A simulation study of commuting alternatives for day care centres

Peer-reviewed author version

CICH, Glenn; BEN ARROYO HARTMAN, Irith; KNAPEN, Luk & JANSSENS, Davy (2020) A simulation study of commuting alternatives for day care centres. In: Future Generation Computer Systems-The International Journal of eScience, 110 , p. 323 -337.

DOI: 10.1016/j.future.2018.05.009
Handle: http://hdl.handle.net/1942/31301

A Simulation Study of Commuting Alternatives for Day Care Centres

Glenn Cich[a,*], Irith Ben-Arroyo Hartman[b], Luk Knapen[a], Davy Janssens[a]

[a] *UHasselt - Hasselt University, Transportation Research Institute (IMOB), Agoralaan, 3590 Diepenbeek, Belgium*
[b] *Caesarea Rothschild Institute, University of Haifa, Haifa, 31905, Israel*

Abstract

In Flanders (Belgium), Mobility impaired people need to travel frequently from their homes to a *Day Care Centre (DCC)*. Currently this is done by *subsidised bus services* but recently a decision was made to cancel these subsidies. The fare the DCC guests will have to pay for transport by bus is too high for most of them.

This paper investigates a solution where voluntary drivers bring as many DCC guests as possible to the DCC by carpooling. These drivers can pick up and drop off DCC guests along the way to their work location or any other destination.

In general it turns out to be impossible to drive all the DCC guests to the DCC by carpooling. The remaining DCC guests will be picked up by dedicated buses. The goal is to keep the bus travel cost as low as possible. The solution is constrained by car capacities, time windows for both drivers and DCC guests, upper bounds for detours and the availability of intermediate transfer locations. The main challenge is the involvement of multiple transportation service providers. Some of these are not under the control of the consultant in charge of finding an efficient solution for the DCC and hence, their operation and cost cannot be included in the objective function. Solving the problem

*Corresponding author
Email addresses: `glenn.cich@uhasselt.be` (Glenn Cich), `irith.hartman@gmail.com` (Irith Ben-Arroyo Hartman), `luk.knapen@uhasselt.be` (Luk Knapen), `davy.janssens@uhasselt.be` (Davy Janssens)

Preprint submitted to Future Generation Computer Systems *18th June 2020*

requires consideration of several cases each leading to a heavy combinatorial computation.

Although it seems to be impossible to find a carpool solution in which all the passengers reach the DCC, the results are promising. In several cases four or more chartered buses can be saved on. However, average results show a saving around one to two chartered buses which represents a cost reduction between 20 % and 30 %.
Keywords: mobility impaired people, optimisation, carpooling, commuting, cost savings, algorithms

---

1. Introduction

Nowadays, carpooling and shared mobility are receiving increasing attention in the literature because of the high potential to reduce vehicle expenses and congestion on the roads. However, these alternatives are mainly aimed at healthy people. Alternatives for mobility impaired people and children going to school are scarce. For carpooling to school in Flanders, a few examples can be found such as the Schoolpool introduced by Taxistop [1]. For mobility impaired people, carpooling is less obvious. Alternatives such as using public transport are often difficult and sometimes even impossible. In many cases the only option for them is (expensive) demand responsive transport.

In this paper, a general carpool-like problem is tackled. This problem consists of people (*facility visitors*) who need to go to a common location (*facility*) where they perform their (daytime) activities and need to be brought by other people (*drivers*). Examples of facility visitors are (i) children going to school, (ii) mentally impaired people going to a day care centre or (iii) children going to the same sports club. They can be transported by their parents, neighbours, public transport or taxi-like approaches. Note that those problems are similar, but also slightly different. Commuting by private car is in most of the cases convenient, but results in unnecessary many vehicles on the road and inefficient use of vehicles. Public transport is sustainable, but not always possible for the target population. Taxi-like approaches are also possible, but expensive.

An obvious solution to this problem is making more efficient use of private cars. Drivers can pick up other facility visitors which are commuting to the same destination. However, this is not easy due to facility visitors' and drivers' constraints. This paper presents, given a set of facility visitors, a set of drivers and a set of constraints, a solution to commute from their homes to a common destination. Note that facility visitors not present in a solution, should still rely on other transport possibilities. Furthermore, some locations can be used as transfer locations, where facility visitors can transfer from one driver to another.

The remaining part of the paper is organised as follows: Section 2 gives an overview of related work. Next, in Section 3, the case study which is investigated in this paper is explained. In Section 4 the concepts that are used in the paper are described, followed by the details about the data collection in Section 5. In Section 6, limitations and assumptions that are used for the algorithms are described. Section 7 defines the discussed problem as a graph theoretical problem. The used algorithms are explained in Section 8. Results can be found in Section 9. Section 10 discusses and compares the results of the proposed methods. Finally, Section 11 concludes the paper.

2. Related Work

The problem that is tackled in this paper is related to *carpooling* which is a special case of *ride-sharing* and is even more related to the *school bus routing problem*.

Authors extensively investigated reasons why people do or do not carpool/share rides. Li et al. [2] conducted a survey in Houston and Dallas to investigate why people do or do not carpool. A main reason why they carpool, which is of interest in our research, is sharing vehicle expenses. Important reasons why they do not carpool are location and schedule limitations and flexibility. Results of our research will show similar findings.

Buliung et al. [3] investigate the factors for a successful carpool with data of a web service Carpool Zone, provided by the Smart Commute in the Greater Toronto and Hamilton area. They found that the main factors where spatial accessibility to carpool matches, car ownership and socio demographics. Factors such as carpool infrastructure and personal attitudes were less important.

The Flemish Government and Traject [4] investigated a variety of measures to promote sustainable commuting trips. They investigated objective and subjective obstacles for the employee as well as for the employer. For employers, flexible working times are one of the main obstacles for not carpooling. Various activities after work and the lack of knowing other employees were other reasons why people do not carpool. The subjective obstacles were in line with the objective ones, namely flexibility and efficiency. Carpooling is also considered as an incursion on one's privacy. People consider their car and the trip from home to work or work to home as a moment to relax. Reasons to carpool are related to financial, social and environmental benefits.

The Belgian Federal Government [5] investigated the effectiveness of different measures for different types of transport for commuting purposes. For carpooling they only investigated objective measures such as infrastructure, cooperation with the government and other companies, information about carpooling etc. Measures which seem to increase carpooling were information sessions about carpooling, a guaranteed ride back home, organising carpools, access to a database and parking spots especially for carpools.

Another research in Belgium by Vanoutrive et al. [6] investigated three characteristics: location, organisational factors and promotion. They found a higher number of carpooling in less accessible regions. Activity sectors such as construction, manufacturing and transport were popular carpool sectors as well. Based on the literature, it seems to be clear that a decreased flexibility always returns as a main reason for not using carpooling. This will be investigated later on in this paper.

Note however that many of the subjective factors mentioned in the research covering *carpooling for commuters* may not hold in the case of *carpooling for facility visitors* because in many cases the drivers are their parents.

As mentioned before, ride-sharing shares characteristics with the facility visitors problem in this paper. Agatz et al. [7] give an overview of ride-sharing. The objectives of ride-sharing include (i) minimising system-wide vehicle-miles, (ii) minimising system-wide travel time and (iii) maximising the number of participants. These objectives are also applicable to the problem addressed in this paper. Ride-sharing takes into account constraints as well. The most important one is the time window. In most of the ride-sharing applications, participants specify an earliest possible departure time and a latest possible arrival time. In other applications it is also possible to specify a maximum excess travel time. Other constraints relate to personal preferences. In the presented paper, even more constraints will be taken into account which are explained in Section 4. In [7], a distinction is made between the *static* and the *dynamic* ride-share applications. Static means that the set of drivers and riders are initially known by the algorithm, while in dynamic applications these sets can change and hence, the algorithm needs to work in real time. In this paper, the focus will be on the static method. If every driver picks up at most one passenger, a *maximum-weight bipartite matching algorithm* can be used, however the problem which is addressed in this paper is more similar to the single driver, multiple rider arrangements and these problems are harder to solve. Our problem can even be classified as a single rider, multiple driver arrangement because facility visitors are allowed to transfer between drivers.

Park and Kim [8] give an extensive overview of the school bus routing problem. Clearly there are several similarities with the problem addressed in this paper. In the school bus routing problem, there are mainly four types of data: (i) students, (ii) schools, (iii) vehicles and (iv) distance matrix. The students can be compared with the facility visitors in our problem, the schools with the facilities, the vehicles with the drivers and the distance matrix is calculated in our problem as well. The different steps to solve a school bus routing problem can be read in [8]. There is a difference between the *single school problem* where students have to be dropped off at one school, whereas *multiple schools* indicates that students can be dropped off at different schools. Clearly, our problem matches the single school problem as well as the subdivision into the *morning* and *afternoon* problems. The school bus problem shares a number of characteristics and constraints with the facility visitors problem, such as *vehicle capacity*, *maximum riding time*, *school time window* (= facility opening times) and earliest pick-up time for facility visitors.

Bögl et al. [9] solve a school bus routing problem (SBRT) that allows for transfers (because pupils for multiple schools are served by a single set of buses). The proposed solution solves the (i) bus stop selection, (ii) pupil to stop assignment, (iii) bus routing and (iv) bus scheduling sub problems in an integrated solver. Sub problems are handled in an iteration of hierarchically organised algorithms that allows for feedback from the bus *routing* and *scheduling* stages to the bus *stop selection* and *pupil assignment*.

Research on multi-modality for people transport focuses on route advisers for individual trips (also referred to as *Traveller Information Services*). In most cases the advice combines car-sharing and public transportation trips. The *SocialCar* EU-project [10] and the *RTA Trip Planner* (Chicago Regional Transport Authority) ([11]) are typical examples. The report by Van Audenhove et al. [12] is a typical report that recognises the need for multi-modality and the opportunities created by the availability of Intelligent Transportation Systems (ITS) and smartphones. It states: '*Urban mobility is one of the toughest system-level challenges facing actors of the mobility ecosystems. In the future, innovative mobility services will be driven less by improvements in single transport modes than by integration. What is needed is system-level collaboration between all stakeholders of the mobility ecosystem to come up with innovative and integrated business models.*' It scores the travel infrastructure for a large list of cities worldwide and investigates future business models. Several similar reports can be found.

However, research that addresses the *operational problem of multi-modal recurrent trips* seems to be absent.

Solutions for *school bus* problems on the one hand and *carpooling* problems on the other hand have extensively been discussed in literature. This paper investigates the feasibility of a multi-modal solution which is required for budgetary and/or sustainability reasons but seems to be more complicated than both unimodal solutions.

Last but not least the financial situation addressed in this paper is of all times. In older research, Oxley et al. [13] attempt to measure additional transport cost for people with a disability. They used Office of Population Censuses and Surveys (OPCS) of Disability, Family Expenditures Survey and Department of Social Security Research Report to investigate this matter. They concluded that the analysis of the extra expenditures of people with a disability was complex and did not show clear evidence to prove that disability introduces extra expenditures for transport. However, they showed a lower income level of impaired people. They showed that when income rises, there is an increased expenditure on transport as well. The expenditures rise substantially faster than the income rate. For impaired people this rate is even faster than for able-bodied people.

Research carried out a few years later by Roberts and Lawton [14] investigated the need for financial assistance for transport costs for families with disabled children. This research in the United Kingdom stated that the Government is aware of the importance of transport for disabled people. However, only one third of the families with disabled people received financial support for transport related costs of the Family Fund Trust grants. This research suggests that the support is insufficient for many families.

Although abundant literature is found on the *carpooling* and *school bus routing* sub-problems, none could be found that covers the combined problem discussed in this paper.

3. Case Study: Day Care Centre Visitors Commuting

3.1. Problem Definition

As a case study, the problem is investigated where a set of people, suffering from mobility and/or intellectual impairment and living in different locations have to be mobilised every morning from their homes to a given *day care centre (DCC)*, and back to their homes in the afternoon. The dedicated buses that used to drive them will no longer be subsidised by the government. The fares the DCC guests need to pay without subsidies is too high for most of them. Neven et al. [15] mentioned high costs for demand responsive transport for mobility impaired people as well. An alternative solution proposed to cut costs is the use of volunteers who drive the DCC guests and if necessary, a few vehicles can be hired by the DCC. The problem has many constraints, both with regard to the drivers, as well as with regard to DCC guests.

Some (but not all) of the volunteers can drive some DCC guests directly to the DCC and back, picking up additional DCC guests along the way. Other drivers can only pick up and drop off DCC guests at certain locations on the way to their work or other errands, provided the detour they make is not too big, and provided the number of DCC guests in their car exceeds neither their car's seating capacity nor their personal capacity of handling several DCC guests at once. Each driver has a *morning time window* within which he/she has to leave home, and arrive at work, and similarly, an *evening time window*. Some locations (homes of DCC guests, or other locations such as a community centre or a sheltered bus stop) serve as *transfer locations*. These are locations where a few DCC guests can be gathered and wait until they are picked up to go to the DCC or dropped off on the way back home. The transfer locations also have a time window within which DCC guests have to be dropped off and picked up, as well as a *capacity constraint* - the maximum number of DCC guests that can stay there at any given time. Some DCC guests can reach the transfer locations independently, by walking or cycling. The DCC guests should not be commuting in either direction for more than T minutes ($T = 90$), and they should not have to wait in more than one transfer location for reasons of convenience.

Finally, dedicated bus services (called *chartered buses* hereafter) are used to transport DCC guests who are not being served by a volunteer driver. The chartered buses have different capacities and costs, proportional to their capacity. The cost of using a chartered bus covers the cost of using the vehicle and hiring the driver. It is the sum of a fixed start fee and a time- (or distance-)dependent fee. This paper aims to find routes so that all the DCC guests reach the DCC in the morning, and return home in the evening in such a way that all constraints are met and the cost of the chartered vehicles is kept as low as possible.

As mentioned in Section 2, the SBRT solved by [9] shares the following properties with the DCC problem which means that one problem can be reduced to the other one by selecting appropriate parameter settings:

1. a subset of the pupils can move autonomously to the pick-up locations,

2. pupils can change vehicle at particular transfer locations and

3. departures and arrivals are subject to time window constraints.

Essential differences are:

1. (electric) wheelchair and person incompatibility (e.g. different intellectual disabilities) constraints in the DCC problem,

2. no detour constraints on individual bus trips in the SBRT as opposed to carpool driver specific upper limits for detours in the DCC problem,

3. the number of buses is minimised by the SBRT and the number of carpool drivers is given in the DCC problem and

4. the cost per unit distance is the same for each partial trip in the SBRT. In the DCC problem the bus can bring DCC guests to the DCC only. Since DCC guests can transfer at most once, the pre-transfer partial trip is by carpooling. The cost per unit distance for the pre-transfer part is much lower than for the chartered bus part because carpool drivers act as volunteers. For this reason, the results obtained by the SBRT and the DCC problem cannot be easily compared. However, the SBRT results show the importance of transfers. In the situation where sufficient carpool drivers are available and no chartered bus is required, the costs for all trip parts are equal and the beneficial effect of transfers shown by [9] will apply.

The SBRT solves the complete set of sub problems required for optimisation. In our case, carpooling trips are used as feeders for transfer locations where DCC guests are picked up by one or more chartered buses that cannot be part of the optimisation for the reasons mentioned in Section 3.2.

3.2. Cooperation between Stakeholders

Three parties are involved: the transport *provider* (the bus operator), the transport *organiser* (the institution, including the volunteering car drivers) and the *passenger* (the DCC guest). Solving the daily commuting problem requires the formulation of an objective function involving elements of all three parties. However, an additional complication arises: cooperating transport providers and transport organisers each provide a part of the solution but act *independently* and hence do not share all available information. Each party tries to optimise its own operations. In the specific Flemish context where the solution will be applied, the size and spatial distribution of *day care centres* and schools may allow/require bus operators to serve passengers for multiple institutions during a particular trip. In order to operate efficiently, the transport provider (bus operator) needs to solve instances of the capacitated vehicle routing problem with time windows (CVRPTW). However, the pick-up locations to be served by the bus depend on the solution found for the carpooling-based problem proposed by each transport organiser (day care centre, school). The set of pick-up locations constitutes the interface between the respective transport *provider* and transport *organiser* problems.

An overall solution is conceived as follows. Each transport organiser proposes several different *sufficiently good* solutions for its own sub-problem. The transport provider determines its own optimal solution to all of these cases. Finally the optimal case is the one for which the combined cost of the transport provider and transport organisers is minimal.

3.3. Solution

This paper focuses on the transport *organiser* problem. Hence, the goal is also to investigate some objective functions in order to be able to find an economically feasible solution. Examples are minimising the number of pick-up locations for the chartered bus or minimising an approximation (based on partial information) for the cost of a chartered bus, which will not necessarily lead to the same results.

The locations to be served by the buses are the DCC guests' *homes* and the *transfer locations* where at least one DCC guest is not served by carpooling (i.e. either not picked up in the morning flow or not dropped off in the evening flow). These DCC guests are considered as being *stuck*.

Solving this combinatorial problem requires heuristics that are efficient with respect to the quality of the solution as well as with respect to the computational effort (time). This paper describes the exploration of the solution space that was carried out to support the design of heuristic solutions. It is interesting to know the level of potential savings in concrete cases in order to find out the suitability of the solutions because implementing them both at practical organisational level and at the level of required computer time is non-trivial.

To give the reader an idea of how a solution can look like, an example is shown in Figure 1. White circles represent non-transfer locations, light grey circles represent transfer locations and the dark grey circle is the DCC. Locations are connected by a directed edge. A continuous line edge indicates the route of a carpool driver, while dashed edges indicate the route of a chartered bus; routes in different colours represent different drivers/chartered buses. For clarity, a filled black dot indicates that a voluntary driver is available in an origin location. All concepts mentioned in the remainder of the text are based on this case study.

3.4. Costs Distribution

Note that the distribution of the cost for the combined carpooling-bus solution over all DCC guests also presents difficult issues to solve. It is to be combined with the refunding of the costs of the carpool drivers. The chartered buses considered in this paper are taxi mini-buses. The fare for a taxi (excluding

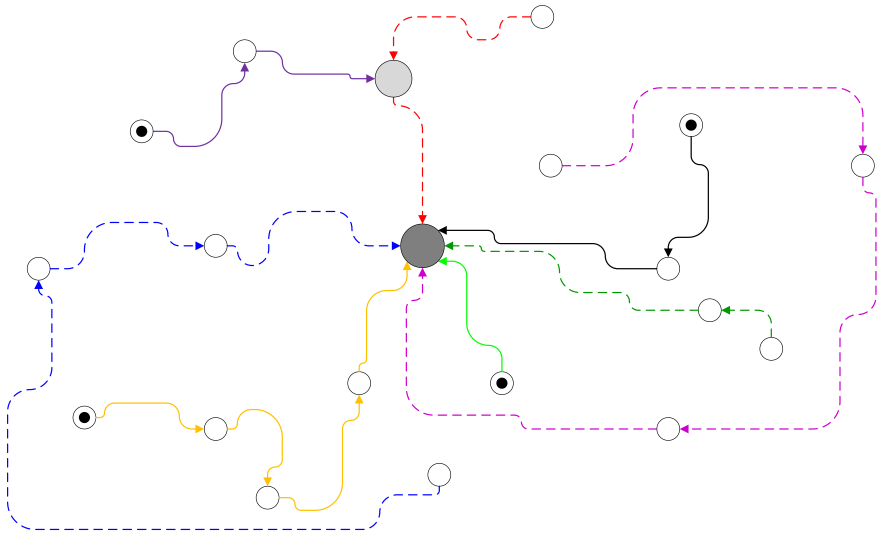


Figure 1: An example of a possible solution for the DCC problem.

start cost) is approximately 2 Euro/km[1,2]. Travel cost refunding to a volunteer for a trip by private car is at most 0.3460 Euro/km [16]. Carpooling costs less per unit distance (by at least a factor 5) than the chartered bus travel but on the other hand consumes time offered voluntarily by the drivers. Research to solve the payment and refunding problem is *not* covered in this paper.

4. Concepts

The concepts described in the following subsections play an important role in this application.

4.1. Flows

The *morning flow* in a day describes the problem of reaching the DCC from the DCC guests' homes, while the *evening flow* describes the trips from the

[1] http://www.taxisverts.be/nl/tarieven-taxi-brussel

[2] http://www.antwerp-tax.be/nl/diensten-prijzen

DCC to the DCC guests' homes. The focus in this paper is on the morning flow.

4.2. Participants

Participants is a collective name for *DCC guests* and *drivers*. *DCC Guests* are persons who need to reach the day care centre on a regular basis. Drivers are persons who are willing to pick up DCC guests and bring them to the day care centre or to locations near to the day care centre. Typically those drivers are the parents, family members or even other volunteers such as neighbours.

4.3. Locations

Four different types of *locations* can be identified: the day care centre (DCC), non-transfer location (NTL), transfer location (TL) and driver destination location. The *day care centre (DCC)* is the location where DCC guests stay during the day; typically those DCC guests travel to the DCC in the morning and leave the DCC in the evening. A *transfer location* is a location where a passenger can transfer from one vehicle to another. A transfer location is subject to capacity (number of people) and time window constraints. Note that a transfer location can coincide with a home location. Transfer locations are the only locations where partial trips are concatenated. They may be served (for both drop-off and pick-up) by a volunteer driver or by a chartered bus. A DCC guest may be dropped off at a transfer location and be picked up by another vehicle. The number of transfers in a passenger trip is limited to one. In the morning (evening) flow, each partial trip by bus ends (starts) in the DCC.

Non-transfer locations (NTL) are homes where the resident DCC guest can be picked up but no transfer is possible. Finally, *driver destination locations* are the end points of the drivers' trips. Typically, destinations are the work locations of the drivers in the morning flow, while in the evening it will be the home locations; however any location could be a destination location (shopping, running errands, leisure, etc.).

A location becomes a *stuck location* if and only if it hosts at least one DCC guest who is not part of a carpool solution and hence, needs to be picked up by a chartered bus.

4.4. Periodic Schemes

Each participant specifies one or more *periodic schemes*: such a scheme is a table which specifies the required trips and their spatio-temporal constraints for a sequence of consecutive days. The number of days involved defines the length of the period. Such a scheme is specified to start at a given calendar date and is then repeated until it is replaced by another one. Each individual specifies a *default* scheme and some special ones. Many people use weekly schemes that differ between working and holiday periods of the year.

For a driver, the periodic scheme specifies the driver's availability, the trip origin and destination, the departure and arrival time windows, the maximal detour (relative to the solo-driven trip) and the capacity of the available car for both autonomous and wheelchair-bound DCC guests.

DCC guests specify trip origin, destination and the associated time windows and add specific requirements: (i) the maximum number of transfers to reach the DCC, (ii) the requirement for a wheelchair, (iii) the incompatibility with specific drivers or DCC guests and (iv) the maximum allowed travel time (currently fixed to 90 minutes). The maximum travel time is based on the current legislation in Flanders with respect to school children transport. Children in special schools are picked up by chartered buses and are allowed to travel at most 220 minutes a day [17, 18].

Availability and time windows for transfer locations are also specified in periodic schemes. Note that in actual practice most of the transfer locations coincide with DCC-guests' homes; household members are prepared to take care of some DCC guests arriving at their location until they are picked up. Hence, the properties for most transfer locations are specified by *personal* periodic schemes. If the transfer location is a public bus stop or community centre (= dedicated transfer location), it has "unlimited" capacity.

Requirements and constraints for a *given calendar day and flow* are derived from the set of periodic schemes that apply for that date. The willingness to extend the time window width and the detour time may have a huge impact on the quality of the solution since at each relevant location the time windows for arrival and departure are *intersections* of time windows specified in the periodic schemes.

4.5. Chartered Buses

DCC Guests who cannot reach the DCC by carpool will be picked up by a *chartered bus* (a bus that is hosted by a commercial provider) either at home or at a transfer location. Buses are operated by one or more providers. Providers can serve DCC guests for multiple clients (day care centres, institutions etc.) during a single trip. Hence, no function is available to map ordered sets of pick-up locations to a client specific cost value.

4.6. Organisation of Transport using Independent Complementary Providers

Solutions to the *school bus routing problem* presented in operations research literature assume (i) long term stable demand, (ii) the use of a single transport mode and (iii) a single operator. On the other hand, the DCC problem is characterised by

- **variable demand**: Demand varies from day to day and if it turns out to be periodic, the period length in the most simple case is the *smallest common multiple* of the length of the default personal periodic scheme lengths which may be very large (people may have two, three and five week periods). In practice the period may be much longer due to the individual specific holiday periods. Consequently, the set of pick-up locations is not constant. Note that this is only a practical but not a fundamental problem.
- **multi-modal travel**: Except for the first part of the trip that may be executed autonomously by DCC guests, trips are either pure bus trips, pure carpool trips or mixed carpool-bus multi-modal trips.

- **multiple operators**: In the DCC problem two types of transport providers cooperate: (i) *carpool drivers* who are assumed to agree on a single globally applicable unit-distance fare (to refund the travel cost) (ii) and one or more *bus operators* offering proposals for transport services. The involvement of independent operators poses a fundamental problem.

For the morning flow, the main goal is to map the set of DCC guest departure (home) locations to a possibly empty set of transfer locations in order to minimise travel costs (this is because DCC guests can have at most one transfer). This coincides with the *pupil-stop assignment* stage mentioned in [9] but in the DCC problem, the support of carpool drivers is required.

The cost to serve all locations of the *mapped* set by buses is unknown to the consultant in charge of finding a solution for the carpooling sub problem. This is because the bus operators are independent and can organise their operations autonomously (for the reason mentioned in Section 3.2).

In practice a request for proposals (RFP) needs to be issued and a tender needs to be organised. For practical reasons a medium to a long term contract is preferred. The contract is based on agreed unit-distance fares: the current practice is characterised by (i) single mode, (ii) single provider and (iii) daily changing demand. The agreement specifies the service level and unit-distance prices.

One way to overcome the problem of variance in the data is to use a large set of typical cases as a basis to set up a tender. A period covering the duration of the contract to be negotiated is to be considered. For each day in that period the DCC guest demand and driver availability are derived from the periodic schemes and the set of pick-up locations is determined using several different criteria (e.g. smallest transfer location set, etc.). The N most frequently occurring sets are passed to providers in the RFP.

Finding transfer locations that remain stable in time needs to be done anyway, in particular in the interest of the DCC guests.

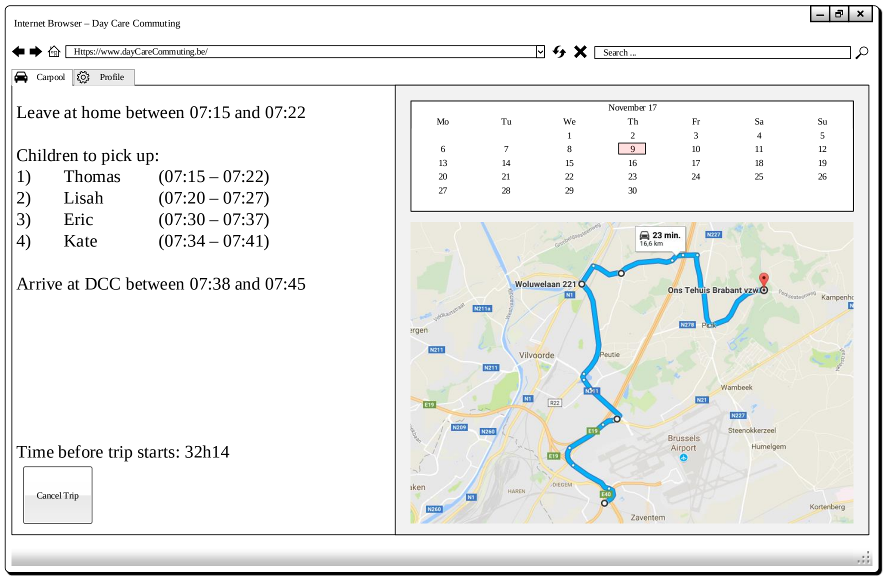


Figure 2: An example of a web application to support (in this case) a driver. The map was taken from Google Maps [19].

5. Data

5.1. Real Situation: Collection by a Web Application

Many challenges arise when developing an application such as this DCC problem. Users of this application should be well informed, and updating information about planned trips and cancellation should be conducted in a very easy and straightforward way. In Figure 2, a possible example (mock-up) of a web application for a trip by a driver can be seen. However, GUI development is not in the scope of this paper.

The idea is to create a platform where participants can provide all the necessary information as was explained in Section 4.

With this information, the application can compute a schedule for every driver and guest. A driver needs to receive information about when she has to leave home, which locations she should visit and in which order she has to do that, while guests need to receive information about who will pick them up (possibly with information about the car or a specific coloured sign to make it easier for (mentally) disabled passengers) and at what time. All this information will be provided according to the participants' preferences, e.g. by email, by text message, etc.

The authors are aware of the fact that, especially for mentally disabled guests, it would be difficult if the daily routing would change a lot. Fundamental research needs to be done before a substantiated reasoning about this issue can be conducted. However, the authors believe that there will be some kind of regularity, not necessarily on a daily base, but on a weekly base. Real challenges will appear when a driver who committed to drive, will cancel this commitment for any reason (e.g. illness). Current technology can be very helpful in this respect such as smartphone apps. Smartphone apps could provide information about the pick-up vehicle such as current location, delay, type of the vehicle, colour, arrival time etc.

5.2. Sensitivity Analysis: Data Sampling

At the moment of writing this paper, there was only data available of origin locations of DCC guests of a specific DCC in Flanders. No data is currently available of individual preferences. Questionnaires are being developed to collect more data in order to be able to compute some realistic cases.

However, data is needed to test the proposed algorithms and to attempt to investigate the feasibility of the solutions. To solve this issue, a *data sampler* was developed. The sampler creates periodic schemes and determines for every participant and location different constraints for every day of the week. A flow chart of the following explanation can be found in Figure 3.

The software samples 30 DCC guests. For every DCC guest a home location is sampled; there is a 40 % probability this will become a transfer location. For every week day (excluding Saturday and Sunday), there is 80 % probability that a DCC guest needs to go to the DCC. For every DCC guest, there is a 25 % probability that a home driver is present. For every week day (excluding Saturday and Sunday), this driver has an 80 % probability of being able to drive.

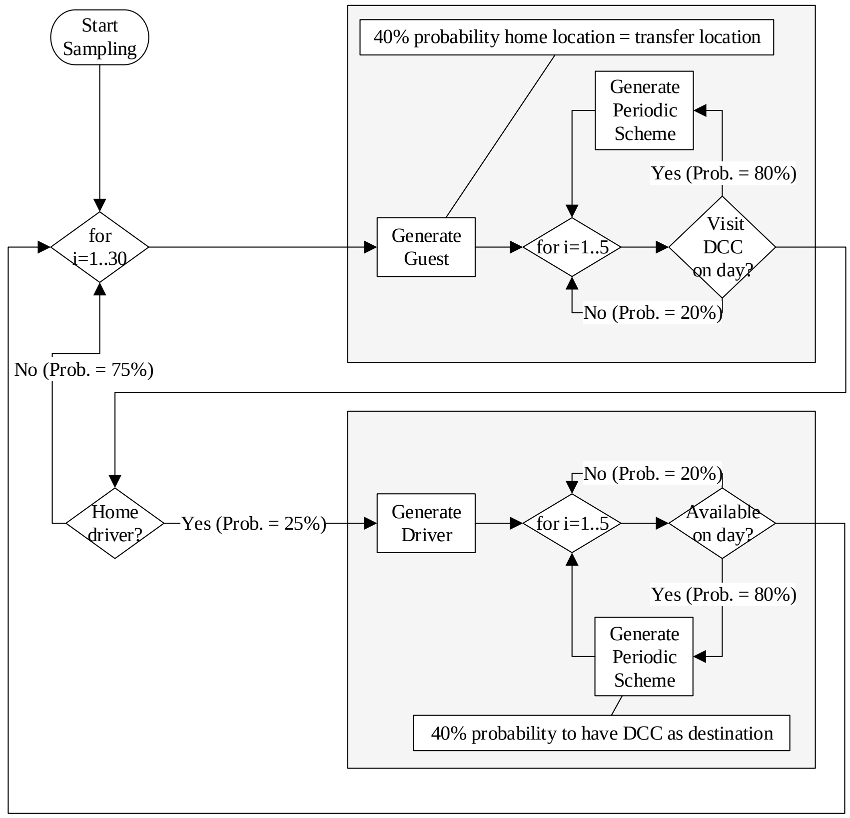


Figure 3: Flow chart describing the process of generating sample data.

In 40 % of the cases the driver's destination is the DCC; in all other cases, the destination is randomly sampled using the distribution for *home-work* distances and are uniformly distributed over space (within the borders of Flanders).

Home-work distances are determined by means of *inverse transform sampling* using the cumulative distribution $D(d)$ for the home-work travel distance found by the Flemish household travel survey (OVG) ([20]). Thereto, the following equations are used:

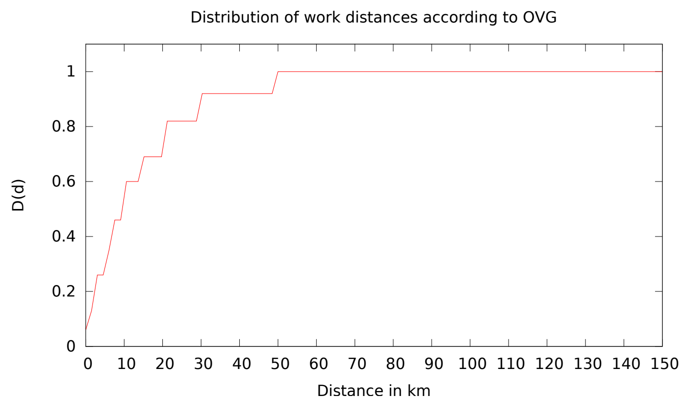


Figure 4: Cumulative relative frequency distribution $D(d)$ for home-work distance d acquired from the OVG survey. The numerical specification is given in Table 1.

Table 1: Numerical specification for the cumulative distribution for the home-work distance in kilometres according to the OVG survey.

d[km]	0	1	2.5	5	7.5	10	15	20	30	50	150
D(d)	0	0.06	0.13	0.26	0.35	0.46	0.60	0.69	0.82	0.92	1

$$p \sim U(0,1) \tag{1}$$

$$p \in [p_i, p_{i+1}[\Rightarrow d \in [d_i, d_{i+1}] \tag{2}$$

$$p \in [p_i, p_{i+1}[\Rightarrow d \sim U(d_i, d_{i+1}) \tag{3}$$

where $U(a,b)$ denotes the uniform distribution with boundaries a and b.

Equation (1) samples a uniformly distributed value that represents a probability. Equation (2) evaluates the inverse $d = D^{-1}(p)$ of the distribution function $p = D(d)$ and delivers a distance range $[d_i, d_{i+1}]$. Finally, Equation (3) samples a distance value from that range.

The values for p_i and D_i are given in Table 1 which is the numerical specification of Figure 4.

Home-DCC distance is estimated using data for a particular Flemish DCC for which all 25 home locations were given. Bird's eye distances from these home locations to the DCC were calculated. Results showed a mean of 5 870.25 metres with a standard deviation of 2 582.05 metres.

The variables *maximum detour time* (MDT) and *time window width* (TWW) are configurable and used for sensitivity analysis. Their impact on the possible solutions will be examined during the experiments in Section 9.

First, MDT is used as follows: arrival times of the participants are sampled in the interval [07:00, 10:00]. Departure times are computed based on the travel times from origin to destination, taking into account the preferred arrival time. In order to get the earliest required departure time, the start time of the departure time window is decreased with the maximum detour time.

Second, in order to ease results interpretation, all applicable time windows have the same length TWW. For participants, time windows are used for (i) trip departure time and (ii) trip arrival time. The same setting is used for the allowed waiting time in home-based TLs. In this case the time window is bound to the preferred departure time of the participant living at that location. Consider an event for which the moment in time having a preferred value t_0 (e.g. trip start time); then, the range for its effective value is given by $[t_0 - TWW/2, t_0 + TWW/2]$.

6. Limitations and Assumptions

The proposed algorithms in Section 8 are based on the following assumptions: (i) If a driver visits locations on his way to his destination, all the DCC guests currently present at that location are picked up by that driver, and (ii) when a trip ends for a driver, all the DCC guests are dropped off at once either at the DCC or at a transfer location.

Obviously, paths in a solution respect all the constraints (as discussed in Section 4) which are related to locations as well as participants.

Due to lack of real surveyed data, the following assumptions are made while generating test cases: (i) the constraints related to wheelchair usage and incompatibilities between DCC guests are ignored (although supported by the model),

(ii) at most one DCC guest is living in every location, (iii) home locations which are also TLs have a capacity of two, (iv) car capacity is always four (excluding the driver) and (v) the arrival time window of the DCC is [07:00, 10:00]. This paper only takes care of the morning flow. The evening flow should be analogously solvable.

7. Graph Theoretical Formulation of the Carpooling Subproblem

We present in this section the DCC problem in graph theoretical tools. This formulation supports in defining our objective function and optimisation problem. Let $G = (V \cup \{0\}, E)$ be a directed complete graph with 0 as the DCC. The vertex set $V = \{V^t \cup V^{nt} \cup V^w\}$ is composed of a subset V^t of *transfer locations (TL)*, a subset V^{nt} of *non-transfer locations*, and a subset V^w of other locations, which we will refer to as driver destination locations. The transfer locations are of two types. $V^t = \{V^t_h \cup V^t_{other}\}$ (Most transfer locations are the *homes* of the DCC guests, denoted by V^t_h but few are not, denoted by V^t_{other}). The set $V^t_h \cup V^{nt}$ represents the *homes* of the DCC guests, and each home contains (in this paper) one DCC guest. Some of the homes have cars (available to be used), but not all of them. We denote by $car(v_i)$ the car associated with vertex v_i. If location v_i does not have a car then $car(v_i) = \emptyset$. Let $l : E \to \mathbb{R}^+$ be the length function that associates a length $l(e)$ to each $e = (v_i, v_j) \in E$, which is the time it takes to travel from v_i to v_j. Each vertex v_i has a *time window* interval $[e_i, l_i]$, meaning a vehicle can reach v_i no earlier than e_i and leave no later than l_i. In addition, each vertex $v^t_i \in V^t$ has a *passenger capacity* $c^t_i \geq 2$ which is the maximum number of DCC guests that can simultaneously be present at the transfer location during its time window. Each vehicle i has a *capacity* c^c_i which does not include the driver. We assume here that all cars have capacity $c^c_i = cc = 4$, except for chartered buses which have a larger capacity $c_B \in \{8, 12, 30\}$. In addition, each car has a *detour constraint*, which is the maximum extra time the driver of the car is willing to be on the road in order to pick up or drop DCC guests. We denote the detour constraint of the car which belongs to home v_i by $\delta(i)$.

We will focus on formulating and solving the problem of planning the routes in the morning, where the DCC guests need to be driven from their homes to the DCC. We will refer to it as the *morning problem*. The *evening problem*, of driving the DCC guests from the DCC back to their homes is not covered by this paper, but can be solved by symmetry.

Every vehicle chooses a *feasible route* which begins at some home vertex in $V^t \cup V^{nt}$, picks up and drops some DCC guests in some vertices on its route, and ends up either in the DCC, or in a transfer or work location, without violating the capacity, time window, and detour constraints.

A *feasible passenger route* is a path from the passenger's home to the DCC that respects all time windows, does not involve waiting in more than one TL, and such that its total length, including waiting times, is less than T. An *optimal solution* to the problem is a planning of feasible routes for the vehicles and DCC guests such that the total cost of chartered buses is kept to a minimum. Two different objective functions are investigated: (i) minimising the number of pick-up locations for the chartered buses and (ii) minimising chartered buses and driven kilometres by using a simple version of a Vehicle Routing Problem solver (See Section 8.4.1).

Each car and its driver is associated with some route which corresponds to a directed path in G. Below are some standard definitions related to paths, and a collection of paths, which are called a *path family*.

Definition 7.1 (Path, initial, terminal, cardinality, length, distance). Let $G = (V, E)$ be a directed weighted graph, with weight function $l : E \to \mathbb{R}^+$. A *path* P in G is a sequence of distinct vertices $(v_1, v_2, \ldots, v_l)$ such that $(v_i, v_{i+1}) \in E$, for $i = 1, 2, \ldots, l-1$. (Note that P is a directed path in G). The set of vertices and edges of a path is denoted by $V(P)$ and $E(P)$, respectively. The initial and terminal vertices of P, v_1 and v_l, are denoted by $ini(P)$ and $ter(P)$, respectively. Other vertices e_i, where $i = 2, \ldots, l-1$ are called *intermediate* vertices of the path. The *cardinality* of a path is defined by

$|P| := |V(P)| = l$, and the *length* of the path is defined by $l(P) = \sum_{e \in E(P)} l(e)$. For vertices u and v, the *distance* from u to v is the length of the shortest path from u to v, and is denoted by $dist(u, v)$. Note that $dist(u, v)$ may be different from $dist(v, u)$ since the graph is directed and not symmetric. If P and Q are paths where the terminal vertex of P equals the initial vertex of Q, then the concatenation of P and Q is denoted by $P * Q$.

Definition 7.2 (Path family). A collection of paths, not necessarily disjoint, is denoted by a script letter $\mathcal{P}$ or $\mathcal{S}$ and is called a *path family*. We write $V[\mathcal{P}] := \bigcup\{V(P) : P \in \mathcal{P}\}$, $E[\mathcal{P}] := \bigcup\{E(P) : P \in \mathcal{P}\}$, i.e. $V[\mathcal{P}]$ and $E[\mathcal{P}]$ are the sets of vertices and edges covered by the paths in $\mathcal{P}$, respectively. Denote by $ter[\mathcal{P}] := \bigcup\{ter(P) : P \in \mathcal{P}\}$ the set of terminal vertices of paths in $\mathcal{P}$ and similarly $ini[\mathcal{P}] := \bigcup\{ini(P) : P \in \mathcal{P}\}$. The *out-degree* of a vertex v in a path family $\mathcal{P}$ is denoted by $deg^+_{\mathcal{P}}(v)$ and it is defined as the out degree of v in the graph induced by $E[\mathcal{P}]$. The term $deg^-_{\mathcal{P}}(v)$ is similarly defined.

In a feasible solution to our problem, each car starts from a location that contains a car, moves along some path in the graph picking up at least one DCC guest at every vertex on the path and ends up either in a TL or the DCC, dropping all the DCC guests in the car there, such that the car capacity constraints, the time window constraints of all the vertices of the path, and the detour constraints are satisfied. The cars that end up in a TL may continue to their driver destination (work or other errands), but as far as the solution is concerned, we ignore those trips from a TL vertex to a driver destination vertex. This motivates the following definitions:

Definition 7.3 (Constraint Feasible Path (CFP)). A *constraint feasible path (CFP)* in G is a path $P = (v_1, v_2, \ldots, v_l)$ where $v_1 \in V^t \cup V^{nt}$, v_1 contains a car, $v_l \in V^t \cup \{0\}$, $l \leq c + 1$, satisfying the time window constraints of all the vertices in the path, as well as the detour constraints. We denote by $car(P)$ the car associated with location $ini(P)$.

Since no DCC guest can wait in more than one TL, and once all DCC guests are picked up from a non-terminal location there is no other driver who should stop at his/her home, a family $\mathcal{P}$ of CFPs should have the property that every pair of paths in $\mathcal{P}$ is either disjoint, or have a unique vertex v_t in common - their common endpoint which is the DCC or a TL (where the drivers drop their DCC guests), or v_t is an endpoint for one of the paths, and the other path contains v_t and ends in the DCC - implying that the second car picks up all the DCC guests in the TL and drives them to the DCC (see Figure 5). This motivates the following definition:

Definition 7.4 (Semi-disjoint Constraint Feasible Paths (SD-CFP)). A path family $\mathcal{P}$ consisting of CFPs is *semi-disjoint* if any pair of paths $P_1, P_2 \in \mathcal{P}$ satisfies one of the following three conditions:

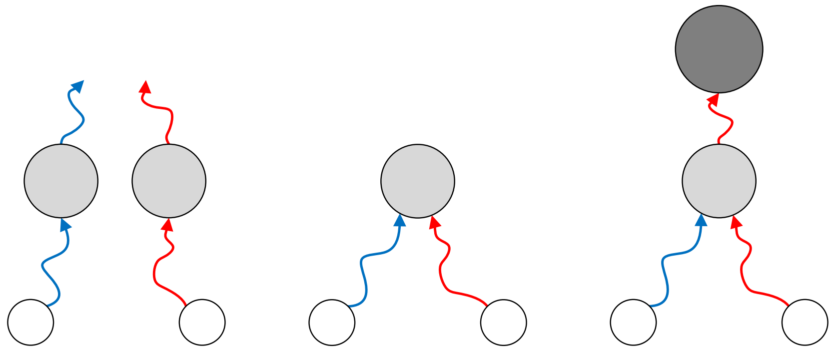
(a) $V(P_1) \cap V(P_2) = \emptyset$ (Figure 5a)

(b) $V(P_1) \cap V(P_2) = \{v_j\}$ for some $v_j \in V^t \cup \{0\}$ and $v_j = ter(P_1) = ter(P_2)$ (Figure 5b)

(c) $V(P_1) \cap V(P_2) = \{v_j\}$ for some $v_j \in V^t$ where $v_j = ter(P_1)$ and $ter(P_2) = \{0\}$ (Figure 5c)

In addition, for any such vertex v_j above, the passenger capacity is not violated, in other words the number of paths in $\mathcal{P}$ terminating at v_j is at most c_j^t, and the amount of time car(P_2) needs to wait for car(P_1) to reach v_j (in case car(P_1) reaches v_j after car(P_2)) is feasible with respect to $\delta(car(P_2))$, the detour constraint for car(P_2).

The problem that we address is finding a family $\mathcal{P}$ of semi-disjoint constraint feasible paths (SD-CFP) which covers as many vertices in $V^t \cup V^{nt}$ as possible. This will guarantee that as many children as possible are picked up by carpooling, and as few as possible need to be picked up by a paid vehicle. This is an NP-hard optimisation problem. The search space of possible solutions is exponentially large.



(a) Two drivers bring guests to distinct transfer locations.

(b) Two drivers bring guests to a joint transfer location. Both guests are dropped at this location.

(c) Two drivers bring guests to a transfer location. One of the drivers picks up these DCC guests and possibly other guests who are already at this location.

Figure 5: Possible relations between two CFPs. White circles represent non-transfer locations, light grey circles represent transfer locations and the dark grey circle is the DCC. Locations are connected by a directed edge which represents the route of a driver; routes in different colours represent different drivers.

7.1. Finding Family of Constraint Feasible Paths

An important step in the algorithm is finding possible paths of drivers. It means finding all CFPs in G, which we call *CFP family*, and denote it by $\mathcal{S}$. A collection of all CFPs is found in two stages:

In the first stage, a pre-process step is conducted; a compatibility digraph $G^c = (V \cup \{0\}, E^c)$ is constructed. For every ordered pair (u, v) of vertices in G where u contains a vehicle, an edge $e = (u, v) \in E^c$ is defined if and only if the vehicle in u can drive to location v considering the time $l(e)$, and if all of the following constraints are met: the time windows in u and v, the detour constraint $\delta(u)$, and the personal incompatibility between DCC guests. (If at least one of these constraints is not met, then (u, v) is not an edge in G^c).

The second stage includes finding all constraint feasible paths $\mathcal{S}$; using G^c we find the set of all CFPs in G of cardinality at most $cc+1$ (where cc denotes the maximum car capacity, and is usually 4) in the following way: for each vertex v with a car, let $N = N^+_{G^c}(v)$ be the set of out neighbours of v in the graph G^c. We check every ordered subset $s \subseteq N$ with a terminal vertex in $V^t \cup \{0\}$, and of maximum size four. (There are at most $k(k-1)\ldots(k-3)$, such subsets, where $k = |N|$). If the combined path $v \star s$ is a CFP we add it to the set $\mathcal{S}$.

8. Algorithmic Solutions

As described in Section 7.1, the CFP family is computed. This CFP family represents all the possible paths drivers can use respecting all driver related constraints. For a solution to the DCC problem, an algorithm should choose at most one path for every driver. It is important to know that when a path is chosen, other paths in the CFP family may be *disabled*, since they do not obey the conditions of SD-CFP - see Definition 7.4. If a path is disabled, it cannot be chosen any more for the current solution. Hence, if path $P \in \mathcal{S}$ is chosen, then disable every $Q \in \mathcal{S}$ for which P and Q do not constitute SD-CFP as described in Definition 7.4. This also means that every driver appears at most once in the solution and non-transfer locations are only visited once, since it is irrelevant to visit a location without a pick-up action.

8.1. Exhaustive Search Method

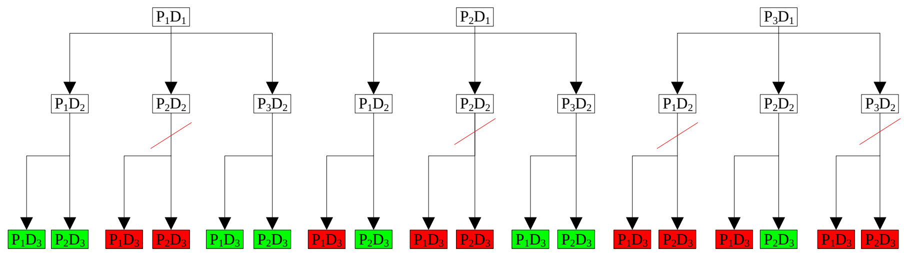
We first use an exhaustive search method to find all families of SD-CFPs and choose a family which delivers the best score for the chosen goal. Goals and scoring functions are detailed in Section 8.4.

The input for the algorithm is a family of CFPs, as described in Section 7.1. Given this path family, a graph is built where a vertex represents a driver and vertices are connected by an edge if and only if they share at least one non-DCC location in one of their paths. This graph is partitioned into *connected components*. Two drivers belonging to different components can be handled independently. Clearly, each component constitutes an independent smaller problem. The more components are found, the better for the (costly) exhaustive search method. Every component can be handled separately by considering all the possible combinations of driver paths.

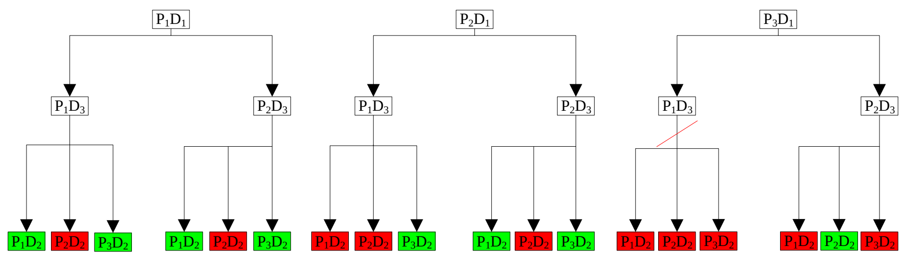
The enumeration of the solutions goes as follows: The algorithm starts by choosing a path for the first driver, next a path for the second driver and so on. When a path is chosen, the current combination of paths is checked on validity. When this combination is not valid, every possible combination with paths of the remaining drivers can be ignored, and the algorithm backtracks to the next CFP of the driver. As soon as a (possibly empty) path is successfully added for every driver, the combination is registered as a solution and scored; the search then continues to find the next one. A driver can appear at most once in each solution and the non-transfer locations will be handled by at most one driver. This means that during the enumeration (in some cases) a large number of paths of other drivers will be cancelled out.

Path enumeration order affects the required computational effort. This is a subject for further research. Two possible orders of search space scanning are represented schematically in Figure 6. P_iD_j stands for path i of driver j. A path from the root to a green leaf represents a feasible solution, while paths ending in a red leaf are infeasible. An arrow represents the feasibility check of the driver path it points to. A red line through an arrow indicates that this check is not necessary any more since the previous feasibility check failed. Figure 6a and Figure 6b use the same input data, but use a different order of combining the paths of the drivers. The order of the chosen paths is not relevant for the solution since (i) every non-empty subset of the drivers and for each of the subset members exactly one of the feasible paths is considered, (ii) in a feasible solution, paths (irrespective of the driver) need to be semi-disjoint and (iii) feasible time windows are predefined (i.e. do not depend on e.g. the first arrival time of a passenger at a transfer location).

In total there are 18 possible combinations, which results in 27 feasibility checks from which 19 are actually conducted for Figure 6a and 24 feasibility checks for Figure 6b from which 21 are actually conducted.



(a) An exhaustive search where the paths of drivers D_1, D_2 and D_3 are combined. With respect to the infeasible combinations, in this case, 19 checks are needed.



(b) An exhaustive search where the paths of drivers D_1, D_3 and D_2 are combined. With respect to the infeasible combinations, in this case, 21 checks are needed.

Figure 6: An example of the exhaustive search algorithm. P_iD_j stands for path i of driver j. Leafs in red are not feasible, while leafs in green are feasible. In total there are 18 possible combinations. Every arrow indicates a feasibility check. A red line through an arrow indicates that this check is not necessary any more since the previous feasibility check failed.

The number of cases to evaluate can be (extremely) large. Given a set of drivers $D = \{d_1, d_2, \ldots, d_n\}$ and the set of paths for a driver d_i, $\mathcal{P}(d_i) = \{P_1^{d_i}, P_2^{d_i}, \ldots, P_m^{d_i}\}$, the total number of cases to evaluate equals

$$\sum_{D_k \in 2^D \setminus \emptyset} \Big(\prod_{d \in D_k} |\mathcal{P}(d)| \Big) \tag{4}$$

where 2^D represents the power set. This may be very large; moreover, every solution is some subset of these paths which is SD-CFP, so the computation time may be very large

In order to avoid infeasible executions times, computations are stopped after

60 minutes.

8.2. Heuristics 1: Focusing first on DCC

This is a two-stage heuristic. In the *first* stage only the CFPs in the SD-CFP family that lead to the DCC are considered. In other words, only the drivers who drive to the DCC are taken into account. The heuristic attempts to find a routing for them that picks up as many DCC guests as possible. For each driver a route is chosen, so that all drivers pick up as many DCC guests as possible. Once such a routing is found for all the drivers who reach the DCC, the set of TLs covered by these routes are considered. In the *second* stage, as many DCC guests as possible are collected at these TLs (respecting the constraints of the path). We use a greedy approach for these subproblems since they are NP-hard.

8.3. Heuristics 2: Focusing first on Transfer Locations

In comparison with the two stage heuristic as described in Section 8.2, the first stage of this algorithm only considers the CFPs leading to dedicated transfer locations (even if the trip could lead to the DCC). In this case the algorithm attempts to bring as many DCC guests to these locations in order to reduce the number of pick-up locations for the chartered buses. In the second stage, remaining CFPs going to the DCC (if any) are taken into account. For the experiments, nine schools in the surrounding area of the DCC were selected to be used as dedicated transfer locations. Other predefined home-based transfer locations were ignored in this solution.

8.4. Scoring

8.4.1. Determination of Bus Trips

As indicated in Section 3.2 it is not possible for legal and operational reasons to solve the problem in an integrated way. In order to find a good solution for the carpooling part it is assumed that the bus operator serves the DCC independently of any other client. For each carpooling proposal, a CVRPTW needs to be solved for the bus service. Open-source solvers such as OptaPlanner [21] and GraphHopper [22] have been considered. It was decided not to use these tools because of the following reasons: (i) it is not known whether these tools are deterministic in the sense that they always give the same results with the same input and (ii) set-up costs and execution times are high.

A non-expensive approximate solution for the CVRPTW is required because many cases need to be evaluated. Therefore, for the morning flow, the cost for each bus is estimated by assuming that it starts at a location where a passenger is to be picked up. This is realistic because the usual contracting rules specify that only the distance driven with at least one passenger on board can be charged. The bus is assumed to move counter-clockwise around the DCC and does not insert wait periods at serviced locations. The DCC location is used as the origin of a polar coordinate system and the angular argument corresponding to the consecutively visited locations is non-decreasing. The bus picks up as many passengers as possible; the trip leads to the DCC and respects the vehicle capacity constraint as well as the timing constraints for each passenger and transfer location. The chronological pick-up order coincides with the counter-clockwise location visit order for each particular bus but not necessarily for the set of all passengers. A bus may need to skip a location in the counter-clockwise order due to timing constraints and because a bus does not wait at service locations. Additional bus trips are scheduled until all passengers reach the DCC and each trip is served by an additional bus.

The quality of the approximation is assessed by considering the total distance driven with passengers on board; this is used to determine the amount to invoice. (It is obvious that the cost per kilometre will increase if more *empty kilometres* are required.) Assume that a solution with N_B buses is found by the proposed approximation. Let $N_L(b)$ denote the number of locations served by bus b. The j-th location served by bus b is denoted by $L_{b,j}$ and the shortest distance between locations L_i and L_j is denoted by $d(L_i, L_j)$. Then the minimum total distance D_{N_B} driven by N_B buses is limited by

$$D_{N_B} \leq \sum_{b=1}^{N_B} \Big(\sum_{j=1}^{N_L(b)-1} d(L_{b,j}, L_{b,j+1}) \Big) \tag{5}$$

Due to the triangle inequality the total length D of all bus trips (irrespective of the number of buses used) cannot exceed the distance for a single-bus star-based solution in which each subtrip between consecutive pick-up locations passes at/near the DCC so that:

$$D \leq d_1 + 2 \cdot \sum_{i=2}^{n} d_i \tag{6}$$

where n is the number of locations served and d_i is the distance between the i-th location and the DCC.

The distance associated with the solution of the CVRPTW is at least

$$\sum_{i=1}^{n} d_i = D_{min} \leq D_{CVRPTW} \tag{7}$$

Due to the triangle inequality and because the number of subtrips driven twice in equation (5) cannot exceed the number of subtrips driven twice in equation (6)

$$D_{min} \leq D_{CVRPTW} \leq D \leq 2 \cdot D_{min} \tag{8}$$

$$D_{min} \leq D_{N_B} \leq D \leq 2 \cdot D_{min} \tag{9}$$

for all values of N_B. Hence, the total distance D_{N_B} found by the proposed approximation and the total distance found by a CVRPTW solver can differ by at most a factor two.

8.4.2. Scoring Functions

Before implementing algorithms and deploying proposed solutions, the economic feasibility is to be investigated. The *effectiveness* of carpooling acting as a feeder for bus stops could be expressed by (i) the fraction of DCC guests delivered by carpools at the DCC, (ii) the fraction of locations where people need to be picked up by the chartered bus and (iii) the number of buses and the amount of kilometres needed to pick up the stuck DCC guests. Two scoring functions are developed, one is based on (i) and (ii) (minimising stuck locations), while the other one is based on (iii) (minimising chartered bus costs).

For the *minimising stuck locations scoring function*, the following variables are taken into account: (i) the number of DCC guests reaching the DCC by voluntary drivers (*nDccGuests*), (ii) the number of transfer locations in which DCC guests are stuck (*nStuckTl*), (iii) the number of origins in which DCC guests are stuck (*nStuckOrigin*), (iv) the number of locations in the input (*nLocations*), (v) the total number of DCC guests in the input (*nGuests*).

This leads to the scoring function represented in Equation 10.

$$score = \frac{nDccGuests}{nGuests} + \left(1 - \left(\frac{nStuckTl + nStuckOrigin}{nLocations}\right)\right) \tag{10}$$

The first term is a measure for the number of DCC guests whose travel problem was completely solved. The second term accounts for the number of locations that need to be visited by the (expensive) bus.

For the *minimising chartered bus costs scoring function* other variables are of interest: (i) the number of chartered buses *nCharteredBuses* and (ii) the average amount of kilometres travelled by the chartered buses (*nAvgDistance*). This leads to the scoring function as can be seen in Equation 11.

$$score = -1 \cdot (nCharteredBuses \cdot 60 + nCharteredBuses \cdot nAvgDistance \cdot 0.5) \tag{11}$$

We assume that 60 €/bus covers the cost of a driver and 0.5 €/km covers the cost of the vehicle per kilometre. Because the algorithm is a minimiser, the score is reversed by multiplying it by -1.

Note that both scoring functions will yield different results for the exhaustive search. As we will see in Section 9.2, minimising the number of stuck locations does not mean minimising driving costs of a chartered bus and vice versa.

9. Experiments

9.1. Scenarios

The goal of this paper is to find out whether cost savings can be achieved by applying carpooling solutions. Since no real case data about participant's preferences are available, a number of combinations should be computed in order to be able to give some advice to DCCs. Two variables will be configurable during the simulation: (i) maximum detour time (MDT) and (ii) departure/arrival time window width (TWW). The used values for the TWW (in minutes) are $\{5, 15, 30, 45\}$ and the used values for MDT (in minutes) are $\{5, 15, 30, 45\}$. This results in 16 different *combinations* to examine.

The flow of the experiments is as follows: first a random *case* is generated based on a given MDT and TWW as described in Section 5.2. Such a case represents a realistic case study for a given DCC. For this random case a solution is computed for every day of the week (except Saturday and Sunday) based on the combination of the three algorithms and the two scoring functions as explained in Section 8.4. Four different experiments are conducted: (i) the exhaustive search method with the minimising stuck locations scoring function, (ii) the exhaustive search method with the minimising chartered bus costs scoring function, (iii) the two-stage heuristic (DCC) with the minimising chartered bus costs scoring function and (iv) the two-stage heuristic (transfer locations) with the minimising chartered bus costs scoring function. This allows us to compare the combination of the algorithm and scoring functions because they are executed on exactly the same data. In our experiments ten random cases are generated per combination MDT and TWW. This means that in total 800 solutions are computed per algorithm and scoring function combination. An overview of the experiment flow can be seen in Figure 7.

9.2. Results

The algorithms and scoring functions are applied to exactly the same data, the results of the cases without voluntary drivers can be found in Table 2.

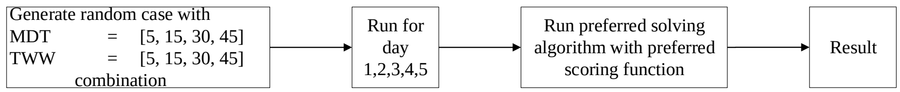


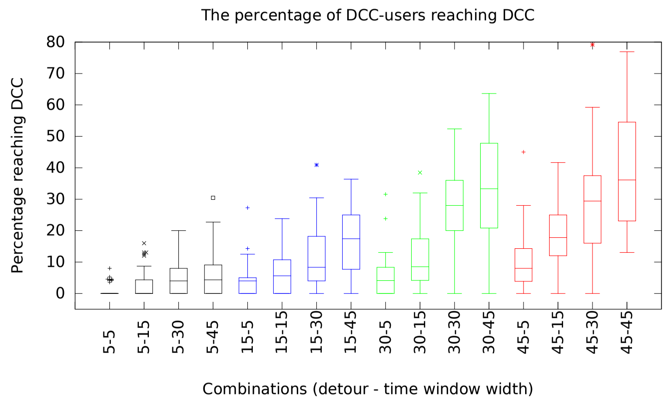
Figure 7: An overview of the execution of the experiments.

Column headers indicate the combination maximum detour time (MDT) and time window width (TWW), the number of stuck locations (nStuckLoc.), the number of buses needed to serve all the DCC guests (nBus) and the average distance travelled per bus (Avg. dist/bus). Note that these numbers are average values. On average around 24 DCC guests need to go to the DCC and depending on the value of the TWW, the number of needed chartered buses vary between 5 and 8. To have a realistic idea, one should take the ceiling of the number (2.4 chartered buses means 3 in reality for example).

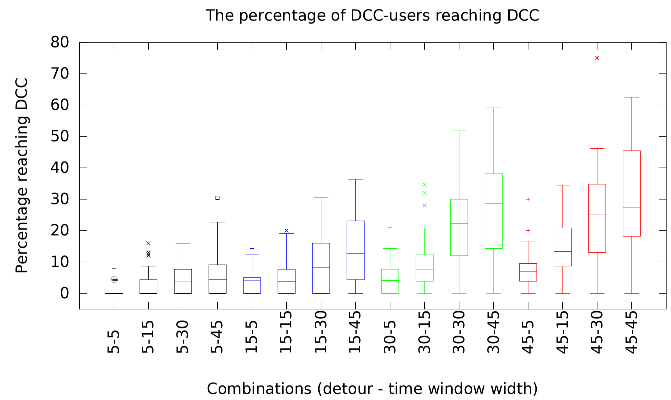
Figure 8 shows box plots for the percentage of DCC guests reaching the DCC by carpooling with voluntary drivers. Each box plot emerges from a set of randomly sampled cases for a particular pair of MDT and TWW values.

Table 2: Results of the cases without voluntary drivers. Note that this are average results of 50 cases (five days per iteration and ten iterations).

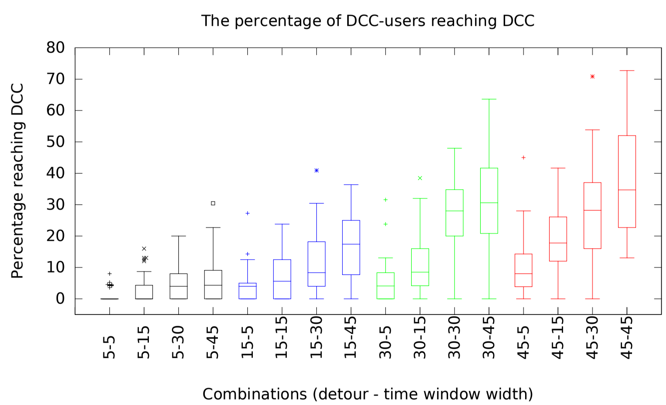
Combination		nStuckLoc.	nBus	Avg. dist/bus
MDT	TWW			
5	5	23.44	5.32	23.27
5	15	23.82	7.20	27.93
5	30	23.96	6.06	35.93
5	45	23.36	4.96	39.60
15	5	24.16	6.28	22.49
15	15	24.72	7.16	28.92
15	30	23.96	5.76	34.63
15	45	23.84	5.02	38.38
30	5	24.52	6.10	24.67
30	15	24.30	7.14	27.27
30	30	23.72	5.94	34.42
30	45	23.96	5.06	39.97
45	5	24.52	6.26	23.39
45	15	24.22	7.06	28.18
45	30	24.26	6.18	33.86
45	45	24.08	5.10	39.94



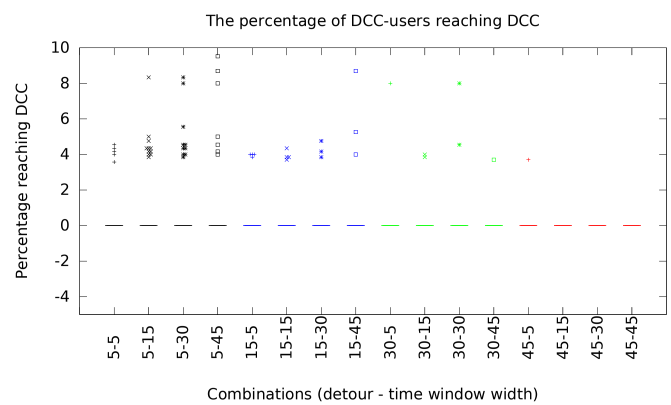
(a) Results of the exhaustive search method with the minimising stuck locations scoring function.



(b) Results of the exhaustive search method with the minimising chartered bus costs scoring function.



(c) Results of the two-stage heuristic (DCC) with the minimising chartered bus costs scoring function.

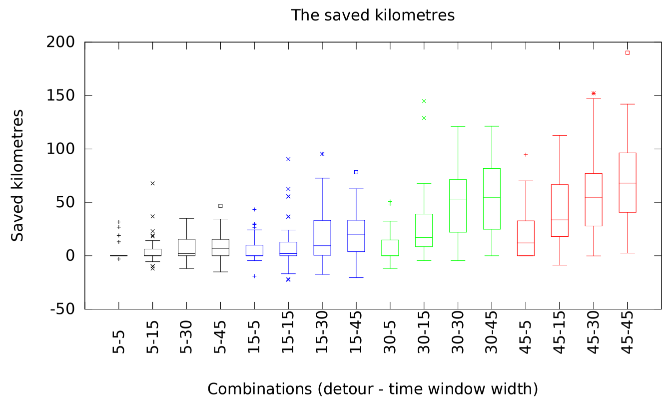


(d) Results of the two-stage heuristic (transfer locations) with the minimising chartered bus costs scoring function.

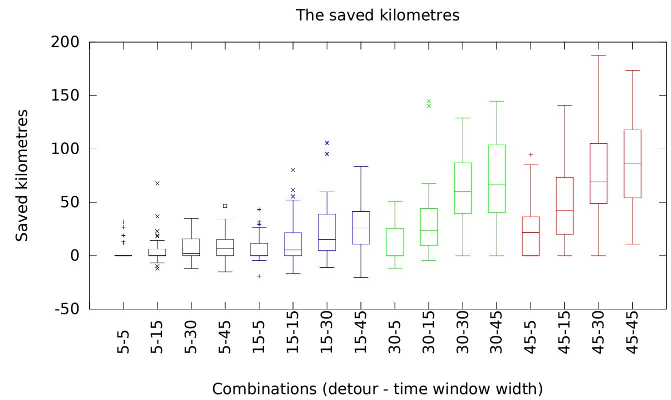
Figure 8: An overview of the results of the percentage of DCC guests actually reaching the DCC by using voluntary drivers.

One can immediately observe a low percentage of DCC guests reaching the DCC by use of volunteers in Figure 8d. This is expected: the heuristic does not aim to bring as many DCC guests to the DCC by volunteers, but aims to minimise the number of stuck locations. Nevertheless, the saved amount of kilometres is comparable with the other simulations as will be discussed later on. For the other three figures, similar results can be observed. There is a large dispersion within the results of each combination of MDT and TWW. For one particular combination even up to 80 % can reach the DCC by volunteers. However in the same combination, there are cases with 15 % as well. One average decent results can be found with a MDT of 30 minutes and TWWs from 30 minutes on. In that particular case, on average 30 % of the DCC guests reach the DCC by using volunteers.

Another interesting measurement is the total saved amount of kilometres; this is a combination of the number of saved chartered buses and the amount of kilometres they travel. These results can be seen in Figure 9.



(a) Results of the exhaustive search method with the minimising stuck locations scoring function.



(b) Results of the exhaustive search method with the minimising chartered bus costs scoring function.

(c) Results of the two-stage heuristic (DCC) with the minimising chartered bus costs scoring function.

(d) Results of the two-stage heuristic (transfer locations) with the minimising chartered bus costs scoring function.

Figure 9: An overview of the results of the saved amount of kilometres for the chartered vehicles.

On overall, one can see that the exhaustive search method and the two-stage heuristic (DCC) have similar results. The two-stage heuristic (DCC) and the two-stage heuristic (transfer location) are performing similarly although almost no DCC guests reach the DCC by carpooling. When taking a closer look at Figure 9b, it is clear that this method is the best. In best cases, up to 200 km can be saved, but on average in realistic cases it will be between 75 km and 100 km. Note that values can have negative results as well. This could happen when more vehicles were needed. This could be the case when particular DCC guests were picked up, which made the route of the chartered buses less efficient. The reference value is the situation were no voluntary drivers are present.

A more detailed overview can be found in Table 3. For the percentage of DCC guests reaching the DCC by volunteers, similar results can be observed for $S1$, $S2$ and $S3$. The results for $S1$ are slightly better because the used scoring function is created to maximise the number of DCC guests reaching the DCC. Results for $S4$ are disastrous for this measurement, but it was expected since the method is not aimed at reaching the DCC by volunteers. The next measurement is about the number of stuck locations. All experiment types have similar results regarding this measurement. In this case, it is hard to find out which experiment type outperforms the others. As can be seen in Table 2, on average the simulations start at around 24 stuck locations. Depending on the MDT and TWW variables, a reduction of up to ten locations can be achieved. Again for the number of buses and the amount of driven kilometres per bus, the four experiment types perform similarly. By comparing it with Table 2, one can see that in many cases up to two vehicles can be saved.

Table 3: Overview of the results. $S1$ represents the exhaustive search method with the minimising stuck locations scoring function, $S2$ represents the exhaustive search method with the minimising chartered bus costs scoring function, $S3$ represents the two-stage heuristic (DCC) with the minimising chartered bus costs scoring function and $S4$ represents the two-stage heuristic (transfer locations) with the minimising chartered bus costs scoring function. Grey cells indicate that at least one of simulations were stopped after an hour of computation. Column headers indicate the combination of maximum detour time (MDT) and time window width (TWW), the percentage of DCC guest which reach the DCC with carpooling (% reach. DCC), the number of stuck locations (nStuckLoc.), the number of buses needed for the stuck passengers (nBus) and the average distance per bus to serve the stuck passengers (Avg. dist/bus).

Combination		% Reach. DCC				nStuckLoc.				nBus				Avg. dist/bus			
MDT	TWW	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
5	5	0.94	0.94	0.94	0.43	23.22	23.22	23.22	23.34	5.30	5.30	5.32	5.38	22.97	22.87	22.86	23.10
5	15	3.02	2.94	3.02	1.01	23.10	23.12	23.10	23.52	7.06	7.06	7.06	7.02	27.84	27.83	27.86	28.03
5	30	4.42	3.84	4.34	1.00	22.88	22.98	22.92	23.48	5.86	5.82	5.86	5.90	35.78	35.88	35.75	36.34
5	45	6.34	6.08	6.34	0.86	21.70	21.72	21.88	22.72	4.72	4.70	4.76	4.80	39.49	39.67	39.28	39.63
15	5	4.14	3.81	4.14	0.33	23.16	23.22	23.16	23.78	5.98	5.96	5.96	6.18	22.74	22.69	22.73	22.55
15	15	6.72	4.61	6.80	0.32	22.94	23.32	23.04	23.70	6.80	6.72	6.84	6.74	28.97	28.59	28.82	29.78
15	30	11.10	9.85	10.93	0.25	21.00	21.24	21.34	22.16	5.32	5.20	5.30	5.40	33.88	33.84	34.13	34.80
15	45	16.61	14.09	16.53	0.34	19.58	19.98	19.90	21.26	4.38	4.26	4.38	4.54	38.90	38.38	38.90	38.14
30	5	4.89	4.24	4.89	0.16	23.32	23.48	23.32	23.90	5.80	5.62	5.74	5.96	24.70	24.98	24.67	24.36
30	15	11.52	9.05	11.36	0.16	21.46	21.86	21.54	22.30	6.28	6.04	6.34	6.24	26.80	27.08	26.78	26.95
30	30	27.23	21.42	26.48	0.25	17.04	18.04	17.44	18.92	4.76	4.32	4.90	4.90	32.33	33.26	31.46	33.28
30	45	32.47	26.29	30.55	0.08	16.12	17.04	16.64	17.78	3.86	3.52	4.14	4.02	37.91	37.00	37.07	38.04
45	5	9.05	7.01	9.05	0.08	22.28	22.74	22.30	22.98	5.52	5.30	5.52	5.76	22.93	23.08	23.03	23.12
45	15	17.84	14.70	17.84	0.00	19.84	20.48	19.90	20.88	5.72	5.50	5.82	6.02	27.74	27.00	27.97	27.96
45	30	30.01	25.06	28.44	0.00	16.88	17.80	17.36	18.24	4.72	4.02	4.84	4.92	32.11	32.79	32.40	32.93
45	45	38.87	30.81	36.71	0.00	14.64	15.74	15.24	16.36	3.58	2.98	3.84	3.74	36.50	37.48	36.65	38.24

10. Discussion

The results of the simulations should be carefully interpreted. In this paper, the results were in most of the cases averaged over the number of runs conducted for every combination of maximum detour time (MDT) and time window width (TWW). It was observed that there was a very large dispersion between the results. In some cases even up to 80 % of the DCC guests could reach the DCC without making use of a chartered bus, however in some cases nobody did. The authors suppose that a MDT of 30 minutes in combination with a 30 minute TWW is realistic for a majority of the participants. The idea of adding fixed bus stops resulted in similar results as for the two stage heuristic (DCC). In order to judge both alternatives better, the amount of driven kilometres of the volunteers should be taken into account. It may be that volunteers would drive significantly less kilometres when they can bring passengers to surrounding transfer locations instead of to the DCC. The exhaustive search method gave the best results depending on the goal. Nevertheless, the developed heuristic method approached this method very well, it is faster and results did not differ too much.

Unfortunately, the research could only make use of a very limited dataset. As described in Section 5.2, only data about home locations were available, other information was sampled based on statistical information about Flanders. Being in possession of the additional needed data could give us a very good insight into our solution space. Due to the large dispersion in the results, the average is quite low.

It was decided to keep the MDT and TWW fixed for all the participants. In daily life, this will not be the case, but it was necessary to be able to produce comparable results. If we were to randomly assign different MDTs and TWWs to participants, it would be very hard to interpret the results.

For the cases presented in Section 9.2, the actual calculation of a solution, given the possible paths of a driver took on average 53.97 seconds (remember that simulations were stopped after 1 hour) for the exhaustive search method, while it only took 0.25 seconds to do the same with a heuristic. In order to have better insight in the results, simulations were conducted for which the optimal result could not be found within an hour. This was done by increasing the number of potential DCC guests from 30 to 150. However, similar results were found. The heuristic methods came close to the approximation for the optimal solution found by the truncated exhaustive search method. It became also clear that a larger number of stuck locations did not mean that more buses were needed. This can be a very useful insight for new heuristics. It could be beneficial to distribute DCC guest as much as possible over various transfer locations in order to be picked up later on by a chartered bus.

11. Conclusion and Future Research

In this paper, the impact of two variables (*maximum detour time* and *time window width*) on the solution of the DCC problem was investigated. Results were somewhat disappointing as a solution to a transportation problem in the sense that it is not possible to take more than 50 % of the DCC guests directly to the DCC without using a chartered bus. The possibility to save chartered buses seemed to be hard as well. The data that was sampled is based on real data and should give a more or less accurate view of the situation.

In this paper, data of a DCC in Flanders was used. On average only 24 DCC guests (30 DCC guests sampled with 80 % probability) would visit the DCC every day. One fourth of them had a voluntary driver. Due to the limited search space, most of the cases could be simulated by the exhaustive search method. For these cases heuristics are not really needed. When the number of DCC guests and/or drivers increases, the exhaustive search method fails to complete in many cases within an hour and hence heuristic solutions are very useful. The already developed heuristics approximate the results of the exhaustive search method quite well.

The final conclusion is that the flexibility (and probably also the number of volunteers) will be a major factor with regards to the feasibility of the different solutions. The goal here is finding a set of volunteers together with specific constraints for which the average results remain more or less stable.

In future research the proposed methods can be applied to other use cases such as schools and companies. Since there will be many more passengers, we anticipate that the exhaustive search technique will not be usable. However, research can be conducted to sort the drivers in a specific order, in order to exploit the pruning steps mentioned in Section 8.1 and hence, reducing the execution time. We could also investigate the influence of the number of voluntary drivers. It is clear that if the number of drivers increases, the solution would be better. The idea here is to find out as of which number of drivers results are getting substantially better. Furthermore, research on the division of the costs between the participants should be conducted.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors for the staff members of UHasselt - Hasselt University. The cooperation with Irith Ben-Arroyo Hartman was funded by UHasselt - Hasselt University with BOF-project R-7238, "Methods from graph theory to support sustainable transport".

The authors would like to thank Nadine Smeyers, Mandy Snoeks and Jan Vuurstaek for reviewing this paper and for their valuable comments.

References

[1] Schoolpool, `https://www.schoolpool.be/nl`, accessed: 2017-11-07 (in Dutch).

[2] J. Li, P. Embry, S. P. Mattingly, K. F. Sadabadi, I. Rasmidatta, M. W. Burris, Who chooses to carpool and why? examination of texas carpoolers, Transportation Research Record: Journal of the Transportation Research Board 2021 (2007) 110–117. `doi:10.3141/2021-13`.

[3] R. Buliung, K. Soltys, C. Habel, R. Lanyon, Driving facotrs behind successful carpool formation and use, Transportation Research Record: Journal of the Transportation Research Board 2118 (2009) 31–38. doi:10.3141/2118-05.

[4] Flemish Government, Traject, Research into obstacles and conditions for the successful promotion of carpooling, http://www.mobielvlaanderen.be/studies/carpoolen02.php?a=18 (2006).

[5] Federal Government, Best mobility practices for the residential work at companies and government institutions: Report based on residential work diagnostics 2008, https://mobilit.belgium.be/sites/default/files/downloads/diagnostique_best_practices_nl.pdf (2008).

[6] T. Vanoutrive, E. Van De Vijver, L. Van Malderen, B. Jourquin, I. Thomas, A. Verhetsel, F. Witlox, What determines carpooling to workplaces in belgium: location, organisation, or promotion?, Journal of Transport Geography 22 (Supplement C) (2012) 77 – 86, special Section on Rail Transit Systems and High Speed Rail. doi:https://doi.org/10.1016/j.jtrangeo.2011.11.006.

[7] N. Agatz, A. Erera, M. Savelsbergh, X. Wang, Optimization for dynamic ride-sharing: A review, European Journal of Operational Research 223 (2) (2012) 295 – 303. doi:10.1016/j.ejor.2012.05.028.

[8] J. Park, B. Kim, The school bus routing problem: A review, European Journal of Operational Research 202 (2) (2010) 311 – 319. doi:10.1016/j.ejor.2009.05.017.

[9] M. Boegl, K. F. Doerner, S. N. Parragh, The school bus routing and scheduling problem with transfers, Networks 65 (2) (2015) 180–203. doi:10.1002/net.21589.

[10] A. Rizzoli, R. Montemanni, C. Serra, F. Minori, D. Kehagias, E. Silk, M. Bulgheroni, C. Bonizzi, SocialCar: Open social transport network

for urban approach to carpooling, EU H2020 Project No 636427 Deliverable Deliverable D2.1, SUPSI, EURECAT, MOVENDA, CERTH, LIFTSHARE, Ab.ACUS (Jan. 2016).

[11] Regional Transportation Authority Chicago, Rta trip planner, http://www.rtachicago.org/plan-your-trip Accessed Jul-11 (2017).

[12] F. Van Audenhove, O. Korniichuk, L. Dauby, J. Pourbaix, The Future of Urban Mobility 2.0 : Imperatives to shape extended mobility ecosystems of tomorrow, Tech. rep., Arthur D. Little (Jan. 2014).

[13] P. R. Oxley, M. J. Richards, Disability and transport: A review of the personal costs of disability in relation to transport, Transport Policy 2 (1) (1995) 57 – 65. doi:10.1016/0967-070X(95)93247-V.

[14] K. Roberts, D. Lawton, Financial assistance for families with severely disabled children and transport costs, Children & Society 13 (5) (1999) 333–345. doi:10.1111/j.1099-0860.1999.tb00130.x.

[15] A. Neven, K. Braekers, K. Declercq, G. Wets, D. Janssens, T. Bellemans, Assessing the impact of different policy decisions on the resource requirements of a demand responsive transport system for persons with disabilities, Transport Policy 44 (2015) 48 – 57. doi:10.1016/j.tranpol.2015.06.011.

[16] Flemish Centre for Volunteering, Kilometre fees, http://www.vlaanderenvrijwilligt.be/wetgeving/kosten-en-vergoedingen/kilometervergoedingen/, accessed: 2017-11-29 (in Dutch).

[17] Committee for Education and Equal Rights, Committee for Mobility and Public Works, Knelpuntnota, http://issuu.com/kinderrechten/docs/8797dd86-fdca-4ced-83c8-61a9503c5228/16?e=6593254/5429626 (2013-2014).

[18] M. Dillen, Draft resolution on improving the transportation of students in special education, https://docs.vlaamsparlement.be/docs/stukken/2013-2014/g2402-1.pdf, (in Dutch) (2014).

[19] Google maps, https://www.google.be/maps/, accessed: 2017-11-07.

[20] K. Declercq, D. Janssens, G. Wets, Onderzoek Verplaatsingsgedrag Vlaanderen 5.1 (2015-2016) Tabellenrapport, Tech. rep., Universiteit Hasselt, IMOB, Hasselt, Belgium (Dec. 2016).

[21] What is optaplanner?, https://www.optaplanner.org/, accessed: 2017-10-16 (2017).

[22] The graphhopper directions api route planning for your application, https://www.graphhopper.com/, accessed: 2017-10-16 (2017).