Benefits of horizontal cooperation in dial-a-ride services

Peer-reviewed author version

Benefits of horizontal cooperation in dial-a-ride services

Yves Molenbruch[*1,2], Kris Braekers[1,2], and An Caris[1]

[1]Hasselt University, Faculty of Business Economics, Research Group Logistics,
Agoralaan Gebouw D, BE-3590 Diepenbeek
[2]Research Foundation Flanders (FWO), Egmontstraat 5, BE-1000 Brussels

August 31, 2017

Abstract

Dial-a-ride services provide collective on-demand transportation, usually tailored to the needs of people with reduced mobility. This paper investigates the operational effects of horizontal cooperation among dial-a-ride providers. The current practice is that users choose a particular provider to submit their requests. Providers operating in the same area create their routing solutions independently of each other, given their own set of customers. In contrast, horizontal cooperation through joint route planning implies that customer requests can be exchanged among providers in order to minimize the overall routing cost. In addition to quantifying the operational benefits generated by such a horizontal cooperation, this paper identifies operational characteristics that influence the magnitude of the savings. A real-life case study reveals the reasons why providers benefit from certain request exchanges, as well as the extent to which these exchanges are predictable in advance. The solutions are obtained using a large neighborhood search algorithm that performs well on benchmark data.

Keywords: dial-a-ride services, demand-responsive transportation, horizontal cooperation, joint route planning

1 Introduction

A dial-a-ride system is an application of demand-dependent, collective passenger transportation (Cordeau and Laporte, 2003). Each customer requests a trip between an origin and a destination of choice, to which a number of service level requirements are linked. The service provider attempts to develop efficient vehicle routes and time schedules, respecting these requirements and the technical constraints of a pickup and delivery problem (Parragh et al., 2008). A frequent objective is to minimize operational costs subject to full demand satisfaction and side constraints, but service level criteria may be optimized as well. Balancing the human

---

[*]corresponding author: yves.molenbruch@uhasselt.be

and economic perspectives involved in solving such a dial-a-ride problem (DARP) is essential for organizing quality-oriented, yet efficient transportation of customers with special needs, such as elderly or disabled. In light of the ageing population, dial-a-ride systems are gaining importance to complement regular transportation modes. They fulfill a social role as well, preventing isolation of vulnerable groups in society.

When a certain area is served by multiple dial-a-ride providers with a comparable quality policy, most users living in that area will intuitively submit their requests to the provider located closest to their homes. The different providers create their vehicle routes and schedules independently of each other, based on the requests they receive. Whereas strategies to share information or resources have been cost-effective practices among logistic service providers for many years (Verdonck et al., 2013), they are completely unexplored in the domain of dial-a-ride services. Also the academic literature lacks insights into the effects of cooperation in the specific context of demand-responsive passenger transportation, e.g. due to tighter quality requirements. The present paper performs an analysis on *joint route planning* in dial-a-ride services. This particular form of horizontal cooperation assumes a centralized decision making, enabling service providers to exchange customer requests such that overall routing costs are minimized. The problem is solved using a large neighborhood search (LNS) metaheuristic whose performance is demonstrated using common benchmark data from the literature.

The contribution of this paper is threefold. First, the potential of horizontal cooperation through joint route planning among dial-a-ride providers is analyzed. Second, the operational characteristics that influence these benefits are identified using artificially constructed data in which the operational setting is varied. This analysis allows to draw general conclusions on a more strategic/tactical level regarding the benefits of cooperation. Third, a real-life case study is performed to discover the reasons why service providers benefit from exchanging certain requests. The predictability of these exchanges influences the extent to which information must be disclosed to enable a successful cooperation. Note that all analyses focus on the joint operational benefits incurred by the overall cooperation. To allocate these benefits among the providers, which falls outside the scope of this paper, gain sharing techniques from existing literature on horizontal cooperation (e.g. Shapley value, alternative cost avoided method, equal profit method) may be used (Verdonck et al., 2016).

The remainder of this paper is structured as follows. Section 2 introduces the standard DARP and provides a mathematical formulation of the variant used in this work. Section 3 summarizes related literature on horizontal cooperation in logistics and translates these principles to the domain of demand-responsive people transportation. Section 4 discusses the structure of the LNS metaheuristic. Section 5 presents the artificial and real-life data sets used for the computational tests and analyzes the results. Section 6 draws conclusions in view of a practical implementation of horizontal cooperation and suggests ideas for future research.

2   Problem statement

Cordeau and Laporte (2003) introduced a standard definition of the DARP. It consists of designing several minimum-cost vehicle routes in a complete graph of nodes and arcs. The nodes represent pickup and delivery locations of customers, as well as a single vehicle depot. The arcs between these nodes have an associated travel time and cost, which is incurred if the arc is traversed by a vehicle. Routes start and end at the depot within fixed time intervals and respect a maximum route duration. The service at each customer location starts within a hard time window. A maximum user ride time cannot be exceeded and a vehicle's load should respect the maximum capacity. To ensure a correct physical route construction, precedence and pairing of a customer's origin and destination must be ensured by visiting them in the right order, using the same vehicle. A service duration indicates the time that may be needed for loading and unloading customers.

The problem studied in this paper involves multiple service providers that may or may not cooperate. Since each participating provider disposes of at least one depot, a multi-depot variant of the DARP (Braekers et al., 2014) needs to be solved. In scenarios with horizontal cooperation among the providers, the principle of joint route planning implies that requests may be served by a vehicle originating from any given depot. In a scenario without cooperation, requests are preassigned to one specific provider, usually based on the customer's geographical situation. In addition, the real-life case study concluding Section 5 involves multiple customer types (Parragh, 2011) and configurable vehicle capacity (Qu and Bard, 2013). The former means that customers are categorized based on their mobility, i.e. whether they use a wheelchair or not. This heterogeneity is reflected in the fact that the vehicles offer a number of standard seats and wheelchair spaces. Their design is instantaneously configurable, meaning that the ratio between both resource types can be modified at any stop in the route according to a limited set of configurations.

A mathematical formulation for this rich problem variant can be composed by building upon the arc-based mixed-binary linear programs of Cordeau (2006), Røpke et al. (2007), Parragh (2011), Qu and Bard (2013) and Braekers et al. (2014). The resulting formulation is shown by equations 1-15. Given an input of n user requests, each node i in the range $1, ..., n$ represents the pickup location of a specific user i, whereas node $n + i$ represents the corresponding delivery location. The four-index binary decision variable x_{ij}^{kf} indicates whether vehicle k traverses the arc between nodes i and j in configuration f, where the set of possible configurations F_k depends on the vehicle k. Each vehicle route is assumed to start at an origin depot and end at a destination depot (eq. 4, 6). One and the same vehicle should reach and leave corresponding pickup and delivery locations i and $n + i$ (eq. 2, 3, 5), which ensures flow conservation and pairing. However, without horizontal cooperation, only a limited set of vehicles (i.e. the vehicles from one specific provider) is eligible to serve a particular user (eq. 7). The use of an index k for parameters related to travel costs/times allows to account for vehicles originating from multiple depots at different physical locations, even though a single node is used to represent the start depot 0 and end depot $2n + 1$. Decision variable L_i^k computes the ride time of user i (eq. 10) and cannot exceed

3

the maximum user ride time L (eq. 13). Explicit precedence constraints become redundant if L_i^k is set at least equal to the associated direct ride time. Decision variable B_i^k tracks the service start in node i (eq. 8) and should respect the time window of this node (eq. 12). A vehicle may idle at any time and place, e.g. when it reaches a node before the start of the time window. Decision variable Q_i^{kr} registers the load of resource r upon leaving node i (eq. 9) and cannot exceed the capacity C_{kf}^r of vehicle k in configuration f for resource r (eq. 14), respectively. The vehicle's configuration can be changed at any node. The time span between the moment a vehicle leaves the origin depot and the moment it returns to the destination depot cannot exceed the maximum route duration T_k (eq. 11). A minimum-cost selection of arcs is made (eq. 1), subject to all constraints and full demand satisfaction. Finally, note that equations 8 and 9 can be linearized as proposed by Cordeau (2006).

$$\text{Minimize} \quad \sum_{k \in K} \sum_{f \in F_k} \sum_{i \in N} \sum_{j \in N} c_{ij}^k x_{ij}^{kf} \tag{1}$$

Subject to

$$\sum_{k \in K} \sum_{f \in F_k} \sum_{j \in N} x_{ij}^{kf} = 1 \qquad \forall i \in P \tag{2}$$

$$\sum_{f \in F_k} \sum_{j \in N} x_{ij}^{kf} - \sum_{f \in F_k} \sum_{j \in N} x_{n+i,j}^{kf} = 0 \qquad \forall i \in P, \, \forall k \in K \tag{3}$$

$$\sum_{f \in F_k} \sum_{j \in N} x_{0j}^{kf} = 1 \qquad \forall k \in K \tag{4}$$

$$\sum_{f \in F_k} \sum_{j \in N} x_{ji}^{kf} - \sum_{f \in F_k} \sum_{j \in N} x_{ij}^{kf} = 0 \qquad \forall i \in P \cup D, \, \forall k \in K \tag{5}$$

$$\sum_{f \in F_k} \sum_{i \in N} x_{i,2n+1}^{kf} = 1 \qquad \forall k \in K \tag{6}$$

$$\sum_{f \in F_k} \sum_{j \in N} x_{ij}^{kf} \leq a_i^k \qquad \forall i \in P, \, \forall k \in K \tag{7}$$

$$B_j^k \geq (B_i^k + d_i + t_{ij}^k) \cdot \sum_{f \in F_k} x_{ij}^{kf} \qquad \forall i \in N, \, \forall j \in N, \, \forall k \in K \tag{8}$$

$$Q_j^{kr} \geq (Q_i^{kr} + q_j^r) \cdot \sum_{f \in F_k} x_{ij}^{kf} \qquad \forall i \in N, \, \forall j \in N, \, \forall k \in K, \, \forall r \in R \tag{9}$$

$$L_i^k = B_{i+n}^k - (B_i^k - d_i) \qquad \forall i \in P, \, \forall k \in K \tag{10}$$

$$B_{2n+1}^k - B_0^k \leq T_k \qquad \forall k \in K \tag{11}$$

$$e_i \leq B_i^k \leq l_i \qquad \forall i \in N, \, \forall k \in K \tag{12}$$

$$t_{i,i+n}^k \leq L_i^k \leq L \qquad \forall i \in P, \, \forall k \in K \tag{13}$$

$$\sum_{f \in F_k} \sum_{j \in N} (x_{ij}^{kf} \cdot max\{0, q_i^r\}) \leq Q_i^{kr} \leq \sum_{f \in F_k} \sum_{j \in N} (x_{ij}^{kf} \cdot min\{m_{kf}^r, m_{kf}^r + q_i^r\}) \qquad \forall i \in N, \, \forall k \in K, \, \forall r \in R \tag{14}$$

$$x_{ij}^{kf} \in \{0,1\} \qquad \forall i \in N, \, \forall j \in N, \, \forall k \in K, \, \forall f \in F_k \tag{15}$$

P = set of pickup nodes $1, ..., n$
D = set of delivery nodes $n+1, ..., 2n$
N = set of all nodes (including the start depot 0 and end depot $2n+1$)
K = set of vehicles

F_k = set of configurations for vehicle k,
R = set of all resource types
c_{ij}^k = cost associated with arc (i,j) for vehicle k
t_{ij}^k = direct travel time associated with arc (i,j) for vehicle k
a_i^k = binary parameter indicating if user i may be served by vehicle k
d_i = service duration in node i
q_i^r = net number of users requiring resource r and boarding in node i
m_{kf}^r = capacity of vehicle k in configuration f for resource r

3 Literature overview

Apart from the problem variants covered in this work, several other models involving additional real-life characteristics have been proposed to enhance the practical applicability of solution methods. A comprehensive typology and literature review on DARPs, based on contributions from the past four decades, is presented by Molenbruch et al. (2017b). Specific reviews on dynamic problem variants and the definition of service quality can be found in Berbeglia et al. (2010) and Paquette et al. (2009), respectively. A structural description of a fully automated dial-a-ride system is given in Dial (2002).

Common extensions to the standard problem include advanced service designs (e.g. transfers in Masson et al. (2014)), alternative and/or multiple objectives (e.g. Lehuédé et al. (2013)) and dynamic or stochastic information (e.g. Xiang et al. (2008)). Metaheuristics based on local search, such as variable neighborhood search (Parragh et al., 2010) and deterministic annealing (Braekers et al., 2014), are very suitable for solving large-scale problem instances within realistic computation times. They typically use combinations of classical neighborhood types, such as relocate and exchange operations. Solution methods for rich problem variants benefit from additional operators taking advantage of the specific structure. A recent shift of attention is the integration of multiple types of solution approaches. For example, several state-of-the-art metaheuristics benefit from integrating a population concept into local search approaches (Chassaing et al., 2016; Masmoudi et al., 2016) or vice versa (Masmoudi et al., 2017). Besides, hybridizations of exact and approximate solution methods have been proposed in the literature (Parragh and Schmid, 2013).

The literature lacks insights into the effects of a horizontal cooperation in the specific context of demand-responsive passenger transportation. In general, Cruijssen et al. (2007b) describe horizontal cooperation in transportation and logistics as the identification and exploitation of win-win situations by companies operating at the same level of the supply chain. The participating parties may apply various strategies to meet their expectations of such a cooperation, often related to an increased productivity for core ac-

tivities (Cruijssen et al., 2007c). Verdonck et al. (2013) present a detailed classification of cooperation techniques in freight transportation. The present work translates one of these techniques, being joint route planning, to the related domain of demand-responsive passenger transportation. Joint route planning is a particular type of request sharing, which means that service providers may decide to exchange their users in order to align their available resources with the demand of users. Providers retain their own set of vehicles and depots, but routes are designed in an overall efficient manner, which implies that users may be served by any given provider. Such a centralized decision making enables a better capacity utilization and savings in transportation costs, e.g. thanks to reductions in unproductive distance traveled by unloaded vehicles (Cruijssen et al., 2007a). In logistics, joint route planning has been applied to the pickup and delivery problem with time windows (PDPTW) by Krajewska et al. (2008). Their computational tests on both artificial and real-life data reveal cost savings between 16% and 31% obtained by the participating carriers. Although the PDPTW and the DARP are interrelated, the latter is more tightly constrained due to the maximum user ride time. The impact of this additional constraint is difficult to predict. On the one hand, it reduces the operational flexibility of service providers and thus increases the benefits of scaling through cooperation. On the other hand, it may impede efficient customer combinations when cooperating.

4 Algorithm

The algorithm used in this paper is based on an efficient implementation of a large neighborhood search (LNS) framework by Pisinger and Røpke (2010), extended with an additional periodic diversification phase. It combines well-known operators from the literature and specific components that are useful to solve the problem at hand. The algorithmic structure is summarized in Algorithm 1 and the design choices are explained below.

The algorithm requires an initial solution s_{init}, constructed by a random order insertion heuristic. This solution is copied to initialize s_{cur}, s_{best} and s_{ovr}. In each iteration, s_{cur} represents the solution on which operations are performed and s_{best} is the best solution found during the current search phase, which ends at the next periodic diversification. By contrast, s_{ovr} represents the best solution found throughout the entire procedure. Each of the n_{iter} iterations starts with the choice of a destroy operator. The four operators introduced in Section 4.1 may be selected with equal probabilities. The selected operator is executed until a random percentage of requests, situated in the interval $[0.01, des_{max}]$, has been removed from s_{cur}. Then, a repair operator is chosen in order to gradually complete s_{cur}. The three operators discussed in Section 4.2 may be selected with equal probabilities. In order to focus on promising parts of the

search area, the newfound objective value is not allowed to deteriorate excessively. Specifically, it cannot exceed det_{max} times the objective value of s_{best}. This approach corresponds to the record-to-record strategy of Dueck (1993). If the solution cannot be completed without violating this threshold or any feasibility constraint, s_{cur} is replaced with s_{best} and the current iteration is terminated. Otherwise, the complete s_{cur} is intensified using two local search operators which never deteriorate the objective value, as explained in Section 4.3. Afterwards, s_{best} and s_{ovr} are updated if necessary. After f_{div} iterations without improvement of s_{best}, a periodic diversification is applied with the intention of escaping the local optimum, as described in Section 4.4. The resulting s_{cur} serves as the first s_{best} of the next search phase.

4.1 Destroy operators

In each iteration, one of the following destroy operators is executed until the required destroy percentage is reached. The first three operators are based on existing destroy strategies from the literature (Røpke and Pisinger, 2006), whereas the fourth operator is a new contribution.

- The *random removal* operator removes random requests from the solution. All requests have equal probabilities to be selected.

- The *worst removal* operator involves a biased random selection procedure that takes into account the saving obtained by removing a request. Removing undesirably positioned requests may cause considerable improvements of the objective value. Following the roulette wheel principle, the probability of removing request $i \in R$ equals $a_i^2 / \sum_{r \in R} a_r^2$, with a_i denoting the corresponding distance saving and R the set of requests. A probalistic selection procedure is preferred to a deterministic variant in order to encourage diversification.

- The *related removal* operator involves a biased random selection procedure that considers spatial and temporal similarities between requests. Removing interchangeable requests may increase the probability of finding a different solution. After removing a first randomly chosen request i, the similarity between this request and each other request j is computed as follows:

$$b_{i,j} = \left(\frac{d_{i,j} + d_{i+n,j+n}}{d_{max}} + \frac{|T_i - T_j| + |T_{i+n} - T_{j+n}|}{t_{max}} \right)^{-1}$$

$d_{a,b}$	Distance between nodes a and b
d_{max}	Length of the longest arc in the network
T_a	Center of the time window of node a
t_{max}	Difference between the largest upper time window bound and the smallest lower time window bound over all nodes
$i, i+n$	Origin and destination nodes for request i

Consequently, for all following removals during the same iteration, the probability of removing request $j \in R$ equals $b_{i,j}^2 / \sum_{r \in R} b_{i,r}^2$,

Algorithm 1 - Structure of the large neighborhood search algorithm

1:

2: set parameters: n_{iter}, des_{max}, det_{max}, f_{div}

3:

4: construct s_{init}

5: initialize $s_{cur}, s_{best}, s_{ovr} \leftarrow s_{init}$

6: set $i_{impr} = 0$ (iteration of last improvement s_{best})

7:

8: **for** $i = 1 \rightarrow n_{iter}$ **do**

9:

10:    **if** $i = i_{impr} + f_{div}$ **then**

11:       $s_{cur}, s_{best} \leftarrow$ *periodic diversification* on s_{ovr}

12:       $i_{impr} = i$

13:    **end if**

14:

15:    select $p = rand_{(0.01, des_{max})}$

16:    randomly select $D_i \in \{D_{rand}, D_{worst}, D_{rel}, D_{prox}\}$

17:    randomly select $R_i \in \{R_{rand}, R_{gree}, R_{regr}\}$

18:    $s_{cur} \leftarrow R_i(D_i(s_{cur}, p))$

19:

20:    **if** s_{cur} complete **and** $f(s_{cur}) \leq f(s_{best}) * det_{max}$ **then**

21:       $s_{cur} \leftarrow$ *relocate* on s_{cur}

22:       $s_{cur} \leftarrow$ *exchange natural sequences* on s_{cur}

23:

24:       **if** $f(s_{cur}) < f(s_{best})$ **then**

25:          $s_{best} \leftarrow s_{cur}$

26:          $i_{impr} = i$

27:          **if** $f(s_{cur}) < f(s_{ovr})$ **then**

28:             $s_{ovr} \leftarrow s_{cur}$

29:          **end if**

30:       **end if**

31:

32:    **else**

33:       $s_{cur} \leftarrow s_{best}$

34:    **end if**

35:

36: **end for**

37:

38: **return** s_{ovr}

with R denoting the set of remaining requests. Thanks to the small time windows involved with dial-a-ride problems, the actual service time in a node can accurately be approximated by the center of the corresponding time window. As a result, the similarities between requests are static and can be precomputed, which is computationally more efficient than the classical related removal strategy and makes the similarities independent of the scheduling procedure used.

- The *proximity removal* operator involves a biased random selection procedure which favors request removals in areas where different routes overlap in time and space, since enhancing the interaction among routes may be interesting to stimulate horizontal cooperation in the context of this paper. Thus, the probability of removing request $i \in R$ equals $c_i^2 / \sum_{r \in R} c_r^2$, with c_i denoting the similarity between request i and the most similar request in a *different* route. Although similarity is measured as for the related removal operator, both strategies are very different. Related removal starts from a single random request and intends to remove requests that are interchangeable with this first request. In contrast, proximity removal does not require relatedness between the selected requests, but focuses on requests that are each individually related to a request served in another route.

The aforementioned operators were selected after testing their contribution to the solution quality of the algorithm, as will be shown in Section 5.1. The tests also reveal that a route removal operator is rather unappropriate within the current framework, despite its common application in the literature. It tends to result in new solutions whose objective value exceed the allowed threshold.

4.2 Repair operators

In each iteration, one of the following repair operators is executed until either a complete solution is reached or no insertion can be performed without exceeding the maximum deterioration threshold or violating a feasibility constraint. The operators are based on common repair strategies from the literature (Røpke and Pisinger, 2006).

- The *random order insertion* operator repeatedly performs the best insertion of a randomly selected request, checking all insertion positions in any route. Requests have equal probabilities to be selected.

- The *greedy insertion* operator repeatedly performs the best insertion over all requests, considering all insertion positions in any route.

- The *2-regret insertion* operator repeatedly performs the best insertion of the request having the largest regret value, considering all insertion positions in any route. A request's regret

value is defined as the difference between the insertion cost at its second-best insertion position and the insertion cost at its best insertion position. This strategy recognizes that postponing the insertion of requests having a large regret value may deteriorate the eventual solution quality. The second-best insertion position is assumed to be in a different route than the best one, such that both positions cannot simultaneously be eliminated by the insertion of another request.

A random noise factor in the interval [0.95, 1.05] is applied to encourage variation in the choice of the insertion positions and routes. The scheduling procedure of Cordeau and Laporte (2003) will be used to perform time-related feasibility checks on the artificial data in Sections 5.1 and 5.2, which ensures the comparability with other solution approaches. The real-life case in Section 5.3 invokes the scheduling procedure introduced in Molenbruch et al. (2017c), which minimizes the total user ride time of a route. This is in line with the service provider's quality policy.

4.3 Additional local search

Two additional local search operators are applied to repaired solutions whose objective value does not exceed the maximum deterioration threshold. The *relocate* operator considers all requests in a random order and moves them to the best position in any route. At worst, a request is reinserted at its original position, which implies that this operation cannot deteriorate the solution quality. The *exchange natural sequences* operator performs the best swap of two natural sequences, being node sequences before and after which a vehicle is empty. They can be exchanged without violating pairing or precedence constraints. Any two natural sequences in a solution may be used, provided that they belong to different routes and at most one of them is empty. If no improving exchange can be found, the original solution is maintained.

4.4 Periodic diversification

After f_{div} iterations without improvement of s_{best}, a periodic diversification is applied in order to guide the search towards a different part of the solution space. This periodic diversification phase affects s_{cur} and s_{best}, but not s_{ovr}, unless the diversification would immediately result in a new overall best solution. The random removal and random order reinsertion operators presented in Sections 4.1 and 4.2 destroy and repair des_{max} percent of all requests in a copy of s_{ovr}. No threshold applies to the newfound objective value, such that a complete solution is always accepted, provided that all feasibility constraints are respected. If not, the entire diversification phase is reapplied until a feasible solution is found. It is stored into s_{cur} and s_{best}, after which the normal LNS procedure resumes from these solutions.

5 Analysis of horizontal cooperation

The LNS algorithm has been implemented in Python 3.4.3. Note that Cython 0.22, a module that runs Python code in C and allows variables to be declared as C types, is invoked to accelerate the numerous executions of feasibility checks. The computational resources and services used in this paper were provided by the VSC (Vlaams Supercomputer Centrum), funded by the Research Foundation Flanders (FWO) and the Flemish Government, department EWI. All experiments are performed using Xeon E5-2680v2 CPUs at 2.8 GHz with 64 GB of RAM.

Section 5.1 uses existing artificial data to obtain a first impression of the performance of the solution method and the operational savings that can be obtained through horizontal cooperation. However, this data set is rather small and ignores a number of real-life characteristics that may influence the findings. Therefore, Section 5.2 introduces a new artificial data set to analyze the effect of different operational settings on the benefits of horizontal cooperation. Finally, Section 5.3 conducts a real-life case study to observe a pattern in the exchanges of requests among providers. This may allow them to predict which requests contribute in a horizontal cooperation.

5.1 Existing artificial data

A first series of experiments is performed on the a-instances introduced by Cordeau (2006) and extended by Røpke et al. (2007). This artificial data set consists of 24 relatively small benchmark instances, including 16 to 96 requests and 2 to 8 vehicles. An instance is structured as follows. Half of all requests are outbound, whereas the others represent inbound requests. Origins and destinations are randomly and independently generated in a square region $[-10, 10]^2$. The depot is located in the center at coordinates $(0, 0)$. A single user needs to be picked up or delivered at each location. Time windows of 15 minutes are created for destinations of an outbound request, as well as for origins of an inbound request. Travel times equal Euclidean distances. A fixed maximum user ride time of 30 minutes and a fixed service duration of 3 minutes are taken into account. Vehicle capacity is fixed at 3 customers. The maximum route duration is dependent on the instance and ranges from 240 to 720 minutes. Braekers et al. (2014) extended this data set to a multi-depot variant, with four depots located at coordinates (5, 5), (-5, -5), (5, -5) and (-5, 5). This variant will be used to mimic the presence of multiple service providers. If they cooperate, all requests may be handled by vehicles originating from any depot. Otherwise, each request is assigned to one specific depot.

Race	Number surviving	Elite candidates		
1	24/47	0.44	1.05	925
		0.20	1.06	581
		0.26	1.07	152
2	11/41	0.41	1.04	1417
		0.40	1.05	1482
		0.37	1.04	480
3	37/37	**0.40**	**1.04**	**1468**
		0.39	1.05	1456
		0.37	1.04	480

Table 1: Parameter tuning using the Irace package of López-Ibáñez et al. (2016).

5.1.1   Parameter tuning

First, these artificial instances are used for tuning the three main parameters of the metaheuristic. These are (1) the maximum destroy percentage des_{max}, (2) the maximum deterioration factor det_{max} and (3) the diversification frequency f_{div}. Prior to performing detailed parameter tuning tests, a range of appropriate values is identified for each parameter using the automatic iterated racing algorithm of López-Ibáñez et al. (2016). The algorithm is initiated with a very broad range of possible settings for each of the three parameters: (1) [0.20, 0.50], (2) [1.01, 1.08] and (3) [100, 5000]. In each race, a sample of candidate configurations is drawn and repeatedly tested on the single-depot and multi-depot variants of the instances a8-64, a8-80 and a8-96[1]. Candidate configurations are discarded as soon as they perform statistically worse than others. The next race starts with a new sample of candidate configurations that is drawn such that promising parameter settings are favored. In this test, a large tuning budget of 1000 experiments is provided, which implies that the algorithm can perform three subsequent races of approximately 333 experiments. The number of candidate configurations decreases in subsequent races, such that more evaluations per configuration can be made. Table 1 summarizes the results of the automatic tuning. After three races, the algorithm identifies three elite configurations, but is not able to discard any of the 34 other configurations investigated during the last race. This implies that the average results found for all 37 candidate configurations in the last race are extremely close. For example, the differences in solution quality found for the three elite candidates do not exceed 0.04%. Therefore, a more in-depth tuning is required within the promising ranges identified by the tuning algorithm. Based on the surviving candidates throughout the entire procedure, three well-performing settings were identified for all three parameters: (1) 0.30, 0.35 and 0.40, (2) 1.04, 1.05, 1.06 and (3) 500, 1000 and 1500.

For each of the 27 resulting combinations, the algorithm is now

---

[1]Tests on smaller instances do not result in any optimality gaps at all.

	Single-depot		Multi-depot		Average	
	dist	time (m)	dist	time (m)	dist	time (m)
Maximum destroy percentage des$_{max}$						
30%	975.99	13.60	933.08	12.73	954.53	13.16
35%	**975.93**	14.85	**932.81**	14.00	**954.37**	14.42
40%	975.95	16.05	932.84	15.20	954.39	15.62
Maximum deterioration factor det$_{max}$						
1.04	975.97	13.96	932.90	13.05	954.44	13.51
1.05	**975.88**	14.89	**932.88**	14.03	**954.38**	14.46
1.06	976.01	15.64	932.94	14.84	954.47	15.24
Diversification frequency f$_{div}$						
500	**975.75**	15.52	**932.69**	14.55	**954.22**	15.03
1000	975.96	14.75	932.96	13.91	954.46	14.33
1500	976.15	14.22	933.07	13.47	954.61	13.85

Table 2: Parameter tuning on the instances of Røpke et al. (2007) and Braekers et al. (2014).

run 20 times (20000 iterations per run) on the selected instances. Table 2 summarizes the average objective value and average computation time (in minutes) for each parameter separately. This analysis confirms that the performance of the LNS algorithm remains remarkably stable and reveals $des_{max} = 0.35$, $det_{max} = 1.05$ and $f_{div} = 500$ as the preferred parameter setting. This combination also generates the best average result of all 27 combinations, with objective values 975.22 and 932.26. The fact that the resulting parameter values are identical for the single-depot and multi-depot instances illustrates the robustness of the parameter tuning. As expected, the solution quality is not very sensitive to small changes in des_{max} and det_{max}, as long as these parameters remain within acceptable ranges. That is, large enough to escape the current solution, but small enough to preserve an efficient route structure. However, rather frequent diversifications through a small value of f_{div} are highly beneficial, which was not apparent from the automatic tuning. All three parameters clearly influence the computation time. In other words, small sacrifices of solution quality may reduce the computation time if needed.

Also note that the automatic tuning does not take into account the spread of the solutions for a particular parameter setting. Figure 1 reveals a more detailed analysis in which information on the average solution quality for each parameter separately (based on Table 2) is supplemented with the worst en the best solution found. This reveals that applying a larger destroy percentage and a larger maximum deterioration may be useful to avoid extreme solutions. The worst solution is - contrary to the average solution - considerably better than for the preferred setting, whereas the best-found solution tends to remain stable. This is also confirmed by an analysis of the corresponding standard deviations, which is not shown here.
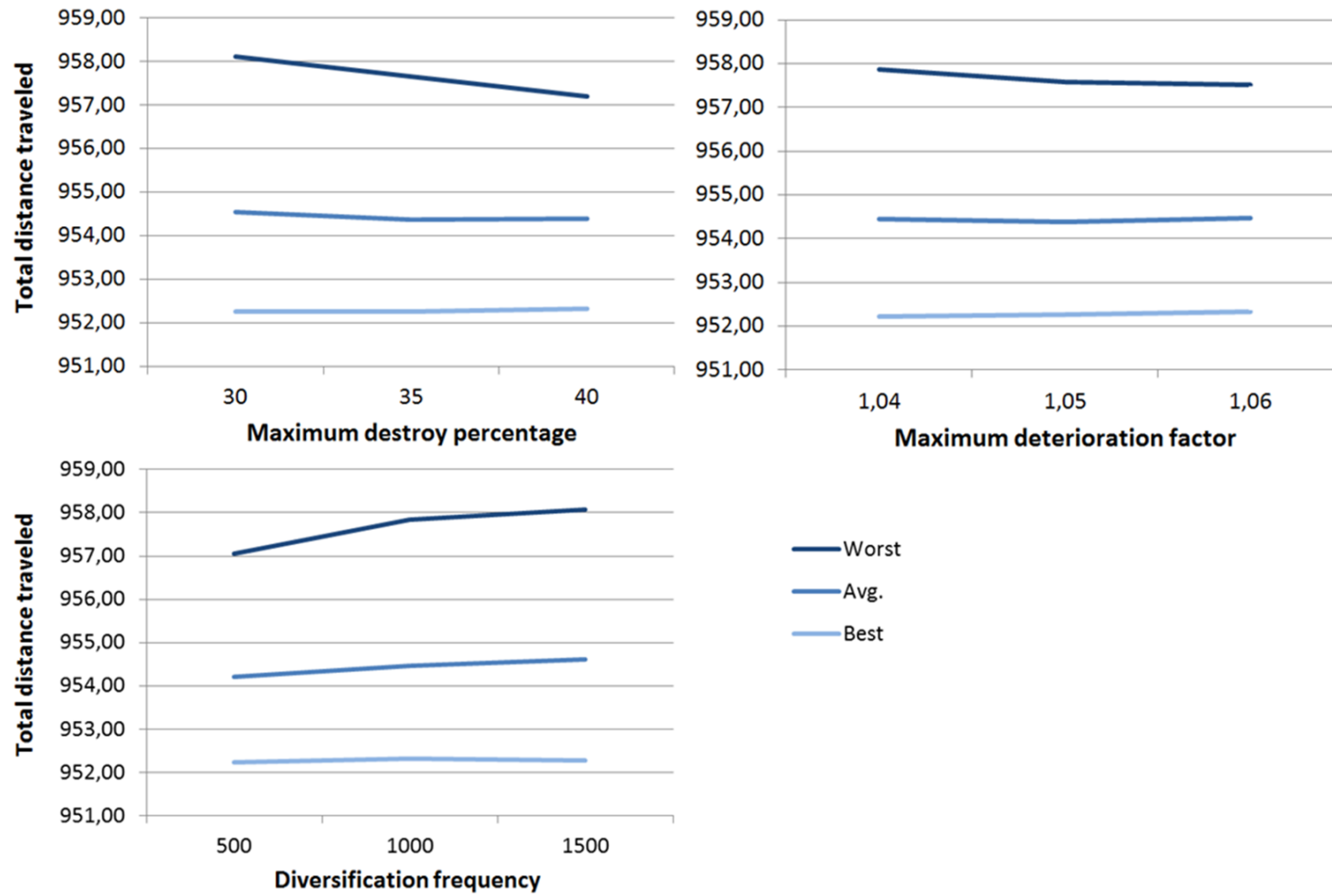
Figure 1: Parameter tuning based on best, average and worst solutions.

Iter.	Single-depot		Multi-depot	
	dist	time (m)	dist	time (m)
5000	977.52	3.95	935.09	3.71
10000	976.44	7.87	933.28	7.39
15000	975.67	11.75	932.61	11.01
20000	**975.22**	**15.66**	**932.26**	**14.65**
25000	975.17	19.61	931.99	18.29
30000	975.09	23.50	931.87	21.91
35000	975.03	27.39	931.71	25.55
40000	974.95	31.29	931.68	29.21

Table 3: Evolution of the average solution quality throughout the iterations.

Even though all experiments performed on the artificial instances are based on 20000 iterations, it is interesting to point out that the LNS metaheuristic tends to converge towards a good solution range rather quickly. Table 3 shows the evolution of the average solution quality and computation time for the single-depot and multi-depot variants of instances a8-64, a8-80 and a8-96 after every 5000 iterations. Note that the solutions for multi-depot instances tend to converge somewhat more slowly, which confirms the observation of Braekers et al. (2014) that these instances are more difficult to solve. In other words, the larger search space offers more degrees of freedom. Nevertheless, after only 5000 iterations, the average objective value already approximates its eventual value. Beyond 20000 iterations, the limited improvements in solution quality do no longer justify the linear increase in computation time.

5.1.2 Design choices

As discussed in Section 4.1, the set of destroy operators used slightly deviates from the common practice in the literature. For example, no route removal is included and an additional proximity removal operator is proposed. This choice was based on preliminary tests of different designs. The upper part of Table 4 summarizes the effect of adding a route removal operator or deleting one of the removal operators currently included. For every design, 20 runs consisting of 20000 iterations were performed on the single-depot and multi-depot variants of instances a8-64, a8-80 and a8-96. Adding a route removal operator, which removes all requests in random routes until the destroy percentage is reached, causes the largest deterioration of the objective value. This operator tends to destroy the solution structure to such an extent that the repaired solution often violates the maximum deterioration threshold. On the other hand, it reduces the total computation time, which indicates that this operator requires little computational effort. Deleting the related removal causes the second largest deterioration of the objective value. As explained in Section 4.1, it is beneficial to remove requests that are easily interchangeable in order to obtain a different repaired solution. The related removal is computationally inexpensive thanks to

the fact that the similarities between requests can be computed in advance. The other operators - including the newly proposed proximity removal - also have a positive, though more limited effect on the solution quality.

Generally, the effects of deleting any operator remain very limited thanks to the additional local search phase, which contributes considerably to the performance of the algorithm. As illustrated in the middle part of Table 4, leaving out these local search operators causes a considerable deterioration of the solution quality, even when the corresponding savings in computation time are spent on additional LNS iterations. In other words, the advantages caused by including the additional local search phase justify its computational requirements amounting up to half of the total computation time. Particularly the *exchange natural sequences* operators has an important impact. Deleting this operator would cause a strong deterioration of the solution quality, whereas its contribution to the total computation time is relatively small. In contrast, the *relocate* involves considerable computational requirements relative to the associated effect on the solution quality. Leaving out the latter operator may be advisable whenever solutions of reasonable quality need to be computed within a short time frame. However, if both local search operators are included, their order of application hardly affects the solution quality or computation time.

Finally, the use of the record-to-record strategy (Dueck, 1993) as an acceptance criterion, meaning that a fixed maximum deterioration factor determines whether new solution are accepted, needs to be justified. To this end, the same instances are solved using the more traditional simulated annealing acceptance criterion based on the Boltzmann function. Its parameters, being (1) the start temperature control parameter and (2) the cooling rate, are set as determined in Røpke and Pisinger (2006) for the closely related PDPTW. Using the Boltzmann function, the computational tests reveal a slight deterioration of the solution quality. In addition, the implementation of the Boltzmann function is inherently more complicated, e.g. since two parameters need to be tuned and repairs should always be entirely performed before an assessment can be made. On the other hand, the computation time is slightly reduced due to the fact that additional local search is only executed on accepted new solutions, which become rare towards the end of the procedure. However, since this may be the very reason why worse solution are obtained, preference is given to the record-to-record strategy.

	dist	time (m)
Current design	**953.74**	**15.15**
... with route removal	954.78	14.47
... without random removal	953.87	15.41
... without worst removal	953.83	15.18
... without related removal	954.32	15.36
... without proximity removal	953.89	14.74
... without LS (20000 LNS iter.)	968.40	7.59
... without LS (40000 LNS iter.)	965.30	15.15
... with LS in reverse order	953.85	15.24
... without exchange natural sequences	963.12	14.16
... without relocate	955.22	8.55
... with Boltzmann acceptance criterion	955.40	11.06

Table 4: Results for different designs of the removal phase.

5.1.3 Computational tests

Tables 5 and 6 present the detailed average results obtained by the LNS metaheuristic for all single-depot instances of Røpke et al. (2007) and multi-depot instances of Braekers et al. (2014) after 20 runs, consisting of 20000 iterations each. For the single-depot instances, the solution quality comes close to the results of the hybrid genetic algorithm of Masmoudi et al. (2017). They published the best-known heuristic solutions on these instances, performing 5 runs consisting of 50000 iterations. The proposed LNS metaheuristic leaves no optimality gap for the smallest half of the instances, whereas the average gap for the largest half of the instances equals a mere 0.05% (varying between 0.00% and 0.26%). Looking at the best solutions, the results for all instances match the optimal solutions reported by Gschwind and Irnich (2014). The results for the multidepot variant are compared with Braekers et al. (2014), performing 5 runs consisting of 350,000 deterministic annealing iterations. The LNS metaheuristic improves 5 of their average solutions, although its overall performance is slightly worse. The average optimality gaps are 0.00% for the smallest half of the instances and 0.19% for the largest half (varying between 0.00% and 0.45%). Looking at the best solutions, the results for all but two instances match the optimal solutions computed by Braekers et al. (2014). The optimal results for instances a8-80 and a8-96 are still unknown, but new best heuristic results are found. The corresponding gap for these instances is expressed relative to the best-known upper bound. Note that the total distance traveled is consistently lower in the presence of multiple depots than in the single-depot case, although the number of requests and the total size of the fleet remains unchanged. This indicates that from a cost perspective, service providers may take advantage of having multiple depots spread across their service area. These operational savings should of course be weighed against the fixed costs of exploiting multiple depots and a potential loss of operational flexibility. For example, a central depot often accommodates a backup driver to deal with unexpected sickness of a driver,

whereas it may be impracticable to provide such a backup in each of the multiple depots.

Similar results are obtained when 50000 iterations of the LNS algorithm are applied to the larger, more difficult benchmark data set of Cordeau and Laporte (2003). It consists of 20 instances, containing between 24 and 144 requests. The requests are generated in a comparable manner as in Cordeau (2006), but clustered around a number of seed points. The coordinates of the depot are the averages over those of the seed points. A single user is picked up or delivered at each location. Half of the instances have time windows of 30 minutes, whereas wider time windows of 60 minutes are defined for the other half. A fixed maximum user ride time of 90 minutes and a fixed service duration of 10 minutes are imposed. Vehicle capacity is fixed at 6 customers and the maximum route duration equals 480 minutes for all instances. Table 7 compares the results of the LNS metaheuristic with the hybrid genetic algorithm of Masmoudi et al. (2017), obtaining the best solution quality for these instances so far. The average optimality gap of the LNS metaheuristic equals 1.02%, which is relatively close to the 0.47% gap found by Masmoudi et al. (2017). The difference is even smaller for the best solutions: 0.29% versus 0.19%. All optimality gaps are expressed relative to the best-known solutions in the literature, since optimal results could not be computed yet for these instances. A new best-known solution is discovered for instance *R10b*.

Inst.	LNS algorithm					Masmoudi et al. (2017)				
	Avg.	Gap	T(m)	Best	Gap	Avg.	Gap	T(m)	Best	Gap
a2-16	294.25*	0.00%	0.57	294.25*	0.00%	294.25*	0.00%	0.21	294.25*	0.00%
a2-20	344.83*	0.00%	0.87	344.83*	0.00%	344.83*	0.00%	0.48	344.83*	0.00%
a2-24	431.12*	0.00%	1.14	431.12*	0.00%	431.12*	0.00%	0.42	431.12*	0.00%
a3-18	300.48*	0.00%	0.98	300.48*	0.00%	300.48*	0.00%	0.22	300.48*	0.00%
a3-24	344.83*	0.00%	1.32	344.83*	0.00%	344.83*	0.00%	0.40	344.83*	0.00%
a3-30	494.85*	0.00%	1.94	494.85*	0.00%	494.85*	0.00%	0.45	494.85*	0.00%
a3-36	583.19*	0.00%	2.54	583.19*	0.00%	583.19*	0.00%	0.67	583.19*	0.00%
a4-16	282.68*	0.00%	0.97	282.68*	0.00%	282.68*	0.00%	0.23	282.68*	0.00%
a4-24	375.02*	0.00%	1.49	375.02*	0.00%	375.02*	0.00%	0.31	375.02*	0.00%
a4-32	485.50*	0.00%	2.59	485.50*	0.00%	485.50*	0.00%	0.61	485.50*	0.00%
a4-40	557.69*	0.00%	3.10	557.69*	0.00%	557.69*	0.00%	0.63	557.69*	0.00%
a4-48	668.82*	0.00%	5.53	668.82*	0.00%	668.82*	0.00%	0.85	668.82*	0.00%
Avg.	**430.27**	**0.00%**	**1.92**	**430.27**	**0.00%**	**430.27**	**0.00%**	**0.46**	**430.27**	**0.00%**
a5-40	498.41*	0.00%	3.89	498.41*	0.00%	498.41*	0.00%	0.52	498.41*	0.00%
a5-50	686.62*	0.00%	6.35	686.62*	0.00%	686.62*	0.00%	0.73	686.62*	0.00%
a5-60	808.42*	0.00%	8.16	808.42*	0.00%	808.42*	0.00%	1.05	808.42*	0.00%
a6-48	604.12*	0.00%	6.06	604.12*	0.00%	604.12*	0.00%	0.73	604.12*	0.00%
a6-60	819.30	0.01%	8.74	819.25*	0.00%	819.25*	0.00%	0.88	819.25*	0.00%
a6-72	916.46	0.05%	11.71	916.05*	0.00%	916.05*	0.00%	1.28	916.05*	0.00%
a7-56	724.04*	0.00%	8.00	724.04*	0.00%	724.04*	0.00%	0.77	724.04*	0.00%
a7-70	889.44	0.04%	11.18	889.12*	0.00%	889.12*	0.00%	1.04	889.12*	0.00%
a7-84	1036.08	0.26%	15.41	1033.37*	0.00%	1033.37*	0.00%	1.39	1033.37*	0.00%
a8-64	747.58	0.02%	10.85	747.46*	0.00%	747.46*	0.00%	0.90	747.46*	0.00%
a8-80	946.37	0.07%	16.09	945.73*	0.00%	945.97	0.02%	1.38	945.73*	0.00%
a8-96	1231.72	0.17%	20.02	1229.66*	0.00%	1231.04	0.11%	1.59	1229.66*	0.00%
Avg.	**825.71**	**0.05%**	**10.54**	**825.19**	**0.00%**	**825.32**	**0.01%**	**1.02**	**825.19**	**0.00%**

∗ = optimal solution

Table 5: Results on the single-depot artificial data of Røpke et al. (2007).

	LNS algorithm					Braekers et al. (2014)				
Inst.	Avg.	Gap	T(m)	Best	Gap	Avg.	Gap	T(m)	Best	Gap
a2-16	284.18*	0.00%	0.57	284.18*	0.00%	284.18*	0.00%	0.09	284.18*	0.00%
a2-20	343.43*	0.00%	0.86	343.43*	0.00%	343.43*	0.00%	0.27	343.43*	0.00%
a2-24	427.17*	0.00%	1.13	427.17*	0.00%	427.17*	0.00%	0.26	427.17*	0.00%
a3-18	289.67*	0.00%	0.83	289.67*	0.00%	289.67*	0.00%	0.13	289.67*	0.00%
a3-24	348.30*	0.00%	1.38	348.30*	0.00%	348.30*	0.00%	0.24	348.30*	0.00%
a3-30	469.16*	0.00%	1.74	469.16*	0.00%	469.16*	0.00%	0.24	469.16*	0.00%
a3-36	592.42*	0.00%	2.57	592.42*	0.00%	592.42*	0.00%	0.41	592.42*	0.00%
a4-16	262.44*	0.00%	0.85	262.44*	0.00%	262.44*	0.00%	0.12	262.44*	0.00%
a4-24	355.72*	0.00%	1.49	355.72*	0.00%	355.72*	0.00%	0.18	355.72*	0.00%
a4-32	461.65*	0.00%	2.48	461.65*	0.00%	461.65*	0.00%	0.30	461.65*	0.00%
a4-40	540.42	0.01%	3.20	540.34*	0.00%	540.34*	0.00%	0.33	540.34*	0.00%
a4-48	631.75*	0.00%	5.20	631.75*	0.00%	632.31	0.09%	0.43	631.75*	0.00%
Avg.	**417.19**	**0.00%**	**1.85**	**417.19**	**0.00%**	**417.23**	**0.01%**	**0.25**	**417.19**	**0.00%**
a5-40	482.20	0.00%	4.12	482.19*	0.00%	482.19*	0.00%	0.33	482.19*	0.00%
a5-50	665.08	0.08%	6.02	664.54*	0.00%	665.17	0.09%	0.45	664.54*	0.00%
a5-60	790.13	0.03%	7.45	789.87*	0.00%	789.87*	0.00%	0.58	789.87*	0.00%
a6-48	586.85	0.13%	5.70	586.08*	0.00%	586.08*	0.00%	0.40	586.08*	0.00%
a6-60	777.99	0.17%	8.07	776.63*	0.00%	776.65	0.00%	0.57	776.63*	0.00%
a6-72	884.38	0.07%	11.29	883.77*	0.00%	883.77*	0.00%	0.83	883.77*	0.00%
a7-56	680.44	0.05%	7.08	680.08*	0.00%	682.14	0.30%	0.47	680.08*	0.00%
a7-70	857.09	0.34%	11.20	854.22*	0.00%	857.67	0.40%	0.70	855.76	0.18%
a7-84	1011.82	0.45%	14.82	1007.33*	0.00%	1009.92	0.26%	0.86	1007.33*	0.00%
a8-64	715.04	0.36%	9.83	713.11*	0.00%	713.11*	0.00%	0.73	713.11*	0.00%
a8-80	891.82	0.28%	14.84	889.29	0.00%	892.79	0.39%	1.19	890.69	0.16%
a8-96	1189.91	0.38%	19.28	1186.63	0.10%	1189.74	0.36%	1.57	1187.26	0.15%
Avg.	**794.39**	**0.19%**	**9.98**	**792.81**	**0.01%**	**794.09**	**0.15%**	**0.72**	**793.11**	**0.04%**

∗ = optimal solution

Table 6: Results on the multi-depot artificial data of Braekers et al. (2014).

Inst.	*LNS algorithm*					*Masmoudi et al. (2017)*				
	Avg.	Gap	T(m)	Best	Gap	Avg.	Gap	T(m)	Best	Gap
R1a	190.02	0.00%	4.93	190.02	0.00%	190.02	0.00%	0.35	190.02	0.00%
R2a	301.34	0.00%	19.68	301.34	0.00%	301.34	0.00%	0.89	301.34	0.00%
R3a	533.75	0.33%	40.27	532.00	0.00%	534.08	0.39%	1.04	532.00	0.00%
R4a	574.29	0.71%	72.22	570.25	0.00%	571.45	0.21%	1.59	570.25	0.00%
R5a	636.00	1.45%	107.80	629.93	0.48%	631.39	0.71%	1.90	628.48	0.25%
R6a	799.52	1.82%	151.77	791.25	0.76%	788.52	0.42%	2.28	787.41	0.27%
R7a	292.43	0.25%	10.60	291.71	0.00%	291.79	0.03%	0.48	291.71	0.00%
R8a	493.69	1.20%	38.20	491.01	0.65%	491.53	0.76%	1.03	488.89	0.22%
R9a	668.85	1.60%	75.20	661.06	0.42%	660.24	0.29%	1.54	658.31	0.00%
R10a	871.20	2.28%	138.24	860.06	0.97%	859.91	0.95%	2.43	853.16	0.16%
R1b	164.46	0.00%	6.16	164.46	0.00%	164.46	0.00%	0.51	164.46	0.00%
R2b	296.12	0.16%	27.75	295.66	0.00%	295.66	0.00%	1.09	295.66	0.00%
R3b	489.82	1.03%	61.77	484.82	0.00%	487.23	0.50%	1.62	484.83	0.00%
R4b	535.48	1.16%	123.63	530.22	0.17%	532.19	0.54%	2.49	531.86	0.48%
R5b	585.23	1.38%	177.01	578.66	0.24%	582.06	0.83%	3.30	579.03	0.30%
R6b	746.02	2.10%	256.92	740.25	1.31%	741.06	1.42%	3.84	737.03	0.87%
R7b	248.21	0.00%	16.23	248.21	0.00%	248.29	0.03%	0.72	248.21	0.00%
R8b	463.67	1.08%	55.64	461.48	0.60%	463.32	1.00%	1.72	461.11	0.52%
R9b	600.10	1.11%	139.98	594.64	0.19%	595.37	0.32%	3.12	593.49	0.00%
R10b	805.31	2.69%	244.83	***784.22***	***0.00%***	793.64	1.20%	3.46	791.01	0.87%
Avg.	**514.78**	**1.02%**	**88.44**	**510.06**	**0.29%**	**511.18**	**0.48%**	**1.77**	**509.41**	**0.20%**

Table 7: Results on the single-depot artificial data of Cordeau and Laporte (2003).

The computation times exceed the ones mentioned by Masmoudi et al. (2017) and Braekers et al. (2014), even though a relatively small number of iterations is performed. In this respect, it should be noted that these authors implemented their algorithms using the C++ programming language, rather than Python. The Python language allows for a rapid development of object-oriented programs, especially suitable to examine algorithm potential. Unfortunately, these programs tend to execute relatively slowly. A fair comparison between the proposed LNS and other metaheuristics can only be based on results obtained upon meeting a criterion that is independent of the processor speed or the coding efficiency (Leyman and De Causmaecker, 2017), such as the number of routes investigated during the procedure (i.e. the number of times a feasibility check has been performed). Anyway, this contribution has no intention to be competitive in terms of computation time, particularly since the analysis of centralized decision making through joint route planning is situated at a strategic level, rather than at a time-critical operational level. The aforementioned results indicate that in terms of solution quality, the LNS metaheuristic can be competitive with the current state of the art within reasonable computation times. Hence, it is an appropriate instrument for assessing the potential of joint route planning in dial-a-ride services.

5.1.4 Horizontal cooperation

The presence of multiple depots in the data of Braekers et al. (2014) allows to analyze the impact of a horizontal cooperation for these artificial instances. In fact, the solution strategy adopted so far assumes a cooperation between four service providers, i.e. the four depots. Each request may be handled by a vehicle originating from any depot, which can be seen as service providers that allow the exchange of any request among each other. The corresponding optimal or best-known results are repeated in the second column of Table 8.

A scenario without horizontal cooperation can be created by assigning each request to one specific service provider. To this end, it is first assumed that a request can only be handled by a vehicle originating from the depot closest to the customer's *home location*, which represents the provider that the customer would logically choose. Each depot disposes of one fourth of the total fleet. If the number of vehicles is not sufficient to serve all customers, an additional vehicle is added. The results are now obtained by solving the problem for each depot separately, given the assignment of requests to depots. The third column of Table 8 presents the joint total distance traveled without cooperation, the associated increase in distance traveled (relative to the second column) and the minimal fleet requirement. No exact solutions are known for this scenario, but the LNS metaheuristic is likely to generate optimal solutions for such small-sized problems. The results show that for this data set, the increase in total distance traveled ranges between 11.46% and

Inst.	Cooperation		No cooperation[2]			No cooperation[3]		
	dist	fleet	dist	dist incr	fleet	dist	dist incr	fleet
a2-16	284.18	2	362.56	27.58%	5	349.25	22.90%	4
a2-20	343.43	2	382.79	11.46%	4	363.37	5.81%	5
a2-24	427.17	2	489.57	14.61%	5	484.27	13.37%	5
a3-18	289.67	3	356.65	23.12%	4	368.80	27.32%	4
a3-24	348.30	3	460.87	32.31%	4	436.73	25.39%	5
a3-30	469.16	3	584.59	24.60%	5	574.66	22.49%	6
a3-36	592.42	3	701.95	18.49%	7	692.68	16.92%	5
a4-16	262.44	4	322.43	22.86%	5	322.40	22.85%	6
a4-24	355.72	4	446.56	25.54%	7	460.58	29.48%	5
a4-32	461.65	4	600.96	23.18%	7	580.03	25.64%	6
a4-40	540.34	4	717.47	30.18%	8	731.19	35.32%	6
a4-48	631.75	4	777.48	23.07%	8	829.87	31.36%	5
Avg.	**417.19**	**3.17**	**516.99**	**23.92%**	**5.75**	**516.15**	**23.72%**	**5.15**
a5-40	482.19	5	654.59	35.75%	6	619.94	28.57%	6
a5-50	664.54	5	943.59	41.99%	6	782.73	17.79%	8
a5-60	789.87	5	1020.23	29.16%	7	1016.31	28.67%	7
a6-48	586.08	6	759.46	29.58%	8	770.44	31.46%	8
a6-60	776.63	6	1011.22	30.21%	8	993.60	27.93%	9
a6-72	883.77	6	1189.43	34.59%	9	1246.96	41.10%	8
a7-56	680.08	7	931.57	36.98%	9	884.40	30.04%	8
a7-70	854.22	7	1111.93	30.17%	8	1172.38	37.25%	8
a7-84	1007.33	7	1368.92	35.90%	8	1399.84	38.97%	7
a8-64	713.11	8	1003.07	40.66%	9	976.66	36.96%	8
a8-80	889.29	8	1192.13	34.05%	10	1129.13	26.97%	8
a8-96	1185.45	8	1569.84	32.43%	10	1557.17	31.36%	9
Avg.	**792.71**	**6.50**	**1063.00**	**34.10%**	**8.17**	**1045.80**	**31.93%**	**7.83**

Table 8: Effect of horizontal cooperation on instances of Braekers et al. (2014).

41.99%, with an average of 22.37%. The fleet requirements are also larger than in the scenario with cooperation, although fleet minimization is not an objective as such. Particularly smaller service providers considerably reduce their fleet requirements by exchanging requests that are inconvenient to include in their own routes.

Finally, a different scenario without horizontal cooperation is created to illustrate the impact of the request assignment rule. The fourth column of Table 8 contains the results when customers are assumed to be served by a vehicle originating from the depot closest to the customer's *outbound location*. Compared with the scenario with cooperation, the increase in joint total distance traveled ranges between 5.81% and 41.10%, with an average of 21.69%. Hence, both assignment rules in the scenario without cooperation give rise to similar benefits when a horizontal cooperation is established, given the characteristics of the data set under consideration. For example, the users in these instances do not submit paired outbound and inbound requests, which will be the case in other experiments in this paper.

---

[2]Assignment to a depot based on the home location of the user
[3]Assignment to a depot based on the outbound location of the user

5.2 New artificial data

Real-life dial-a-ride systems are often characterized by additional features not included in the data used in Section 5.1. First, requests are often submitted in pairs. A customer traveling from his home location to a certain event (e.g. a medical appointment) is likely to make the reverse trip later that day. Although the existing artificial instances consist of an equal number of outbound and inbound requests, their geographical and time-related pairing is ignored. Second, requests tend to be clustered in space. Most origins and destinations of customers will be located in densely populated areas or at specific attraction poles, such as hospitals. The existing artificial instances do not take into account this observation. Finally, most real-life systems serve a larger number of customers. Particularly in the multi-depot case, even the largest instance reaches an average of only 24 requests per depot (= 96 requests for four depots). Therefore, a more extensive artificial data set with additional real-life characteristics is introduced in Section 5.2.1. Section 5.2.2 presents six hypotheses regarding the effect of the operational setting on the benefits of horizontal cooperation, which are then tested in Section 5.2.3 using the new artificial instances.

5.2.1 Data set

In order to analyze the effect of horizontal cooperation in the presence of somewhat more realistic features, 10 new artificial multi-depot instances have been created. All instances include a total number of 400 requests, originating from 200 customers submitting a corresponding outbound and inbound request. Random time windows of 15 minutes are imposed for the arrival of outbound trips and for the departure of inbound trips. The time span between the latest arrival for an outbound trip and the earliest departure for the corresponding inbound trip ranges between one and four hours. The service area $[-10, 10]^2$ is assumed to consist of four densely populated cities or clustering poles. Each clustering pole comprises a vehicle depot possessing 8 vehicles, which results in a total number of four depots (i.e. four different service providers) and 32 vehicles. This fleet size is sufficient to serve all requests in all scenarios that will be investigated. The depots are located at coordinates (-5, -5), (5, -5), (-5, 5) and (5, 5). An equal number of 50 customers (100 requests) is assigned to each depot. This is reflected by the fact that their home location, being the origin of their outbound trip and the destination of their inbound trip, is clustered around a particular depot using the procedure of Cordeau et al. (1997), with clustering parameter $\phi = 0.75$. Based on real-life experience, it is assumed that 25% of the customers request a long-distance trip, in which case the destination of their outbound trip (also the origin of their inbound trip) is clustered around a different depot than their home location. For all other customers, both locations are clustered around the same depot. The maximum user ride time and maximum

24

route duration equal 30 minutes and 720 minutes, respectively. The service at each stop requires three minutes and the vehicle capacity is limited to three customers.

5.2.2 Effect of operational setting on cooperation benefits

The impact of the operational setting on the benefits of horizontal cooperation is analyzed by applying minor changes to the instances introduced in the previous section. Six groups of alternative operational settings are considered based on the following hypotheses:

1. The relative benefits of horizontal cooperation increase as the *width of the customers' time windows decreases*. This hypothesis is based on the fact that a reduced number of feasible customer combinations decreases the operational flexibility of a service provider (Molenbruch et al., 2017a), such that the benefits of a horizontal cooperation may become more important. Reductions of the time window width from 15 minutes to (a) 10, (b) 5 and (c) 0 minutes are investigated.

2. The relative benefits of horizontal cooperation increase as *more users request long-distance trips*. This hypothesis assumes that a frequent presence of service providers in each other's service areas enables more request exchanges and increases the potential of horizontal cooperation. The percentage of long-distance requests is increased from 25% to (a) 50%, (b) 75% and (c) 100%.

3. The relative benefits of horizontal cooperation increase as the *strength of the clustering around the clustering poles weakens*. This hypothesis assumes that overlapping service areas of different providers enable more potential request exchanges and increase the benefits of horizontal cooperation. The clustering parameter ϕ is reduced from 0.75 to (a) 0.60, (b) 0.45 and (c) 0.30[2].

4. The relative benefits of horizontal cooperation increase as the *clustering poles are located closer to each other*. This hypothesis may again be explained by an increased overlap of the providers' service areas. The coordinates of the depots are changed from -5.00/5.00 to (a) -3.33/3.33, (b) -1.66/1.66 and (c) a single central depot at (0, 0).

5. The relative benefits of horizontal cooperation increase as the *size of the problem reduces*. This hypothesis assumes that small-sized service providers have fewer possibilities to efficiently combine their own customers, such that they gain considerable operational flexibility through horizontal cooperation. The size of the instances is reduced from 400 requests and 32 vehicles

---

[2]Scenarios with lower values of ϕ have not been considered, since they are unrealistic. In the most extreme scenario ($\phi = 0.00$), customer locations would be spread in a completely random manner, not influenced by the distance to any depot. As a result, a customer living near a particular service provider would have a 75% probability of contacting one of the three other providers, which does not correspond to realistic behavior.

to (a) 300 requests and 24 vehicles, (b) 200 requests and 16 vehicles and (c) 100 requests and 8 vehicles.

6. The relative benefits of horizontal cooperation either increase or decrease as *the sizes of the service providers become unbalanced*. It is difficult to predict whether the relatively large benefits obtained by small providers or the relatively small benefits obtained by large providers will prevail. Starting from equally-sized providers handling 100 requests each, the balance is changed into (a) 125-75-75-125, (b) 150-50-50-150 and (c) 175-25-25-175. The large-sized providers exploit depots (5, 5) and (-5, -5) and the fleet is allocated proportionally: (a) 10-6-6-10, (b) 12-4-4-12 and (c) 14-2-2-14.

To investigate these hypotheses, adaptations are applied to the current operational setting for each hypothesis separately. The relative benefits of horizontal cooperation are computed for the adapted operational setting and then compared to the baseline scenario. The results are obtained after 2 runs for each instance (20 runs in total), with each run consisting of 20000 iterations. It should be mentioned that, in order to deal with the large scale of the instances, operations in each iteration of the LNS algorithm are performed on a selection of eight related routes, rather than on the complete set of routes (e.g. similar to the principle of parallel computing (Taillard, 1993)). For large-scale instances with many vehicles, requests that are distant from each other are indeed unlikely to be successfully interchangeable. The selection of related routes prior to each iteration requires the following steps. First, a random route is selected based on equal probabilities. Second, the relatedness between this route and all other routes is assessed. For this purpose, the distances between each node in the first route and its closest node in the second route (which may also be the depot node) are summed, resulting in a measure that is small for related routes. Third, the seven routes being most related to the first route are selected, provided that at least two routes in the total selection are not empty. This selection procedure is repeated prior to each iteration. The choice of the subset's size is based on computational tests, of which the results are shown in Table 9. This table shows the average total distance traveled for different sizes of the subset, as well as the corresponding computation time in minutes. Solution quality improves at a declining rate as the size of the subset increases, whereas computation times increase at a growing rate. Subsets consisting of 8 routes provide a reasonable equilibrium between solution quality and computation time.

5.2.3 Results

Table 10 presents the average results in the baseline scenario and the six alternative operational settings. The first column refers to the scenario under consideration. The second column gives the relative saving in joint distance traveled thanks to horizontal cooperation. Detailed information on the distance traveled and the required fleet

26

Size	dist	time
4 routes	2191.60	15.72
6 routes	2154.08	26.69
8 routes	**2130.24**	**39.95**
10 routes	2114.63	57.10
12 routes	2107.81	75.77

Table 9: Results for different sizes of the selected subset of routes.

with and without cooperation is displayed in the next four columns. The seventh column shows whether the relative benefits of horizontal cooperation in a certain scenario are statistically different from the 'previous' scenario. For example, scenario *50% long trips* is compared with the baseline scenario, scenario *75% long trips* is compared with scenario *50% long trips*, etc. The corresponding p-values are indicated. For hypotheses (1) and (5), the assessment is based on the *Wilcoxon signed-ranks test*, since the locations of the customer requests remain unchanged from the baseline scenario. The weaker, but less restricted *Mann-Whitney U-test* is used for the other hypotheses (Anderson et al., 2010). Finally, the last column in Table 10 mentions the percentage of requests that are exchanged to another depot in the scenario with cooperation.

Scenario	Benefits	Cooperation		No cooperation		Sign.	Exch.
	dist	dist	fleet	dist	fleet	p-val	pct.
Baseline	5.61%	2130.24	30.20	2256.96	32.00	-	43.34%
Hypothesis 1							
- time window width 10	6.70%	2195.39	31.45	2353.15	32.00	0.005*	41.61%
- time window width 5	7.85%	2302.46	31.95	2498.56	32.00	0.005*	42.19%
- time window width 0	10.33%	2489.76	32.00	2776.47	32.00	0.005*	42.63%
Hypothesis 2							
- 50% long trips	8.01%	2563.82	29.80	2786.95	32.00	0.000*	53.38%
- 75% long trips	10.62%	3001.90	29.80	3358.43	32.00	0.000*	57.69%
- 100% long trips	11.56%	3349.86	29.95	3787.91	32.00	0.063	57.44%
Hypothesis 3							
- clustering $\phi = 0.60$	6.27%	2337.91	29.95	2494.23	32.00	0.436	47.06%
- clustering $\phi = 0.45$	9.00%	2601.31	30.50	2858.70	32.00	0.001*	51.23%
- clustering $\phi = 0.30$	12.67%	2925.11	30.55	3349.34	32.00	0.001*	57.61%
Hypothesis 4							
- depots -3.33/3.33	7.17%	1856.75	29.15	2000.23	31.95	0.003*	48.75%
- depots -1.66/1.66	11.53%	1559.06	26.40	1762.28	31.80	0.000*	63.60%
- central depot	16.73%	1390.84	21.00	1670.31	31.35	0.000*	74.33%
Hypothesis 5							
- 300 requests, 24 vehicles	8.42%	1673.09	23.55	1826.84	24.00	0.005*	43.58%
- 200 requests, 16 vehicles	11.10%	1181.27	15.90	1328.83	16.00	0.007*	42.90%
- 100 requests, 8 vehicles	14.41%	669.27	8.00	781.92	8.00	0.005*	38.30%
Hypothesis 6							
- providers 125-75-75-125	5.83%	2164.61	29.45	2298.59	30.40	-	-
- providers 150-50-50-150	7.02%	2095.32	29.50	2253.55	28.15	-	-
- providers 175-25-25-175	6.29%	2074.89	28.35	2214.25	25.30	-	-

Table 10: Effect of operational scenarios on benefits of horizontal cooperation.

The results show that the benefits of horizontal cooperation, ranging from 5.61% to 16.73%, strongly depend on the operational setting. More specifically, in the given experimental setting, hypotheses (1), (2), (3), (4) and (5) are essentially confirmed by the results. For example, reducing the width of the time windows in hypothesis (1) increases the total distance traveled, due to the fact that service providers have fewer possibilities to combine customers. However, this impact is less noticeable when service providers take advantage of the operational flexibility created in a horizontal cooperation, which explains why the benefits of cooperating become more important as time windows become tighter. Note that for hypothesis (2), the benefits of horizontal cooperation do no longer increase significantly once a level of 75% long trips is reached. For hypothesis (3), the benefits of cooperation only increase significantly once the clustering becomes weak ($\phi = 0.45$ or lower). Hypothesis (6) is not confirmed, as there is no clear pattern across the scenarios and the differences are very small. Consequently, in the given experimental setting, only the global size of the horizontal cooperation is determinative of the operational benefits, not the size differences between the participating service providers. However, one must be aware of potential interaction effects. For example, hypothesis (6) may be confirmed for a smaller total number of requests in the data set.

As mentioned before, this analysis focuses on the overall savings obtains by the cooperation, regardless of the strategy according to which these savings are distributed among the participating providers. However, by means of example, the *alternative cost avoided method (ACAM)* (Tijs and Driessen, 1986) is applied to the results in the baseline scenario. According to Verdonck et al. (2016), this method is intuitively appealing and transparent because the total cost, i.e. the joint distance traveled, is divided into separable and non-separable costs. The separable cost of each provider corresponds to the increase in joint distance traveled when this provider enters the cooperation. The remaining, non-separable cost is divided among the providers using weights based on the difference between the individual cost and the separable cost of the provider. The resulting cost allocation is displayed in Table 11 and allows to identify the savings that each individual provider obtains by participating in the cooperation. Since all providers have an equal size and hence a comparable contribution, it is not surprising that the savings are approximately evenly divided.

$$y_i = m_i + \frac{c(i) - m_i}{\sum\limits_{j=1}^{n} [c(j) - m_j]} * \left(c(N) - \sum\limits_{j=1}^{n} m_j\right)$$

y_i = cost allocated to provider i (with $i \in \{1, ..., n\}$)
m_i = separable cost of provider $i = c(N) - c(N \backslash i)$
$c(N)$ = cost incurred with cooperation among all providers
$c(i)$ = cost incurred by provider i operating individually

Provider	Without cooperation	With cooperation	Saving
Provider 1	579.14	541.64	37.50 (6.48%)
Provider 2	553.60	521.74	31.86 (5.76%)
Provider 3	565.72	542.65	23.07 (4.08%)
Provider 4	558.50	524.21	34.29 (6.14%)
Total	2256.96	2130.24	126.72 (5.61%)

Table 11: Operating costs per provider without cooperation and with cooperation, using the alternative cost avoided method as an allocation strategy.

Hypotheses (2), (3) and (4) assumed that the benefits of horizontal cooperation may be affected by an increased potential of request exchanges, e.g. because of overlapping service areas or an increased presence of providers in each other's areas. This is confirmed by the percentage of requests exchanges, particularly for hypothesis (4). Although the percentage of requests exchanges is certainly not an accurate indicator of the benefits of horizontal cooperation (e.g. it remains unchanged for hypothesis (1)), this finding confirms the underlying mechanism assumed for hypothesis (2), (3) and (4).

As a final remark, note that the solution approach may slightly underestimate the actual operational savings caused by horizontal cooperation. The average total distance traveled without cooperation, obtained by solving four separate small-sized routing problems, is presumably closer to optimality than the solution with cooperation. From this perspective, the benefits caused by horizontal cooperation may be slightly larger than computed. The optimality gaps cannot be displayed, since the optimal solutions for these large instances are unknown.

5.3 Real-life case study

The artificial data sets used in Sections 5.1 and 5.2 are suitable for comparing algorithmic performances or demonstrating the relative effects of different operational settings. However, they inevitably ignore some real-life complexities that may influence the absolute size of the benefits obtained through horizontal cooperation. First, real-life systems operate on a road network, such that distances and travel times may not be directly proportional to Euclidean distances. Second, case-specific problem characteristics may apply, giving rise to additional types of constraints. Third, both from a cost-related and a quality-related perspective, service providers usually pursue more complicated objectives (measurable or not) than a mere minimization of total distance traveled. The objective of this section is to quantify the benefits of horizontal cooperation in the context of a real-life case study. In addition, solutions with and without cooperation are analyzed to identify specific conditions under which request exchanges among participating providers may be beneficial.

5.3.1 Data set

The data of this real-life case study is provided by a service provider in Belgium, having branches in Wilrijk and Melsbroek. Currently, these branches operate independently, meaning that there is no exchange of information or resources at all. Each branch disposes of one depot that is strategically situated near a large rehabilitation center or hospital, notwithstanding the fact that customers may request rides to any destination of choice. The current practice is that users are assigned to the branch of which the depot is closest to their destination location, regardless of their home location. In Figure 2, the home locations of the customers assigned to the branches in Wilrijk and Melsbroek (which are approximately 35 km apart) are represented by red dots and blue diamonds, respectively. It is visually clear that the service areas of both branches overlap to a certain extent, such that a horizontal cooperation between these branches may be worth investigating.

The data set covers two consecutive operating days, being Monday May 18 and Tuesday May 19, 2015. On the first day, the data consists of 296 private requests served in both branches. Without cooperation, 134 and 162 requests are assigned to Wilrijk and Melsbroek, respectively. On the second day, a total number of 241 (124 + 117) private requests should be satisfied. Note that a single request may involve multiple customers traveling together, such that the total number of individual private requests equals 444 on Monday, which is typically a rather busy day, and 319 on Tuesday. The vast majority of the requests come in pairs. Customers are heterogeneous with respect to their mobility, since approximately 70% of them are wheelchair users. Requests tend to be clustered around peak hours (8-10 am, 3-5 pm) and mainly represent trips to/from a limited set of rehabilitation centers or hospitals around Antwerp and Brussels.

The fleet consists of 69 vehicles, offering a configurable ratio of normal seats and wheelchair spaces. In one extreme configuration, a maximum number of four wheelchair users can be transported at the same time, leaving three normal seats available. The other extreme configuration provides seven normal seats, leaving space for a single wheelchair user. Several intermediate combinations are eligible as well. Routes investigated by the LNS algorithm satisfy the load constraints if at least one feasible configuration exists for each of its arcs separately. Currently, 23 vehicles are assigned to the depot in Wilrijk, whereas 46 are assigned to Melsbroek. At first sight, the fleet size seems more than sufficient to serve all requests. However, apart from the private customer requests mentioned before, the service provider also performs fixed pickup and delivery tours, typically at the request of daycare centers. They pick up customers and deliver them at the daycare center in the morning, and bring them back home in the afternoon. For contractual reasons, private customer re-
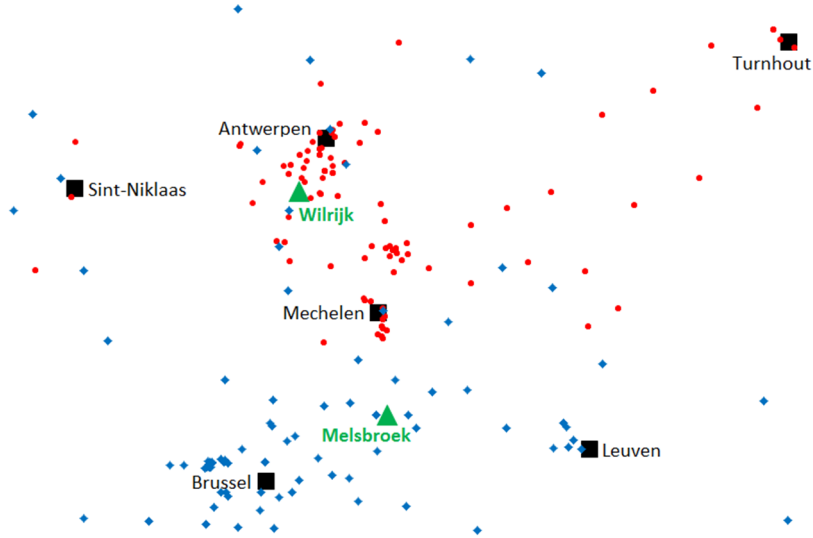
Figure 2: Location of the customer nodes.

quests cannot be served during a fixed pickup and delivery tour. As a result, the vehicles performing such a tour are only available for private customers in the time span between their arrival and their departure at the daycare center, which is often the off-peak period between the morning peak and the afternoon peak. Therefore, the optimization problem addressed in this paper, i.e. related to private requests, is quite tightly constrained during peak hours and rather easy during the off-peak. More specifically, a total number of 41 vehicles (12 in Wilrijk, 29 in Melsbroek) is not available during peak hours. Finally, a vehicle is not necessarily physically stationed at the depot to which it is assigned. Drivers living relatively far from the depot are asked to take their vehicles home during the night if this allows a cost-efficient service of customers in that region during the early morning or late evening.

5.3.2 Problem characteristics

In consultation with the service provider, the impact of three different objective functions is analyzed:

1. The first objective function corresponds to the typical objective used for artificial data, which is a minimization of the total traveling cost. This objective assumes that the costs incurred by a service provider are exclusively related to the distance traveled by its vehicles, which is a substantial simplification of the service provider's real cost structure. Based on the service provider's information, a variable vehicle cost of €0.30 is incurred per kilometer traveled. Only the costs for serving private customers are taken into account, since the fixed pickup and delivery tours are invariable.

2. The second objective function is based on a more comprehensive computation of costs, considering both the distance traveled and the wage payed to the driver. According to the service provider, the hourly driver wage amounts to €25 including taxes. A driver's wage cost is computed based on the time spent in the vehicle. This includes potential waiting time faced at any location, be it between two private requests, after a fixed pickup tour or before a fixed delivery tour. To avoid excessive driver waiting time during the off-peak, a shift can be split into two short shifts at most, provided that the intermediate break (staring and ending at the vehicle's depot) lasts for at least 3 hours. As the driver may freely dispose of his time during such a break, the service provider does not incur wage costs meanwhile. This operational flexibility is useful to cope with demand differences during peak and off-peak hours.

3. In order to support tactical decisions, fleet size may be included in the optimization process. Although the available number of vehicles is fixed on the short term, the service provider is particularly interested in knowing the minimum fleet size required to meet all customer demand. Given certain fixed costs for acquiring, insuring and maintaining vehicles, this allows him to find a balance between a medium-term reduction in fixed vehicle costs and the associated increase in operating costs due to a reduced flexibility in serving customers. More specifically, the third objective function consists of a hierarchical minimization of (1) the number of active vehicles and (2) the operating costs as defined in the second objective function. The first part only relates to those vehicles not performing a fixed pickup and delivery tour. If a vehicle is activated for a fixed pickup and delivery tour anyway, it would not make sense not to use it for private requests during the intermediate time span.

Furthermore, the heterogeneity of customers and the configurability of vehicles were already discussed in Section 5.3.1. All other problem characteristics largely correspond to the standard problem (Cordeau and Laporte, 2003). A time window is imposed for every request, though always on the moment of pickup. The maximum user ride time is expressed as 1.5 times the direct ride time between the customer's pickup and delivery location. A fixed service duration of five minutes is taken into account for wheelchair users only. The maximum route duration is limited to 10 hours, even for split shifts. An exception to this rule is allowed to ensure that the same driver performs corresponding fixed pickup and delivery tours.

5.3.3 Results

Solving the real-life case requires the following conceptual adjustments. First, the *Graphhopper* API is invoked to geocode all nodes and precompute distances and travel times on the road network, using *OpenStreetMap* data. Based on realistic estimates, the resulting

travel times are multiplied by a factor 1.25 to provide sufficient margin in the schedules. Second, the scheduling procedure of Cordeau and Laporte (2003) is replaced by the one discussed in Molenbruch et al. (2017c). The latter scheduling procedure delivers schedules in which total user ride time is minimized, which is fully in line with the service policy of the provider. Last, additional depots disposing of a single vehicle are added to represent the addresses of drivers who may take their vehicles home. An additional constraint prevents the total number of vehicles from exceeding the actual fleet size. Without cooperation, the sum of all vehicles stationed at the depot in Wilrijk (resp. Melsbroek) or at the home address of a driver related to this depot cannot exceed 23 (resp. 46). With cooperation, the total fleet size is limited to 69 vehicles. No constraint is imposed on the distribution of these vehicles among both depots.

Table 12 shows the average results with and without horizontal cooperation for both operating days, optimizing the different objectives. A total number of 20 runs was performed, each consisting of 20000 iterations. The first column indicates the operating day and the applicable scenario (with/without cooperation). The second column shows the total cost for a given scenario, which consists of the variable vehicle cost (€0.30 per km traveled) and the wage cost (€25 per hour, if applicable) displayed in the next two columns. The fifth column gives the total required fleet size, summing the active vehicles in both depots as mentioned in the next two columns. The eighth and ninth column indicate the number of requests served by both branches. The last two columns indicate the number of requests exchanged from Wilrijk to Melsbroek and from Melsbroek to Wilrijk, respectively.

Optimizing the first objective, the benefits of horizontal cooperation are similar to the baseline scenario for the artificial data in Section 5.2. A 5.64% (May 18) and 3.88% (May 19) average reduction in the variable vehicle cost is found. This is a plausible outcome, given the rather limited degree of overlap between both service areas. In the scenario with cooperation, only very small net transfers of vehicles[3] and customers are suggested. However, the absolute number of request exchanges is very different on both operating days, even though the degree of overlap between the service areas is similar. This reconfirms the finding from Section 5.2.3 that the number of requests exchanged is not a reliable indicator of the benefits of a horizontal cooperation.

As mentioned before, a mere minimization of the total traveling cost does not comprise the cost-related approach envisaged by the service provider. It is therefore important to investigate how

---

[3]On May 19, the available fleet of 23 vehicles is insufficient to find a feasible solution for the branch of Wilrijk in the scenario without cooperation, such that an additional vehicle was added during the solution procedure. In reality, the service provider would allow slight violations of time-related constraints to find a feasible solution requiring only 23 vehicles.

the benefits of horizontal cooperation change when a more realistic objective is introduced. Results for the second objective show that, due to high labor costs in Belgium, the wage payed to drivers considerably outweighs the variable vehicle cost. The relative importance of the latter becomes so small that the efficiency of the vehicle routing (i.e. combining customers as efficiently as possible) is partly sacrificed whenever another routing plan shortens the overall work duration of the drivers. Horizontal cooperation turns out to contribute relatively little in shortening the drivers' shifts. The average reduction in total cost is limited to 3.17% (May 18) and 1.69% (May 19) when the second objective is investigated. There is a net transfer of customers towards Melsbroek, particularly on May 18. This may be explained by the computation method of the wage cost. In Melsbroek, many drivers perform a fixed pickup and delivery tour at the beginning/end of their shift. Unproductive (but costly) waiting time after the end of the fixed pickup tour or before the start of the fixed delivery tour may be filled with requests transferred from the branch in Wilrijk. This is particularly beneficial for requests which would otherwise be a driver's first or last job of the day, thanks to the direct effect on the corresponding route duration and thus on the wage cost.

Optimizing the third objective, solutions involving an average of 64.90 (May 18) and 68.00 (May 19) instead of 69 vehicles can be obtained in the scenario without cooperation. Horizontal cooperating further reduces the average fleet size to 63.75 (May 18) and 66.40 (May 19). This reduction mainly concerns the depot in Melsbroek, in combination with a net transfer of customers from Melsbroek to Wilrijk. The corresponding increase in average total cost relative to objective 2 is only 1.63% (May 18) and 1.30% (May 19), which is a particularly interesting finding from the service provider's point of view. Of course, it should be stressed that medium-term decisions on the fleet size cannot be based on results originating from two operating days. Ideally, this decision is based on data over a longer period of time, such that the service provider can determine his requirement of vehicles in normal circumstances. Unfortunately, since the data collection requires lots of manual interventions by the provider, it was not possible to provide additional data.

Scenario		Total	Veh	Wage	F_t	F_W	F_M	W	M	W→M	M→W
Objective 1: minimization of traveling cost											
18.05	coop.	2136.47	2136.47	-	68.95	24.70	44.25	126.05	169.95	60.10	52.15
	no coop.	2264.21	2264.21	-	68.95	22.95	46.00	134.00	162.00		
19.05	coop.	2114.19	2114.19	-	69.00	23.60	45.40	118.65	122.35	26.75	21.40
	no coop.	2199.63	2199.63	-	69.85	24.00	45.85	124.00	117.00		
Objective 2: minimization of distance and wage cost											
18.05	coop.	8556.61	2512.85	6043.76	69.00	22.15	46.85	105.45	190.55	76.60	48.05
	no coop.	8836.29	2532.01	6304.28	69.00	23.00	46.00	134.00	162.00		
19.05	coop.	8391.43	2521.63	5869.80	69.00	25.05	43.95	121.95	119.05	26.90	24.85
	no coop.	8535.94	2509.08	6026.86	70.00	24.00	46.00	124.00	117.00		
Objective 3: hierarchical minimization of (1) fleet size and (2) distance and wage costs											
18.05	coop.	8695.80	2460.48	6235.33	63.75	25.95	37.80	163.60	132.40	54.50	84.10
	no coop.	8924.59	2473.68	6450.91	64.90	21.00	43.90	134.00	162.00		
19.05	coop.	8500.42	2497.40	6003.02	66.40	25.20	41.20	137.65	103.35	19.95	33.60
	no coop.	8575.29	2491.58	6083.71	68.00	24.00	44.00	124.00	117.00		

Table 12: Effect of horizontal cooperation on real-life case for different objectives.

From a quality-related perspective, it is important to verify whether the horizontal cooperation causes any increase in the average ride time of customers, since a larger number of customers may be transported jointly. Even though a minimal service level is guaranteed through an upper bound on each customer's ride time, the average ride time remains a relevant quality measure. Analyzing the minimal-cost solution on each operating day, horizontal cooperation is found to increase the average customer ride time from 21.30 to 21.58 minutes (May 18) and from 24.55 to 24.89 minutes (May 19), which is negligible. This implies that the greatest merit of horizontal cooperation lies in a reduction of empty rides. For example, on May 18, empty distance traveled reduces from 4179.38 to 3886.29 km (-7.01%). Nevertheless, empty rides still account for almost half of the total distance traveled, since drawing efficient vehicle routes is complicated due to the tight quality constraints that characterize DARPs.

5.3.4 Reasons for request exchanges

The present case study deals with different branches belonging to the same service provider, which facilitates the establishment of a horizontal cooperation. When two separate service providers decide to cooperate, they may not be willing to share all information on their customer requests, both for privacy reasons and from a strategic perspective. Therefore, it would be interesting to know which requests are most valuable in a horizontal cooperation, i.e. which requests will be exchanged among providers. However, it is unclear whether this assessment can be made in advance and even if so, whether similar operational benefits can be obtained by sharing only those requests. The remainder of this section aims to identify reasons why requests are exchanged, analyzing the detailed route structures in the solutions obtained before.

Figure 3 is similar to Figure 2, but only shows the customers being exchanged in the cost-minimal solution (objective 2). Hence, the red dots (resp. blue diamonds) are the home locations of customers who were previously assigned to the branch in Wilrijk (resp. Melsbroek), but now served by a vehicle origination from the depot in Melsbroek (resp. Wilrijk). Three particular groups of customers are highlighted, as they are examples of the three main reasons for exchanges that are identified. They are ranked in descending order of predictability:

1. A first group of exchanges is explained directly by the geographical location of the customers' home addresses. For example, consider customers A1 and A2, who are living close to Antwerp and both request a trip to the rehabilitation center in Melsbroek early in the morning. Thanks to their coinciding time windows, it is immediately obvious why these customers will be combined in a single ride. However, in this case, it is more
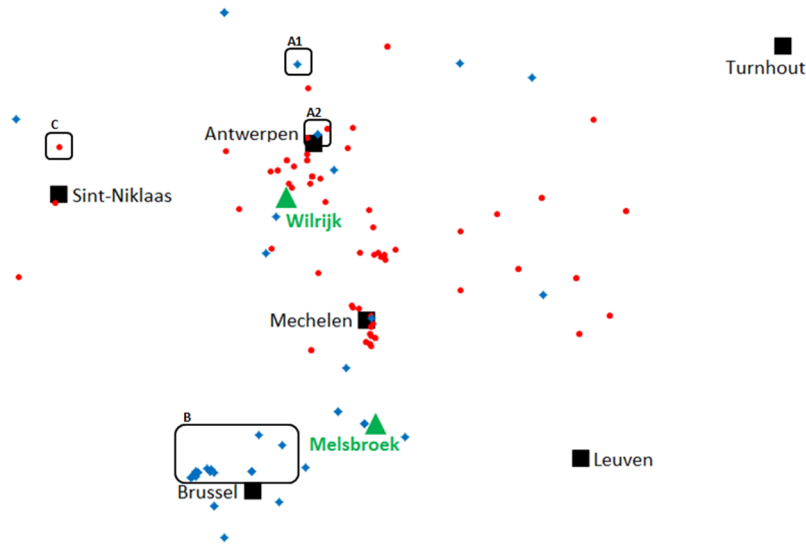
Figure 3: Location of the exchanged customer nodes.

efficient to deviate from the current assignment rules and serve these customers using a vehicle that originates from the depot in Wilrijk, since this vehicle covers less empty distance to reach the customers' home addresses. This type of exchange is easily predictable in advance. Unfortunately, relatively few requests belong to this category due to the limited overlap between both service areas. Nevertheless, in a setting where service providers prefer to have their own set of customers, one might think of an agreement to preassign them based on their geographical characteristics. Note that this corresponds to the current practice of the real-life service provider considered in this study, who - in the scenario without cooperation - assigns customers to the branch that is located closest to their outbound location. In this respect, it is interesting to investigate whether it would be more efficient to systematically assign customers based on (1) their *home* location or (2) the *geographical center* between their home and outbound location. Table 13 compares the three different assignment strategies for the second objective function. Similar results are obtained for the other objectives. On both operating days, the assignment based on the geographical center is strongly outperformed, but it is unclear which of the two other strategies is preferable. None of them matches the results of a full horizontal cooperation, which indicates that joint route planning is a prerequisite for obtaining substantial operational benefits.

2. A second group of exchanges can be explained indirectly, based on the fact that customer requests come in pairs. For example, consider the group of customers with label B in Figure 3. Based on their home location near Brussels, it might seem strange that

38

Scenario		Total	Veh	Wage
18.05	cooperation	8556.61	2512.85	6043.76
	assign by destination	8836.29	2532.01	6304.28
	assign by home	8764.87	2505.98	6258.89
	assign by average	8933.25	2586.28	6346.97
19.05	cooperation	8391.43	2521.63	5869.80
	assign by destination	8535.94	2509.08	6026.86
	assign by home	8732.30	2563.97	6168.34
	assign by average	8862.98	2549.22	6313.77

Table 13: Effect of different assignment strategies (cost minimization)

in the scenario with cooperation, these customers are picked up by a vehicle from the branch in Wilrijk. This is explained by the fact that vehicles from the depot in Wilrijk deliver some customers at the rehabilitation center in Melsbroek (such as A1 and A2 in the previous example). While these customers undergo their treatment, it would not be efficient for these vehicles to idle at the rehabilitation center, nor to return to their own service area being unloaded. Therefore, they serve one or several local requests in the area of Melsbroek, i.e. the ones with label B. However, since multiple local request(s) might be eligible, this type of exchanges is difficult to predict, despite being easy to explain in retrospect.

3. A last group of exchanges are impossible to predict, since they are only convenient relative to the actual structure of the routes. For example, consider customer C, living north of Sint-Niklaas (which is close to Wilrijk and Antwerp, but far from Melsbroek). This customer is picked up at a hospital in Antwerp by an empty vehicle originating from the depot in Melsbroek. The reason for this unexpected finding is that after the delivery of this customer, the vehicle picks up another customer living south of Sint-Niklaas and requesting a ride to the rehabilitation center in Melsbroek. Obviously, in this case, the exchange of customer C cannot be explained taking an isolated view on his request. It can only be interpreted in a broader context, optimizing the entire set of routes in the solution. Therefore, it seems an impossible task for service providers to predict this type of exchanges in advance.

6   Conclusions and future research

Although horizontal cooperation through joint route planning has been a cost-effective practice among logistic service providers for many years, this strategy remains completely unexplored in the domain of demand-responsive passenger transportation. This paper analyzes the potential of horizontal cooperation among dial-a-ride providers, as well as the operational characteristics that influence the benefits. In addition, a real-life case study discovers the reasons

why providers benefit from exchanging particular requests among each other. The extent to which these exchanges are predictable determines how much information must be disclosed in order to make the cooperation succeed. From a technical point of view, the analysis of horizontal cooperation is approached as a multi-depot variant of the dial-a-ride problem, in which each service provider exploits one or more depots. Joint route planning implies that requests may be served by vehicles originating from any depot. Without cooperation, requests are preassigned to a specific provider based on the customer's geographical location. A large neighborhood search metaheuristic with additional periodic diversification is applied to solve this problem. Its excellent performance on single-depot and multi-depot instances is illustrated through tests on existing artificial data.

To analyze the impact of horizontal cooperation, a larger artificial data set with more realistic characteristics is introduced. The comparison of scenarios with and without horizontal cooperation reveals a total distance gain between 5.61% and 16.73%, which is more modest than reported in freight transportation. The operational setting is shown to have a substantial impact on the magnitude of the benefits. Statistical analyses confirm that the potential of horizontal cooperation increases for problem variants with smaller time windows, more long-distance trips, weaker clustering of customers around their closest depot, more overlap between service areas or fewer customers. Further research may focus on analyzing potential interactions between these effects. In addition, the influence of other operational characteristics on the impact of horizontal cooperation may be identified, such as heterogeneity of customers or combination constraints.

The concept of horizontal cooperation is also applied to a real-life case study, involving a service provider considering a cooperation between two of his branches. Several conclusions can be drawn that go beyond the specific characteristics of this case study. For example, the benefits of horizontal cooperation are mainly explained by a reduction in empty distance, without affecting the service level offered to customers. Unfortunately, the variable costs related to the distance traveled only account for a small part of the total cost incurred by the provider. Taking driver wages into account, rather than a mere minimization of the total distance traveled, the impact of horizontal cooperation becomes smaller. In addition, real-life cooperations between separate service providers may be hampered by a reluctance to share information for several reasons. The solutions with and without horizontal cooperation are analyzed in order to discover the underlying reasons why requests are exchanged among the participating parties. Few of the exchanges turn out to be easily predictable in advance, which implies that the potential of a horizontal cooperation can only be fully exploited if providers share complete information on their requests. However, the observed exchange patterns may provide useful information to develop new local search

operators which are tailored to the context of horizontal cooperation.

In order to tailor the proposed solution approach to specific real-life systems that differ from the one considered in the real-life case study, the LNS metaheuristic may need to be extended with additional problem characteristics. For example, it may be interesting to analyze the impact of a dynamic problem setting, where requests may be submitted or canceled during the execution of the service, on the benefits of horizontal cooperation. It is likely that cooperation will create more flexibility for service providers to respond to additional requests, as these requests may be served by any vehicle. The same reasoning applies to a request that should be rescheduled to another vehicle in real time, e.g. when the intended vehicle cannot reach the customer in time as a result of a breakdown or a traffic jam. In other words, the benefits of cooperation may be larger in a dynamic setting. This assumption is to be confirmed by future research.

Note that this paper mainly focuses on the operational effects of joint route planning. In reality, other factors may impede the implementation of such a centralized decision making strategy (Cruijssen et al., 2007c). The initialization phase often requires considerable investments to align information systems and policies. For example, the proposed form of cooperation has little chance of success if the participating providers offer a different quality of service. Moreover, trust and commitment are required among all participating parties. In this respect, joint route planning would be facilitated by the creation of an independent body that collects all user requests and divides them in a globally optimal manner among the participating service providers.

Finally, note that the analysis in this paper is executed in terms of the joint benefits obtained by the participating service providers. This disregards the fact that all parties have to agree on a strategy to divide these benefits, which is necessary to guarantee the stability of the cooperation in the long term. Considerable research has already been conducted into profit sharing techniques in logistics, which may also be applied to demand-responsive passenger transportation. For example, Verdonck et al. (2016) compare three common sharing techniques (Shapley value, alternative cost avoided method, equal profit method) and find that the participating providers may prefer different techniques, e.g. depending on the size of their contribution. Moreover, the first two techniques do not guarantee to deliver a stable solution in which no subcoalition would benefit from leaving the grand coalition, whereas the latter technique may only be acceptable in the early phases of a growing cooperation. In summary, it is not straightforward for the participating providers to agree on the choice of a particular profit sharing technique.

41

References

Anderson, D., Sweeney, D., Williams, T., Freeman, J., Shoesmith, E., 2010. Statistics for Business and Economics. CENGAGE Learning Business Press.

Berbeglia, G., Cordeau, J.-F., Laporte, G., 2010. Dynamic pickup and delivery problems. European Journal of Operational Research 202 (1), 8–15.

Braekers, K., Caris, A., Janssens, G. K., 2014. Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots. Transportation Research Part B: Methodological 67, 166–186.

Chassaing, M., Duhamel, C., Lacomme, P., 2016. An ELS-based approach with dynamic probabilities management in local search for the dial-a-ride problem. Engineering Applications of Artificial Intelligence 48, 119–133.

Cordeau, J.-F., 2006. A branch-and-cut algorithm for the dial-a-ride problem. Operations Research 54 (3), 573–586.

Cordeau, J.-F., Gendreau, M., Laporte, G., 1997. A tabu search heuristic for periodic and multi-depot vehicle routing problems. Networks 30 (2), 105–119.

Cordeau, J.-F., Laporte, G., 2003. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. Transportation Research Part B: Methodological 37 (6), 579–594.

Cruijssen, F., Bräysy, O., Dullaert, W., Fleuren, H., Salomon, M., 2007a. Joint route planning under varying market conditions. International Journal of Physical Distribution & Logistics Management 37 (4), 287–304.

Cruijssen, F., Cools, M., Dullaert, W., 2007b. Horizontal cooperation in logistics: opportunities and impediments. Transportation Research Part E: Logistics and Transportation Review 43 (2), 129–142.

Cruijssen, F., Cools, M., Dullaert, W., 2007c. Horizontal cooperation in logistics: opportunities and impediments. Transportation Research Part E: Logistics and Transportation Review 43 (2), 129–142.

Dial, R. B., 2002. Autonomous dial-a-ride transit: Software functionality and architecture overview. In: Applications of Advanced Technologies in Transportation (2002). pp. 297–304.

Dueck, G., 1993. New optimization heuristics: The great deluge algorithm and the record-to-record travel. Journal of Computational physics 104 (1), 86–92.

Gschwind, T., Irnich, S., 2014. Effective handling of dynamic time windows and its application to solving the dial-a-ride problem. Transportation Science 49 (2), 335–354.

Krajewska, M. A., Kopfer, H., Laporte, G., Ropke, S., Zaccour, G., 2008. Horizontal cooperation among freight carriers: request allocation and profit sharing. Journal of the Operational Research Society 59 (11), 1483–1491.

Lehuédé, F., Masson, R., Parragh, S. N., Péton, O., Tricoire, F., 2013. A multi-criteria large neighbourhood search for the transportation of disabled people. Journal of the Operational Research Society 65 (7), 983–1000.

Leyman, P., De Causmaecker, P., 2017. Termination criteria for metaheuristics: Is computation time worth the time? In: Booklet of abstracts of the 31st annual conference of the Belgian Operational Research Society, Brussels. pp. 89–90.

López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., Stützle, T., 2016. The irace package: Iterated racing for automatic algorithm configuration. Operations Research Perspectives 3, 43–58.

Masmoudi, M. A., Braekers, K., Masmoudi, M., Dammak, A., 2017. A hybrid genetic algorithm for the heterogeneous dial-a-ride problem. Computers & Operations Research 81, 1–13.

Masmoudi, M. A., Hosny, M., Braekers, K., Dammak, A., 2016. Three effective metaheuristics to solve the multi-depot multi-trip heterogeneous dial-a-ride problem. Transportation Research Part E: Logistics and Transportation Review 96, 60–80.

Masson, R., Lehuédé, F., Péton, O., 2014. The dial-a-ride problem with transfers. Computers & Operations Research 41, 12–23.

Molenbruch, Y., Braekers, K., Caris, A., 2017a. Operational effects of service level variations for the dial-a-ride problem. Central European Journal of Operations Research 25 (1), 71–90.

Molenbruch, Y., Braekers, K., Caris, A., 2017b. Typology and literature review of dial-a-ride problems. Annals of Operations Research (forthcoming).

Molenbruch, Y., Braekers, K., Caris, A., Vanden Berghe, G., 2017c. Multi-directional local search for a bi-objective dial-a-ride problem in patient transportation. Computers & Operations Research 77, 58–71.

Paquette, J., Cordeau, J.-F., Laporte, G., 2009. Quality of service in dial-a-ride operations. Computers & Industrial Engineering 56 (4), 1721–1734.

Parragh, S. N., 2011. Introducing heterogeneous users and vehicles into models and algorithms for the dial-a-ride problem. Transportation Research Part C: Emerging Technologies 19 (5), 912–930.

Parragh, S. N., Doerner, K. F., Hartl, R. F., 2008. A survey on pickup and delivery problems. Part II: Transportation between pickup and delivery locations. Journal für Betriebswirtschaft 58 (2), 81–117.

Parragh, S. N., Doerner, K. F., Hartl, R. F., 2010. Variable neighborhood search for the dial-a-ride problem. Computers & Operations Research 37 (6), 1129–1138.

Parragh, S. N., Schmid, V., 2013. Hybrid column generation and large neighborhood search for the dial-a-ride problem. Computers & Operations Research 40 (1), 490–497.

Pisinger, D., Røpke, S., 2010. Large neighborhood search. In: Handbook of metaheuristics. Springer, pp. 399–419.

Qu, Y., Bard, J. F., 2013. The heterogeneous pickup and delivery problem with configurable vehicle capacity. Transportation Research Part C: Emerging Technologies 32, 1–20.

Røpke, S., Cordeau, J.-F., Laporte, G., 2007. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. Networks 49 (4), 258–272.

Røpke, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science 40 (4), 455–472.

Taillard, É., 1993. Parallel iterative search methods for vehicle routing problems. Networks 23 (8), 661–673.

Tijs, S.H., Driessen, T.S.H., 1986. Game theory and cost allocation problems. Management Science 32 (8), 1015–1028.

Verdonck, L., Beullens, P., Caris, A., Ramaekers, K., Janssens, G. K., 2016. Analysis of collaborative savings and cost allocation techniques for the cooperative carrier facility location problem. Journal of the Operational Research Society 67 (6), 853–871.

Verdonck, L., Caris, A., Ramaekers, K., Janssens, G. K., 2013. Collaborative logistics from the perspective of road transportation companies. Transport Reviews 33 (6), 700–719.

Xiang, Z., Chu, C., Chen, H., 2008. The study of a dynamic dial-a-ride problem under time-dependent and stochastic environments. European Journal of Operational Research 185 (2), 534–551.