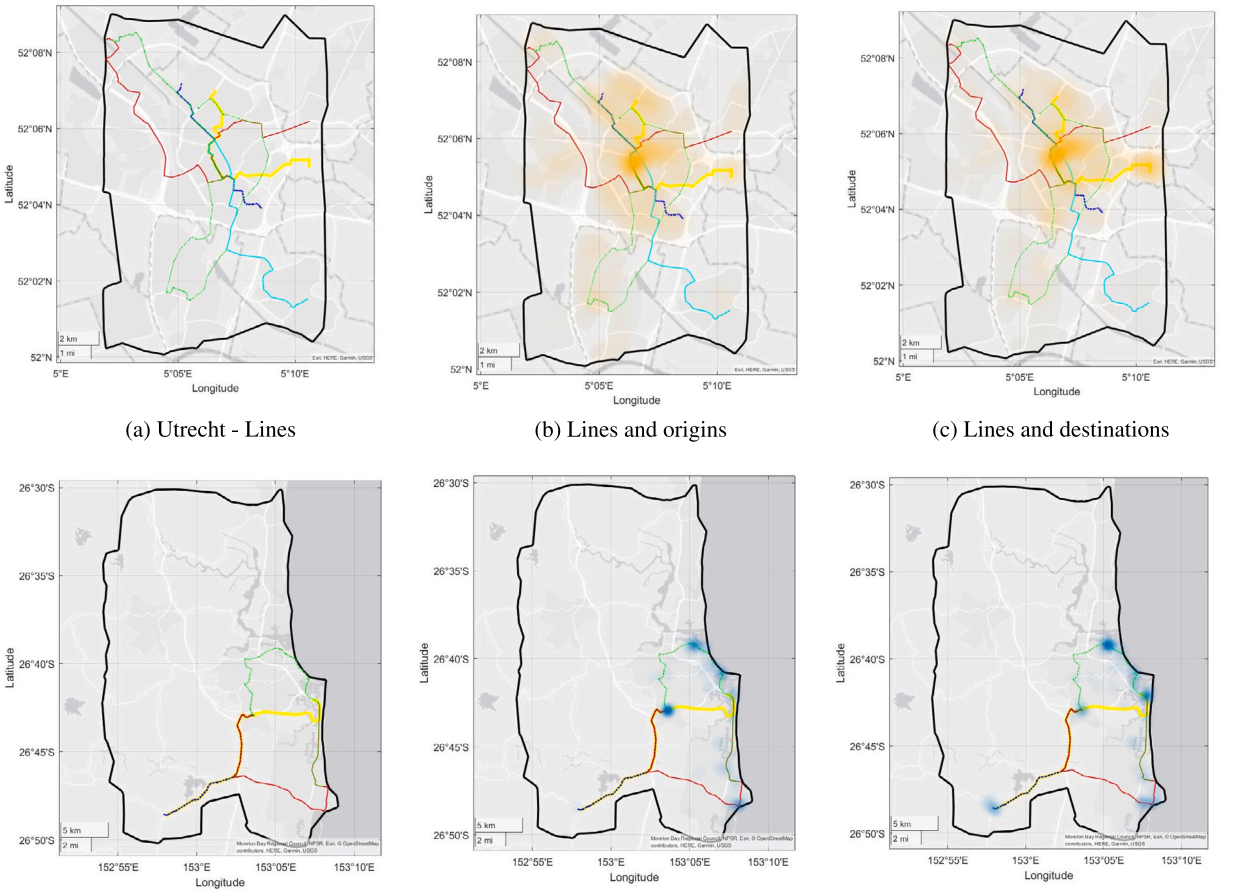
- The results of our method outperform all the benchmarks in all of the scenarios, with the sole exception of Utrech/human-driven, where the F-Benchmark is more convenient. Recall that the F-Benchmark is also based on the fixed-routes outputted by Algorithm 1. These results show that our method, purely based on the trajectories followed by the flexible vehicles, entails a competitive way to decide on a whole mixed network, including the fixed-route lines.
- The G-Benchmark is always worse than ours. In all of the scenarios, it requires more lines and serves fewer passengers. Again, this implies that tracking the routes of the ODRP vehicles when the service is not door-to-door works very efficiently, finding paths that are useful to many passengers, which translates into a small number of fixed-routes that carry a large demand.

Table 2
Results of our method compared to several benchmarks.

Index	Mixed network	G-Benchmark	F-Benchmark	Only ODRP
Utrecht/Human-driven				
On-demand fleet	889	1097	510	1725
Fixed fleet	209	235	408	–
N° of fixed lines	5	11	5	–
Passengers served via fixed lines	74.7%	60%	100%	–
Av. Walk [min]	11.4	9.8	10.1	1.02
Av. Wait [min]	3.18	3.18	1.77	2.53
Av. In-Vehicle [min]	23.8	20.64	25.4	16.3
N° of transfers	–	–	7027	–
Av. cost [US$]	5.79	6.18	5.41	6.69
Utrecht/Automated vehicles				
On-demand fleet	934	1111	522	1829
Fixed fleet	295	326	473	–
N° of fixed lines	5	11	5	–
Passengers served via fixed lines	74.7%	60%	100%	–
Av. Walk [min]	11.4	9.75	10.1	0.9
Av. Wait [min]	1.2	1.58	1.53	2.72
Av. In-Vehicle [min]	17	17.9	24.9	17.3
N° of transfers	–	–	7027	–
Av. cost [US$]	3.7	3.76	4.1	4.02
Sunshine Coast/Human-driven				
On-demand fleet	91	132	79	258
Fixed fleet	28	22	38	–
N° of fixed lines	4	7	4	–
Passengers served via fixed lines	76.15%	57.4%	100%	–
Av. Walk [min]	10.7	6.77	5.8	1.99
Av. Wait [min]	7.6	6.87	11.8	3.35
Av. In-Vehicle [min]	25.9	23.2	29.3	20.9
N° of transfers	–	–	664	–
Av. cost [US$]	6.3	6.75	6.84	9.03
Sunshine Coast/Automated vehicles				
On-demand fleet	91	139	82	258
Fixed fleet	51	26	44	–
N° of fixed lines	4	7	4	–
Passengers served via fixed lines	76.15%	57.4%	100%	–
Av. Walk [min]	7.5	6.7	5.8	2
Av. Wait [min]	6.2	6.2	10.5	3.4
Av. In-Vehicle [min]	25.5	23.2	29.2	20.9
N° of transfers	–	–	664	–
Av. cost [US$]	4.35	5.23	5.74	5.23

- The F-Benchmark reduces drastically the required number of ODRP vehicles, which happens because now those trips are shorter (towards the closest bus stop instead of the destination). On the other hand, the number of buses is increased. The resulting total number of vehicles is lower, which is why this method is competitive when vehicles require a driver.
- For those remaining users that cannot commute just through the fixed lines, we have proposed two alternatives, namely (i) being fully served via ODRP, or (ii) using ODRP to serve the first and last legs of the trip. This suggests a natural question: how to decide which of these two alternatives (full-trip or first-and-last-mile) should be assigned to each of the remaining users?

  Our method and the F-Benchmark can be seen as two extreme answers to that question. While in our method the answer is "all of them", in the F-Benchmark, the answer is "none". The fact that each method can be more convenient depending on the circumstances suggests that some intermediate alternative could be even better, which poses an interesting direction for further research.
- Utilising only ODRP is never optimal, due to the substantial operational costs incurred from the increased number of vehicles required.black
- In general, results are better in Utrecht, which does not come as a surprise as its demand is much larger and both fixed-route lines and ODRP present scale economies.
- Automated vehicles generate the same set of fixed lines, but the optimisation of their frequencies varies significantly. As a result, operational costs are substantially reduced despite the increased fleet sizes. Using more buses provides a better quality of service to the users. This dual benefit to users and operators is aligned with the previous literature on public transport automation (Fielbaum, 2020; Tirachini and Antoniou, 2020).



(a) Utrecht - Lines (b) Lines and origins (c) Lines and destinations

(d) Sunshine Coast - Lines (e) Lines and origins (f) Lines and destinations

Fig. 6. The lines generated by our methods (left), drawn together with a heatmap of the requests' origins (centre) and destinations (right). The top row shows the results in Utrecht and the bottom row in the Sunshine Coast.

Results are promising. They clearly show that the accumulation of flexible vehicles in some streets works as an effective mark to decide where to allocate public transport lines. Algorithm 1 is already an efficient way to do so, and enhanced versions might obtain even better results. On the other hand, transfers might play a useful role, but there is room to serve some of the trips directly by ODRP, especially if vehicles are automated.

To obtain a better intuition of the results created by our method, in Fig. 6 we depict the routes of the fixed lines in both scenarios. They are drawn by themselves, and also coupled with heatmaps of the origins and destinations. In both cases, different routes coincide in some segments, whereas other segments cover specific areas of the map. The most demanded sectors of the maps, both as origins and destinations, are strongly served as we now detail:

- In Utrecht, there is one zone close to the centre of the graph that is both a strong attractor and generator of trips, and all of the five lines pass through there. There is one additional strong destination at the right of the network, which is served by the yellow line. In general, there is an evident visual correlation between the orange areas of the map and the routes of the fixed lines.
- In the Sunshine Coast the demand is much more sparse, represented by some separate blue points in the map. Crucially, all such blue points are covered by at least one fixed line, and each fixed line serves at least one blue point.

As such, the visualisations shown in Fig. 6 verify that our method effectively identifies the hot zones as origins and destinations, and allocate fixed lines to connect them. Going further back, this means that flexible vehicles accumulated in those areas; as users walk, some vehicles could follow repeated paths to connect nodes that are different but located close to each other, belonging to the same hot zone.

7. Conclusions

On-demand ridepooling systems route vehicles without the constraints imposed by fixed routes. Nevertheless, as street networks are typically hierarchical, vehicles tend to concentrate in the most important streets and arcs, especially if the system is not door-to-door so that users can be told to walk towards the higher-hierarchy arcs. In this paper, we take this simple idea to propose a method that builds mixed public transport networks. Namely, we simulate as if all of the demand was to be served by ODRP only, and develop a tailored algorithm that allocates fixed lines on the paths that were followed by the largest number of users. The remaining users are served with on-demand ridepooling in the mixed network.

We study the merits of this method utilising the real-life public transport demand datasets from Utrecht, The Netherlands, and The Sunshine Coast, Australia. Our results show that the method we propose to build the fixed-routes outperform a benchmark method found in the literature, and that serving the remaining passengers via direct trips is usually, but not always, better than a feeder-trunk scheme.

The last commented aspect suggests a relevant question for future research: Which users should be transported towards the fixed lines to make a transfers, and which should travel directly from origin to destination via on-demand ridepooling? Other relevant questions to extend this paper are including transfers within the fixed network, considering users' mode and route choice in real time, developing more sophisticated algorithms that might outperform the greedy one we have proposed in this paper, maximising profit with elastic demand, and studying the robustness of the method against unexpected events or variations in the demand.

CRediT authorship contribution statement

Andres Fielbaum: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Javier Alonso-Mora:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing.

Acknowledgements

The authors thank Dr. Javier Duran-Micco, from Vrije Universiteit Brussel, and the QBuzz company, for their help to obtain the Utrech data. The comments and suggestions of the anonymous reviewers significantly helped improve the scientific quality and readability of this paper.

Appendix A

See Table 3.

Table 3
Numerical value of the parameters used in the simulations. In the scenario of only ODRP in Utrecht the max waiting time was further reduced to 5 min, as in Alonso-Mora et al. (2017), to reduce the computational burden; the same reason is why we use a shorter rolling horizon in Utrecht. Operators' and users' costs parameters are taken from Fielbaum et al. (2023a).

Parameter	Meaning	Numeric value
General parameters		
Ω_a	Max walking time	20 [min]
Ω_w	Max waiting time	10 [min]
Ω_v	Max detour time	15 [min]
α_a	Cost of one walking hour	4.64 [US$]
α_w	Cost of one waiting hour	4.64 [US$]
α_v	Cost of one in-vehicle hour	2.32 [US$]
c_{BO}	Operational cost per hour-vehicle	1.13 [US $]
c_{KO}	Operational cost per hour-seat	0.074
H	Graph-reduction parameter	100
Human-driven		
c_{BC}	Fixed cost per vehicle	78.1 [US$]
c_{KC}	Fixed cost per seat	1.2 [US $]
Automated vehicles		
c_{BC}	Fixed cost per minute-vehicle	24.6 [US$]
c_{KC}	Fixed cost per minute-seat	2.1 [US$]

(*continued on next page*)

Table 3 (*continued*).

Parameter	Meaning	Numeric value
Scenario-specific parameters: Utrecht		
δ_t	Rolling horizon	30 [s]
κ	ODRP vehicles' capacity	4
Scenario-specific parameters: Sunshine Coast		
δ_t	Rolling horizon	60 [s]
κ	ODRP vehicles' capacity	3

References

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., Rus, D., 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proc. Natl. Acad. Sci. 114 (3), 462–467.
Arriagada, J., Munizaga, M.A., Guevara, C.A., Prato, C., 2022. Unveiling route choice strategy heterogeneity from smart card data in a large-scale public transport network. Transp. Res. C 134, 103467.
Badia, H., Estrada, M., Robusté, F., 2014. Competitive transit network design in cities with radial street patterns. Transp. Res. B 59, 161–181.
Badia, H., Estrada, M., Robusté, F., 2016. Bus network structure and mobility pattern: A monocentric analytical approach on a grid street layout. Transp. Res. B 93, 37–56.
Banerjee, S., Hssaine, C., Luo, Q., Samaranayake, S., 2021. Plan your system and price for free: Fast algorithms for multimodal transit operations. Available at SSRN 3972423.
Barabino, B., 2009. Transit bus route network design: A model and its application in a real network. In: 15th International Conference on Urban Transport and the Environment, Urban Transport 2009, June 22, 2009-June 24, 2009, Bologna, Italy, vol. 107, pp. 369–382.
Basciftci, B., Van Hentenryck, P., 2023. Capturing travel mode adoption in designing on-demand multimodal transit systems. Transp. Sci. 57 (2), 351–375.
Bertsimas, D., Sian Ng, Y., Yan, J., 2020. Joint frequency-setting and pricing optimization on multimodal transit networks at scale. Transp. Sci. 54 (3), 839–853.
Borndörfer, R., Grötschel, M., Pfetsch, M.E., 2007. A column-generation approach to line planning in public transport. Transp. Sci. 41 (1), 123–132.
Bürstlein, J., López, D., Farooq, B., 2021. Exploring first-mile on-demand transit solutions for North American suburbia: A case study of Markham, Canada. Transp. Res. Part A: Policy Pract. 153, 261–283.
Calabrò, G., Araldo, A., Oh, S., Seshadri, R., Inturri, G., Ben-Akiva, M., 2021. Integrating fixed and demand-responsive transportation for flexible transit network design. In: TRB 2021: 100th Annual Meeting of the Transportation Research Board. Transportation Research Board, pp. TRBAM–21.
Cancela, H., Mauttone, A., Urquhart, M.E., 2015. Mathematical programming formulations for transit network design. Transp. Res. B 77, 17–37.
Chen, J., Liu, Z., Wang, S., Chen, X., 2018. Continuum approximation modeling of transit network design considering local route service and short-turn strategy. Transp. Res. Part E: Logist. Transp. Rev. 119, 165–188.
Daganzo, C.F., 2010. Structure of competitive transit networks. Transp. Res. B 44 (4), 434–446.
Dubois, D., Bel, G., Llibre, M., 1979. A set of methods in transportation network synthesis and analysis. J. Oper. Res. Soc. 30 (9), 797–808.
Durán-Micco, J., van Kooten Niekerk, M., Vansteenwegen, P., 2022. Designing bus line plans for realistic cases-the Utrecht case study. Expert Syst. Appl. 187, 115918.
Durán-Micco, J., Vansteenwegen, P., 2022. A survey on the transit network design and frequency setting problem. Public Transp. 14 (1), 155–190.
Fielbaum, A., 2020. Strategic public transport design using autonomous vehicles and other new technologies. Int. J. Intell. Transp. Syst. Res. 18 (2), 183–191.
Fielbaum, A., Alonso-Mora, J., 2020. Unreliability in ridesharing systems: Measuring changes in users' times due to new requests. Transp. Res. C 121, 102831.
Fielbaum, A., Bai, X., Alonso-Mora, J., 2021. On-demand ridesharing with optimized pick-up and drop-off walking locations. Transp. Res. C 126, 103061.
Fielbaum, A., Jara-Diaz, S., Alonso-Mora, J., 2022a. Beyond the last mile: Different spatial strategies to integrate on-demand services into public transport in a simplified city. Available at SSRN 4240986.
Fielbaum, A., Jara-Diaz, S., Gschwender, A., 2016. Optimal public transport networks for a general urban structure. Transp. Res. B 94, 298–313.
Fielbaum, A., Jara-Diaz, S., Gschwender, A., 2017. A parametric description of cities for the normative analysis of transport systems. Netw. Spat. Econ. 17, 343–365.
Fielbaum, A., Jara-Díaz, S., Gschwender, A., 2018. Transit line structures in a general parametric city: The role of heuristics. Transp. Sci. 52 (5), 1092–1105.
Fielbaum, A., Jara-Diaz, S., Gschwender, A., 2020. Beyond the mohring effect: Scale economies induced by transit lines structures design. Econ. Transp. 22, 100163.
Fielbaum, A., Kronmueller, M., Alonso-Mora, J., 2022b. Anticipatory routing methods for an on-demand ridepooling mobility system. Transportation 49 (6), 1921–1962.
Fielbaum, A., Tirachini, A., Alonso-Mora, J., 2023a. Economies and diseconomies of scale in on-demand ridepooling systems. Econ. Transp. 34, 100313.
Fielbaum, A., Tirachini, A., Alonso-Mora, J., 2023b. Improving public transportation via line-based integration of on-demand ridepooling. submitted for publication.
Figueiredo, L., Amorim, L., 2007. Decoding the Urban Grid: or Why Cities are Neither Trees nor Perfect Grids. ITU Faculty of Architectur.
Garcia-Martinez, A., Cascajo, R., Jara-Diaz, S.R., Chowdhury, S., Monzon, A., 2018. Transfer penalties in multimodal public transport networks. Transp. Res. Part A: Policy Pract. 114, 52–66.
Grahn, R., Qian, S., Hendrickson, C., 2023. Environmental impacts of first-mile-last-mile systems with shared autonomous electric vehicles and ridehailing. Transp. Res. Part D: Transp. Environ. 118, 103677.
Guihaire, V., Hao, J.-K., 2008. Transit network design and scheduling: A global review. Transp. Res. Part A: Policy Pract. 42 (10), 1251–1273.
Gurumurthy, K.M., Kockelman, K.M., Zuniga-Garcia, N., 2020. First-mile-last-mile collector-distributor system using shared autonomous mobility. Transp. Res. Rec. 2674 (10), 638–647.
He, P., Jin, J.G., Schulte, F., Trépanier, M., 2023. Optimizing first-mile ridesharing services to intercity transit hubs. Transp. Res. C 150, 104082.
Iliopoulou, C., Kepaptsoglou, K., Vlahogianni, E., 2019. Metaheuristics for the transit route network design problem: A review and comparative analysis. Public Transp. 11 (3), 487–521.
Jara-Díaz, S.R., Gschwender, A., 2003. From the single line model to the spatial structure of transit services: corridors or direct? J. Transp. Econ. Policy (JTEP) 37 (2), 261–277.
Jara-Diaz, S., Monzon, A., Cascajo, R., Garcia-Martinez, A., 2022. An international time equivalency of the pure transfer penalty in urban transit trips: Closing the gap. Transp. Policy 125, 48–55.
Jha, S.B., Jha, J.K., Tiwari, M.K., 2019. A multi-objective meta-heuristic approach for transit network design and frequency setting problem in a bus transit system. Comput. Ind. Eng. 130, 166–186.
Kim, M.E., Schonfeld, P., 2014. Integration of conventional and flexible bus services with timed transfers. Transp. Res. B 68, 76–97.

Kim, J., Tak, S., Lee, J., Yeo, H., 2023. Integrated design framework for on-demand transit system based on spatiotemporal mobility patterns. Transp. Res. C 150, 104087.
Kucharski, R., Cats, O., 2022. Hyper-pool: pooling private trips into high-occupancy transit-like attractive shared rides. arXiv preprint arXiv:2206.05940.
Ma, T.-Y., Rasulkhani, S., Chow, J.Y., Klein, S., 2019. A dynamic ridesharing dispatch and idle vehicle repositioning strategy with integrated transit transfers. Transp. Res. Part E: Logist. Transp. Rev. 128, 417–442.
Maheo, A., Kilby, P., Van Hentenryck, P., 2019. Benders decomposition for the design of a hub and shuttle public transit system. Transp. Sci. 53 (1), 77–88.
Masucci, A.P., Stanilov, K., Batty, M., 2014. Exploring the evolution of London's street network in the information space: A dual approach. Phys. Rev. E 89 (1), 012805.
Mauttone, A., Cancela, H., Urquhart, M.E., 2021. Public transportation. In: Network Design with Applications To Transportation and Logistics. Springer, pp. 539–565.
Mohamed, M., Rye, T., Fonzone, A., 2020. The utilisation and user characteristics of uber services in London. Transp. Plan. Technol. 43 (4), 424–441.
Nnene, O.A., Zuidgeest, M.H., Joubert, J.W., 2023. Optimising transit networks using simulation-based techniques. In: Transportation Systems Technology and Integrated Management. Springer, pp. 317–345.
Perera, S., Ho, C., Hensher, D., 2020. Resurgence of demand responsive transit services–Insights from BRIDJ trials in inner west of Sydney, Australia. Res. Transp. Econ. 83, 100904.
Pinto, H.K., Hyland, M.F., Mahmassani, H.S., Verbas, I.Ö., 2020. Joint design of multimodal transit networks and shared autonomous mobility fleets. Transp. Res. C 113, 2–20.
Porta, S., Crucitti, P., Latora, V., 2006. The network analysis of urban streets: A dual approach. Physica A 369 (2), 853–866.
Ronald, N., Thompson, R., Winter, S., 2015. Simulating demand-responsive transportation: A review of agent-based approaches. Transp. Rev. 35 (4), 404–421.
Sanders, P., Schultes, D., 2005. Highway hierarchies hasten exact shortest path queries. In: European Symposium on Algorithms. Springer, pp. 568–579.
Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S.H., Ratti, C., 2014. Quantifying the benefits of vehicle pooling with shareability networks. Proc. Natl. Acad. Sci. 111 (37), 13290–13294.
Shen, Y., Zhang, H., Zhao, J., 2018. Integrating shared autonomous vehicle in public transportation system: A supply-side simulation of the first-mile service in Singapore. Transp. Res. Part A: Policy Pract. 113, 125–136.
Tachet, R., Sagarra, O., Santi, P., Resta, G., Szell, M., Strogatz, S.H., Ratti, C., 2017. Scaling law of urban ride sharing. Sci. Rep. 7 (1), 1–6.
Tang, X., Yang, J., Lin, X., He, F., Si, J., 2023. Dynamic operations of an integrated mobility service system of fixed-route transits and flexible electric buses. Transp. Res. Part E: Logist. Transp. Rev. 173, 103081.
Thorhauge, M., Jensen, A.F., Rich, J., 2022. Effects of autonomous first-and last mile transport in the transport chain. Transp. Res. Interdiscip. Perspect. 15, 100623.
Tirachini, A., Antoniou, C., 2020. The economics of automated public transport: Effects on operator cost, travel time, fare and subsidy. Econ. Transp. 21, 100151.
Tirachini, A., Hensher, D.A., 2011. Bus congestion, optimal infrastructure investment and the choice of a fare collection system in dedicated bus corridors. Transp. Res. B 45 (5), 828–844.
Vansteenwegen, P., Melis, L., Aktaş, D., Montenegro, B.D.G., Vieira, F.S., Sörensen, K., 2022. A survey on demand-responsive public bus systems. Transp. Res. C 137, 103573.
Zuniga-Garcia, N., Gurumurthy, K.M., Yahia, C.N., Kockelman, K.M., Machemehl, R.B., 2022. Integrating shared mobility services with public transit in areas of low demand. J. Public Transp. 24, 100032.