mathematics

MDPI

Article

Autonomous Ride-Sharing Service Using Graph Embedding and Dial-a-Ride Problem: Application to the Last-Mile Transit in Lyon City

Omar Rifki

Laboratoire d'Informatique Signal et Image de la Côte d'Opale (LISIC), University Littoral Côte d'Opale, UR 4491, F-62100 Calais, France; omar.rifki@univ-littoral.fr

Abstract: Autonomous vehicles are anticipated to revolutionize ride-sharing services and subsequently enhance the public transportation systems through a first–last-mile transit service. Within this context, a fleet of autonomous vehicles can be modeled as a Dial-a-Ride Problem with certain features. In this study, we propose a holistic solving approach to this problem, which combines the mixed-integer linear programming formulation with a novel graph dimension reduction method based on the graph embedding framework. This latter method is effective since accounting for heterogeneous travel demands of the covered territory tends to increase the size of the routing graph drastically, thus rendering the exact solving of small instances computationally infeasible. An application is provided for the real transport demand of the industrial district of "Vallée de la Chimie" in Lyon city, France. Instances involving more than 50 transport requests and 10 vehicles could be easily solved. Results suggest that this method generates routes of reduced nodes with lower vehicle kilometers traveled compared to the constrained K-means-based reduction. Reductions in terms of GHG emissions are estimated to be around 75% less than the private vehicle mode in our applied service. A sensitivity analysis is also provided.

Keywords: shared autonomous mobility; first–last-mile service; vehicle routing; node embeddings

MSC: 90B20; 68R12

Citation: Rifki, O. Autonomous Ride-Sharing Service Using Graph Embedding and Dial-a-Ride Problem: Application to the Last-Mile Transit in Lyon City. *Mathematics* **2024**, *12*, 546. https://doi.org/10.3390/math12040546

Academic Editor: Andrea Scozzari

Received: 29 December 2023
Revised: 3 February 2024
Accepted: 8 February 2024
Published: 10 February 2024

1. Introduction

The urban expansion in most cities around the world followed an automobile-oriented pattern, leading to the emergence of several low-density metropolitan areas, thus increasing the overall infrastructure costs for public transportation (PT) (Sinha, 2003 [1]). Since then, and due to this reason, PT systems have been relegated to a secondary role behind private car use in urban mobility. The PT cover for low-density areas and commercial and industrial districts outside the city is limited. Shared mobility, which is the shared use of vehicles, bicycles, and other means of transport to access PT networks, started to appear to be a viable solution to this issue, known as the first- and last-mile commute. The concept of shared mobility is not novel in itself and could be traced back to the 1990s when it was popular, and even to the year 1948 for the first car-sharing program. We refer to Shaheen & Chan (2016) [2] for a complete history of this matter. Yet, the major shift in shared mobility is still expected by many to happen with the advent of shared autonomous vehicles (SAV). Several AV manufacturers and service transportation provider companies have heavily invested in and adopted specific ride-sharing plans, e.g., Waymo in the Phoenix area (Waymo, 2020 [3]), Jaguar Land Rover (Jaguar Land Rover, 2020 [4]), and, eventually, Amazon, a fresh new competitor in this market share (Amazon, 2020 [5]).

The factors making SAV an attractive transit mode are mainly economic for both sides of customers and PT and transportation network entities. Due to lower operational and investment costs over fixed routes, transportation companies could propose services at

fair prices with a higher frequency. Higher customer satisfaction and time reductions may be equally obtained compared with other transit modes such as park-and-ride and active modes. Another argument in favor of SAV lies in the environmental sustainability of this technology, which is mainly due to the application of electric vehicles in SAV services. Autonomous fleets could reduce greenhouse gas (GHG) emissions by up to 73% compared to conventional taxi fleets (Bauer et al., 2018 [6]). It was shown that each SAV could replace around 11, respectively, from 5 to 10 privately owned vehicles, with an increase of 10%, resp., from 6 to 89% travel distance via agent-based simulations of the city of Austin, Texas (Fagnant & Kockelman, 2014 [7]), resp., and Lisbon (International Transport Forum, 2015 [8]). The existence of empty trips justifies this latter increase. All these favorable reports led to a resurgence of small-size experimentations of SAV services in Europe, the U.S., and globally. Just in Lyon city in France, where the case study of this paper is located, there are currently three experimentations: Navya shuttle buses in the Confluence area, around Groupama stadium, and Mia buses in the Meyzieu-Décine suburb area. Therefore, there is a great need for efficient design models for the short-time deployment of SAVs, especially to handle the first–last-mile transit issue (see Hyland & Mahmassani (2017) [9] for a taxonomy of SAV fleet management problem classes to accompany this current mobility shift). This study is not concerned with the SAV experimentations in Lyon city, but rather targets an area of the city with a great need for a last-mile solution.

A critical component of the design of an SAV service is the vehicle assignment or the process of matching customer requests to vehicles. Most SAV approaches rely on rule-based assignment methods, e.g., Gurumurthy & Kockelman (2018) [10]. Optimization models are almost not used here. This is because SAV studies are mostly on-demand systems, which are dynamic systems that require solving approaches that are computationally cost efficient and easy to implement (Narayanan et al., 2020 [11]). On the other hand, SAV models that are reservation based or, in other terms, static systems that could be solved beforehand rely, in general, on optimization. However, they are quite few in number, e.g., Levin (2017) [12] and Ma et al. (2017) [13]. Both Levin (2017) and Ma et al. (2017) proposed a linear programming model to modelize an SAV fleet. In this study, we propose using the famous routing problem of the Dial-a-Ride Problem (DARP), which is derived from the classical vehicle routing problem (VRP) with additional constraints to account for pickup and drop-off requests. VRP and DARP are both computationally intractable and can be pinned down as mixed-integer linear programming models. According to Ho et al. (2018) [14] and Gschwind & Irnich (2014) [15], the largest instance solved to optimality for the basic DARP with time windows is up to 8 vehicles and 96 requests, which is quite small for practical deployment of an SAV service. Therefore, a mechanism has to be found in order to reduce the dimension of the routing graph to be scalable for exact methods. To address this research gap, we develop a new reduction method based on the graph embedding framework, specifically around the node2vec algorithmic framework (Grover and Leskovec, 2016 [16]), which will be seen later. The choice of node2vec is due to its property to capture the topology of any given graph, which can provide helpful information to locate nodes to be merged in the routing problem graph. Thus, the main objective of this study is to propose a mechanism to reduce the size of the routing problem, in order to be solved by an exact algorithm. Specifically, we propose a matheuristic to solve a reservation-based SAV service.

A second objective is to design an SAV service that is as close as it would work in a real environment. First, this was carried out by the choice of an application territory. We chose an industrial district in Lyon city (France) called "Vallée de la Chimie" (VC). Although VC is one of the biggest chemical and petrochemical parks in Europe, it is poorly served by public transport systems. It presents a concrete application of the last-mile transit. This study focuses on a classical commute problem between the city center and suburbs where two transportation alternatives compete: highway and rapid transit system. Second, we aim to generate realistic traffic demand accounting for intermodality, i.e., using multiple transport modes on the same trip. The traditional four-step travel models could only

produce demand matrices for separate modes (McNally, 2007 [17]). We have relied on a Land Use and Transport Interactions (LUTI) model-based tool called OPTIREL to account for intermodality, without using agent-based simulations. OPTIREL has been previously applied to design new subway lines in Paris among other projects. The data we use in this study are real and based on OPTIREL software version 2020. Therefore, the problem we solve is tested on VC for the commute problem with a local rapid transit system.

The remainder of the paper is organized as follows. A brief literature review on the integration of the PT-SAV system, the DARP problem, and the graph embedding framework is presented in the next section. Section 3 introduces the optimization model and the solving approach, while Section 4 describes the study case and the generation of traffic demand. In Section 5, we experimentally evaluate the benefit of the designed last-mile service, before concluding with some final thoughts.

2. Literature Review

In this section, we briefly review the literature of interest in this paper within the following three subsections.

2.1. Integration of PT-SAV Systems

Several recent studies are starting to explore the benefits of integrating SAV services with existing PT systems. The most commonly adopted approach in this regard is agent-based simulations, whether it concerns the integration, e.g., Shen et al. (2018) [18], Pinto et al. (2020) [19], and Shan et al. (2021) [20], or the intersection of PT-SAV systems, e.g., Fagnant & Kockelman (2014) [7], or the replacement of PT by SAV, e.g., Yantao et al. (2021) [21]. Liang et al. (2016) [22] proposed an optimization model for an automated taxi service serving the last-mile transit of a train system. Their results are shown for a train station in Delft and include comparisons with human-driven taxis. Shen et al. (2018) [18] simulate several scenarios of integrating bus lines with SAVs for first-mile connectivity during the morning peak hours. Their results based on the Singapore PT show significant savings if low-demand buses are substituted by SAVs, savings that go up to 860 passenger car unit kilometers. Pinto et al. (2020) [19] proposed a bi-level optimization framework to simultaneously modelize the joint transit network parameters and the SAV mobility service. The lower-level problem is a dynamic combined mode choice–traveler assignment problem, while the upper lever is a modified transit network frequency setting problem. Their results are given for the Chicago metropolitan area. Shan et al. (2021) [20] developed a fixed-point algorithm to solve a joint system of railway transport and SAV. They provided a case study for the Melbourne railway extension. Al Maghraoui et al. (2020) [23] presented different attributes that affect travelers for SAV adoption depending on their current mode of transport. Several parameters enter into the integrated design of SAV. For instance, the locations of pickup and drop-off points can have a great impact on dynamic ride-sharing by SAV, as shown by Gurumurthy and Kockelman (2022) [24]. Special care has to be given to the design of the integration PT-SAV. More detailed reviews of overall SAV services are provided by Narayanan et al. (2020) [11] and Golbabaei et al. (2021) [25].

2.2. Dial-a-Ride Problems

Designing vehicle routes and schedules for collective people transportation such that each user request has a pickup and a drop-off point locations are referred to as Dial-a-Ride systems, as in early versions of this service, customers had to phone their requests. From a modeling perspective, DARP belongs to the family of routing problems. It is a variant of the vehicle routing problem (VRP), precisely the capacitated VRP with pickup and delivery and time windows, with hands-on transporting passengers rather than goods. Early solving approaches include Psaraftis (1980) [26], who solved the single-vehicle DARP using dynamic programming, and Jaw et al. (1986) [27], who developed one of the first heuristics to the multi-vehicle DARP. For the exact approaches, the Branch-and-Bound method and its derived algorithms are the most applied, e.g., Ropke et al. (2007) [28] and Gschwind &

Irnich (2014) [15]. Concerning metaheuristics, those based on local search, especially those combining local search elements with other metaheuristics currently constitute state-of-the-art solvers, e.g., Masmoudi et al. (2016) [29]. Applications of DARPs are overall real-life problems with a diverse range of backgrounds, starting from the standard application to the elderly and disabled people's transportation to healthcare services to the current emerging applications in public transportation and shared mobility (Ho et al. 2018 [14], Mourad et al. 2019 [30]). For a more comprehensive review of the problem classes, solving algorithms, and the applications of the DARP, see Cordeau & Laporte (2007) [31], Molenbruch et al. (2017) [32], and Ho et al. (2018) [14].

2.3. Graph Embeddings

Graph embedding is a powerful method to represent a graph into a low-dimensional vector space by preserving as much as possible its topological structure. It addresses the research question of efficient graph analytics since traditional methods suffer from the curse of dimensionality and space cost. Essential graph analytic tasks, such as graph classification, node clustering, and link prediction, become scalable to large instances. Representation learning and visualization is another research problem tackled by graph embeddings. There are three main categories of graph embeddings: probabilistic, matrix factorization-based, and deep learning-based methods (see Goyal & Ferrara, 2018 [33] for a survey on the topic). Probabilistic models learn the embedding of graphs through random walks. The samplings given by those walks can capture the neighborhood structure of nodes, connectivities, and other graph properties such as node centrality. Compared with other probabilistic models like DeepWalk (Perozzi et al., 2014 [34]) and LINE (Tang et al., 2015 [35]), the probabilistic model of node2vec (Grover & Leskovec, 2016 [16]) is shown to perform better. This is why we chose node2vec implementation for the graph embedding in the current study. The random walk exploration of this algorithm interpolates between breadth-first (BFS) and depth-first (DFS) searches in order to construct a more informative embedding.

3. Proposed Method

Figure 1 illustrates the flowchart of the proposed approach starting from the input data to the generated routes. The data collection is discussed in Section 4. Our approach is composed of two main distinct blocks: 1/optimization according to the DARP formulation, and 2/the graph reduction mechanism. Given a set of stations in the studied area, the method's inputs are the origin–destination (OD) matrix of the traffic demand, i.e., the number of customers traveling between any two stations, the duration of travel, and the traveled distance between all couples of stations. The earliest visit time, service time, and latest visit time for each OD couple of the demand matrix are also input data. Routing problems in their general formulation are defined on an underlying graph where the set of vertices corresponds to the depot and points to visit, and the set of arcs weighted with costs to travel, generally distances or travel times, represent the shortest paths between stations. On this defined routing graph, each point has to be visited only once by precisely one vehicle. Since each initial station could be a pickup and a drop-off point to all the remaining stations, and taking into account all traffic demand inputs of the OD matrix of the area, the dimension of the routing graph to be solved can quickly increase. If n is the total number of stations, the graph dimension could attain a maximum of $n(n-1)$ points to visit without counting the depots. We assume that the vehicle's capacity is greater or equal to every entry of the OD demand matrix. This will prevent us from duplicating the same O-D couple of stations in the routing graph to account for the total demand. The graph reduction mechanism we propose aims to detect groups of pickup and drop-off couple points to be merged, and output a reduced graph that can be easily solvable by a mixed-integer programming solver. Note that clustering and community detection algorithms could be used as well in the step of forming groups (This is different from cluster and route methods, which first apply a clustering algorithm to the input visiting points, and second generate optimal routes for each cluster, e.g., Chen et al. (2020) [36]).

We use the constrained K-means clustering method proposed by Bradley et al. (2000) [37] for comparison matter. Our method is overall a model-based heuristic around a central component of mathematical programming. More details of the two main parts of our model are presented in the following.

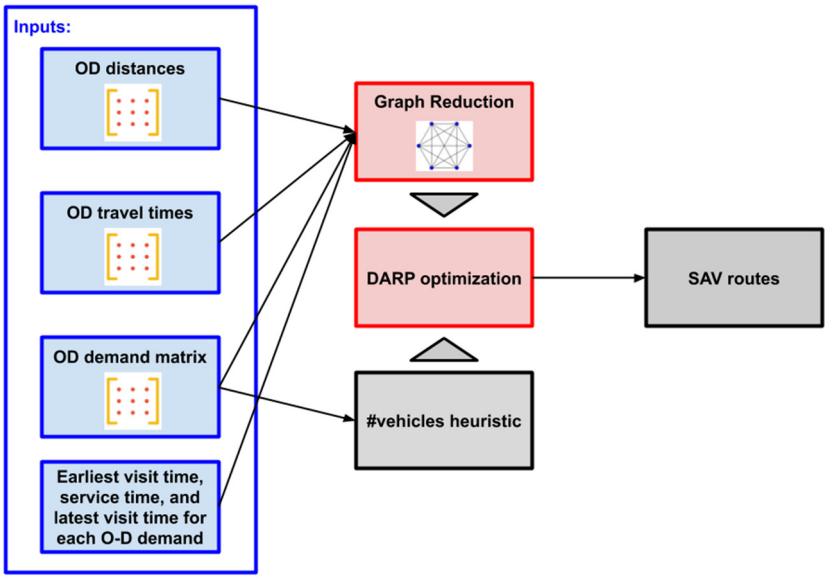

Figure 1. Flowchart of the proposed method.

3.1. Mathematical Formulation

In this study, we build on Cordeau's (2006) [38] three-index formulation x_{ij}^k of the DARP by considering a new objective function and adding new variables. The points to visit are partitioned into two sets: the set of pickup points P and the set of drop-off points D, such that each demand request of the OD demand matrix has a pickup point $i \in P$, a corresponding delivery point which will be denoted $deliv(i) \in D$, a maximum riding time L_i, and a maximum waiting time M_i. Each point $i \in P \cup D$ has a service time ser_i, the earliest time to visit tw_i^{early}, the latest time to visit tw_i^{late}, and a demand load denoted req_i, such that for $i \in P, req_i \geq 0$ and $req_{deliv(i)} = -req_i$.

K is the set of vehicles. Each vehicle $k \in K$ has a capacity $Q_k \in N$, GHG emissions up to E_k by kilometer, starts from the depot denoted s_k and ends at the depot denoted e_k. Let $D_{start} = \{s_k : k \in K\}$ be the set of all start depots and $D_{end} = \{e_k : k \in K\}$ the set of all end depots. For all $j \in D_{start} \cup D_{end}$, the values $serv_j = req_j = 0$ are fixed. We associate different start and end depots to each vehicle to make the model more general. However, all these depots may correspond to the same geographical location, which coincides with the case of our application to the transit station of the PT network.

The underlying routing graph $G = (V, A)$ has a vertex set equal to $V = P \cup D \cup D_{start} \cup D_{end}$. The arc set is defined as $A = \{(i,j) : (i \in D_{start}, j \in P) \vee (i \in D, j \in D_{end}) \vee (i, j \in P \cup D, i \neq j, i \neq deliv(j))\}$, and the weights of the graph are given by c_{ij} and d_{ij}, which denote, respectively, the duration of travel and the distance from i to j.

The decision variable x_{ij}^k indicates whether the arc (i,j) is traversed by vehicle $k \in K$ or not, r_i and w_i gives, respectively, the riding time and the waiting time for the request associated with pickup $i \in P$, t_i^k is the arrival time at point i using vehicle k, and q_i^k is the load of the vehicle k when leaving the point i.

In addition to minimizing total travel times for all vehicles, the degree of customer dissatisfaction in the sense of Psaraftis (1980) [26] is also minimized, plus the greenhouse gas emissions due to the transport of passengers. Plausible values of E_k are drawn from the

life-cycle assessment (LCA) conducted by Gawron et al. (2019) [39]. $(\omega_1, \omega_2, \omega_3)$ gives the weights between the three quantities in the objective function. The model is as follows.

$$\text{Min } \omega_1 \sum_{k\in K} \sum_{i,j\in V} c_{ij} x_{ij}^k + \omega_2 \sum_{i\in P} (r_i + w_i) + \omega_3 \sum_{k\in K} \sum_{i\in V} \sum_{j\in V} E_k d_{ij} x_{ij}^k \quad (1)$$

Subject to:

$$\sum_{i\in P} x_{s_k i}^k = \sum_{i\in D} x_{i e_k}^k = 1 \quad (k \in K), \quad (2)$$

$$\sum_{k\in K} \sum_{j\in V} x_{ij}^k = 1 \quad (i \in P), \quad (3)$$

$$\sum_{j\in V} x_{ij}^k - \sum_{j\in V} x_{deliv(i)j}^k = 0 \quad (i \in P, k \in K), \quad (4)$$

$$\sum_{j\in V} x_{ji}^k - \sum_{j\in V} x_{ij}^k = 0 \quad (i \in P \cup D, k \in K), \quad (5)$$

$$t_j^k \geq \left(t_i^k + ser_i + c_{ij}\right) x_{ij}^k \quad (i, j \in V, k \in K), \quad (6)$$

$$t_{deliv(i)}^k \geq t_i^k + ser_i + c_{ideliv(i)} \quad (i \in P, k \in K), \quad (7)$$

$$tw_i^{early} \leq t_i^k \leq tw_i^{late} \quad (i \in P \cup D, k \in K), \quad (8)$$

$$q_j^k \geq \left(q_i^k + req_i\right) x_{ij}^k \quad (i, j \in V, k \in K), \quad (9)$$

$$r_i \geq t_{deliv(i)}^k - \left(t_i^k + ser_i\right) \quad (i \in P, k \in K), \quad (10)$$

$$w_i \geq t_i^k - tw_i^{early} \quad (i \in P, k \in K), \quad (11)$$

$$c_{ideliv(i)} \leq r_i \leq L_i \quad (i \in P), \quad (12)$$

$$0 \leq w_i \leq M_i \quad (i \in P), \quad (13)$$

$$0 \leq q_i^k \leq Q_k \quad (i \in V, k \in K), \quad (14)$$

$$x_{ij}^k = 0 \vee 1 \quad (i, j \in V, k \in K). \quad (15)$$

The routing constraints (2)–(5) ensure that each vehicle starts and ends at its corresponding depot points in constraint (2), each request is answered in constraint (3), the same vehicle is used for pickup and drop-off in constraint (4), and the flow conservation in constraint (5). Constraint (6) tracks the service time, while constraint (7) ensures that pickup points are visited before their delivery points, and time window constraints are given in (8). Constraint (9) tracks the load of vehicles, which is needed later to compute the vehicle occupancy rate (VOR) indicator. Constraints (10) and (11) are, respectively, relative to the riding time and the waiting time for each request. Constraints (12)–(15) represent the binary and the bounding restrictions for the decision variables. Note that the current mixed-integer programming solvers can efficiently handle indicator constraints as in (6) and (9). Other useful constraints to add concern the battery management aspect, which includes the recharge time, the battery energy consumption, and detours to recharge stations as in Bongiovanni et al. (2019) [40]. Recent simulations (Vosooghi et al., 2020 [41]) suggest that the best strategy for SAV could be deploying batteries with more charging space to avoid charging the battery within the rush hours of the morning and the evening

peaks; otherwise, alternating charging and routing could drastically increase the passenger kilometer traveled. For this reason and for the time being, we suppose that battery charging operations are set outside the routing process and that vehicles are set to be initially charged before transporting customers.

For the heuristic to determine the number of vehicles $|K|$ (mentioned in Figure 1), we use the following formula for vehicles with an identical capacity $Q = Q_k$:

$$vehicles(m) = \sum_{i,j \in V^2, i \neq j} demand(i,j)/(n \times Q). \quad (16)$$

m is an estimate of the average number of O-D requests each vehicle is expected to answer during the time horizon (usually $m = 3$ or 4).

3.2. Graph Reduction Mechanism

The steps of the graph reduction are listed in the below Algorithm 1. At first, *node2vec* is applied to the routing graph G, which is introduced in the previous section and is weighted by the travel durations $\left(c_{ij}\right)_{(i,j)\in A}$, for the following inputs: the desired dimension size of the embedding d, the number of random walks to launch from each node and their length, plus other parameters related to the guided exploration strategy of the walks. Thereafter, each node of G can be represented by a feature vector of dimension d. To obtain a similarity matrix S for the nodes of G, we apply the cosine similarity, which is defined for any two real-valued vectors v_1 and v_2 as the cosine of the angle θ between them, i.e., $cos(\theta) = v_1 \cdot v_2/(\|v_1\| \times \|v_2\|)$. Other measures of similarity and distances could be also used at this step. This step is important because it gives a quantification of the similarity between any two nodes of G while taking into account as much as possible the topology structure of the graph.

The remaining steps concern merging the points to visit. We opt for merging pickup and drop-off (P-D) couples two by two. This gives us more flexibility in the rate of contraction of graph G and allows us to quickly compute the optimal order within the merged P-D couples. For each two P-D couples $(p_1, deliv(p_1), p_2, deliv(p_2))$, we compute the following average similarity using the similarity matrix S:

$$\begin{aligned} simil(p_1,p_2) \quad &= (S(p_1,p_2) + S(deliv(p_1),deliv(p_2)) + S(p_1,deliv(p_1)))/6 \\ &+(S(p_2,deliv(p_2)) + S(p_2,deliv(p_1)) + S(p_1,deliv(p_2)))/6. \end{aligned} \quad (17)$$

Which indicates how close the two couples are. Note that we affect higher input parameters for BFS over DFS in the guided search of *node2vec* to favor closer nodes within the vicinity of each node of G. After ranking pairs of the P-D couples by their similarity, we choose the top $r \times (|P| \cup |D|)/2$ pairs, with r is the rate of contraction of the graph. The following step proceeds by computing the optimal order of visiting points within each of the chosen pairs. There are only six possible orders respecting the pickup and drop-off constraints among the total 32, which are shown in Figure 2a. For each optimal order, we output two aggregated nodes in the novel graph, i.e., if an optimal order is for instance $(p_1, deliv(p_1), p_2, deliv(p_2))$, the new nodes will correspond to $p' = (p_1, deliv(p_1))$ and $deliv(p') = (p_2, deliv(p_2))$. The final step updates the graph weights involving the new nodes, as shown in Figure 2b. Distances between nodes are updated in the same manner. The graph reduction algorithm has an overall time complexity of $O\left(|V|^2\right)$, since *node2vec* runs in $O\left(|V|^2\right)$ and the number of pairs of P-D couples processed in the merging steps is at maximum $COMBIN(n = (|P| \cup |D|)/2, k = 2) \leq |V|^2/8$. This limits the total reduction rate of the routing graph to 50%, as only pairs of actual P-D couples are merged (transforming four points to two points).

Algorithm 1: ***node2vec*-based Graph Reduction**

Inputs: The routing graph $G = (V, A)$, the contraction rate r.
1. Apply the *node2vec* algorithm to G.
2. Compute the similarity matrix S of G using the cosine measure on the embedding of 1.
3. Compute for each two P-D couples $(p_1, deliv(p_1), p_2, deliv(p_2))$ the value $simil(p_1, p_2)$.
4. Select the top $r \times (|P| \cup |D|)/2$ similar pairs of P-D couples.
5. Compute the optimal order of points to visit within each chosen pair.
6. Update the graph's costs for the arcs going to/coming from the newly merged nodes.

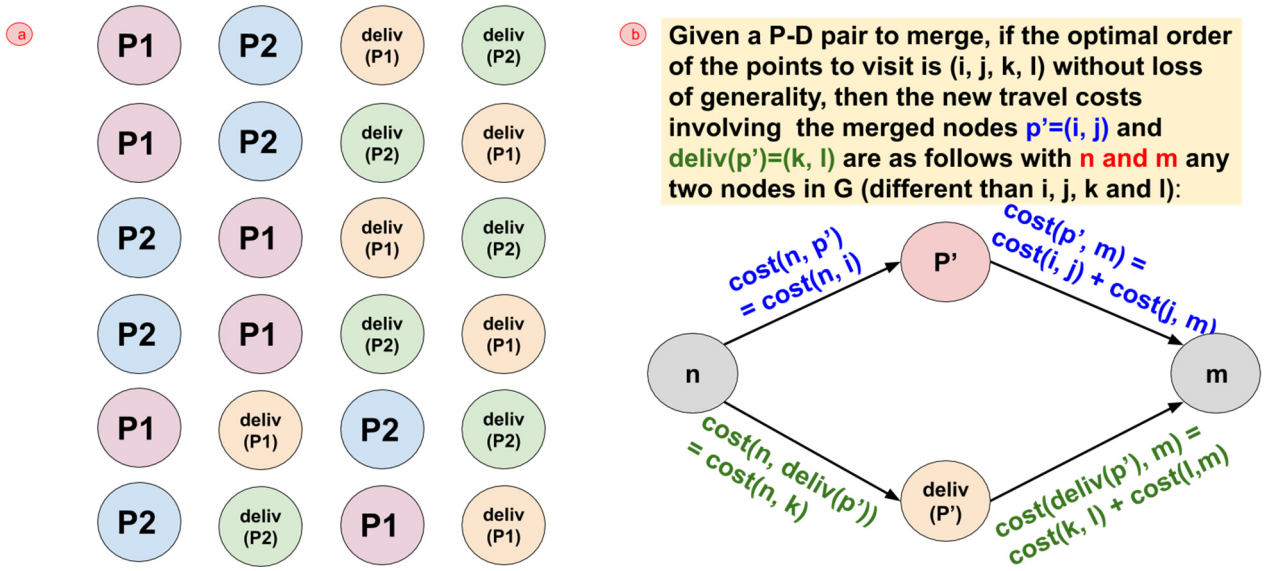

Figure 2. Possible arrangements for a pair of P-D couples (**a**), and the updated travel costs involving merged P-D pairs (**b**).

4. Case Study: The Industrial District of "Vallée De La Chimie", Lyon City

In this section, we describe the studied territory and the process used to generate the input demand data: the OD matrices of Figure 1. The number of vehicles is set to four, as will be seen in the next section.

4.1. Territory

Lyon is the second-largest urban area in France and is home to several industrial hubs. The largest of them is "Vallée de la Chimie" (VC) with approximately 17km^2. Situated about 10km away from the city center, VC has currently over 1000 companies and 6 research and development centers, mostly working in the chemical, energy, and environmental sectors, and it attracts over 20,000 jobs (Métropole de Lyon, 2023 [42]). To summarize, this area is the main industrial sector of Lyon city and currently is in great need of a last-mile transport solution. It provides a relevant application case to test our reduction method for SAV deployment. The second reason for choosing this area is the availability of the demand data tuned for VC according to OPTIREL software (see next section).

In terms of mobility infrastructure, a rapid transit system (RTS) links VC to the city center on a regular schedule (each 30 min in rush hours, otherwise once per hour). There are quite a few buses connecting VC to Lyon city (in total, three, and two are drawn in the right map of Figure 3 in light red lines). The highway A7 also crosses the territory. This artery is essential for logistic transportation and becomes severely congested in the morning rush hours, as A7 is also the south entry point to the city. Commuting to work accounts for a significant part of mobility flows from/to VC. By considering the low PT supply and the highway's high saturation, a typical last-mile transit issue occurs. A number of experimentations to overcome this issue are currently being conducted in this territory: Personal Rapid Transit (PRT) from the Esprit project (Esprit, 2019 [43]) and a demand-

responsive transit (DRT) service from the local PT service (TCL, 2020 [44]). VC constitutes a good study example for SAV deployment.

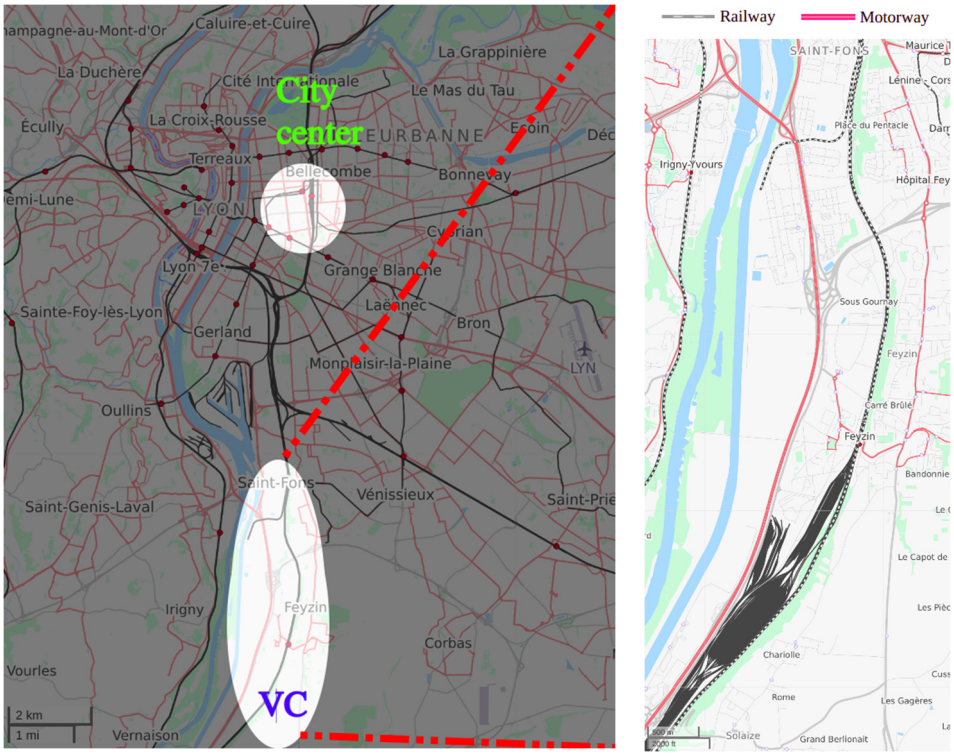

Figure 3. The placement of VC within the Lyon urban area in the left map, and the transport network connecting VC in the right map. Source: www.openstreetmap.org/ (accessed on 15 January 2021).

Finally, this test case is classic and generalizable to several other mobility situations all over the world. Indeed, it corresponds to the framework of a commute problem with two main alternatives: PT and a private vehicle, which experience bottlenecks.

4.2. Demand Generation

To generate the traffic demand, we apply a Land Use and Transport Interaction (LUTI) model. LUTI models are spatial interaction models that integrate the socio-demographic and economic features of the population into the transport infrastructure and mobility features to answer questions such as what is the transport modal share of a middle-sized income household living in the suburb to commute to work in the city center. With LUTI models, the cause/effect relationships between land use and transport become well understood, and the mobility patterns are well identified. It is noteworthy that LUTI models are extensively used for urban and transport planning (see Acheampong et al., 2015 [45] for a review of the topic).

We use a LUTI model called OPTIREL, which has the advantage of accounting for intermodality. It is a proprietary software, which relies on French governmental data and surveys for the socio-demographic, economic, and mobility features of the population. OPTIREL proceeds by decomposing the area's space according to the transport mode used from/to the target territory. For instance, the first zone of the studied territory is solely accessible by walking, the second zone is larger and is accessible by bus or RTS system (from/to the territory), and the third zone is even larger. It delimits areas accessible by private vehicles combined with relay parking. In this manner, it is possible to define mobility solutions combining several modes in the step of the modal decomposition. The classical four-step travel models define zones according to the administrative or population census-based spatial decomposition, which turns out to be a rigid assumption for combining

several modes in the third step of modal share. The traffic demand considered in our application corresponds to the PT demand arriving in the territory by train during the morning peak (7 h–10 h). This interval will set our time horizon. Ten stations are defined for the planned SAV service, which corresponds to stations of the Esprit project (Esprit, 2019 [43]) and covers the northern part of VC. They are shown in Figure 4. The main transit station is "Gare Feyzin", which will coincide with the depots of all vehicles. We can notice from our input OD demand matrix, generated by OPTIREL, and shown in Figure 4, that traffic coming from "Gare Feyzin" is the highest. Interestingly, the traffic is also high from the stations "Hector Berlioz" and "Thomas", which are the closest stations to the transit station. Finally, duration and distances between stations are generated using web mapping services. OPTIREL provides OD matrices of distances and travel times as well. The OD distances are calculated according to the shortest paths between the points, and the OD travel times are based on average values for a weekday.

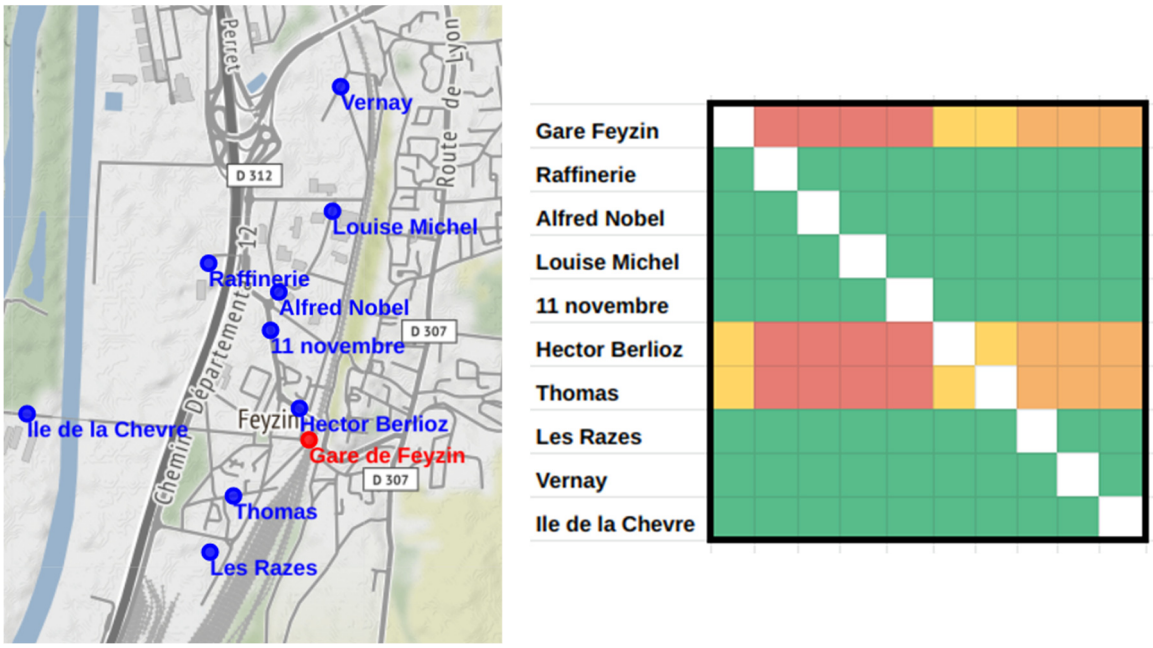

Figure 4. The locations of the ten stations considered in the study (**left map**), and the input OD demand matrix (**right plot**), which is colored according to the demand intensity: green = low, yellow and orange = middle, and red = high.

5. Results

In this section, we test the proposed method on the study case and show its usefulness.

5.1. Parameters Setting

For solving the DARP, the ILOG CPLEX 12.10 solver was used, with the termination criteria set to a relative gap of 1%. Although the routing graph was reduced, we noticed that problem (1)–(15) takes an extended time to find a first solution, which is not the case for problem (1)–(15), excluding the precedence constraint (7). Thus, we solved the latter problem in the first step, rearranged the points to visit for each vehicle by solving the single-vehicle DARP problem (1)–(15), and then considered this solution as a starting point for our main optimization. The effect of adding an initial good quality starting point was beneficial in our CPLEX solving process. In the absence of precise data about the customers' time windows, we set the constraints (8) to two time windows, each with a one-hour duration, and uniformly at random partition P and D points on the two time windows, with the condition that each drop-off point has the same time window of its pickup point. We did not consider customers' waiting times w_i with this configuration of time windows. A fixed service time $ser_i = 1$ min was associated with each point to visit $i \in P \cup D$. In the baseline

scenario, we set the capacity of vehicles to $Q = 25 > 20 = max_{i,j \in V^2, i \neq j} demand(i,j)$, the number of vehicles to $vehicles(m = 3) = 4$, the maximum riding time to $L = 15$ min, and the weights of the objective function to $(\omega_1, \omega_2, \omega_3) = (1, 1, 0.1)$. The GHG emission rate was chosen to be equal to $E_k = 0.171 KgCO_2 - eq/km$, which corresponds to the short-range electric autonomous taxi scenarios in Gawron et al. (2019) [39]. For the input parameters of *node2vec*, the dimension size was set to 30, while the walk length was equal to 6 for 1000 random walks launched from each node. Since *node2vec* is a stochastic method, we launched this method multiple times, exactly 50 times for each configuration of Q and L, and then chose the instance with the minimal inside travel time. This latter value was given by the summation of travel times inside the reduced nodes. For instance, the inside travel time of the graph of Figure 2b was equal to $c_{ij} + c_{kl}$. The calculation was performed for each configuration of Q and L as the possible arrangements of P-D couples could be less than six (Figure 2a). All experiments were performed on an Intel(R) Core(TM) i7-8750H CPU @ 2.20 GHz machine with 32 GB RAM memory.

5.2. Constrained K-Means

The constrained K-means clustering proposed by Bradley et al. (2000) [37] extends the traditional algorithm ensuring that each cluster contains at least a minimum number of elements. The problem of minimum cost flow optimization is used to solve the new formulation. Similarly, it is possible to add a maximum size constraint to clusters. In order to have a fair comparison with *node2vec*, we associate with each pickup node a value similar to the average of the expression (17):

$$\begin{aligned} cost_{avg}(p) = \sum \Big(& c_{pq} + c_{deliv(p)deliv(q)} + c_{pdeliv(p)} + c_{qdeliv(q)} + c_{pdeliv(q)} + c_{qdeliv(p)} \Big) / 12 \\ + \sum \Big(& c_{qp} + c_{deliv(q)deliv(p)} + c_{deliv(p)p} + c_{deliv(q)q} + c_{deliv(q)p} + c_{deliv(p)q} \Big) / 12. \end{aligned}$$

This expression has twelve terms for each $q \in P$ to account for the asymmetry of the travel times c_{ij}. Clusters are generated with a maximum size of two and a minimum size of one for the clustering with the best inside travel time among 50 runs.

5.3. Indicators

Two main performance measures are used: the vehicle kilometers traveled (VKT), and the vehicle occupancy rate (VOR). Two additional measures are extracted from them. VKT indicates the total distance traveled by all vehicles of K to satisfy the demand and is equal to

$$VKT(x) = \sum_{k \in K} \sum_{i,j \in V^2, i \neq j} d_{ij} x^k_{i,j}.$$

We also use a correlated measure to VKT, which is the total GHG emissions generated for all trips to provide an order of magnitude of SAV emissions: $GHG(x) = E_k VKT(x)$. To obtain a glimpse of the GHG environmental gain of using SAV, which is previously established in the literature, we show the gap:

$$GHG_{gain}(x) = \frac{GHG_{pv}(demand) - GHG(x)}{GHG_{pv}(demand)},$$

With $GHG_{pv}(demand)$ represents GHG emissions estimated for the input demand and the personal vehicle mode. By drawing on the estimates of the vehicle occupancy rate and the GHG emissions of private vehicles computed for the Lyon urban area (François et al., 2017 [46]), we obtain the input demands:

$$GHG_{pv}(demand) = \sum_{i,j \in V^2, i \neq j} demand(i,j) \times d_{ij} \frac{175 gCO_2 - eq/Km}{1.33 person/car} = 43.8 KgCO_2 - eq.$$

The third measure VOR is an average of the load rate of all vehicles at each step of the routing and is equal to

$$VOR(x,q) = \frac{1}{|K|}\frac{1}{|P| \cup |D|}\sum_{k\in K}\sum_{i,j\in P\cup D, i\neq j} x_{ji}^{k}\frac{q_i^k}{Q}.$$

VOR can also be seen as an index of the comfort of the mobility service. The last measure, the zero occupancy rate (ZOR) is equal to the percentage of trips between any two stations that have an occupancy rate equal to zero and is given by

$$ZOR(x,q) = \frac{1}{|K|}\sum_{k\in K}\sum_{i,j\in P\cup D, i\neq j, q_i^k=0}\frac{x_{ji}^{k}}{(|P| \cup |D|)-1}.$$

All measures are computed on the original non-reduced problem.

5.4. Baseline Scenario

Accounting for the total demand of VC leads to solving a problem of 54 points to visit (The number of points to visit for the full OD demand matrix of Figure 4 is 180 points. We have only 54 points in our input data since some OD couples have a null demand). Plus, the dummy start and end vehicle depots, which is a computationally hard problem for $k > 2$ vehicles. The graph reduction mechanism makes this optimization more manageable. CPLEX solver could attain a solution for the baseline scenario within a relative MIP gap of 23% in 0.28, 277.17, and 3609.62 s when the graph contraction rate is, respectively, 50%, 45%, and 40%. In the rest of the study, we set the contraction rate of the routing graph at $r = 50\%$. Figure 5 illustrates the SAV routes obtained for the baseline scenario. This scenario has a total traveled distance of $VKT = 65.46$ (respectively, 74.44 km) for an average occupation rate of $VOR = 42.1\%$ (resp., 31.6%), and a total emissions of $GHG = 11.12$ (resp., $12.73 KgCO_2 - eq$) for *node2vec* (resp., constrained K-means)-based node reduction.

Compared to the private vehicle mode ($43.8 KgKgCO_2 - eq$), the SAV service leads to a reduction of 74.61% of GHG emissions for the territory of VC when relying on the *node2vec* reduction.

Figure 5. *Cont.*

K-means

Figure 5. SAV routes for the baseline scenario with the graph embedding reduction (**first row**) and the K-means-based reduction (**second row**).

5.5. Sensitivity Analysis

Figure 6 and Table 1 show the sensitivity analysis performed on the baseline scenario by varying the fleet size $|K|$, the maximum riding time L, and the vehicle's capacity Q. For a number of vehicles equal to 1, 5, and 10, the total traveled distance VKT slowly decreases to be, resp., 67.70, 64.53, and 59.88 km for the *node2vec* reduction. K-means clustering generates tours in the same decreasing trend, but with higher VKT, which is an indication of a lower performance to find the best pairs of P-D couples to merge. While $|K|$ is multiplied by 10, VKT of *node2vec* is divided by 1.13. Having more vehicles actually allows more flexibility for the routing optimization to produce efficient assignments of vehicles to the transport requests. VKT is decreasing in function of Q, which is expected since larger capacities imply smaller tours. *Node2vec* is again much more efficient than K-means reduction in this regard. VKT is overall stable under varying L. As some values of L and Q may constrain the number of possible arrangements of Figure 2a, an oscillatory effect when varying L and Q could be seen for the VKT and VOR index as well. GHG_{gain} plots on the other hand show a substantial cut in terms of GHG emissions if SAVs are deployed instead of private vehicles, especially for the *node2vec* method. The reduction corresponds to, resp., 73.57% and 76.62% of GHG emissions if the fleet size is equal to, resp., 1 and 10. Concerning the vehicle occupation rate, we notice that this rate is not affected much by the size $|K|$ and the riding time L, and oscillates around the mean value of 35%. The vehicle's capacity Q, on the other hand, has some influence on VOR. VOR decreases from 38.5 (resp., 43.3%) to 24.9 (resp., 22.7%) when the capacity increases from 20 to 50 for *node2vec* (resp., K-means) reduction. This particular point should be carefully considered by the designers of SAV services to achieve a fair tradeoff between users' comfort and the costs for an additional capacity of automated vehicles and buses. As for empty trips, ZOR is solely impacted by the size $|K|$. This index decreases for an increase in $|K|$; otherwise, it has a constant value of around 10%.

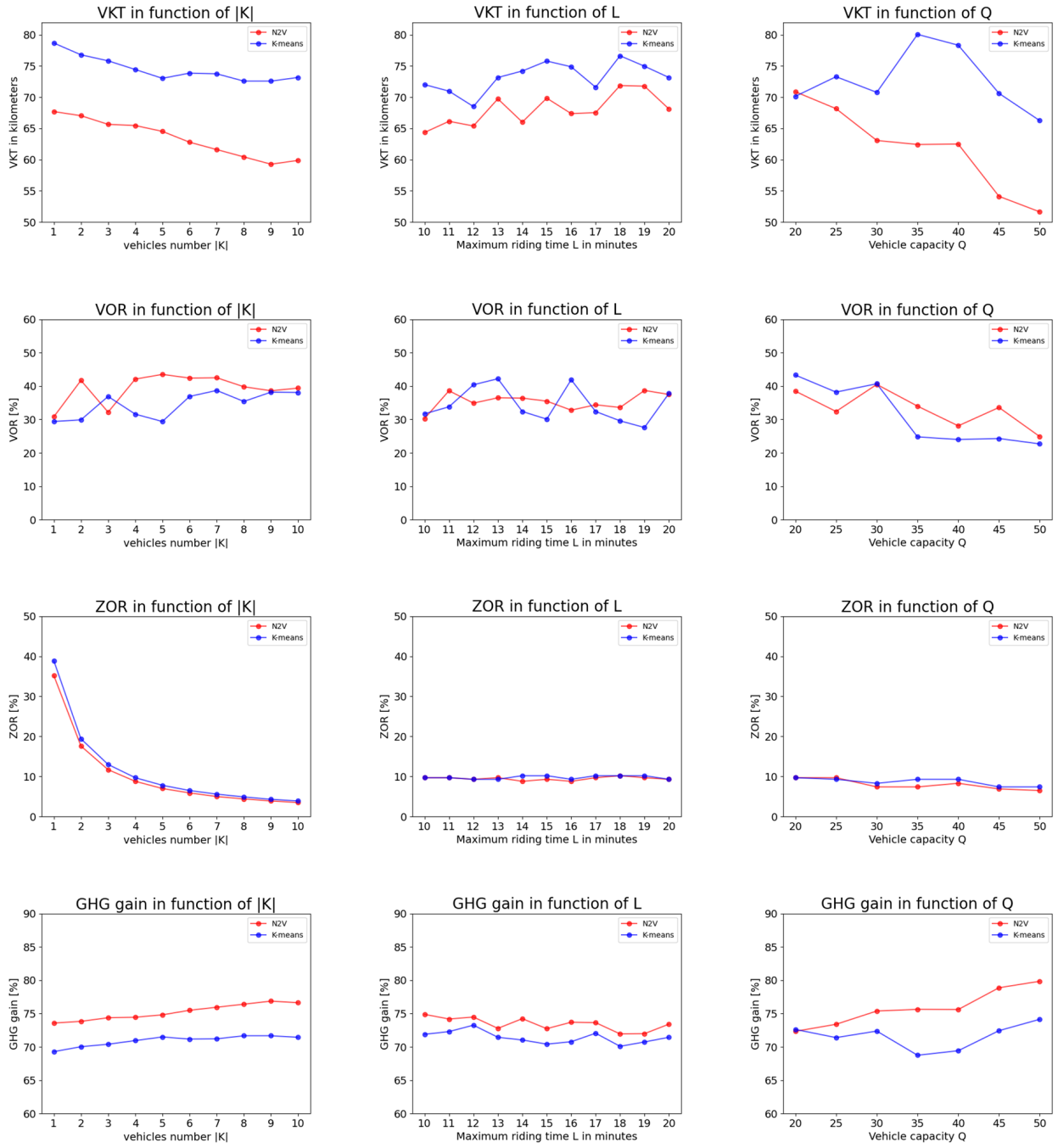

Figure 6. Sensitivity analysis on the baseline scenario of the fleet size (first column), the clients' maximum riding time (second column), and the vehicles' capacity (third column) for the indicators VKT (first row), VOR (second row), ZOR (third row), and GHG gain (fourth row).

Table 1. Numerical values of the sensitivity analysis (plotted in Figure 6).

Fixed Parameters	Variable Parameter	VKT [Km]		VOR [%]		ZOR [%]		GHG [Kg CO_2-eq]	
		N2V	K-Means	N2V	K-Means	N2V	K-Means	N2V	K-Means
Q = 25 L = 15 min	|K| = 1	67.7	78.7	30.8	29.4	35.2	38.9	11.6	13.45
	|K| = 2	67.0	76.8	41.7	29.9	17.6	19.4	11.5	13.13
	|K| = 3	65.6	75.8	32.2	36.9	11.7	13.0	11.2	12.97
	|K| = 4	65.5	74.4	42.1	31.6	8.8	9.7	11.2	12.73
	|K| = 5	64.5	73.0	43.5	29.4	7.0	7.8	11.0	12.49
	|K| = 6	62.8	73.9	42.4	36.9	5.9	6.5	10.7	12.63
	|K| = 7	61.6	73.7	42.5	38.7	5.0	5.6	10.5	12.61
	|K| = 8	60.4	72.6	39.8	35.4	4.4	4.9	10.3	12.41
	|K| = 9	59.3	72.6	38.6	38.2	3.9	4.3	10.1	12.41
	|K| = 10	59.9	73.2	39.4	38.1	3.5	3.9	10.2	12.51
|K| = 4 Q = 25	L = 10 min	64.3	72.0	30.3	31.7	9.7	9.7	11.0	12.31
	L = 11 min	66.1	70.9	38.6	33.8	9.7	9.7	11.3	12.1
	L = 12 min	65.4	68.5	34.9	40.4	9.3	9.3	11.2	11.7
	L = 13 min	69.7	73.2	36.5	42.2	9.7	9.3	11.9	12.5
	L = 14 min	66.0	74.2	36.4	32.4	8.8	10.2	11.3	12.7
	L = 15 min	69.8	75.8	35.5	30.1	9.3	10.2	11.9	13.0
	L = 16 min	67.4	74.9	32.8	41.9	8.8	9.3	11.5	12.8
	L = 17 min	67.5	71.6	34.4	32.4	9.7	10.2	11.5	12.2
	L = 18 min	71.9	76.6	33.6	29.6	10.2	10.2	12.3	13.1
	L = 19 min	71.8	74.9	38.7	27.6	9.7	10.2	12.3	12.8
	L = 20 min	68.1	73.2	37.5	37.8	9.3	9.3	11.6	12.5
|K| = 4 L = 15 min	Q = 20	70.9	70.1	38.5	43.3	9.7	9.7	12.1	11.0
	Q = 25	68.2	73.3	32.4	38.2	9.7	9.3	11.7	12.5
	Q = 30	63.1	70.8	40.5	40.7	7.4	8.3	10.8	12.1
	Q = 35	62.4	80.1	34.0	24.8	7.4	9.3	10.7	13.7
	Q = 40	62.5	78.3	28.1	24.0	8.3	9.3	10.7	13.4
	Q = 45	54.1	70.6	33.6	24.3	6.9	7.4	9.3	12.1
	Q = 50	51.6	66.2	24.9	22.7	6.5	7.4	8.8	11.3

We must also mention that the results here are obtained according to data constructed according to the LUTI model. A validation in a real-life context is still needed to confirm these positive results.

6. Conclusions

In this study, we propose a model-based heuristic design for a shared autonomous vehicle (SAV) service by combining the formulation of the Dial-a-Ride Problem (DARP) and a graph reduction mechanism. This latter procedure is based on the graph embedding framework *node2vec* and allows us to merge similar couples of pickup and drop-off points, hence solving instances with a large number of O-D requests and fleet size. A study case analysis is provided for an industrial district of the Lyon urban area (France), wherein the traffic demand is generated by a LUTI model. In this paper, we address a classical commute problem between the city center and the suburbs where two transportation alternatives compete: highway and rapid transit system (RTS). The SAV service is deployed as a last-mile transit for the RTS. Our results suggest that *node2vec* is more efficient for node reduction than the constrained K-means (in terms of vehicle traveled distance); in addition, around a 75% reduction of GHG emissions is gained by SAV service when compared to the private vehicle choice.

The limitations of this study mainly concern the maximum reduction rate, which is currently 50%, as reduction is carried out by merging couples of pickup and drop-off pairs. One natural improvement is to enlarge the reduction mechanism to three pickup and drop-off pairs. We also think that integrating the current approach with the LUTI model would generate precise partitions of the mode share for the studied territory and population, which would in return help to fix the adequate parameters of the SAV service, i.e., load capacity, fleet size, and maximum riding time, in a cause-and-effect loop. This study can also benefit from extending the application to much larger target areas to provide more validation.

Funding: This research was funded by "Lyon Urban School" (ANR-17-CONV-0004) of University of Lyon.

Data Availability Statement: The data will be made available by the authors on request.

Acknowledgments: We thank the creator of the software OPTIREL for allowing us to use it to generate the traffic input demand data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sinha, K.C. Sustainability and urban public transportation. *J. Transp. Eng.* **2003**, *129*, 331–341. [CrossRef]
2. Shaheen, S.; Chan, N. Mobility and the sharing economy: Potential to facilitate the first-and last-mile public transit connections. *Built Environ.* **2016**, *42*, 573–588. [CrossRef]
3. Waymo. Autonomous Ridesharing Isn't Dead: How Waymo Is Adapting to the Post-COVID Era. By Ronan Glon. 3 July 2000. Available online: https://www.digitaltrends.com/cars/waymo-in-post-covid-era/ (accessed on 7 February 2024).
4. Amazon. Amazon Buys Autonomous Taxi Company Zoox. Look Out, Uber and Lyft. By Eric J. Savitz. 26 June 2020. Available online: https://www.barrons.com/articles/amazon-buys-autonomous-taxi-company-zoox-51593187053 (accessed on 7 February 2024).
5. Jaguar Land Rover. Jaguar Land Rover Reveals Project Vector Autonomous Ride-Share Vehicle. By Alistair Charlton. 19 February 2020. Available online: https://www.gearbrain.com/jaguar-land-rover-project-vector-2645188684.html (accessed on 7 February 2024).
6. Bauer, G.S.; Greenblatt, J.B.; Gerke, B.F. Cost, energy, and environmental impact of automated electric taxi fleets in Manhattan. *Environ. Sci. Technol.* **2018**, *52*, 4920–4928. [CrossRef] [PubMed]
7. Fagnant, D.J.; Kockelman, K.M. The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios. *Transp. Res. Part C Emerg. Technol.* **2014**, *40*, 1–13. [CrossRef]
8. International Transport Forum. *Organisation for Economic Co-Operation and Development. Urban Mobility System Upgrade: How Shared Self-Driving Cars Could Change City Traffic*; OECD Publishing: Paris, France, 2015.
9. Hyland, M.F.; Mahmassani, H.S. Taxonomy of shared autonomous vehicle fleet management problems to inform future transportation mobility. *Transp. Res. Rec.* **2017**, *2653*, 26–34. [CrossRef]
10. Gurumurthy, K.M.; Kockelman, K.M. Analyzing the dynamic ride-sharing potential for shared autonomous vehicle fleets using cellphone data from Orlando, Florida. *Comput. Environ. Urban Syst.* **2018**, *71*, 177–185. [CrossRef]
11. Narayanan, S.; Chaniotakis, E.; Antoniou, C. Shared autonomous vehicle services: A comprehensive review. *Transp. Res. Part C Emerg. Technol.* **2020**, *111*, 255–293. [CrossRef]
12. Levin, M.W. Congestion-aware system optimal route choice for shared autonomous vehicles. *Transp. Res. Part C Emerg. Technol.* **2017**, *82*, 229–247. [CrossRef]
13. Ma, J.; Li, X.; Zhou, F.; Hao, W. Designing optimal autonomous vehicle sharing and reservation systems: A linear programming approach. *Transp. Res. Part C Emerg. Technol.* **2017**, *84*, 124–141. [CrossRef]
14. Ho, S.C.; Szeto, W.Y.; Kuo, Y.H.; Leung, J.M.; Petering, M.; Tou, T.W. A survey of dial-a-ride problems: Literature review and recent developments. *Transp. Res. Part B Methodol.* **2018**, *111*, 395–421. [CrossRef]
15. Gschwind, T.; Irnich, S. Effective handling of dynamic time windows and its application to solving the dial-a-ride problem. *Transp. Sci.* **2015**, *49*, 335–354. [CrossRef]
16. Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
17. McNally, M.G. *The Four-Step Model*; Emerald Group Publishing Limited: Leeds, UK, 2007.
18. Shen, Y.; Zhang, H.; Zhao, J. 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.* **2018**, *113*, 125–136. [CrossRef]
19. Pinto, H.K.; Hyland, M.F.; Mahmassani, H.S.; Verbas, I.Ö. Joint design of multimodal transit networks and shared autonomous mobility fleets. *Transp. Res. Part C Emerg. Technol.* **2019**, *113*, 2–20. [CrossRef]

20. Shan, A.; Hoang, N.H.; An, K.; Vu, H.L. A framework for railway transit network design with first-mile shared autonomous vehicles. *Transp. Res. Part C Emerg. Technol.* **2021**, *130*, 103223. [CrossRef]
21. Yantao, H.; Kockelman, K.M.; Truong, L.T. SAV Operations on a Bus Line Corridor: Travel Demand, Service Frequency, and Vehicle Size. *J. Adv. Transp.* **2021**, *2021*, 5577500. [CrossRef]
22. Liang, X.; de Almeida Correia, G.H.; Van Arem, B. Optimizing the service area and trip selection of an electric automated taxi system used for the last mile of train trips. *Transp. Res. Part E Logist. Transp. Rev.* **2016**, *93*, 115–129. [CrossRef]
23. Al Maghraoui, O.; Vosooghi, R.; Mourad, A.; Kamel, J.; Puchinger, J.; Vallet, F.; Yannou, B. Shared autonomous vehicle services and user taste variations: Survey and model applications. *Transp. Res. Procedia* **2020**, *47*, 3–10. [CrossRef]
24. Gurumurthy, K.M.; Kockelman, K.M. Dynamic ride-sharing impacts of greater trip demand and aggregation at stops in shared autonomous vehicle systems. *Transp. Res. Part A Policy Pract.* **2022**, *160*, 114–125. [CrossRef]
25. Golbabaei, F.; Yigitcanlar, T.; Bunker, J. The role of shared autonomous vehicle systems in delivering smart urban mobility: A systematic review of the literature. *Int. J. Sustain. Transp.* **2021**, *15*, 731–748. [CrossRef]
26. Psaraftis, H.N. A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transp. Sci.* **1980**, *14*, 130–154. [CrossRef]
27. Jaw, J.J.; Odoni, A.R.; Psaraftis, H.N.; Wilson, N.H. A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transp. Res. Part B Methodol.* **1986**, *20*, 243–257. [CrossRef]
28. Ropke, S.; Cordeau, J.F.; Laporte, G. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Netw. Int. J.* **2007**, *49*, 258–272. [CrossRef]
29. Masmoudi, M.A.; Hosny, M.; Braekers, K.; Dammak, A. Three effective metaheuristics to solve the multi-depot multi-trip heterogeneous dial-a-ride problem. *Transp. Res. Part E Logist. Transp. Rev.* **2016**, *96*, 60–80. [CrossRef]
30. Mourad, A.; Puchinger, J.; Chu, C. A survey of models and algorithms for optimizing shared mobility. *Transp. Res. Part B Methodol.* **2019**, *123*, 323–346. [CrossRef]
31. Cordeau, J.F.; Laporte, G. The dial-a-ride problem: Models and algorithms. *Ann. Oper. Res.* **2007**, *153*, 29–46. [CrossRef]
32. Molenbruch, Y.; Braekers, K.; Caris, A. Typology and literature review for dial-a-ride problems. *Ann. Oper. Res.* **2017**, *259*, 295–325. [CrossRef]
33. Goyal, P.; Ferrara, E. Graph embedding techniques, applications, and performance: A survey. *Knowl. -Based Syst.* **2018**, *151*, 78–94. [CrossRef]
34. Perozzi, B.; Al-Rfou, R.; Skiena, S. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710.
35. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1067–1077.
36. Chen, S.; Wang, H.; Meng, Q. Solving the first-mile ridesharing problem using autonomous vehicles. *Comput. -Aided Civ. Infrastruct. Eng.* **2020**, *35*, 45–60. [CrossRef]
37. Bradley, P.S.; Bennett, K.P.; Demiriz, A. Constrained k-means clustering. *Microsoft Res. Redmond* **2000**, *20*, 1–8.
38. Cordeau, J.F. A branch-and-cut algorithm for the dial-a-ride problem. *Oper. Res.* **2006**, *54*, 573–586. [CrossRef]
39. Gawron, J.H.; Keoleian, G.A.; De Kleine, R.D.; Wallington, T.J.; Kim, H.C. Deep decarbonization from electrified autonomous taxi fleets: Life cycle assessment and case study in Austin, TX. *Transp. Res. Part D Transp. Environ.* **2019**, *73*, 130–141. [CrossRef]
40. Bongiovanni, C.; Kaspi, M.; Geroliminis, N. The electric autonomous dial-a-ride problem. *Transp. Res. Part B Methodol.* **2019**, *122*, 436–456. [CrossRef]
41. Vosooghi, R.; Puchinger, J.; Bischoff, J.; Jankovic, M.; Vouillon, A. Shared autonomous electric vehicle service performance: Assessing the impact of charging infrastructure. *Transp. Res. Part D Transp. Environ.* **2020**, *81*, 102283. [CrossRef]
42. Métropole de Lyon. Vallée de la Chimie Presentation. 2021. Available online: https://lyonvalleedelachimie.fr/en/home/ (accessed on 15 January 2021).
43. ESPRIT Project. 2019. Available online: http://www.esprit-transport-system.eu/ (accessed on 23 November 2020).
44. TCL. Transport à la Demande. 2020. Available online: https://www.tcl.fr/transport-a-la-demande-vallee-de-la-chimie (accessed on 27 July 2020). (In French).
45. Acheampong, R.A.; Silva, E.A. Land use–transport interaction modeling: A review of the literature and future research directions. *J. Transp. Land Use* **2015**, *8*, 11–38. [CrossRef]
46. François, C.; Gondran, N.; Nicolas, J.P.; Parsons, D. Environmental assessment of urban mobility: Combining life cycle assessment with land-use and transport interaction modelling—Application to Lyon (France). *Ecol. Indic.* **2017**, *72*, 597–604. [CrossRef]