Erasmus University Rotterdam

Erasmus School of Economics

Bachelor Thesis

Bachelor^2 Econometrics/Economics

A branch-and-cut algorithm and a branch-and-cut based sweep heuristic for the heterogeneous fleet vehicle routing problem

On the efficiency-equity trade-off of taxation of downtown large vehicle usage

---

Author:
Marten Koole
Student number:
450807mk

Supervisor:
Naut Bulten MSc
Second assessor:
dr. Krzysztof Postek

Abstract

This thesis extends the branch-and-cut algorithm by Achuthan, Caccetta and Hill (2003) for the capacitated vehicle routing problem (CVRP) to two algorithms applicable to solve heterogeneous fleet vehicle routing problems (HVRP). Using these algorithms, this thesis shows that taxation can lead to the usage of smaller vehicles near city centers, which reduces, among others, downtown congestion. However, such taxes lead to excessive costs for delivery companies. Those costs can be offset, for example by subsidizing the purchase of smaller vehicles, to limit unfair government intervention. Meanwhile, this thesis offers new cutting planes for the HVRP, but shows that a branch-and-cut algorithm based on those cutting planes is unable to solve HVRPs within reasonable computation time. Finally, a sweep heuristic based on the CVRP algorithm leads to low computation times, but the optimality gap could not be determined adequately.

July 5, 2020

ERASMUS UNIVERSITEIT ROTTERDAM

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Table of Contents

1 Introduction

Short haul transportation plays a crucial role in the global economy. It is estimated that around €70 billion was spent on last mile transportation in 2016 (Joerss, Schröder, Neuhaus, Klink, & Mann, 2016), which refers solely to delivering parcels from a warehouse. Joerss et al. (2016) also indicate that the industry of last mile transportation expands over time. In such an extensive industry, efficient planning can lead to substantial reductions in terms of costs and greenhouse gas emissions, even if they only lead to minor reductions on a per parcel basis. Hence, finding efficient algorithms to increase the efficiency of parcel delivery can have a massive economic and environmental impact. Examples thereof in earlier research are provided by, among others, Figliozzi (2010).

This thesis focuses on efficient parcel delivery planning in urban environments, since rising urbanisation was found in several studies (Angel, Parent, Civco, Blei, & Potere, 2011; Jiang & O'Neill, 2017). This focus is further motivated by the fact that large cities often have high congestion (Schrank & Lomax, 2009), causing relatively high greenhouse gas emissions (Stopher & Stanley, 2014), as well as other (economic) costs. This thesis shows that government intervention (e.g., via the utilization of taxation) can stimulate the usage of smaller vehicles in urban environments.

Traditionally, urban parcel delivery is often performed using delivery vans. Due to rising environmental awareness and urban congestion, the parcel delivery industry diverges to different modes of transport, such as delivery bicycles, which have smaller transport capacities but experience less congestion and emit less greenhouse gasses than delivery vans. In particular for such small vehicles, vehicle capacity often plays a crucial role in planning delivery routes.

An efficient branch-and-cut algorithm to solve capacitated vehicle routing problems (CVRP) was presented by Achuthan, Caccetta and Hill (2003). In CVRPs, one is given $n-1$ potential clients, each with known locations and demand for a commodity, that must be served from a depot using (at most) m vehicles and tasked to find the minimum cost routes to do so. Each of these vehicles has a capacity to transport the commodity that may not be exceeded.

Since the paper by Achuthan et al. (2003) is over fifteen years old, it is time to re-evaluate the algorithm's efficiency using modern equipment. Furthermore, Achuthan et al. (2003) present an algorithm that relies on a homogeneous delivery vehicle fleet, in which all vehicles have the same capacity. This thesis will relax that assumption and extend the aforementioned branch-and-cut algorithm to heterogeneous fleets, in which varying vehicle capacity plays a major role. The algorithm's relative performance for heterogeneous and homogeneous fleets will then be analyzed.

A further intention of this extension is to analyze the effects of taxation to reduce the usage of large delivery vehicles in or near city centers. In Section 2, the economic background to this problem is defined more extensively.

The computational results presented by Achuthan et al. (2003) are based on 1620 test problems and 24 literature problems. In this thesis, the data will solely be derived from simulated problems, which include most settings applied by Achuthan et al. (2003), while also covering a variety of other settings. The details hereof are outlined in Section 6.

The remainder of this thesis is structured as follows: In Section 2, the economic background to the (potential) influence of taxation of the downtown usage of large vehicles is outlined. In Section 3, the literature related to capacitated and heterogeneous fleet vehicle routing problems is discussed, followed by a translation of these problems into a mathematical model in Section 4. Then, Section 5 presents the techniques implemented to solve the problems, which are simulated according to the details provided in Section 6. Then, the (computational) results are shown in Section 7. Finally, Section 8 provides the conclusions of this thesis along with a discussion thereof.

2 Economic background

2.1 The economics of externalities

Neoclassical economics relies on the assumption that each individual makes his or her choices such as to maximize individual well-being, which is commonly referred to as *utility*. Generally, such choices made by one individual influence the utility of another, be it positively or negatively. Such *externalities* exist in urban traffic too. Particularly, the choice of some individuals and companies to engage in urban traffic induces the negative externalities of congestion (Schrank & Lomax, 2009; Qingyu, Zhicai, Baofeng, & Hongfei, 2007) and discomfort (Ossokina & Verweij, 2005).

Obviously, people and companies can take others' preferences into account, but the aforementioned existence of externalities in urban traffic shows that this is generally insufficient to lead to efficient outcomes. According to the Coase theorem, assigning property rights will lead to an efficient distribution of resources, provided that parties can freely negotiate (Coase, 1960). An alternative thereof is taxation, which forces both individuals and companies to take the effects they have on others into account. Such taxes are known as Pigouvian taxes (Pigou, 1932). This thesis offers an analysis of the effects induced by governments imposing such taxes on companies driving with large vehicles in or near the city center, as opposed to having equal costs of driving with larger or smaller vehicles to the preference of the company. On the one hand, such taxes can contribute to achieving more efficient infrastructure usage in or near the city center as they likely lead to a reduction of negative externalities. On the other hand, such taxes can be (perceived) 'unfair' if the magnitude thereof is too large due to which the government massively obstructs the free market. The focus on traffic in or near the city center is explained in Section 2.2.

2.2 Varying urban density

A typical city has higher density near its center(s) than in its suburbs. This principle can be explained from basic economic theory, such as the Von Thünen model which states that land near the city center is more productive than land further away from the city center (Warf, 2010). Furthermore, it can be explained from Christaller's theory, which states that different goods and services require service areas of different sizes. It can also be explained via polarization theory and arguments related to agglomeration benefits, since all of these theories show that it is beneficial for companies to be clustered together in a central business district (CBD) (Wegener, 2013).

Related to these models, it can be argued that land is used at high density near the CBD and other business districts while land close to the edge of cities can be dedicated to (large) suburban gardens as it has a lower opportunity cost. This leads to an inverse relationship between distance from the (central) business district and urban density.

Partially due to the high densities in the city centers, the downtown infrastructure is not built as to accommodate high-intensity parcel delivery with larger delivery vans. For example, in the Dutch city Delft (with approximately 100,000 citizens), it is seemingly infeasible to arrange down town parcel delivery using delivery vans. Also in Rotterdam, a Dutch city with over half a million citizens, parcel delivery near the city center is frequently performed using delivery bicycles rather than delivery vans. Contrarily, it seems acceptable to utilize delivery vans for suburban parcel delivery. As the primary reason thereof, the urban density in the suburbs is likely to be substantially lower than in the CBD, as argued above. Therefore, the infrastructure is likely to be less occupied. This is amplified by the fact that the infrastructure is used by individual households rather than by organized business people.

When multiple distinct types of vehicles are used for parcel delivery, this leads to the Heterogeneous Fleet Vehicle Routing Problem (HVRP). The HVRP is defined similarly to the CVRP, except for a set of vehicle types with a distinct capacity for each vehicle type. In the HVRP, all customers' demand must be satisfied using (at most) m_k vehicles of type k.

3 Literature review

3.1 Capacitated vehicle routing problem

The capacitated vehicle routing problem (CVRP) is an NP-hard problem, since it is an extension of the NP-hard Traveling Salesman Problem. For the CVRP, many formulations exist. Some rely on flow conservation and subtour elimination constraints of some sort (e.g., Achuthan et al., 2003), whereas others rely on a modification of the Miller–Tucker–Zemlin-constraints (MTZ-constraints). An example thereof is provided by Solano-Charris, Prins, and Santos (2015). Finally, a set-covering formulation for the CVRP is often used (e.g., Balinski and Quandt, 1964).

Furthermore, there exists a diversity of exact and heuristic solution methods for the CVRP. Laporte (2009) offers a clear overview thereof. He points to exact branch-and-bound or dynamic programming algorithms as well as a diversity of heuristics, including the savings heuristic (Paessens, 1988), sweep heuristics and some local search (meta)heuristics. Another example is provided by Ralphs, Kopman & Pulleyblank (2003), who provide a branch-and-cut algorithm to solve the CVRP.

3.2 Heterogeneous fleet vehicle routing problem

The heterogeneous fleet vehicle routing problem (HVRP) extends the CVRP by allowing route scheduling of vehicles with varying capacities. Therefore, the HVRP is also NP-hard. Also for the HVRP, multiple formulations exist. Some formulations are based on flow conservation and subtour elimination constraints, while others rely on a modification of the MTZ-constraints (see, for example, Yaman, 2006). Apart from that, a variety of heuristics exist, most of which are based on local search or tabu search procedures (e.g., Gendreau, Laporte, Musaraganyi, and Taillard, 1999; Penna, Subramanian, and Ochi, 2013). Also, there exists a sweep heuristic for the HVRP (Renaud & Boctor, 2002), but it does not account for varying relative edge weights. Taxation of the usage of large vehicles in and near the city center would, in fact, ask for such edge weights to be taken into account in the heuristic.

Exact algorithms

Most closely related to this research are the diverse branch-and-cut algorithms for the HVRP presented in the literature. Yaman (2005) introduced diverse inequalities based on the HVRP's MTZ-formulation, but he did not implement a branch-and-cut algorithm. Others derived a branch-and-cut algorithm for which the cuts were based on feasibility requirements (see, for example, Baldacci, Battarra, and Vigo, 2009). Furthermore, the literature offers diverse column generation implementations to find lower bounds to the HVRP (e.g., Choi and Tcha, 2007), or to derive branch-and-price-and-cut algorithms (see, for example, Betinelli, Ceselli, and Righini, 2011; Pessoa, Sadykov, and Uchoa, 2018; Pessoa, Uchoa, and Poggi de Aragão, 2009). Notably, none of these algorithms rely on cuts based on optimality considerations, which are, in fact, offered by Achuthan et al. (2003) for the CVRP. Rather, feasibility cuts are applied. To the best of my knowledge, there do not exist exact algorithms for the HVRP that use cutting planes based on optimality considerations. This thesis fills that gap in the literature. To that extent, the general purpose of cutting planes is explained in Section 4, where new cutting planes for the HVRP are introduced.

4 Model

This section offers the mathematical model and cutting planes used in the branch-and-cut algorithm, which is introduced in Section 5. Throughout this section, the mathematical model and cutting planes for the heterogeneous fleet vehicle routing problem are taken as the basis. The mathematical model and cutting planes for the capacitated vehicle routing problem are simply a subclass thereof in which only one type of vehicle is available. Nonetheless, the cutting planes and constraints are also provided for the CVRP as to increase the readability of the remainder of this thesis.

4.1 Mathematical model

As indicated above, many formulations for the HVRP and CVRP exist. In analogy to the model for the CVRP used by Achuthan et al. (2003), define graph $G = (V, E)$, in which $V = \{0, \dots, n-1\}$ and $E = \{(i, j) | 0 \leq i \neq j \leq n-1\}$. Furthermore, $\mathcal{L} = \{1, \dots, n-1\} \subset V$ is used to represent the set of customers, each associated with a given demand q_j. Finally, 0 is used to denote the depot.

Now, the set of available vehicle types is denoted by $K = \{0, 1, \dots, t-1\}$ and for each vehicle $k \in K$, m_k represents the number of available delivery vehicles of that type. Also, each vehicle has a capacity $Q_k \in \mathbb{N}$ to transport the commodity. Here, it is required that $\max\{Q_k\} \geq \max\{q_j\}$ in order to preserve that all customers can be brought their demand by some vehicle. Finally, $\ell_k(S)$ denotes the greatest lower bound on the number of vehicles of type $k \in K$ required to visit all locations of S in an optimal solution, for $S \subseteq \mathcal{L}$ such that $\max_{j \in S} q_j \leq Q_k$, for a vehicle of type k cannot be used to serve customers with a demand larger than Q_k.

For the CVRP, $|K| = 1$ and the subscript k is redundant for m_k, Q_k and $\ell_k(S)$.

For any pair of vertices $i, j \in V$ and each vehicle $k \in K$, the 'relative' distance between i and j using vehicle k is denoted by c_{ijk}. These distances are assumed to satisfy the triangle inequality. That is, it is assumed that $c_{ijk} \leq c_{ihk} + c_{hjk},\ \forall k \in K; i, j \in V$. For the CVRP, the subscript k is again redundant. Now, the HVRP relies on the following decision variables for $i < j$ and $k \in K$:

$$x_{ijk} = \begin{cases} 1 & \text{if a vehicle of type } k \text{ visits both } i \text{ and } j \text{ on a single trip;} \\ 2 & \text{if } i = 0 \text{ and } (0, j, 0) \text{ is a route for a vehicle of type } k\text{;} \\ 0 & \text{otherwise.} \end{cases}$$

Since a pair of customers cannot be served by the same vehicle of type k if their joint demand exceeds Q_k, these decision variables are only defined when $q_i + q_j \leq Q_k$. Furthermore, the binary decision variables y_{ik} are introduced, indicating whether a vehicle of type k serves customer $i \in \mathcal{L}$.

Apart from the notation introduced above, $\overline{S}$ is used to denote the complement of S with respect to $\mathcal{L}$. Hence, $\overline{S} = \mathcal{L} \backslash S$. Also, $E[S, T]$ is used to denote the total edge flow between S and T. More formally, $E[S, T] = \{(i, j) | i < j \text{ and } i \in S, j \in T\} \cup \{(i, j) | i < j \text{ and } j \in S, i \in T\}$. An overview of all notation used throughout this thesis is provided in Appendix A.

These definitions bring about the Integer Linear Program (ILP) formulation introduced below, which can be interpreted in analogy to the CVRP formulation provided by Achuthan et al. (2003):

$$\underset{x}{\text{minimize}} \quad \sum_{k \in K} \sum_{i < j \in V} c_{ijk} x_{ijk} \tag{1a}$$

$$\text{subject to} \quad \sum_{j \in \mathcal{L}} x_{0jk} = 2m_k \qquad \forall k \in K \tag{1b}$$

$$\sum_{i < h \in V} x_{ihk} + \sum_{j > h \in V} x_{hjk} = 2y_{hk} \qquad \forall h \in \mathcal{L}, \quad \forall k \in K \tag{1c}$$

$$\sum_{k \in K} y_{ik} = 1 \qquad \forall i \in \mathcal{L} \tag{1d}$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - \ell_k(S) \qquad \forall S \subseteq \mathcal{L} \text{ for which } 3 \leq |S| \leq n-1, \quad \forall k \in K \tag{1e}$$

$$x_{0jk} \in \{0, 1, 2\} \qquad \forall j \in \mathcal{L}, \quad \forall k \in K \tag{1f}$$

$$x_{ijk} \in \{0, 1\} \qquad \forall i, j \in \mathcal{L}, \quad \forall k \in K \tag{1g}$$

$$m_k \in \mathbb{N} \qquad \forall k \in K \tag{1h}$$

$$y_{ik} \in \{0, 1\} \qquad \forall i \in \mathcal{L}, \quad \forall k \in K \tag{1i}$$

The objective (1a) of this Integer Linear Program (ILP) is to minimize the total (weighted) distance driven. Constraints (1b) preserve that all vehicles used in a feasible solution of the HVRP enter and leave the depot, whereas Constraints (1c) and (1d) jointly guarantee that each other node is entered and left exactly once by a vehicle of the same type. Constraints (1e) preserve that subtours cannot exist and that the vehicle capacity is not exceeded by limiting the number of connections within $S \subseteq \mathcal{L}$. This guarantees that the Constraints (1b) can only be satisfied if a sufficient number of vehicles is used and there are (eventually) connections to the depot. The last four sets of Constraints (1f-1g) are required to preserve the variable domains.

In this formulation, m_k can be variable or fixed by additional constraints for all vehicle types. In this thesis, however, m_k is variable throughout. Furthermore, $l_k(S)$ needs only be calculated in the cases where it is required[1] (Achuthan et al., 2003). It can be computed as $\ell_k(S) = \left\lceil \frac{\sum_{j \in S} q_j}{Q_k} \right\rceil$.

For the CVRP, this model simplifies to the model provided by Achuthan et al. (2003):

$$\begin{aligned}
\underset{x}{\text{minimize}} \quad & \sum_{i<j \in V} c_{ij} x_{ij} && (2a)\\
\text{subject to} \quad & \sum_{j \in \mathcal{L}} x_{0j} = 2m && (2b)\\
& \sum_{i<h \in V} x_{ih} + \sum_{j>h \in V} x_{hj} = 2 \quad \forall h \in \mathcal{L} && (2c)\\
& \sum_{i,j \in S} x_{ij} \leq |S| - \ell(S) \qquad \forall S \subseteq \mathcal{L} \text{ for which } 3 \leq |S| \leq n-1 && (2d)\\
& x_{0j} \in \{0,1,2\} && (2e)\\
& x_{ij} \in \{0,1\} \qquad \text{for } i \in \mathcal{L} && (2f)\\
& m \in \mathbb{N}^+ && (2g)
\end{aligned}$$

For both models, specific vehicles are not assigned to routes because vehicles of the same type can be used interchangeably.

In model (1), the subtour elimination constraints (1e) can be replaced by Constraints (3). A proof of this equivalence is provided in Appendix C.

$$\sum_{h \in K} \sum_{i,j \in S} x_{ijh} - \sum_{h \neq k \in K} \sum_{i,j \in \overline{S}} x_{ijh} + \sum_{h \in K} \left(\sum_{j \in S} x_{0jh} - m_h \right) \leq |S| - l(\overline{S}) \quad \forall S \subseteq \mathcal{L} \text{ for which } 1 \leq |S| \leq n-2, \ \forall k \in K \tag{3}$$

[1] Note that the problems introduced by Achuthan, Cacetta and Hill (1996) do not apply here, because there is no limit on the length of routes driven by vehicles.

For the CVRP, these Constraints reduce to (4), which can be used to replace Constraints (2d):

$$\sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{0j} - m \leq |S| - l(\overline{S}) \quad \forall S \subseteq \mathcal{L} \text{ for which } 1 \leq |S| \leq n-2 \tag{4}$$

Both sets of constraints limit the number of edges within a subset to the number of nodes thereof plus the number of vehicles available to service that subset, which are all vehicles that need not be reserved to deliver parcels in $\overline{S}$. It follows from elementary graph theory that disconnected cycles cannot exist with this number of edges for $S \subseteq \mathcal{L}$. Furthermore, since the equation holds for all such subsets S, the capacity of vehicles is not exceeded. For the HVRP, the interpretation is similar but slightly more complicated due to the additional terms that appear in the equation.

4.2 Cutting planes

Apart from the aforementioned sets of constraints, Achuthan et al. (2003) introduce cutting planes for the CVRP. Such cutting planes are valid inequalities that can be added to the model with the purpose of reducing the size of the search tree and hence, of boosting the performance of the branch-and-cut algorithm as compared to a 'standard' branch-and bound algorithm. Namely, such cutting planes reduce the problem's feasible region, without eliminating all optimal solutions.

For each of the aformentioned cutting planes for the CVRP (Achuthan et al., 2003), there exists similar cutting planes for the HVRP that can be interpreted analogously. These cutting planes are presented in this section and a proof of the validity thereof is presented in Appendix C.

First, Achuthan et al. (2003) argue that it can occur that a Constraint of type (1e) (2d for the CVRP) is not violated for a set $S \subset \mathcal{L}$ while it is violated for $P \subset S$. In such a case, the following provides a valid cutting plane for $S \backslash P$:

$$\sum_{h \in K} \sum_{i,j \in P} x_{ijh} - \sum_{h \neq k \in K} \sum_{i,j \in S \backslash P} x_{ijh} + \sum_{h \in K} \left(\sum_{i,j \in S} x_{ijh} + \sum_{j \in P} x_{0jh} + \sum_{E(P,\overline{S})} x_{ijh} \right) \leq t|S| + |P| - \ell_k(S \backslash P) \tag{5}$$

For the CVRP, this cutting plane reduces to the following:

$$\sum_{j \in P} x_{0j} + \sum_{i,j \in P} x_{ij} + \sum_{i,j \in S} x_{ij} + \sum_{E[P,\overline{S}]} x_{ij} \leq |S| + |P| - \ell(\overline{P}) \tag{6}$$

These sets of cutting planes can be interpreted fairly similarly to Constraints (3) and (4), respectively. Namely, consider S as the set of customers to be served from the 'virtual depot', which is everything outside of S. Then, a valid inequality for P can be derived, similar to Constraints (3) and (4). These inequalities can be rewritten into Cutting Planes (5) and (6), respectively. For that reason, these cutting planes can be interpreted in analogy to Constraints (3) and (4).

For the CVRP, Achuthan et al. (2003) provide a further set of valid inequalities, which is outlined in Theorem 1 below, along with its HVRP equivalent that is proven in Appendix C:

Theorem 1. *For any vehicle type $k \in K$, let $D_0, D_1, D_2, \ldots, D_r \subseteq \mathcal{L}$ with the properties 1 - 3:*

1. *$r \geq 2$ and $\sum_{i \in D_0 \cup D_p \cup D_q} q_i > Q_k \; \forall p, q \in \{1, \ldots, r\}$ such that $p \neq q$;*
2. *$\forall i, j \in \{0, \ldots, r\}$ such that $i \neq j$, D_i and D_j are disjoint. That is, $D_i \cap D_j = \emptyset$;*
3. *$D = \cup_{i=1}^{r} D_i$.*

Then, any feasible solution of the HVRP satisfies:

$$3 \sum_{i,j \in D_0} x_{ijk} + \sum_{E[D_0,D]} x_{ijk} + \sum_{p=1}^{r} \sum_{i,j \in D_p} x_{ijk} \leq 3|D_0| + |D| - (r+2) \tag{7}$$

For the CVRP, this simplifies to:

$$3 \sum_{i,j \in D_0} x_{ij} + \sum_{E[D_0,D]} x_{ij} + \sum_{p=1}^{r} \sum_{i,j \in D_p} x_{ij} \leq 3|D_0| + |D| - (r+2) \tag{8}$$

This theorem relies on the fact that there cannot be cycles in the subgraph of G that is defined by the node set D_0 and the edges chosen accordingly, because of the subtour elimination constraints. Furthermore, it uses the property that the number of cross-edges from D to D_i is limited as a consequence of Constraints (1d) (2c for the CVRP), for $i \in \{1, \ldots, r\}$. This leads to an effective upper bound on the number of traversed edges in $D_0 \cup D$.

For $r = 3$, Cutting Planes (7) can be rewritten into Cutting Planes (9):

$$2 \sum_{i,j \in D_0} x_{ijk} + \sum_{E[D_0,D]} x_{ijk} + \sum_{p=1}^{3} \sum_{i,j \in D_p} x_{ijk} \leq 2|D_0| + |D| - 4 \;\; \forall k \in K \tag{9}$$

Similarly, for the CVRP, Cutting Planes (8) simplify to (10), as provided by Achuthan et al. (2003):

$$2 \sum_{i,j \in D_0} x_{ij} + \sum_{E[D_0,D]} x_{ij} + \sum_{p=1}^{3} \sum_{i,j \in D_p} x_{ij} \leq 2|D_0| + |D| - 4 \tag{10}$$

All cutting planes presented thus far are based on the required feasibility of the CVRP and HVRP. Apart from these cutting planes, Achuthan et al. (2003) also present two sets of cutting planes that are based on optimality considerations. Namely, they show that there exists an optimal solution to the CVRP that satisfies the criteria in Cutting Planes (12) and (13). These are introduced below along with their respective HVRP equivalents, Cutting Planes (11) and (15).

For the HVRP, there exists an optimal solution which satisfies:

$$\sum_{h \leq k \in K} \left(\sum_{i,j \in S} x_{ijh} + \sum_{j \in S} x_{0jh} \right) \leq |S| + 1 \ \ \forall k \in K, \ \forall S \subseteq \mathcal{L} \text{ such that } 2 \leq |S| \leq |\mathcal{L}| \text{ and } \sum_{i \in S} q_i \leq Q_k \tag{11}$$

This reduces to the following for the CVRP:

$$\sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{0j} \leq |S| + 1 \ \ \forall S \subseteq \mathcal{L} \text{ such that } 2 \leq |S| \leq |\mathcal{L}| \text{ and } \sum_{i \in S} q_i \leq Q \tag{12}$$

Namely, if this were not the case, there must be multiple vehicles that start and end at the depot and that visit only customers in S. Therefore, a cost reduction can be achieved by merging these vehicles' routes. This leads to a feasible solution to the CVRP because the total demand in S does not exceed the vehicle capacity and to a cost deduction because the distances satisfy the triangle inequality for all vehicle types.

Furthermore, Achuthan et al. (2003) prove that there exists an optimal solution of the CVRP with variable m which satisfies Cutting Planes (12) and the condition in equation (13).

$$\sum_{i,j \in S} x_{ij} + \sum_{i \in S} x_{0i} \leq |S| + \left\lfloor \frac{2(\sum_{i \in S} q_i + \delta)}{Q + 1 + \delta} \right\rfloor \ \ \forall S \subseteq \mathcal{L} \text{ such that } 2 \leq |S| \leq |\mathcal{L}| \text{ and } \sum_{i \in S} q_i > Q.$$

$$\text{Furthermore, } \delta = 0 \text{ if Q is odd or } \delta = 1 \text{ if Q is even.} \tag{13}$$

This equation follows from algebraic manipulation of a solution that satisfies equation (12). For the cases in which m is variable, Cutting Planes (13) lead to an upper bound on m (Achuthan et al., 2003). Namely:

$$m \leq \begin{cases} 1, & \text{if } \sum_{i \in \mathcal{L}} q_i \leq Q \\ \min\left(n, \left\lfloor \frac{2(\sum_{i \in S} q_i + \delta)}{Q + 1 + \delta} \right\rfloor \right) & \text{otherwise.} \end{cases} \tag{14}$$

In equation (14), δ is defined as in Cutting Planes (13). Along with this upper bound, a lower bound on m can be introduced, which is equal to $\left\lceil \frac{\sum_{i \in \mathcal{L}} q_i}{Q} \right\rceil$.

Analogously, for the HVRP, there exists an optimal solution which satisfies Cutting Planes (11) and the Cutting Planes presented in (15), for all vehicle types $k \in K$:

$$\sum_{i,j \in S} x_{ijk} + \sum_{i \in S} x_{0ik} \leq |S| + \left\lfloor \frac{2(\sum_{i \in S} q_i + \delta)}{Q_k + 1 + \delta} \right\rfloor \ \ \forall S \subseteq \mathcal{L} \text{ such that } 2 \leq |S| \leq |\mathcal{L}| \text{ and } \sum_{i \in S} q_i > Q_k.$$

$$\text{Furthermore, } \delta = 0 \text{ if } Q_k \text{ is odd or } \delta = 1 \text{ if } Q_k \text{ is even.} \tag{15}$$

Again, these cutting planes lead to upper bounds on variable m_k. However, these upper bounds are not as strong as they are for the CVRP in general, since fewer vehicles of one type will be used if vehicles of another type are used. Nonetheless, the following set of upper bounds can be found:

$$m_k \leq \begin{cases} 1, & \text{if } \sum_{i \in \mathcal{L}} q_i \leq Q_k \\ \min\left(n, \left\lfloor \dfrac{2(\sum_{i \in S} q_i + \delta)}{Q_k + 1 + \delta} \right\rfloor\right) & \text{otherwise.} \end{cases} \tag{16}$$

In equation (16), δ is defined as in Cutting Planes (15). As mentioned above, this upper bound is less tight than the upper bound for the CVRP. A similar argument goes for the lower bound. As vehicles of different types can be used interchangeably, it is possible to use only vehicles of the largest type and no other vehicle. In certain cases, it is also possible to use no vehicles of the largest type. Therefore, the lower bound on m_k must be set to its trivial value, 0, for all vehicle types.

5 Methodology

5.1 Branch-and-cut algorithm

Throughout this section, a branch-and-cut algorithm for the HVRP is outlined, which simplifies to a corresponding branch-and-cut algorithm for the CVRP for $|K| = 1$. The algorithm for the CVRP is the algorithm provided by Achuthan et al. (2003), whereas the algorithm for the HVRP is an extension thereof. Both the structure of the algorithm in its entirety as well as the structure of its components must be attributed to the work of Achuthan et al. (1996, 2003), or even to Laporte et al. (1985). The modifications made to that algorithm are first presented in this thesis.

Generally, branch-and-cut algorithms extend branch-and-bound algorithms in the sense that cutting planes are used to reduce the number of branches needed. In particular, pseudocode for this branch-and-cut algorithm is presented in Algorithm 1 in Appendix B or as a flowchart by Achuthan et al. (1996). In this algorithm, Z^* is used to denote the current best-known objective value and $\overline{Z}$ denotes the objective value of the current linear programming relaxation.

Fixing forced variables

Evidently, for each subproblem there can exist constraints on edges that must or may not be included in the solution, for example because the corresponding variables x_{ijk} are fixed due to branches that were created in the algorithm's branching fase. Similarly, there may exist constraints on the binary decision variables y_{ih}, forcing them to be equal to 0 or 1.

Both for the HVRP and for the CVRP, requiring one or multiple edges to be included in a subproblem's solution can force other edges to be excluded. Such situations are incorporated as subtour prevention constraints in the following manner:

For each vehicle type $k \in K$, consider the graphs $H = (V_H, E_H)$ and $H' = (V_{H'}, E_{H'})$ such that $V_H = V_{H'} = V$, $E_H = \{(i,j)|x_{ijk} \text{ is a free variable}\}$ and $E_{H'} = \{(i,j)|x_{ijk} = 1\}$. By definition, H' is a union of paths. Now, let $P_1 = \{i_1, \ldots, i_r\}$ and $P_2 = \{j_1, \ldots, j_s\}$ be any pair of paths in H'.

- If $\sum_{u=1}^{r} q_{i_u} + \sum_{u=1}^{s} q_{j_u} > Q_k$, any free variable involving the end vertices of P_1, P_2 and vehicle k is set to 0, since one vehicle of type k cannot serve all customers in these paths.

- No vehicle $h \neq k$ can visit any vertex $i \in P_1 \cup P_2$ unless i is the depot. That is, $y_{ih} = 0$ for all $h \neq k \in K$. As a direct consequence thereof, it follows that $y_{ik} = 1$ by Constraints (1d). Furthermore, all edges involving any vertex of P_1 or P_2 are forbidden to be driven with any vehicle $h \neq k \in K$, unless the vertex is the depot.

- If $r \geq 3$, the edge (i_1, i_r) cannot be driven with vehicle k unless i_1 or i_r is the depot and if $s \geq 3$, (j_1, j_s) is forbidden to be driven with vehicle k unless j_1 or j_s is the depot.

- Any free variable involving internal vertices of P_1 or P_2 is set to 0.

Even though none of these situations can arise in the presence of the constraints presented in model 1 (or model 2 for the CVRP), it is computationally more efficient to force the corresponding variables to be equal to 0 ex ante (Achuthan et al., 1996).

Finally, if a vehicle of type k visits customer i (i.e., $y_{ik} = 1$) for some $i \in \mathcal{L}$, $k \in K$, it is known that no other vehicle can visit this customer. Therefore, y_{ih} is set to 0 for any $h \neq k \in K$. This is a consequence of Constraints (1d).

Purging ineffective constraints

For the case at hand, Constraints 1e and 3 as well as Cutting Planes 5, 7, 9, 11 and 15 (or Constraints 2d and 4 as well as Cutting Planes 6, 8, 10, 12 and 13 for the CVRP) are ineffective if their corresponding slack variable is a basic variable. If that is the case, the constraint is not binding and hence, does not influence the current solution. It is computationally efficient to remove such constraints from the problem, even if they must be reintroduced in a later stage (Achuthan et al., 1996). It may be noted that the Constraints 1b, 1c and 1d (or 2b and 2c for the CVRP) are always tight as they must hold at equality.

Searching capacity constraint violations

For any solution to the current linear program of the HVRP and the CVRP, it must be verified whether none of the constraints or cutting planes are violated. Since constraints are only added when they are violated and removed from the program when they are no longer necessary to maintain, one or multiple constraint violations can exist, after which these violated constraints are added to the (linear) program. The algorithm to find violated constraints and cutting planes is valid both for the HVRP as for the CVRP for which the procedure was originally developed.

Initially, a set $\mathcal{S}_k = \{S_{1k} \dots S_{m_k k}\}$ of the m_k tours in the current solution is defined for each vehicle type. Each of these tours is connected internally but none are connected to other tours, except possibly via the depot. Two points i and j are considered to be connected for vehicle $k \in K$ if $x_{ijk} > 0$ (Achuthan et al., 1996). If such a tour S_{lk} is not connected to the depot, it will violate a Constraint of type 1e (2d for the CVRP) since it must adhere to Constraints 1d and 1c (2c for the CVRP), for all vertices in the tour so that a Constraint of type 1e (2d) can be added for S_{lk}.

Before the (remaining) search procedures are introduced, it is worth noting that Constraints 1e (2d) are equivalent to Constraints 3 (4), as was outlined in Section 4. Nonetheless, searching for violations of both types of constraints can reduce the number of branches substantially (Achuthan et al., 1996). This is similar to the reduction of the number of branches that can be achieved by adding suitable cutting planes, which is the core of any branch-and-cut algorithm. Because the HVRP extends the CVRP, this is assumed to be valid for the HVRP too. Finally, it can be noted that any integer solution for which no violation for Constraints 1e (2d) are found, is feasible. Namely, each tour is then connected to the depot and vehicle capacity is never exceeded. All other constraints and cutting planes are added solely to gain computational efficiency.

For the reason described above, it is only necessary to verify whether a direct violation of Constraints 1e (2d) exists for integer valued solutions. If this is the case, the constraint can be added and it is verified whether the solution is now infeasible, or not. If such a violation does not exist, the solution is feasible and hence optimal for the subproblem. Therefore, the following, more complicated search procedures are only executed if the current solution is fractional.

For tours S_{lk} that are connected to the depot, the primary focus is on direct violations of Constraints 1e, 3 and Cutting Planes 11 (2d, 4 and 12 for the CVRP) for S_{lk} and proper subsets thereof. The following steps are taken to search for such violations for the HVRP, with the CVRP equivalent in parentheses. This is analogous to the procedure defined by Achuthan et al. (2003):

1. A direct violation of Constraints 1e (2d) for S_{lk};
2. A violation of Constraints 1e (2d) for a proper subset of S_{lk} using search 1 as defined below;
3. A direct violation of Constraints 3 (4) or Cutting Planes 11 (12) for S_{lk};
4. A violation of Cutting Planes 5 (6) or 11 (12) for a proper subset of S_{lk} using search 2 as defined below.

If the total demand of all customers in the tour exceeds the capacity of vehicle k, (i.e., $\sum_{j \in S_{lk}} q_j > Q_k$), the order above is adhered to. Otherwise, the order 3, 4, 1, 2 is chosen, because Constraints 1e and 2d relate to the number of vehicles needed to satisfy the demand in S_{lk} and Constraints 3 and 4 to the number of vehicles needed to satisfy the demand of customers that are not in S_{lk}. Therefore, these orderings generally increase the algorithm's computational performance (Achuthan et al., 2003). Furthermore, violated constraints evidently are only added if S_{lk} satisfies the requirements for the validity thereof. Finally, at most one constraint or cutting plane is added for each tour S_{lk}.

Direct violations of any constraint by any tour S_{lk} are easily found by computing the left hand side and right hand side of the constraint. However, it is computationally inefficient to do so for all $R \subset S_{lk}$ since many such subsets will not yield violated constraints. Therefore, search heuristics are implemented, which offer a more adequate trade-off between yielding a reduction in the size of the search tree on the one hand and computation time on the other.

In the remainder of this subsection, six such search procedures are introduced by explaining the intuition behind them. Appendix B offers detailed pseudocode for each of these search procedures.

Search 1 mentioned above considers slack functions $Z_1^{HVRP}(R)$ for $S_{lk} \backslash R \subseteq S_{lk}$, as follows:

$$Z_1^{HVRP}(R) = |S_{lk} \backslash R| - \ell(S_{lk} \backslash R) - \sum_{i,j \in S_{lk} \backslash R} x_{ijk} \tag{17}$$

For the CVRP this reduces to the slack function $Z_1(R)$, as provided by Achuthan et al. (2003):

$$Z_1(R) = |S_{lk} \backslash R| - \ell(S_{lk} \backslash R) - \sum_{i,j \in S_{lk} \backslash R} x_{ij} \tag{18}$$

The search procedure incrementally adds elements from S_{lk} to R to construct a subset $R \subset S_{lk}$. Then, the slack on the Constraint of type 1e (or 2d) is computed for $S_{lk} \backslash R$. If this slack does not decrease after an iteration, the search is terminated since it is unlikely to find a violated constraint (Achuthan et al., 1996) and hence, the computational cost of continuing the search outweighs the expected advantage of reducing the size of the search tree. If negative slack is found after an iteration, the violated constraint is added to the model and the search is terminated.

Search 2 relies on a procedure that is only marginally different from Search 1. In particular, search 2 relies on $Z_2^{HVRP}(R)$ and $Z_3^{HVRP}(R)$, which are the slack functions for Cutting Planes (5) and (11) as defined below:

$$Z_2^{HVRP}(R) = t|S_{lk}|+|R|-\ell_k(S_{lk}\backslash R)-\sum_{h\in K}\left(\sum_{i,j\in R} x_{ij} + \sum_{j\in R} x_{0j} + \sum_{i,j\in S_{lk}} x_{ij} + \sum_{E[R,\overline{S_{lk}}]} x_{ij}\right) + \sum_{h\neq k\in K}\sum_{i,j\in \overline{P}} x_{ijh} \tag{19}$$

$$Z_3^{HVRP}(R) = \begin{cases} |R| + 1 - \sum_{h\leq k\in K}\sum_{i,j\in R} x_{ijh} - \sum_{h\leq k\in K}\sum_{j\in R} x_{0jh} & \text{if } \sum_{j\in R} q_j \leq Q_k; \\ \infty & \text{otherwise.} \end{cases} \tag{20}$$

These slack functions reduce to the slack functions $Z_2(R)$ and $Z_3(R)$ as they were introduced by Achuthan et al. (2003). These slack functions provide the slack on Cutting Planes (6) and (12), respectively.

$$Z_2(R) = |S_{lk}| + |R| - \ell(S_{lk}\backslash R) - \sum_{j\in R} x_{0j} - \sum_{i,j\in R} x_{ij} - \sum_{i,j\in S_{lk}} x_{ij} - \sum_{E[R,\overline{S_{lk}}]} x_{ij} \tag{21}$$

$$Z_3(R) = \begin{cases} |R| + 1 - \sum_{i,j\in R} x_{ij} - \sum_{j\in R} x_{0j} & \text{if } \sum_{j\in R} q_j \leq Q; \\ \infty & \text{otherwise.} \end{cases} \tag{22}$$

Apart from the slack functions, the procedure of search 2 differs from that of search 1, because the search continues to expand R as long as this does not lead to an increase of the slack function. Afterwards, a violated constraint is added if the slack found is negative. Here, a Constraint of type 1e (2d) is added for $S_{lk}\backslash R$ when negative slack is found on Cutting Planes 5 (6), since both equations are equivalent but the former contains fewer terms than the latter (Achuthan et al., 2003). When negative slack is found on Cutting Planes 11 (12), a Cutting Plane of this type is added to the model instead when it has smaller slack than 5 (6).

After steps 1-4 are executed, it is likely that there exist tours for which no violations have been found thus far. Assume, without loss of generality, that this set is $\mathcal{S}'_k = \{S_{1k} \ldots S_{p_k k}\}$, with $p_k < m_k$. Moreover, define the following (non-negative[2]) weight for each $S_{ik} \in \mathcal{S}'_k$:

$$\alpha_k(S_{ik}) = \sum_{h\in K}\sum_{i,j\in S_{ik}} x_{ijh} + \sum_{j\in S_{ik}} x_{0jk} - |S_{ik}| - \ell(S_{ik}) \tag{23}$$

[2] A proof thereof is provided in Appendix C

Again, these weights extend the definition of the weights $\alpha(S_i)$ assigned to $S_i \in \mathcal{S}'$ for the CVRP (Achuthan et al., 2003) which are provided in equation (24):

$$\alpha(S_i) = \sum_{i,j \in S_i} x_{ij} + \sum_{j \in S_i} x_{0j} - |S_i| - \ell(S_i) \tag{24}$$

It is now assumed, without loss of generality, that $\mathcal{S}'_k$ is sorted such that $\alpha_k(S_{ik}) \leq \alpha_k(S_{jk})$, whenever $i > j$. Over this set, a third search procedure is executed. This search procedure is based on $Z_3^{HVRP}(R)$ ($Z_3(R)$) as defined in equation 20 (22) and $Z_4^{HVRP}(R)$ ($Z_4(R)$) as defined below, which represents the slack on Constraints 3:

$$Z_4^{HVRP}(R) = |R| - \ell_k(\overline{R}) - \sum_{h \in K} \left(\sum_{i,j \in R} x_{ijh} + \sum_{j \in R} x_{0j} - m_h \right) + \sum_{h \neq k \in K} \sum_{i,j \in \overline{R}} x_{ijh} \tag{25}$$

This simplifies to the following for the CVRP, representing the slack on Constraints (4) which is also provided by Achuthan et al. (2003):

$$Z_4(R) = |R| + m - \ell(\overline{R}) - \sum_{i,j \in R} x_{ij} - \sum_{j \in R} x_{0j} \tag{26}$$

This search procedure adds entire sets S_{ik} to an initially empty set R. Sets with the largest weight $\alpha_k(S_{ik})$ are likely to have low values $Z_3^{HVRP}(S_{ik})$ and $Z_4^{HVRP}(R)$ since terms that appear with a positive sign in equation (23) appear with negative signs in equations (20) and (25) and vice-versa, which holds for the CVRP analogously. Therefore, this ordering increases the chance to merge sets that jointly provide a violated constraint or cutting plane. As for search 2, this search procedure continues to add sets S_{ik} to R while this does not cause the slack function to increase (Achuthan et al., 1996). If negative slack is found, the constraint or cutting plane that yields the smallest slack is added to the model.

After Search 1, 2 and 3 have been executed for all $S_{ik} \in \mathcal{S}_k$, one can distinguish sets $C_{1k} \dots C_{pk}$ which violated Constraints 1e (2d) and sets $D_{1k} \dots D_{qk}$ which violated Constraints 3 or Cutting Planes 11 (4 and 12, respectively). For these sets, define $\mathcal{W} = \cup_{i=1}^{p} C_{ik}$ and $\mathcal{W}' = \cup_{i=1}^{q} D_{ik}$. If $p \geq 2$, $q \leq 1$ and $|\mathcal{W}| \leq |\mathcal{W}'|$ or if $p, q \geq 2$ and $|\mathcal{W}| \leq |\mathcal{W}'|$, an additional Constraint of type 1e (2d) is added to the model. Likewise, if $p \leq 1$, $q \geq 2$ and $|\mathcal{W}| > |\mathcal{W}'|$ or if $p, q \geq 2$ and $|\mathcal{W}| > |\mathcal{W}'|$, an additional Constraint of type 3 (4) or an additional Cutting Plane of type 11 (12) is added to the model. For the HVRP, a Constraint of type 3 is added if $Z_4^{HVRP}(\mathcal{W}') < Z_3^{HVRP}(\mathcal{W}')$ as defined in equations (20) and (25). A Cutting Plane of type 11 is added otherwise. Likewise, for the CVRP, a Constraint of type 4 is added if $Z_4(\mathcal{W}') < Z_3(\mathcal{W}')$ and a Cutting Plane of type 12 is added otherwise. Here, all constraints are added over $\mathcal{W}$ or $\mathcal{W}'$ in its entirety.

Achuthan et al. (2003) introduce these additional constraints, that need not be violated, for the CVRP because Laporte et al. (1985) have shown that this substantially reduces the size of the search tree. For the HVRP, analogous additional constraints are added to the model. Namely, the HVRP extends the CVRP and hence, it is reasonably likely that the search tree reduction can also be achieved for the HVRP. Note that these constraints can be purged later in the algorithm.

After this point, the search for violated constraints is only continued if no violated constraints were identified thus far, for any of the vehicle types. Otherwise, it is computationally more efficient to resolve the model and re-apply searches 1, 2 and 3 before (potentially) continuing towards searches 4, 5 and 6, which aim to identify further violations of Constraints 1e and 3 and Cutting Planes 5, 7 and 9 (or, for the CVRP, 2d and 4 and Cutting Planes 6, 8 and 10). Violations for Cutting Planes 15 (13) are not incorporated in any search procedure, since Achuthan et al. (2003) noted that this constraint is hardly ever violated for the CVRP. Again, this is assumed to be valid for the HVRP analogously. This is likely to be partially caused by the tight upper bound imposed on the number of vehicles used (m_k) that is introduced in equation 16 (14), which provides a further reason not to search for violations of Cutting Planes 15 (13).

If we continue towards searches 4, 5 and 6 (and hence, no violated constraints were found yet), searches 4 and 5 are executed for all vehicle types $k \in K$ first and search 6 is only executed if searches 4 and 5 do not yield any violated constratins for any of the vehicle types. Now, define $x_{\alpha\beta k} = \max\left[x_{ijk} | i < j \in \mathcal{L}\right]$. If this maximum is attained for multiple pairs of customers, the pair with the largest aggregate demand is chosen. Search 4 is executed if $x_{\alpha\beta k} = 1$ and Search 5 is executed otherwise. If either of these searches fails to introduce a violated constraint, the other search procedure is executed, in analogy to search procedures 4 and 5 for the CVRP as defined by Achuthan et al. (2003).

Search 4 relies on another definition of the slack[3] on Constraints 1e for $R \subset \mathcal{L}$:

$$Z_1'^{HVRP}(R) = \sum_{h \in K} \left(\sum_{j \in R} x_{0jh} + \sum_{E[R,\overline{R}]} x_{ijh} \right) + 2 \sum_{h \neq k \in K} \sum_{i,j \in R} x_{ijh} - 2\ell_k(R) \tag{27}$$

For the CVRP, this slack function simplifies to the following (Achuthan et al., 2003):

$$Z_1'(R) = \sum_{j \in R} x_{0j} + \sum_{E[R,\overline{R}]} x_{ij} - 2\ell(R) \tag{28}$$

This search procedure relies on the argument that a subtour elimination constraint over two sets that are directly connected, meaning that the edge flow between them sums to (at least) 1, is as

[3] The validity thereof is shown in Appendix C

valid as two seperate subtour elimination constraints (Crowder & Padberg, 1980). Following this argument, the search defines adequate sets D'_l over which a potential violation of such subtour elimination constraints must be verified. However, to preserve the validity of this argument, it is required that both sets are directly connected by vehicles of the same type.

If search 4 fails to produce any violated constraint for the CVRP, $\sum_{i,j\in D'_l} x_{ij} = |D'_l| - 1$ for all D'_l (Achuthan et al., 2003). The HVRP equivalent reads: $\sum_{i,j\in D'_l} x_{ijk} = |D'_l| - 1$, which is also valid. If this were not the case, either $\sum_{i,j\in D'_l} x_{ijk} < |D'_l| - 1$ or $\sum_{i,j\in D'_l} x_{ijk} > |D_i| - 1$. In the first case, a merge occurred that was not allowed since the internal edge flow must be at least equal to the number of merges that occurred, which is $|D'_l| - 1$. In the second case, there must be a subtour so that a Constraint of type 1e was violated. For that reason, $D' = \{D'_l\}$ as the output of search 4 can be taken as input to search 5. Otherwise, search 5 is initiated with $D_i = \{i\}, \forall i \in \mathcal{L}$.

The fifth search procedure merges sets with positive edge flows for vehicle type k until its capacity is exceeded. Namely, if there is a positive edge flow between customers, they must be served using the same vehicle and hence, the aggregate demand cannot exceed the vehicle capacity.

At this point, it is again evaluated whether constraints have been added to the model. If so, the linear program is resolved. Otherwise, a final search is performed. This search aims to find violations for Cutting Planes 11 (12) or Constraints 3 (4). To this extent, different slack functions for Cutting Planes 11 (12) and Constraints 3 (4) are introduced[4]:

$$Z'^{HVRP}_3(R) = \begin{cases} \sum_{h\in K}\sum_{E(R,\overline{R})} x_{ijh} - \sum_{h\leq k\in K}\sum_{j\in R} x_{0jh} + \sum_{h>k\in K}\left(2\sum_{i,j\in R} x_{ijh} + \sum_{j\in R} x_{0jh}\right) + 2 & \text{if } \sum_{i\in R} q_i \leq Q_k \\ \infty & \text{otherwise} \end{cases} \quad (29)$$

$$Z'^{HVRP}_4(R) = \sum_{h\in K}\left(2m_h - \sum_{j\in R} x_{0jh} + \sum_{E(R,\overline{R})} x_{ijh}\right) - 2\ell_k(\overline{R}) + 2\sum_{h\neq k\in K}\sum_{i,j\in\overline{R}} x_{ijh} \quad (30)$$

For the CVRP, these alternative slack functions reduce to the versions thereof as provided by Achuthan et al. (2003):

$$Z'_3(R) = \begin{cases} \sum_{E(R,\overline{R})} x_{ij} - \sum_{j\in R} x_{0j} + 2 & \text{if } \sum_{i\in R} q_i \leq Q \\ \infty & \text{otherwise} \end{cases} \quad (31)$$

[4] A proof of the validity of these slack functions is provided in Appendix C.

$$Z'_4(R) = 2m - 2\ell(\overline{R}) + \sum_{E(R,\overline{R})} x_{ij} - \sum_{j \in R} x_{0j} \tag{32}$$

Now, define $Z(D) = \min[Z_3'^{HVRP}(D), Z_4'^{HVRP}(D)]$ (or $Z(D) = \min[Z'_3(D), Z'_4(D)]$ for the CVRP). To initiate the search procedure, define $D_i = \{i\}$, $\forall i \in \mathcal{L} \backslash P$, where P is the set of single-customer tours for vehicle k. Finally, let $T = \cup_{i \in \mathcal{L} \backslash P} D_i \cup P$.

The procedure for this search is similar to that of search 4. Here, a violated Constraint of type 11 (12) is added when $Z(D_i) = Z'_3(D_i)$ and a violated Constraint of type 3 (4) is added otherwise. Hence, the constraint with the smallest slack is added.

Applying Gomory cuts

Ultimately, the goal of a branch-and-cut algorithm is to yield an integer valued optimal solution. Generally, this cannot be achieved solely by solving a linear program (LP) as the solutions thereof are often fractional. Gomory cuts intend to cut off some fractional solutions that include the current optimal LP-solution without eliminating any feasible integral solution. The rationale is described below and a mathematical argumentation is outlined in Appendix D.

For any solution to a linear program, satisfying all constraints thereof in standard form (i.e., including slack variables, yielding constraints $Ax = b$), a Constraint of type (33) can be added to the model to cut-off some non-integral solutions while preserving all feasible integral solutions:

$$\sum_j (A^*_{ij} - \lfloor A^*_{ij} \rfloor) x_j \geq (b^*_i - \lfloor b^*_i \rfloor) \tag{33}$$

If x^* is an integer-valued solution to the linear program, namely, the fractional aspects of the right hand side of the equation b^*_i must be caused by the fractional part of the coefficients as x^*_j is not fractional for any j. This implies the validity of Constraints (33).

Generally, multiple Gomory cuts can be added to a model. However, Laporte, Nobert and Desrochers (1985) argue that one should not add more than five Gomory cuts for CVRPs. Furthermore, they argue that it is computationally inefficient to add Gomory cuts in other nodes than the root node. Such cuts, namely, are only valid below the node in which they are added. Hence, adding Gomory cuts in other nodes in the search tree leads to an iterative procedure, of which the benefits generally do not outweigh the disadvantages (Laporte et al., 1985). Therefore, maximally five Gomory cuts are added in the root node of the search tree. This is similar to the implementation by Achuthan et al. (2003). For the HVRP, this principle is adhered to analogously.

Identifying branching variables

Generally, Achuthan et al. (2003) use the Land-Powell rule to determine the variable to branch upon. According to this rule, one selects the x_{ijk} variable with the potential to produce the largest increase in the LP-objective function value and hence, one selects the variable that is most likely to generate the largest optimality gap decrease. Decision variables that include the depot are excluded from this search as this strategy is found to be likely to force more variables to zero, via subtour prevention constraints (Achuthan et al., 1996). When a branching variable is identified, Achuthan et al. (2003) opt to branch in the direction opposite to the direction that leads to the maximum objective function increase since this leads to an increased likelihood of finding a good feasible solution and hence, to a good upper bound. Therefore, branches are cut off more easily in later stages of the algorithm.

The aforementioned potential change in objective value is found by resolving the model with an additional (temporary) constraint on the decision variable. Such a solve preferably occurs using the dual simplex method since a dual feasible solution already exists. If the potential change is at least $Z^* - \overline{Z}$, no branch is needed since a branch going in this direction can never yield a better solution than the solution that is currently available. Therefore, if such a situation occurs, the variable is fixed in the opposite direction and a branch is prevented.

Furthermore, if the potential increase is smaller than $Z^* - \overline{Z}$ for all variables and $\overline{Z} \geq 0.95Z^*$, all non-basic variables that are not fixed are considered. If the cost of fixing such a variable at a different value than its current value equals at least $Z^* - \overline{Z}$, it is never optimal to fix the variable at a different value and hence, we fix it at its current value.

It must be noted that Laporte et al. (1985) found this to be ineffective and they opted to branch at all times as to reduce the time necessary to search for situations in which branching could be prevented. Achuthan et al. (1996, 2003) chose differently and their choice is followed in this thesis.

Apart from these branches, the variables y_{ik} need to be integer valued, for the x_{ijk} can be integer valued even when y_{ik} are not. The branching procedure for these variables is similar to the branching procedure for the x_{ijk} variables. However, the search for potential branches on y_{ik} variables is only executed when all x_{ijk} are integer valued and when no variable x_{ijk} can be fixed without creating a branch. This order is chosen, because fixing a variable x_{ijk} is more likely to introduce a large number of subtour prevention constraints and to fix variables y_{ik} than fixing a variable y_{ik}. Therefore, this order is more likely to reduce the computation time of the algorithm.

Backing up in the search tree

As was noted by Achuthan et al. (1996), all vehicle capacity constraints that were added as a consequence of an earlier violation can be kept in the model. Namely, these constraints are valid throughout the search tree. Obviously, if these constraints become redundant at a later stage, they can be removed from the model when ineffective constraints are purged.

5.2 Sweep heuristic

Since a potential tax causes larger vehicles to be expensive to drive in or near the city center and relatively cheap further away from the city center, it seems appropriate to first define the vehicle to serve each customer and the routes afterwards. To this extent, compute the weight of driving with a vehicle of each type near each customer $i \in \mathcal{L}$. Denote this weight by c_{ik}. Evidently, such weights exist for each arbitrary pair of a customer and a vehicle. Furthermore, each vehicle has a capacity, Q_k. Now, one can define the relative weight of transporting one unit of cargo with a vehicle of type k near customer i as $\frac{c_{ik}}{Q_k}$. Finally, one selects the vehicle with the lowest relative weight to serve each customer with. However, when another vehicle has a relative weight to serve a particular customer that is at most $100 * \gamma$ percent[5] larger than the minimum weight, both scenarios are considered separately. This leads to (at least) $|K|$ CVRP problems that are solved to optimality using the branch-and-cut algorithm. Pseudocode for this heuristic is included in Appendix B.

In the presence of taxation, this procedure likely causes small vehicles to drive towards the city center, have a route there and drive back to the depot. Slightly larger vehicles will drive routes surrounding the center while the largest vehicles will have an encircling route. Each of these sub-CVRPs is relatively small in size and can be solved to optimality rapidly.

Finally, it must be noted that it might be more efficient to merge routes and drive them with a larger vehicle than originally scheduled or to split routes and serve them with smaller vehicles. Therefore, there will be a quick search to see whether routes can be merged to gain a total cost decrease. This search will consider merging the start of one route to the end of another, splitting existing routes, allowing one customer to switch routes and the switch of the order of customers within a route[6]. In each iteration, the cheapest vehicle is selected to drive each route. The search terminates as soon as a local optimum is identified, meaning that no cheaper HVRP solution can be found in the neighbourhood of the current solution.

[5] In this thesis, a value $\gamma = 0.1$ is chosen arbitrarily.

[6] Pseudocode of this local search is included in Appendix B.

6 Simulated problem instances

Achuthan et al. (2003) test their algorithm on 1620 simulated problems and 24 literature problems. In this thesis, the main focus will be on simulated problems of which the details are outlined below.

Capacitated vehicle routing problem instances

For the capacitated vehicle routing problem, Achuthan et al. (1996, 2003) generate problems with 15, 20, ..., 100 customers. Each of these customers is defined by two coordinates x and y, such that x, y are drawn uniformly from $\{0, \ldots, 1000\}$ and the distance between two customers is given by the Euclidean distance between them. Furthermore, each customer has a demand of q units, where q is drawn uniformly from $\{1, \ldots, 100\}$. Finally, Achuthan et al. (1996, 2003) use the following vehicle capacity, Q (Laporte et al., 1985), for $\alpha \in \{0; 0.33; 0.67; 1\}$:

$$Q = (1 - \alpha) \max_{i \in \mathcal{L}} q_i + \alpha \sum_{i \in \mathcal{L}} q_i$$

In this thesis, the depot will be located at (500, 500) for the 'standard' CVRP to generate results that can be used to compare the results presented in this thesis with the results found by Achuthan et al. (2003). The results found by Achuthan et al. (2003) will also be compared to those of a CVRP with a depot that is non-centered. The details thereof are explained below.

Heterogeneous fleet vehicle routing problem instances

For the HVRP, the customers are simulated as described above. However, the depot is located with coordinates x_0, y_0 drawn uniformly from $\{0, \ldots, 150\}$, as a depot at or near (500, 500) cannot help reducing traffic near the virtual city center for this location is likely to be very close to that city center. Furthermore, this definition allows for customers to be further removed from the city center than the depot, which can be seen to represent customers in an agricultural area.

For the HVRP, the problems are considered with three vehicle types: $\{0, 1, 2\}$. The capacity of a vehicle of type k is taken as $Q_k = \dfrac{Q}{k+1}$. Initially, these problems will be solved using Euclidean distances, as a standard CVRP. Then, weights for all edges are computed as follows:

Let c_{vk} be the marginal cost of driving vehicle $k \in K$ at point $v = (x, y)$, μ the distance between the depot and the centre of gravity and $d(g, v)$ the distance between v and the center of gravity c, and define the following relationship[7]:

[7] Appendix E provides a graphical indication of this relationship.

$$c_{vk} = \max\left(1, \left[(\frac{Q_k}{Q_2})^2 - \beta_k \frac{d(g,v)}{\mu}\right]\right) \tag{34}$$

In this equation β_k is the slope, which is chosen such that the medium vehicle has relative costs $c_{v1} = 1$ when $\frac{d(g,v)}{\mu} \geq 0.2$ and the largest vehicle has relative costs $c_{v0} = 1$ when $\frac{d(g,v)}{\mu} \geq 0.5$. Finally $\beta_2 = 0$ since no additional tax is levied on driving with the smallest vehicle. Appendix E provides a graphical indication of this relationship

To assign these weights to edges, three reference points of each edge are chosen: One at both end-points and one at the *bottleneck*, which is the point closest to the center of gravity representing the city center, where the externalities of urban traffic are most likely to be large. The weight assigned to each edge is the average of those three weights. This definition may lead to violations of the required triangle inequality for any vehicle except the smallest. If such a violation occurs, so that $c_{ipk} + c_{pjk} < c_{ijk}$ for some $i, j, p \in V$ and some $k \in K$, we set $c_{ijk} = c_{ipk} + c_{pjk}$, since it is possible to drive from i to j via p. This procedure is repeated until all such violations are resolved.

Finally, the problems are solved as HVRP and as CVRP with the additional weights to compute the direct differences in the costs that can be associated to the 'tax' in case the company opts to keep using the larger vehicle.

7 Results

The algorithms described in this thesis were implemented on a dual core HP laptop with an Intel Core i7-6500U processor operating at 3.0 GHz. It was implemented in Java, using CPLEX 12.6.3.0 only to solve linear programs. Furthermore, the Jama package that has been released to the public domain has been used for the computation of matrix inverses to generate Gomory cuts and a binary linked tree implementation by Goodrich, Tamassia and Goldwasser has been used as the framework for the search tree under the GNU General Public License. Throughout, it seemed reasonable to run two sets of problems simultaneously without structurally influencing the results.

7.1 The capacitated vehicle routing problem

First, Table 1 introduces the results of the 'standard' CVRP with a depot at (500, 500) and below, those results are compared to the results found by Achuthan et al. (2003). Here, the algorithm was terminated after 3600 seconds of computation time. This convention was taken from Achuthan et al. (2003). If more problems were solved now than in 2003, the table shows the result on the fastest

x runs in parentheses, corresponding to the number of problems solved by Achuthan et al. (2003). Furthermore, details regarding the search procedures are shown in Table 2 for the CVRP instances with a centered depot. Those settings are likely similar to the settings Achuthan et al. (2003) used to generate their results. For smaller instances, the current implementation of the algorithm outperforms the original implementation, while the original algorithm was generally faster for larger problems. Obviously, a substantial part of this first observation must be attributed to an increase in computer speed since 2003. This, in particular, makes the second observation more striking. First, it is noted that the difference between the linear programming relaxation in the root node and the final objective of the CVRP is similar in both implementations of the algorithm. Second, the number of Gomory cuts added to the model is slightly larger in the current implementation, which is probably caused by the fact that a minimum improvement in the linear programming objective was not implemented. On the contrary, the number of child nodes found in the current implementation of the algorithm was generally smaller than that of the original implementation. This is partially caused by a larger number of saved branches, but it could also point towards a more efficient cutting procedure. Here, one should mostly consider smaller problem instances as the results of the larger instances are biased towards 'easier' problems.

Nevertheless, the number of constraints and cutting planes added in the current implementation is well below that of Achuthan et al. (2003). The reason thereof is probably twofold. First, the implementation of Gomory cuts for this thesis required explicit slack variables to be included in the model, for all constraints and cutting planes added before the inclusion of the Gomory cuts. Such slack variables can be incorporated in the Gomory cuts and for simplicity reasons, these variables were not rewritten into their original constraints. Therefore, all constraints and slack variables that were added before the Gomory cuts were defined needed to be kept in the CPLEX model. This leads to a situation in which certain constraints are kept throughout, rather than being constantly removed and re-added to the model, hence reducing double counting of such constraints. Note that these constraints were valid in the root node and that they are thus probably valid in large part of the search tree. Therefore, if they would be removed from the model at some stage, they are probably added back fairly quickly. Second, it was mentioned in Section 5 that constraints were added when slack was negative. However, CPLEX does not offer exact results, but it provides results with a rounding margin of $1.0 * 10^{-5}$. In this thesis, constraints and cutting planes were added whenever slack was smaller than $-1.0 * 10^{-5}$. Not taking this factor into account leads to the inclusion of more (acutally redundant) constraints and a higher success rate on the search procedures, while also leading to multiple iterations in which the LP-objective does not increase and hence, to a larger number of branches. Namely, after 6 iterations in which the LP-objective does not increase sufficiently, one branches rather than continuing the search. In this thesis, this situation was very uncommon, but Achuthan et al. (2003) did not make any such claim. A different choice with respect to handling CPLEX rounding can also have an influence on both the efficiency of the search procedures and the generated number of child nodes. Finally, the constraints added after searches 1, 2 and 3 (as described on pages 16 and 17) could be purged. In contrast to the results described above, this would hinder the intended reduction in the search tree size.

Since more constraints are kept in the model throughout, the overall computation time may increase. First, the linear program that is solved repeatedly will generally have more constraints and hence, take slightly longer to solve. Second, the search procedures become less efficient and hence, are run with less purpose. This likely contributes to the decreased efficiency of the algorithm for larger problem instances. Because Achuthan et al. (1996, 2003) offer detail pseudocode for each of the search procedures, these procedures are probably implemented similarly for this thesis and by the original authors. Since the success rate is fairly low, it is thus relatively likely that the search procedures are run more often than necessary. Finally, the problems are simulated and hence, there may be some stochastic influence on the results. However, the aforementioned pattern in the differences seems too structural to search for an explanation of those differences in this direction.

For 20 problems with a non-centered depot, the results are presented in Table 3. Here, not all original settings for n and α have been considered, due to the limited time available to write this thesis. On average, the objective value has increased, which follows from the depot being further away from the 'average' customer. This effect diminishes when vehicle capacity increases. Then, less vehicles are needed to serve all customers, which leads to a larger influence of the routes driven between customers and a smaller influence of driving from and to the depot. Notably, the number of child nodes and branches saved have slightly increased as compared to the 'standard' CVRP. Probably, these problems are more difficult as most customers are on 'the same side of the depot'. For a centered depot, namely, it is intuitively plausible that one vehicle serves customers on one side of the depot, while another serves customers on the other side. When the depot is located outside of the center, this simplification is no longer present and hence, the number of branches needed is slightly larger, along with the number of branches saved and the computation time. Table 4 shows the specific details on the search procedures, which do not structurally deviate from the results in Table 2, indicating that the search procedures are equally efficient for both scenarios.

Table 1: CVRP computational results with a centered depot

Number of customers	α	Average objective	Root (%)	CPU time Average	CPU time Minimum	CPU time Maximum	Average number Elimination constraints	Average number Child nodes	Average number Gomory's cuts	Problems solved	Branches saved
15	0.00	8889.9	98.8	0.2	0.0	1.3	3.6	11.5	1.9	30	1.9
	0.33	4281.1	99.7	0.1	0.0	3.1	1.1	3.1	0.3	30	0.6
	0.67	3803.4	99.8	0.0	0.0	0.1	0.1	1.1	0.9	30	0.0
	1.00	3451.8	99.9	0.0	0.0	0.1	0.0	0.7	0.5	30	0.0
20	0.00	10706.1	97.6	5.4	0.1	38.0	12.0	49.7	3.3	30	20.1
	0.33	4761.3	99.3	0.3	0.0	3.0	1.4	3.9	1.9	30	0.3
	0.67	4135.1	99.4	0.1	0.0	0.4	0.7	4.2	1.5	30	0.4
	1.00	4011.5	99.6	0.3	0.0	9.1	0.0	2.1	0.7	30	0.2
25	0.00	12688.1	97.0	115.4	0.3	1755.3	23.6	505.7	4.0	30	249.5
		(12817.6)	(97.6)	(24.8)	(0.3)	(128.2)	(21.5)	(133.8)	(3.8)	(24)	(70.0)
	0.33	5090.2	98.6	11.7	0.0	142.3	6.5	45.8	3.1	30	19.4
	0.67	4608.8	99.6	0.1	0.0	0.7	1.3	3.9	1.9	30	0.2
	1.00	4303.5	99.5	0.7	0.0	12.6	0.0	5.1	0.7	30	1.7
30	0.00	15409.6	95.7	783.8	0.7	3490.9	45.9	1791.9	4.3	28	936.5
		(16222.6)	(97.5)	(67.5)	(0.7)	(243.1)	(38.9)	(188.2)	(4.2)	(10)	(97.7)
	0.33	5454.8	98.3	37.8	0.0	433.6	9.0	48.4	3.5	30	28.6
	0.67	4982.0	99.7	0.1	0.0	0.3	0.2	3.1	0.8	30	0.2
	1.00	4634.4	99.5	0.6	0.0	8.9	0.0	5.5	1.0	30	1.5
35	0.00	19192.5	95.2	1327.0	82.3	3411.0	63.3	2156.3	4.3	7	948.9
		(18985.6)	(96.7)	(321.1)	(82.3)	(559.9)	(33.5)	(438.0)	(2.5)	(2)	(290.5)
	0.33	5822.5	97.7	196.5	0.0	732.7	13.0	182.5	3.3	30	70.9
	0.67	5184.1	99.6	5.3	0.0	73.1	1.2	10.7	2.5	30	1.3
	1.00	4937.3	99.5	1.3	0.0	17.6	0.0	6.9	1.2	30	0.8
40	0.33	6063.2	97.7	440.7	0.0	2018.9	12.8	205.0	4.0	30	80.3
		(6064.1)	(98.0)	(228.6)	(0.0)	(855.7)	(10.4)	(132.2)	(3.8)	(25)	(38.2)
	0.67	5456.6	99.2	33.0	0.0	311.9	2.2	27.7	3.0	30	8.4
	1.00	5240.8	99.3	8.0	0.0	98.4	0.0	9.4	1.8	30	3.6
45	0.33	6257.1	98.1	712.4	0.0	2493.5	12.7	154.5	4.2	26	80.3
		(6224.7)	(98.4)	(225.7)	(0.0)	(1365.4)	(9.8)	(67.6)	(3.9)	(19)	(33.9)
	0.67	5692.4	99.1	26.2	0.0	430.3	2.4	21.9	3.3	30	4.1
	1.00	5512.1	99.3	19.3	0.0	169.6	0.1	13.8	1.7	30	4.5
50	0.33	6514.4	98.4	985.3	0.2	2672.7	12.4	177.3	4.4	18	79.4
		(6528.9)	(98.5)	(567.3)	(0.2)	(2174.3)	(8.7)	(82.3)	(4.3)	(14)	(43.5)
	0.67	5955.0	99.2	72.2	0.0	1076.6	2.3	27.1	3.2	30	10.5
	1.00	5707.2	99.3	14.1	0.0	262.9	0.0	15.9	2.0	30	3.8
55	0.33	6632.9	98.6	594.2	0.7	3557.3	10.9	109.6	4.3	14	29.3
		(6711.8)	(99.0)	(17.2)	(0.7)	(106.3)	(6.9)	(25.5)	(4.4)	(8)	(5.9)
	0.67	6252.3	99.2	189.3	0.0	1366.9	2.0	33.6	2.8	30	14.3
	1.00	6030.3	99.4	94.1	0.0	1434.8	0.0	14.9	3.0	30	3.8
60	0.33	6835.2	97.9	1084.5	53.6	2296.9	19.5	180.5	5.0	4	39.5
	0.67	6307.0	99.2	368.7	0.1	3471.9	2.6	49.2	3.7	26	12.5
	1.00	6270.2	99.2	285.7	0.0	2227.9	0.1	22.6	3.3	29	12.3
65	0.67	6672.0	99.1	667.3	0.5	3125.8	3.0	53.5	3.8	24	16.9
	1.00	6386.4	99.4	278.4	0.2	3400.1	0.1	19.9	3.0	28	8.1
70	0.67	6874.0	99.4	455.3	0.1	3436.3	1.2	27.6	4.1	22	9.6
	1.00	6684.9	99.4	384.2	0.1	3267.1	0.0	20.5	2.4	25	7.4
75	0.67	7071.4	99.4	534.0	0.7	3510.9	1.9	37.7	4.0	20	7.0
	1.00	6946.0	99.4	463.8	0.4	3384.8	0.1	17.5	3.5	26	6.0
80	0.67	7245.4	99.5	122.7	2.4	2017.8	2.3	24.3	3.8	20	5.7
	1.00	7049.4	99.6	241.0	1.1	3154.8	0.0	19.3	3.8	25	3.1
85	0.67	7346.1	99.4	402.2	3.7	3526.3	1.5	29.6	4.4	16	8.1
	1.00	7102.9	99.4	220.7	6.0	3010.7	0.0	26.2	4.5	19	7.7
90	0.67	7583.2	99.3	31.3	1.8	106.3	1.8	21.5	4.6	12	8.3
	1.00	7417.0	99.6	16.5	3.6	47.6	0.2	13.3	4.7	16	2.6
100	0.67	7921.0	99.5	36.6	7.6	78.7	1.4	24.9	4.7	16	2.3
	1.00	7803.3	99.5	43.9	9.3	157.1	0.0	24.0	4.0	21	6.9

Table 2: CVRP search procedure results with a centered depot

Number of		Problems	Average number of constraints added			% Succes of searches			
customers	α	solved	1.4 and 1.7	2.3 and 2.8	2.9	1-3	4	5	6
15	0.00	30	52.5	4.0	1.4	23.6	0.2	21.9	0.3
	0.33	30	23.5	4.0	0.1	13.5	1.0	18.8	0.2
	0.67	30	8.2	0.1	0.0	28.5	5.7	8.0	0.0
	1.00	30	5.5	0.0	0.0	3.4	13.0	0.0	0.0
20	0.00	30	376.1	68.2	7.7	13.7	0.2	29.7	0.2
	0.33	30	27.8	3.0	0.0	11.3	3.0	15.2	2.6
	0.67	30	12.6	1.0	0.0	15.9	3.9	22.1	4.3
	1.00	30	8.5	0.0	0.0	0.4	1.2	0.0	0.0
25	0.00	30 (24)	5141.4 (1160.9)	1413.7 (194.4)	96.6 (28.3)	11.7 (14.6)	0.5 (0.3)	31.5 (24)	0.0 (0.0)
	0.33	30	131.4	55.8	0.1	2.7	1.5	10.1	0.6
	0.67	30	14.4	2.2	0.0	12.2	14.1	37.0	2.9
	1.00	30	11.0	0.0	0.0	0.3	0.9	0.0	0.0
30	0.00	28 (10)	20238.4 (1885.9)	5293.3 (810.3)	556.8 (45.2)	10.9 (11.4)	0.5 (0.4)	26.7 (47)	0.0 (0.0)
	0.33	30	129.7	74.5	0.1	1.4	1.2	7.8	0.4
	0.67	30	15.6	0.2	0.0	15.0	12.8	9.5	3.5
	1.00	30	14.7	0.0	0.0	0.5	4.0	0.0	0.0
35	0.00	7 (2)	33278.4 (4549.0)	7765.7 (1458.5)	730.1 (137.5)	12.8 (10.1)	0.2 (0.4)	33.6 (29)	0.0 (0.0)
	0.33	30	377.7	204.1	4.7	1.4	1.0	6.1	0.4
	0.67	30	23.0	5.7	0.0	0.9	1.7	4.4	0.5
	1.00	30	16.7	0.0	0.0	0.2	2.7	0.5	0.0
40	0.33	30 (25)	438.9 (234.3)	271.6 (124.0)	0.2 (0.2)	1.0 (0.7)	1.1 (1.1)	5.3 (4)	0.3 (0.3)
	0.67	30	41.5	12.4	0.0	0.3	1.1	2.3	0.3
	1.00	30	20.1	0.0	0.0	0.1	1.8	0.0	0.0
45	0.33	26 (19)	327.7 (176.7)	207.0 (90.9)	0.1 (0.1)	0.6 (0.7)	0.9 (1.1)	4.3 (5)	0.2 (0.3)
	0.67	30	36.1	6.4	0.0	0.5	1.6	3.0	0.2
	1.00	30	23.0	0.1	0.0	0.1	1.5	0.1	0.0
50	0.33	18 (14)	604.6 (208.4)	254.1 (109.4)	0.8 (0.4)	0.7 (0.7)	1.1 (0.9)	10.7 (4)	0.3 (0.2)
	0.67	30	41.5	8.6	0.0	0.3	0.8	1.5	0.2
	1.00	30	25.6	0.0	0.0	0.1	2.7	0.0	0.0
55	0.33	14 (8)	181.2 (71.9)	102.2 (16.6)	0.6 (0.0)	0.5 (3.5)	1.1 (8.7)	4.3 (20)	0.2 (2.2)
	0.67	30	48.1	10.9	0.0	0.1	0.7	1.1	0.1
	1.00	30	30.0	0.0	0.0	0.0	1.1	0.6	0.0
60	0.33	4	251.8	138.8	0.0	0.6	1.5	4.9	0.3
	0.67	26	59.2	10.3	0.0	0.1	0.6	0.9	0.1
	1.00	29	37.3	0.1	0.0	0.0	0.6	0.0	0.0
65	0.67	24	67.3	14.1	0.0	0.1	0.6	0.8	0.1
	1.00	28	33.3	0.1	0.0	0.0	0.7	0.0	0.0
70	0.67	22	44.1	5.0	0.0	0.1	0.7	0.5	0.1
	0.67	22	44.1	5.0	0.0	0.1	0.7	0.5	0.1
75	0.67	20	57.7	5.3	0.0	0.1	0.9	0.7	0.1
	1.00	26	37.6	0.1	0.0	0.0	0.8	0.1	0.0
80	0.67	20	52.7	4.4	0.0	0.3	3.5	2.5	0.5
	1.00	25	35.0	0.0	0.0	0.0	2.0	0.0	0.0
85	0.67	16	57.3	3.4	0.0	0.4	2.0	0.9	0.1
	1.00	19	42.2	0.0	0.0	0.1	3.4	0.0	0.0
90	0.67	12	53.8	3.8	0.0	2.7	26.3	15.2	4.0
	1.00	16	39.9	0.2	0.0	0.8	34.5	1.8	0.0
100	0.67	16	51.9	2.1	0.0	2.5	25.0	12.8	1.2
	1.00	21	50.8	0.0	0.0	0.4	27.6	0.0	0.2

Table 3: CVRP computational results with a non-centered depot

Number of customers	α	Average objective	Root (%)	CPU time Average	Minimum	Maximum	Average number Elimination constraints	Child nodes	Gomory's cuts	Problems solved	Branches saved
15	0.33	5737.6	97.7	2.5	0.0	38.8	5.9	39.3	1.4	20	20.5
	0.67	4663.2	99.4	0.0	0.0	0.2	0.6	3.3	0.9	20	0.1
	1.00	3502.8	100.0	0.0	0.0	0.0	0.0	0.0	0.0	20	0.0
20	0.33	6076.6	97.0	9.0	0.0	25.6	14.2	68.2	3.7	20	21.5
	0.67	4700.8	98.8	0.6	0.0	4.5	1.9	9.6	0.8	20	1.0
	1.00	4113.7	99.8	0.0	0.0	0.1	0.0	0.6	0.5	20	0.1
25	0.33	6531.7	96.3	94.1	0.2	501.9	19.1	265.8	3.6	20	121.4
	0.67	5277.1	98.6	1.0	0.0	8.5	2.4	12.1	2.8	20	3.5
	1.00	4426.6	99.8	0.0	0.0	0.1	0.0	1.6	1.3	20	0.0
30	0.33	6248.7	95.6	608.9	0.3	2659.1	28.6	673.3	3.7	18	391.3
	0.67	5509.6	98.2	8.0	0.0	61.5	2.7	24.2	2.3	20	4.8
	1.00	4731.8	99.4	1.3	0.0	12.4	0.0	6.5	1.8	20	1.6
40	0.67	5976.5	98.3	89.7	0.0	882.1	2.8	64.3	3.6	20	24.7
	1.00	5329.8	99.1	5.5	0.0	101.4	0.0	10.3	3.3	20	5.8
50	0.67	6416.4	98.9	371.2	0.0	2686.5	4.4	80.2	3.0	20	35.0
	1.00	5789.7	99.4	9.7	0.0	176.3	0.1	12.8	2.8	20	3.2
60	0.67	4419.1	98.5	618.0	0.4	2671.8	4.3	121.1	3.5	13	27.8
	1.00	6252.9	99.4	311.2	0.1	3136.1	0.1	18.2	2.0	20	9.7
70	1.00	6350.6	99.4	160.2	0.8	2176.2	0.3	22.0	3.4	19	8.1

7.2 The heterogeneous fleet vehicle routing problem

As mentioned above, Table 4 holds data on the search procedures for the CVRP, but it also contains results on the solutions generated by the sweep heuristic. First, it shows that the heuristic is fast in computing a solution, in which (on average) roughly 2.5 vehicle types are used. This shows that taxation can, in fact, reduce the usage of large vehicles near the city center. Next, the column 'Relative savings' shows the number of times the heuristic objective value was saved by using a heterogeneous fleet of vehicles, rather than solving a CVRP while taxes must be paid. This shows that the usage of different vehicle types substantially reduces the costs of parcel delivery under such taxes. These savings correspond to (roughly) 40 to 67 percent of the original CVRP costs, which are achieved even though the routing of the vehicles is less efficient. Namely, the CVRPs were solved to optimality, which is not the case for the heuristically solved HVRPs. Nonetheless, the costs to deliver all parcels under such taxes is substantially larger than that of delivering all parcels with larger vehicles in the absence of taxation. In particular, those costs roughly double. The reason thereof is twofold. First, delivering parcels using smaller vehicles evidently causes more vehicles to be needed and hence, by the triangle inequality, larger distances to be driven. Second, taxes must be paid on some edges included in the final solution. Hence, taxation can reduce the usage of large vehicles near the city center, but it comes at a large costs for parcel delivery companies.

Table 4: CVRP search procedure results with a non-centered depot and the influence of taxation

Number of customers	α	Problems solved	Average number of constraints added: 1.4 and 1.7	2.3 and 2.8	2.9	% Succes of searches: 1-3	4	5	6	Heuristic objective	Number of vehicles	Relative savings	Computation time heuristic (s)
15	0.33	20	182.4	88.7	0.2	7.5	1.5	29.3	0.3	10525.2	2.7	0.8	0.4
	0.67	20	15.0	0.8	0.0	16.8	11.1	19.4	1.7	8821.5	2.6	1.2	0.2
	1.00	20	7.3	0.0	0.0	6.0	25.9	0.0	0.0	7983.8	1.9	1.3	0.2
20	0.33	20	215.7	133.6	1.0	4.8	2.1	20.8	0.8	12293.5	2.6	1.0	0.1
	0.67	20	26.6	4.3	0.0	2.4	1.8	8.1	1.2	9957.3	2.6	1.4	0.1
	1.00	20	8.2	0.0	0.0	3.3	9.7	0.0	0.0	10178.9	2.1	1.2	0.0
25	0.33	20	799.9	568.3	0.3	3.2	1.9	16.4	0.6	14511.3	2.7	1.2	1.4
	0.67	20	34.3	7.1	0.0	1.5	5.0	11.5	1.0	11984.6	2.4	1.4	0.9
	1.00	20	12.0	0.0	0.0	2.1	23.4	0.0	0.0	11307.7	2.2	1.5	0.6
30	0.33	18	2322.4	2164.4	1.4	2.3	2.5	16.9	0.4	16492.9	2.9	1.1	1.0
	0.67	20	48.8	8.1	0.0	0.7	2.1	3.3	0.5	14413.0	2.8	1.4	0.5
	1.00	20	14.9	0.0	0.0	0.0	3.4	0.0	0.0	12993.1	2.0	1.6	0.5
40	0.67	20	98.3	25.9	0.0	0.3	1.6	2.3	0.2	17093.6	2.6	1.6	8.0
	1.00	20	23.0	0.0	0.0	0.1	3.3	0.3	0.1	16485.1	2.1	1.7	4.9
50	0.67	20	151.4	65.5	0.0	0.1	1.1	3.0	0.1	20498.3	2.8	1.7	1.7
	1.00	20	25.7	0.1	0.0	0.1	3.6	0.1	0.1	17987.2	2.1	2.0	25.7
60	0.67	13	120.3	35.1	0.0	0.1	1.4	2.0	0.1	25560.9	2.8	0.9	36.2
	1.00	20	34.6	0.2	0.0	0.0	0.4	0.0	0.0	23457.7	2.1	1.8	40.6
70	1.00	19	39.4	0.5	0.0	0.0	1.8	0.2	0.0	28799.1	2.1	1.5	38.0

The branch-and-cut algorithm for the HVRP failed to solve all 60 problem instances with 15 customers. Primarily, this occurs because the constraints and cutting planes for the HVRP are less tight than those of the CVRP, in particular when some y_{ik} are still fractional. Then, namely, 'tours' exist that are driven partially by a vehicle of one type and partially by a vehicle of another. Subsidiary, the branch-and-cut algorithm for the CVRP was transformed for the HVRP in its entirety, while certain components may be less good a fit. Both aspects may compromise the computational performance of the HVRP algorithm. Now, the maximum computation time for the algorithm was set to 4 hours and 'unstructured' problems were simulated, in which both the number of customers and α are randomly drawn from a uniform distribution, with a maximum of 50 customers. The specific results thereof are found in Appendix F. None of these problems were solved to optimality within 4 hours. However, the results indicate that the search procedures for the HVRP were less effective than for the CVRP, which was anticipated as the cutting planes for the HVRP are less tight. Furthermore, these results stipulate that the the heuristic's performance varies substantially. Occasionally, it outperforms the exact algorithm after 4 hours. At other times, the exact algorithm produces substantially better results. Nonetheless, there is no structural pattern indicating that the heuristic uses more or less vehicle types than the exact algorithm. This reinforces the conclusion that taxation can reduce the usage of large vehicles near the city center. Meanwhile, it expounds on the remark that the additional costs thereof are prohibitively unfair as adequate routing seems capable to further reduce the costs of parcel delivery with a heterogeneous fleet.

8 Conclusion and discussion

In summary, this thesis shows that taxation can help to reduce the usage of large vehicles in downtown parcel delivery, which leads to an increase in the (economic) efficiency of urban traffic by reducing the negative externalities thereof. Nonetheless, the introduction of such taxes leads to a heavy burden on parcel delivery companies by substantially increasing the costs of their daily operations. On the one hand, it must be noted that the specifics of the simulated data contribute largely to the size of this burden. Therefore, further research into a specific tax structure on (large) vehicle usage near the city center is recommended. On the other hand, governments can offer a lump sum compensation to parcel delivery companies, which would offset the financial unfairness while maintaining the motivation to use smaller rather than larger vehicles for downtown parcel delivery. Such a compensation can, for example, be offered as lump sum or as a subsidy on the purchase of smaller delivery vehicles. Finally, it must be noted that the customers' location coordinates and demands were drawn from a uniform distribution, which needs not adequately reflect a realistic setting for urban parcel delivery and hence, may influence the effect of taxation on vehicle usage.

Furthermore, this thesis offers new cutting planes for the HVRP. However, the branch-and-cut algorithm based on those cutting planes fails to produce optimal solutions within a reasonable computation time. This is partially caused by a suboptimal implementation of the algorithm, a presumption supported by the fact that the corresponding branch-and-cut implementation for the CVRP was seemingly suboptimal too. Meanwhile, the complexity of the HVRP also contributes heavily to the computation time of any exact solution algorithm thereof. Nonetheless, these cutting planes provide a genesis for future research, which can identify the extent to which the cutting planes presented in this thesis can contribute to an adequate exact or heuristic algorithm for the HVRP.

Finally, this thesis offers a sweep heuristic to solve the HVRP, in which clusters are based largely on the customers' relative distance to a virtual city center. For each cluster, a CVRP arises that is solved using the corresponding branch-and-cut algorithm. This heuristic is quite fast, because the CVRPs that arise generally have relatively few customers to be served using relatively large vehicles. Even though the heuristic's solution quality has not been adequately determined, it can potentially be improved upon by solving the CVRP is to optimality in each iteration. Since the heuristic generates solutions rapidly, such a modification probably does not lead to prohibitive computation times, while it may substantially improve the heuristic's results.

References

Achuthan, N., Caccetta, L., & Hill, S. (1996). A new subtour elimination constraint for the vehicle routing problem. *European Journal of Operational Research*, *91*, 573-586.

Achuthan, N., Caccetta, L., & Hill, S. (2003). An improved branch-and-cut algorithm for the capacitated vehicle routing problem. *Transportation Science*, *37*, 153-169.

Angel, S., Parent, J., Civco, D., Blei, A., & Potere, D. (2011). The dimensions of global urban expansion: Estimates and projections for all countries, 2000-2050. *Progress in Planning*, *75*, 52-107.

Baldacci, R., Battarra, M., & Vigo, D. (2009). Valid inequalities for the fleet size and mix vehicle routing problem with fixed costs. *Networks: An International Journal*, *54*, 178-189.

Balinski, M., & Quandt, R. (1964). On an integer program for a delivery problem. *Operations research*, *12*, 300-304.

Betinelli, A., Ceselli, A., & Righini, G. (2011). A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, *19*, 723-740.

Choi, E., & Tcha, D. (2007). A column generation approach to the heterogeneous fleet vehicle routing problem. *Computers & Operations Research*, *34*, 2080-2095.

Coase, R. (1960). The problem of social cost. *Journal of Law and Economics*, *3*, 1-44.

Crowder, H., & Padberg, M. (1980). Solving large-scale symmetric travelling salesman problems to optimality. *Management Science*, *26*, 495-509.

Figliozzi, M. (2010). Vehicle routing problem for emissions minimization. *Transporation Research Record*, *2197*, 1-7.

Gendreau, M., Laporte, G., Musaraganyi, C., & Taillard, E. (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers & Operations Research*, *26*, 1153-1173.

Gomory, R. (1958). Outline for an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, *64*, 275-278.

Jiang, L., & O'Neill, B. (2017). Global urbanization projections for the Shared Socioeconomic Pathways. *Global Environmental Change*, *42*, 193-199.

Joerss, M., Schröder, J., Neuhaus, F., Klink, C., & Mann, F. (2016). *Parcel delivery the future of last mile* (Tech. Rep.). McKinsey & Company.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation science*, *43*, 408-416.

Laporte, G., Nobert, Y., & Desrochers, M. (1985). Optimal routing under capacity and distance restrictions. *Operations research*, *33*, 1050-1073.

Ossokina, I., & Verweij, G. (2005). Urban traffic externalities: Quasi-experimental evidence from housing prices. *Regional Science and Urban Economics*, *55*, 1-13.

Paessens, H. (1988). The savings algorithm for the vehicle routing problem. *European Journal of Operational Research*, *34*, 336-344.

Penna, P., Subramanian, A., & Ochi, L. (2013). An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *Journal of Heuristics*, *19*, 201-232.

Pessoa, A., Sadykov, R., & Uchoa, E. (2018). Enhanced branch-cut-and-price algorithm for heterogeneous fleet vehicle routing problems. *European Journal of Operational Research"*, *270*, 530-543.

Pessoa, A., Uchoa, E., & Poggi de Aragão, M. (2009). A robust branch-cut-and-price algorithm for the heterogeneous fleet vehicle routing problem. *Networks: An International Journal*, *54*, 167-177.

Pigou, A. (1932). *The economics of welfare* (4th ed.). London: Macmillan.

Qingyu, L., Zhicai, J., Baofeng, S., & Hongfei, J. (2007). Method research on measuring the external costs of urban traffic congestion. *Journal of Transportation Systems Engineering and Information Technology*, *7*, 9-12.

Ralphs, T., Kopman, L., Pulleyblank, W., & Trotter, L. (2003). On the capacitated vehicle routing problem. *Mathematical programming*, *94*, 343-359.

Renaud, J., & Boctor, F. (2002). A sweep-based algorithm for the fleet size and mix vehicle routing problem. *European Journal of Operational Research*, *140*, 618-628.

Schrank, D., & Lomax, T. (2009). *2009 urban mobility report* (Tech. Rep.). U.S. DEPARTMENT OF TRANSPORTATION.

Solano-Charris, E., Prins, C., & Santos, A. (2015). Local search based metaheuristics for the robust vehicle routing problem with discrete scenarios. *Applied Soft Computing*, *32*, 518-531.

Stopher, P., & Stanley, J. (2014). *Introduction to transport policy – a public policy view.* Edward Elgar Publishing.

Warf, B. (2010). Thünen model. In B. Warf (Ed.), *Encyclopedia of geography* (p. 2833-2834). Thousand Oaks, CA: SAGE Publications, Inc. doi: 10.4135/9781412939591.n1138

Wegener, M. (2013, June). *Polycentric europe: More efficient, more equitable and more sustainable*

(Vol. 7). International seminar on welfare and competitiveness in the European polycentric urban structure.

Yaman, H. (2006). Formulations and valid inequalities for the heterogeneous vehicle routing problem. *Mathematical Programming*, *106*, 365-390. doi: 10.1007/s10107-005-0611-6

Appendix A: Overview of notation

Table 5 provides an overview of the mathematical notation used in this thesis. The diverse symbols are introduces on a per topic basis and presented where that is most appropriate.

Table 5: CVRP computational results centered depot

Symbol	**Definition**
General graph and set notation	
$H = (V_H, E_H)$	A graph with vertex set V_H and edge set E_H.
$E[S, T]$	The set of all edges that start in S and end in T, or vice-versa.
$\overline{S} = \mathcal{L} \backslash S$	The complement of set S with respect to $\mathcal{L}$.
Problem definition	
$V = \{0, \ldots, n-1\}$	The set of vertices, representing the customers and the depot.
$0 \in V$	The depot vertex.
$\mathcal{L} = \{1, \ldots, n-1\} \subset V$	The set of $n-1$ customers.
q_j	The demand of customer $j \in \mathcal{L}$.
$E = \{(i,j) \mid 0 \leq i \neq j \leq n-1\}$	The set of edges.
$K = \{0, \ldots, t-1\}$	The set of t vehicle types.
m_k	The number of vehicles of type k used.
Q_k	The capacity of vehicle k.
$\ell_k(S)$	The greatest lower bound on the number of
	vehicles of type k to serve all customers in S.
c_{ijk}	The 'relative' distance between vertices
	$i, j \in V$ for a vehicle of type k.
x_{ijk}	A decision variable indicating whether edge
	$(i,j) \in E$ is used by a vehicle of type k.
y_{ik}	A decision variable indicating whether a vehicle
	of type k serves customer $i \in \mathcal{L}$.
c_{ik}	The weight of driving vehicle k near customer i.

Table continues on the next page.

Symbol	Definition
Branch-and-cut	
Z^*	The best upper bound known in the algorithm.
$\overline{Z}$	The LP-relaxation of the current subproblem.
$S_k = \{S_{1k}, \ldots, S_{m_k k}\}$	The set of tours driven with vehicle k.
Slack functions	
$Z_1^{HVRP}(R)$	The slack on Constraints (1e).
$Z_1(R)$	The slack on Constraints (2d).
$Z_2^{HVRP}(R)$	The slack on Cutting Planes (5).
$Z_2(R)$	The slack on Cutting Planes (6).
$Z_3^{HVRP}(R)$	The slack on Cutting Planes (11).
$Z_3(R)$	The slack on Cutting Planes (12).
$Z_4^{HVRP}(R)$	The slack on Constraints (3).
$Z_4(R)$	The slack on Constraints (4).
$Z_1'^{HVRP}(R)$	An alternative slack function for Constraints (1e).
$Z_1'(R)$	An alternative slack function for Constraints (2d).
$Z_3'^{HVRP}(R)$	An alternative slack function for Cutting Planes (11).
$Z_3'(R)$	An alternative slack function for Cutting Planes (12).
$Z_4'^{HVRP}(R)$	An alternative slack function for Constraints (3).
$Z_4'(R)$	An alternative slack function for Constraints (4).

Appendix B: Algorithm pseudocode

Branch-and-cut algorithm

Pseudocode of the branch-and-cut algorithm for the HVRP and CVRP is provided in Algorithm 1.

Algorithm 1 Branch-and-cut

1: Initialize L as a list of active (sub)problems.

2: Add the original model (2) to L.

3: Initialize x^* as the solution found using the savings heuristic for the vehicle routing problem (Paessens, 1988) (CVRP), or for each of the clusters generated according to steps 1-6 of Algorithm 7 (HVRP); Also, initialize Z^* as the costs of this initial solution and $\overline{Z}$ as the objective value of the LP-relaxation of the original model (1 for the HVRP and 2 for the CVRP).

4: **while** L is not empty **do**

5:     Select P as a problem from L.

6:     Remove P from L.

7:     Define and fix the forced variables.

8:     Solve the model (for example using CPLEX).

9:     **if** $\bar{z} \geq Z^*$ **then**

10:         This branch cannot yield an optimal solution. Thus, go to Step 4.

11:     **end if**

12:     Purge ineffective constraints.

13:     **if** There exists a capacity constraint violation **then**

14:         Add the violated constraint to the model and go back to Step 8.

15:     **end if**

16:     **if** $|L| = 1$ and no Gomory cuts were added yet **then**

17:         Add at most 5 Gomory's cuts to the model and go back to Step 8.

18:     **end if**

19:     **if** The solution x is integer valued **then**

20:         The solution is feasible and better than Z^*. Thus, update Z^* and store the solution.

21:         Go to Step 4.

22:     **end if**

23:     Identify the next branching variable, generate two new subproblems and add them to L.

24: **end while**

Identifying violated constraints

Search 1 is executed according to the following pseudocode (Achuthan et al., 1996, 2003), in which $Z(R)$ represents $Z_1(R)$ as defined in equation (18) for the CVRP and in equation (17) for the HVRP:

Algorithm 2 Search 1

Initialize $R = \emptyset$
while $|R| < |S| - 2$ **do**
    Select $r \in S \backslash R$ such that $Z(R \cup \{r\})$ is minimized.
    if $Z(R \cup \{r\}) \geq Z(R)$ **then**
        Stop the search.
    end if
    if $Z(R \cup \{r\}) < 0$ **then**
        Add a Constraint of type 1e (or 2d for the CVRP) for $S \backslash R$ and stop the search.
    end if
    Set $R = R \cup \{r\}$.
end while

Search 2 is executed as detailed in Algorithm 3 (Achuthan et al., 1996, 2003), in which $Z(R) = \min\left[Z_2(R), Z_3(R)\right]$ for the CVRP and $Z(R) = \min\left[Z_2^{HVRP}(R), Z_3(R)^{HVRP}\right]$ for the HVRP. $Z_2(R)$ and $Z_3(R)$ are defined in equations (21) and (22), while $Z_2^{HVRP}(R)$ and $Z_3(R)^{HVRP}$ are defined in equations (19) and (20).

Search 3 is executed based on $Z(R) = \min\left[Z_3(R), Z_4(R)\right]$ for the CVRP. For the HVRP, it is based on $Z(R) = \min\left[Z_3^{HVRP}(R), Z_4^{HVRP}(R)\right]$. The search is executed according to the pseudocode introduced in Algorithm 4 (Achuthan et al., 1996, 2003). Again, $Z_3(R)^{HVRP}$ and $Z_2(R)$ are defined in equations (20) and (22), while $Z_4^{HVRP}(R)$ and $Z_4(R)$ are defined in equations (25) and (26), respectively. Note that $\mathcal{S} = \{S_i\}$ is assumed to be sorted as described in Section 5.

Algorithm 3 Search 2

Initialize $R = \emptyset$ and $MIN = \infty$.
while $|R| < |S| - 1$ **do**
  Select $r \in S \backslash R$ such that $Z(R \cup \{r\})$ is minimized.
  if $Z(R \cup \{r\}) \leq MIN$ **then**
    $MIN = Z(R \cup \{r\})$ and $R = R \cup \{r\}$.
  else
    break
  end if
end while
if $MIN < 0$ **then**
  Add a Constraint of type 1e (2d for the CVRP) for $\overline{R}$ if $Z(R) = Z_2^{HVRP}(R)$ $(Z(R) = Z_2(R))$ or a Cutting Plane of type 11 (12 for the CVRP) if $Z(R) = Z_3^{HVRP}(R)$ $(Z(R) = Z_3(R))$.
end if

Algorithm 4 Search 3

Initialize $R = \emptyset$, $i = 1$ and $MIN = \infty$.
while $i < l$ **do**
  Set $R = R \cup S_i$ and $MIN = Z(R)$.
  if $MIN < 0$ **then**
    if $Z(R) = Z_3^{HVRP}(R)$ $(Z(R) = Z_3(R)$ for the CVRP) **then**
      Introduce the violated Cutting Plane of type 11 (12) for R.
      Set $R = \emptyset$ and $MIN = \infty$.
    else
      Introduce a violated Constraint of type 3 (4) for R.
      Set $R = \emptyset$ and $MIN = \infty$.
    end if
  end if
  i=i+1.
end while

With slack function Z_1' for the CVRP and $Z_1'^{HVRP}$ for the HVRP (as defined in equations (28) and (27), respectively), search 4 is executed as outlined in Algorithm 5 (Achuthan et al., 2003).

Algorithm 5 Search 4

Initialize $T_i = \{i\}, \forall i \in \mathcal{L}$ and initialize succesfulMerge=true.
If $x_{\alpha\beta} < 1$, let $T_\alpha = \{\alpha, \beta\}$ and remove T_β.
while succesfulMerge **do**
    if $\exists(r, s)$ such that $\sum_{E[T_s,T_r]} x_{ij} \geq 1$ **then**
        Merge sets T_r and T_s into one set T_{rs}[8].
    else
        Set succesfulMerge=false.
    end if
end while
Define $T' = \{T'_1 \dots T'_k\}$ as the set of sets found following this merge procedure.
for $T'_i \in T'$ **do**
    if $Z'_1(T'_i) < 0$ **then**
        Add a violated constraint for T'_i.
    end if
end for

Search 5 is implemented as outlined in Algorithm 6, based on an initial set $T = \{T_1 \dots T_p\}$ (Achuthan et al., 2003).

Finally, search 6 is implemented in a manner similar to search 4 (Achuthan et al., 2003). However, when there exist multiple pairs (r, s) for which $Z(T_r \cup T_s)$ is minimized, a pair (r^*, s^*) is chosen for a merge such that $Z(T_{r^*} \cup T_{s^*}) = Z'_3(T_{r^*} \cup T_{s^*})$ and subsidiary a pair that maximizes

$$\frac{1}{Q} \sum_{i \in T_{r^*} \cup T_{s^*}} q_i - \left\lfloor \frac{1}{Q} \sum_{i \in T_{r^*} \cup T_{s^*}} q_i \right\rfloor$$

The search procedure is based on $Z^{HVRP}(R) = \min\left[Z_3'^{HVRP}(R), Z_4'^{HVRP}(R)\right]$ for the HVRP and $Z(R) = \min\left[Z'_3(R), Z'_4(R)\right]$ for the CVRP. $Z_3'^{HVRP}(R)$ and $Z'_3(R)$ are defined in equations (29) and (31), respectively. Furthermore, $Z_4'^{HVRP}(R)$ and $Z'_4(R)$ are defined in equations (30) and (32), respectively.

[8] If multiple such sets exist, pick the pair (r, s) that minimizes $Z'_1(T_r \cup T_s)$.

Algorithm 6 Search 5

for i=1 ... p **do**

  Let $S = T_i$, $R = \{T_1 \dots T_i\} \backslash S$, *TFound*=true and $Q = \emptyset$.

  while *TFound* **do**

    Find $T_r \in R$ such that $\sum_{E[S,T_r]} x_{rs} > 0$ and $\max_{T_l \in R} \sum_{l \in T_l} q_l$ is attained for T_r. In case of a tie, opt T_r that maximixes $\sum_{E[S,T_r]} x_{rs}$.

    if T_r was found **then**

      Add T_r to T.

    elseSet *TFound*=false.

    end if

  end while

  if $\sum_{T_j \in T} \sum_{i \in T_j} q_i > Q$ **then**

    if $\sum_{E[S,T]} x_{rs} > 1$ **then**

      Add the violated Cutting Plane of type 7 or 9 (or 8 or 10 for the CVRP) or the violated Constraint of type 1e (2d) for T if $|T| > 3$, $|T| = 3$ or $|T| = 2$, respectively.

    end if

  end if

end for

Sweep heuristic

The general pseudocode for the sweep heuristic is introduced in Algorithm 7, while the local search procedure is outlined in Algorithm 8. In the implementation of the sweep heuristic in this thesis, $\gamma = 0.1$.

Algorithm 7 Sweep heuristic

for Customer $i \in \mathcal{L}$ **do**
    Add i to the cluster of vehicle $k \in K$ such that $k = \arg\min_{k \in K} \frac{c_{ik}}{Q_k}$.
    if $\exists h \neq k \in K$ such that $\frac{c_{ih}}{Q_h} \leq (1 + \gamma) * \frac{c_{ik}}{Q_k}$ **then**
        Generate a new set of clusters in which i is added to the cluster of vehicle h instead of vehicle k and treat both sets separately.
    end if
end for
for Vehicle $k \in K$ **do**
    Solve the CVRP for the cluster associated with k.
end for
Apply local search to (potentially) improve upon this solution.

Local search

Algorithm 8 Local search sweep heuristic

Define *CurrentCost* to be the current cost of the HVRP solution *CurrentSolution*, as found after step 6 in Algorithm 7. Set *Improved* to true.
while *Improved* **do**
    Get the neighbourhood of *CurrentSolution* as defined below.
    Find the neighbourhood element (x) with lowest costs, $c(x)$.
    if $c(x) <$ *CurrentCost* **then**
        Set *CurrentSolution* $= x$ and *CurrentCost* $= c(x)$.
    else
        Set *Improved* to false.
    end if
end while
After the local search, *CurrentSolution* is found to be the local optimum with corresponding cost *CurrentCost*.

In the pseudocode above, the neighbourhood of *CurrentSolution* is needed. For any feasible solution of the HVRP, the neighbourhood thereof consists of any element that can be found by applying one of the following actions to one (pair of) route(s) of that solution.

1. Merge two routes, which is allowed if the largest vehicle available can serve the aggregate demand of the original two routes.

2. Split one route into two routes, provided that (at least) one of the emerging routes can be served by a smaller vehicle than the vehicle used to serve the original route. Otherwise, namely, the new solution can never lead to a cost decrease (as follows from the interpretation of Cutting Planes 11).

3. Swap two customers within their current route.

4. Add one customer to a different route.

For simplicity of the local search procedure, scenarios in which multiple modifications are made simultaneously are excluded from this local search.

Appendix C: Cutting plane validity

Cutting planes for the CVRP

In this Appendix, the validity of each of the cutting planes is proven, except for the cases in which a direct proof is provided by Achuthan et al. (2003). That is the case for Constraints (4) and Cutting Planes (8), (10), (12) and (13). Also, slack functions (18), (21), (22) and (26) follow trivially from the constraints. However, this is not necessarily the case for slack functions (28), (31) ad (32). The validity of these slack functions is also shown in this Appendix.

Throughout this (sub)section, the following property, that holds for all $S \subseteq \mathcal{L}$ in a feasible solution of the CVRP, is of much importance:

$$2|S| = 2\sum_{i,j\in S} x_{ij} + \sum_{E(S,\overline{S})} x_{ij} + \sum_{i\in S} x_{0j} \tag{35}$$

This property is found by summing Constraint (2c) over all vertices $i \in S$.

The validity of Constraints (4)

Even though the validity of these Constraints is also proven in (Achuthan et al., 1996), I opt to provide a proof for this validity in this appendix too. The proof, namely, contains one equivalence that is also highly relevant in other proofs in this section.

As a starting point, take a set S such that $1 \leq |S| \leq n-2$ and $\overline{S} = \mathcal{L}\backslash S$, which implies that $1 \leq |\overline{S}| = n-1-|S| \leq n-2$. Therefore, the Constraint of type (2d) must be satisfied for $\overline{S}$ for each feasible solution of the CVRP, either because $3 \leq |\overline{S}| \leq n-2$, or trivially if $|\overline{S}| < 3$. Hence, for $\overline{S}$, we have

$$\sum_{i,j\in\overline{S}} x_{ij} \leq |\overline{S}| - \ell(\overline{S})$$

By the property in equation (35), this is equivalent to

$$\sum_{i,j\in\overline{S}} x_{ij} \leq \sum_{i,j\in\overline{S}} x_{ij} + \frac{1}{2}\left(\sum_{E(S,\overline{S})} x_{ij} + \sum_{i\in\overline{S}} x_{0j}\right) - \ell(\overline{S})$$

Deducting $\sum_{i,j\in\overline{S}} x_{ij}$ on both sides, multiplying the entire equation by 2 and rearranging the terms yields the identity:

$$\sum_{E(S,\overline{S})} x_{ij} + \sum_{i\in\overline{S}} x_{0j} \geq 2\ell(\overline{S}) \quad \forall\overline{S} \text{ such that } 1 \leq |\overline{S}| \leq n-2 \tag{36}$$

Adding $2\sum_{i,j\in S} x_{ij} + \sum_{i\in S} x_{0i}$ to both sides of the equation and applying property (35) yields:

$$2|S| + \sum_{j\in\overline{S}} x_{0j} \geq 2\ell(\overline{S}) + 2\sum_{i,j\in S} x_{ij} + \sum_{j\in S} x_{0j}$$

Evidently, this is equivalent to:

$$\sum_{i,j\in S} x_{ij} + \frac{1}{2}\left(\sum_{j\in S} x_{0j} - \sum_{j\in\overline{S}} x_{0j}\right) \leq |S| - \ell(\overline{S})$$

Now, since $S \cup \overline{S} = \mathcal{L}$ by definition and $2m = \sum_{i\in\mathcal{L}} x_{0j}$ per Constraints (2b) for the CVRP, we have:

$$\sum_{i,j\in S} x_{ij} + \frac{1}{2}\left(\sum_{j\in S} x_{0j} - (2m - \sum_{j\in S} x_{0j})\right) \leq |S| - \ell(\overline{S}) \;\; \forall S \subset \mathcal{L} \text{ such that } 1 \leq |S| \leq n-2$$

This, trivially, is equivalent to Constraints (4), which concludes the proof.

The validity of Cutting Planes (6)

Cutting Planes (6) are fairly similar to Constraints 4. To prove the validity of these Cutting Planes, consider S to be the set of customers whose demand must be satisfied and do not consider the customers in $\overline{S} = \mathcal{L}\backslash S$. Then, the virtual depot 'for this subset' $0_S = \{0\} \cup \overline{S}$ consists of the actual depot and all vertices outside of S. Namely, vehicles that enter S from 0 or $\overline{S}$ can be treated similarly when one only focusses on serving the customers in S. Defining m_S as the number of vehicles visiting S, Constraints (4) can be written as follows, for $P \subset S$:

$$\sum_{i,j\in P} x_{ij} + \sum_{i\in 0_S, j\in P} x_{ij} - m_S \leq |P| - \ell(S\backslash P) \iff \sum_{i,j\in P} x_{ij} + \left(\sum_{j\in P} x_{0j} + \sum_{E(P,\overline{S})} x_{ij}\right) - m_S$$
$$\leq |P| - \ell(S\backslash P)$$

Since the Constraint of type (2d) is not violated for S, we can safely add $\sum_{i,j\in S} x_{ij} + s$ to the left hand side of the inequality and $|S| - \ell(S)$ to the right hand side, where $s \geq 0$ equals the slack variable for the Constraint of type (2d). This yields:

$$\sum_{i,j\in P} x_{ij} + \sum_{i,j\in S} x_{ij} + \sum_{j\in P} x_{0j} + \sum_{E(P,\overline{S})} x_{ij} + s \leq |S| + |P| - \ell(S\backslash P) + m_S - \ell(S)$$

If $m_S = \ell(S)$, the Constraint of type (2d) holds at equality ($s = 0$) so that this equation reduces to a Cutting Plane of type (6). If $m_S > \ell(S)$, there is slack on the Constraint of type (2d) since less cross-edges are used for S and more edges that enter/leave S are used. In that case $s = m_S - \ell(S)$, so that the Cutting Plane of type (6) still follows from this inequality.

The validity of slack function (28)

As per identity (36), the following two equations are equivalent, for all S such that $3 \leq |S| \leq n-2$:

$$|S| - \ell(S) - \sum_{i,j \in S} x_{ij} \geq 0$$

$$\sum_{E(S,\overline{S})} x_{ij} + \sum_{i \in S} x_{0j} - 2\ell(S) \geq 0$$

This directly shows why equation (28) is an adequate slack function for Constraints (2d).

The validity of slack function (31)

The trivial slack function on Cutting Planes (12) for set T with $2 \leq |T| \leq n-1$ and $\sum_{i \in T} q_i \leq Q$ reads:

$$Z_3(T) = |T| + 1 - \sum_{i,j \in T} x_{ij} - \sum_{j \in T} x_{0j} \geq 0$$

For other sets T, such a cutting plane is not defined and hence, it cannot be violated.

Multiplying this function by 2 and applying property (35) for T yields an equivalence with

$$Z_3'(T) = 2\sum_{i,j \in T} x_{ij} + \sum_{E(T,\overline{T})} x_{ij} + \sum_{j \in T} x_{0j} + 2 - 2\left(\sum_{i,j \in T} x_{ij} - \sum_{j \in T} x_{0j}\right) \geq 0$$

After cancelling out the duplicate terms, this yields equation (31) and hence, it shows the validity thereof.

The validity of slack function (32)

For Constraints (4), the regular slack function reads:

$$Z_4(T) = |T| + m - \ell(\overline{T}) - \sum_{i,j \in T} x_{ij} - \sum_{j \in T} x_{0j}$$

Multiplying this function by two and applying property (35) for T yields an equivalence with

$$Z_4'(T) = 2\sum_{i,j \in T} x_{ij} + \sum_{E(T,\overline{T})} x_{ij} + \sum_{j \in T} x_{0j} - 2\ell(\overline{T}) + 2m - 2\left(\sum_{i,j \in T} x_{ij} - \sum_{j \in T} x_{0j}\right)$$

After cancelling out the duplicate terms, this yields equation (32) and hence, it shows the validity thereof.

Proof $\alpha(S) \geq 0$

Since S is one of the connected components of the resulting graph, and the Constraint of type (2d) is not violated for S, it must be true that all customers in S can be served by one vehicle. Furthermore, for $i \in S$, Constraints (2c) must be satisfied and hence, it can be concluded that $\sum_{i,j \in S} x_{ij} = |S| - 1$. Therefore,

$$\alpha = \sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{0j} - |S| - \frac{1}{Q} \sum_{j \in S} q_j \geq \sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{0j} - |S| - \ell(S)$$
$$= \sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{0j} - |S| - 1 = \sum_{j \in S} x_{0j} \geq 0$$

Here, the last inequality is valid since $x_{0i} \in \{0, 1, 2\}$ for all $i \in \mathcal{L}$.

Cutting planes for the HVRP

In the remainder of this appendix, the validity of all cutting planes for the HVRP is proven, along with a proof of the validity of Constraints (3) and slack functions (27), (29) and (30), which are the only slack functions whose validity does not directly follow from the constraints or cutting planes they refer to. Finally, a proof that $\alpha_k(S_{ik}) \geq 0$ is provided. It must be noted that most of these proofs are very similar to/rely heavily on the proofs provided by Achuthan et al. (1996, 2003) for the analogous CVRP cutting planes and constraints.

To start, identify the following property for any $S \subseteq \mathcal{L}$ in a feasible solution of the HVRP, which is of high importance in some proofs:

$$2|S| = \sum_{h \in K} \left(2 \sum_{i,j \in S} x_{ijh} + \sum_{E(S,\overline{S})} x_{ijh} + \sum_{i \in S} x_{0jh} \right) \tag{37}$$

This property follows from summing Constraints (1c) for all vertices $i \in S$ along with Constraints (1d).

The validity of Constraints (3)

To start, pick an arbitrary vehicle type $k \in K$ and define a set S such that $1 \leq |S| \leq n-2$ and $\overline{S} = \mathcal{L} \backslash S$, implying that $1 \leq |\overline{S}| \leq n-2$. Therefore, Constraints (1e) must be satisfied for $\overline{S}$ for all feasible solutions of the HVRP, either since $3 \leq |\overline{S}| \leq n-2$, or trivially when $|\overline{S}| < 3$. Therefore, for $\overline{S}$, it must hold that:

$$\sum_{i,j \in \overline{S}} x_{ijk} \leq |\overline{S}| - \ell_k(\overline{S})$$

By the property in equation (37), this is equivalent to:

$$\sum_{i,j\in\overline{S}} x_{ijk} \le \sum_{h\in K}\sum_{i,j\in\overline{S}} x_{ijh} + \frac{1}{2}\sum_{h\in K}\sum_{E(S,\overline{S})} x_{ijh} + \frac{1}{2}\sum_{h\in K}\sum_{i\in\overline{S}} x_{0jh} - \ell_k(\overline{S})$$

This is trivially equivalent to:

$$2\sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh} + \sum_{h\in K}\sum_{E(S,\overline{S})} x_{ijh} + \sum_{h\in K}\sum_{i\in\overline{S}} x_{0jh} \ge 2\ell_k(\overline{S}) \tag{38}$$

Adding $2\sum_{h\in K}\sum_{i,j\in S} x_{ijh} + \sum_{h\in K}\sum_{j\in S} x_{0jh}$ to both sides and using identity (37) for S, this yields:

$$2|S| + 2\sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh} + \sum_{h\in K}\sum_{i\in\overline{S}} x_{0jh} \ge 2\ell_k(\overline{S}) + 2\sum_{h\in K}\sum_{i,j\in S} x_{ijh} + \sum_{h\in K}\sum_{j\in S} x_{0jh}$$

Rearranging the terms and dividing both sides of the equation by 2 gives:

$$\sum_{h\in K}\sum_{i,j\in S} x_{ijh} + \frac{1}{2}\left(\sum_{h\in K}\sum_{j\in S} x_{0jh} - \sum_{h\in K}\sum_{j\in\overline{S}} x_{0jh}\right) - \sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh} \le |S| - \ell_k(\overline{S})$$

Now, since $S\cup\overline{S} = \mathcal{L}$ and because $\sum_{i\in\mathcal{L}} x_{0ih} = 2m_h$, for all $h\in K$ per Constraints (1b), this can be rewritten into:

$$\sum_{h\in K}\sum_{i,j\in S} x_{ijh} + \frac{1}{2}\left(\sum_{h\in K}\sum_{j\in S} x_{0jh} - \sum_{h\in K}(2m_h - \sum_{j\in S} x_{0jh})\right) - \sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh} \le |S| - \ell_k(\overline{S})$$

Constraints (3) are easily obtained from this expression by rearranging the terms of this expression.

The validity of Cutting Planes (5)

As for the proof of Cutting Planes (6) for the CVRP, we must define S as the set of customers whose demand must be satisfied and exclude the customers in $\overline{S} = \mathcal{L}\backslash S$ from consideration. Now, S has virtual depot $0_S = \{0\}\cup\overline{S}$, which is identical to the definition of 0_S in the proof of Cutting Planes (6) as derived above. Defining $m_k(S)$ as the number of vehicles of type $k\in K$ used to serve S, one can write Constraints (3) for all $P\subset S$ as follows:

$$\sum_{h\in K}\sum_{i,j\in P} x_{ijh} - \sum_{h\neq k\in K}\sum_{i,j\in S\backslash P} x_{ijh} + \sum_{h\in K}\left(\sum_{i\in 0_S}\sum_{j\in S} x_{ijh} - m_h(S)\right) \le |P| - \ell(\overline{P}) \iff$$

$$\sum_{h\in K}\sum_{i,j\in P} x_{ijh} - \sum_{h\neq k\in K}\sum_{i,j\in S\backslash P} x_{ijh} + \sum_{h\in K}\left((\sum_{j\in S} x_{0jh} + \sum_{E(P,\overline{S})} x_{ijh}) - m_h(S)\right) \le |P| - \ell(\overline{P})$$

Since the Constraint of type (1e) is not violated for S for any vehicle type $k \in K$, it is safe to add $\sum_{i,j \in S} x_{ijh} + s_h$ to the left hand side of the equality and $|S| - \ell_h(S)$ to the right hand side of the inequality, for all $h \in K$. Here, $s_h \geq 0$ represents the slack on the Constraint for vehicle type $h \in K$. This yields the following expression:

$$\sum_{h \in K} \sum_{i,j \in P} x_{ijh} - \sum_{h \neq k \in K} \sum_{i,j \in S \setminus P} x_{ijh} + \sum_{h \in K} \left(\sum_{j \in S} x_{0jh} + \sum_{E(P,\overline{S})} x_{ijh} + \sum_{i,j \in S} x_{ijh} + s_h - m_h(S) \right)$$
$$\leq \sum_{h \in K} (|S| - \ell_h(S)) + |P| - \ell(\overline{P})$$

Rearranging these terms yields:

$$\sum_{h \in K} \sum_{i,j \in P} x_{ijh} - \sum_{h \neq k \in K} \sum_{i,j \in S \setminus P} x_{ijh} + \sum_{h \in K} \left(\sum_{j \in S} x_{0jh} + \sum_{E(P,\overline{S})} x_{ijh} + \sum_{i,j \in S} x_{ijk} \right)$$
$$\leq t|S| + \sum_{h \in K} (m_h(S) - \ell_h(S) - s_h) + |P| - \ell(\overline{P})$$

Now, either $m_h \leq \ell_h(S)$, so that vehicles of other vehicle types are also used to satisfy the aggregate demand in S. Then, $(m_h(S) - \ell_h(S) - s_h) \leq 0$ (as $s_h \geq 0$ so that it can safely be left out of the inequality as it does not harm the validity thereof. If, however, $m_h > \ell_h(S)$, the slack on the Constraint of type(1e) for set S and vehicle type h is (at least) equal to $m_h(S) - \ell_h(S)$ and hence, $(m_h(S) - \ell_h(S) - s_h) \leq 0$. Therefore, $\sum_{h \in K} (m_h(S) - \ell_h(S) - s_h) \leq 0$ can be left out of this equation, proving the validity of the cutting planes.

The validity of Cutting Planes (7) **and** (9)

For the CVRP, Achuthan et al. (2003) prove the validity of Cutting Planes (8) and (10). The following proof of the validity of these cutting planes for the HVRP is very similar to that proof for the CVRP. For the validity of this cutting plane, the following is, again, required:

Let $S, D_1, D_2, \ldots, D_r$ be given such that

- $r \geq 2$ and $\sum_{i \in S \cup D_p \cup D_q} q_i > Q_k$, for all $1 \leq p \neq q \leq r$;
- $D_i \cap D_j = \emptyset$ for $i \neq j$;
- $S \cap D_i = \emptyset$, $\forall i \in \{1, \ldots, r\}$.

The proof is as follows: First, define $D = \cup_{i=1}^{r} D_i$. Then, take (x_{ijk}) as a the component of the feasible solution of the HVRP for vehicle $k \in K$ and consider the subgraph $H = G(S)$, with vertices $i \in S$ and edges $\{(i,j) \in E | x_{ijk} = 1\}$. Obviously, H consists of a number of components (p), each of which is a path. Per elementary graph theory, it follows that:

$$\sum_{i,j \in S} x_{ijk} = \epsilon(H) \leq |S| - p$$

By a similar argument, it must hold that:

$$\sum_{s=1}^{r} \sum_{i,j \in D_s} x_{ijk} \leq |D| - r$$

Here, equality is only possible if the subgraph $G(T_s)$ is a path for all s.

Furthermore, If $p = 1$, S is served by one vehicle of type k and hence, $\sum_{E(S,T)} x_{ijk} \leq 2$.

When these latter two identities hold simultaneously, S and two sets T_s, $T_{s'}$ are all served by the same vehicle of type k, which is not possible since $\sum_{i \in S \cup D_p \cup D_q} q_i > Q_k$, for all $1 \leq p \neq q \leq r$. In analogy to the argument by Achuthan et al. (2003) for the CVRP, this yields:

$$3 \sum_{i,j \in S} x_{ijk} + \sum_{E(S,D)} x_{ijk} + \sum_{s=1}^{r} \sum_{i,j \in D_s} x_{ijk} \leq 3(|S| - 1) + |D| - k + 1$$

If $p \geq 2$, the number of edges between D and each component of S is at most $2p$ since all components in S are served by one vehicle and subtours are not allowed. Therefore, $\sum_{E(S,T)} x_{ijk} \leq 2p$.

This shows the validity of Cutting Planes (7), of which Cutting Planes (9) are a special case.

The validity of Cutting Planes (11)

The proof of the validity of these Cutting Planes are very similar to the proof of the validity of Cutting Planes (12) for the CVRP that is provided by Achuthan et al. (2003). As it is slightly different, the proof is provided below.

Consider $X_1 = [x_{ijk}]$ to be an optimal solution of the HVRP with cost Z_1. Suppose, for the sake of contradiction, that the Cutting Plane of type (11) is not satisfied, and hence that there exists an arbitrary vehicle k and an arbitrary customer set S whose aggregate demand is, at most, Q_k, for which:

$$\sum_{h \leq k \in K} \left(\sum_{i,j \in S} x_{ijh} + \sum_{j \in S} x_{0jh} \right) > |S| + 1$$

Now, consider G to be the graph formed by taking the edges given by the solution X_1. Clearly, G consists of $\sum_{k \in K} m_k$ cycles, with the depot vertex 0 appearing in every cycle. Consider, in particular, the subgraph H which consists of the vertices $\{0\} \cup S$ and any edge between two of those vertices that is assigned to a vehicle $h \leq k \in K$. Evidently, H will be a union of cycles and paths.

Since the Cutting Plane of type (11) is violated, H contains at least two and at most S cycles. Let $\tau_1 = (0, i_1, \dots, i_u, 0)$ and $\tau_2 = (0, j_1, \dots, j_v, 0)$ be two arbitrary cycles in H, driven with a vehicle of an arbitrary type $h \leq k \in K$. It may be noted that u or v can be 1. Now, consider route $\tau = (0, i_1, \dots, i_u, j_1, \dots, j_v, 0)$. This route can be driven with a vehicle of type k, since the aggregate of all customers within the route does not exceed this vehicle's capacity ($\sum_{j \in S} q_j \leq Q_k$), while the original routes were driven with vehicles of type k or larger vehicles.

Define the costs of driving route τ_1 to be driven with its current vehicle $c(\tau_1)$ and the costs of driving route τ_2 to be driven with is current vehicle $c(\tau_2)$.

Therefore, driving route τ with a vehicle of type k is feasible and it has the following costs:

$$c_k(\tau) = c_k(\tau_1) + c_k(\tau_2) - c_{0i_uk} - c_{0j_vk} + c_{i_uj_1k} \leq c(\tau_1) + c(\tau_2) - c_{0i_uk} - c_{0j_vk} + c_{i_uj_1k} \leq c(\tau_1) + c(\tau_2)$$

The first inequality follows since the routes were driven by vehicle k or a larger vehicle and driving a route with a smaller vehicle is never more expensive than driving it with a larger vehicle. The second inequality follows from the triangle inequality.

Now, replacing τ_1 and τ_2 by route τ, driven with vehicle k leads to a solution X_2 that is feasible and has costs at most equal to the costs of X_1. If a Cutting Plane of type (11) was satisfied for any $S \subseteq \mathcal{L}$, it is also satisfied with reference to X_2. This procedure can be repeated until all customers of S are grouped under one tour, if necessary. Repeating this procedure for all $S \subset \mathcal{L}$ that violate Cutting Planes (11) shows that an optimal solution satisfying these Cutting Planes must exist.

The validity of Cutting Planes (15)

The proof of the validity of these Cutting Planes follows directly from the proof of the validity of Cutting Planes (13) for the CVRP that is provided by Achuthan et al. (2003). It follows from considering the proof provided for all vehicle types $k \in K$ separately.

The validity of slack function (27)

It was already shown that inequality (38) is equivalent to Constraints (1e). From this equivalence, it follows that the slack on Constraints (1e) can be rewritten into the slack on inequality (38).

It may be noted that the latter corresponds to the former multiplied by a factor 2. This is not problematic, since we are only interested in the question whether the slack is negative, or not. Therefore, (27) can adequately be used as a slack function for Constraints (1e).

The validity of slack function (29)

The trivial slack function on Cutting Planes (11) for any set S such that $\sum_{i\in S} q_i \leq Q_k$ is given by:

$$Z_3(S) = |S| + 1 - \sum_{h\leq k\in K}\sum_{i,j\in S} x_{ijh} - \sum_{h\leq k\in K}\sum_{i\in S} x_{0ih}$$

Applying identity (37) shows the equivalence to:

$$Z_3'(S) = \sum_{h\in K}\left(\sum_{i,j\in S} x_{ijh} + \frac{1}{2}(\sum_{E(S,\overline{S})} x_{ijh} + \sum_{i\in S} x_{0ih})\right) + 1 - \sum_{h\leq k\in K}\sum_{i,j\in S} x_{ijh} - \sum_{h\leq k\in K}\sum_{i\in S} x_{0ik}$$

Canceling out opposite terms shows that this is equivalent to the following:

$$Z_3'(S) = \sum_{h>k\in K}\left(\sum_{i,j\in S} x_{ijh} + \frac{1}{2}\sum_{i\in S} x_{0ih}\right) + \frac{1}{2}\sum_{h\in K}\sum_{E(S,\overline{S})} x_{ijh} + 1 - \frac{1}{2}\sum_{h\leq k\in K}\sum_{i\in S} x_{0ik}$$

Since it is only of interest whether this slack is negative, or not, this slack function can be multiplied by 2 to yield the desired result.

The validity of slack function (30)

The trivial slack function for Constraints (3) for a given set S reads:

$$Z_4(S) = |S| - \ell_k(\overline{S}) - \sum_{h\in K}\left(\sum_{i,j\in S} x_{ijh} + \sum_{j\in S} x_{0jh} - m_h\right) + \sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh}$$

Applying identity (37), gives:

$$Z_4'(S) = \sum_{h\in K}\left(\sum_{i,j\in S} x_{ijh} + \frac{1}{2}(\sum_{E(S,\overline{S})} x_{ijh} + \sum_{i\in S} x_{oih})\right) - \ell_k(\overline{S})$$
$$- \sum_{h\in K}\left(\sum_{i,j\in S} x_{ijh} + \sum_{j\in S} x_{0jh} - m_h\right) + \sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh}$$

After canceling out the opposite terms, this reduces to the following:

$$Z_4'(S) = \sum_{h\in K}\left(\frac{1}{2}(\sum_{E(S,\overline{S})} x_{ijh} - \sum_{j\in S} x_{0jh}) + m_h\right) - \ell_k(\overline{S}) + \sum_{h\neq k\in K}\sum_{i,j\in\overline{S}} x_{ijh}$$

Since, again, it is only of interest whether this slack function is negative, the function can be multiplied by 2 in its entirety, giving the desired result.

Proof $\alpha_k(S_{lk}) \geq 0$

Since S_{lk} is one of the connected components of the resulting graph for vehicle $k \in K$, and Constraint (1e) are not violated for S_{lk}, it must be true that the aggregate demand in S_{lk} can all be served by one vehicle. Namely, $i \in S_{lk}$ with $q_i > Q_k$ cannot exist since the decision variables on the edges incident to i are not defined for vehicle k. Furthermore, for $i \in S_{lk}$, Constraints (1d) must be satisfied and hence, it can be concluded that $\sum_{h\in K}\sum_{i,j\in S} x_{ijh} = |S| - 1$. Therefore,

$$\alpha_k(S_{lk}) = \sum_{h\in K}\sum_{i,j\in S_{lk}} x_{ijh} + \sum_{j\in S_{lk}} x_{0jk} - |S_{lk}| - \frac{1}{Q_k}\sum_{j\in S_{lk}} q_j \geq \sum_{h\in K}\sum_{i,j\in S_{lk}} x_{ijh} + \sum_{j\in S_{lk}} x_{0jk} - |S_{lk}| - \ell_k(S_{lk})$$

$$= \sum_{h\in K}\sum_{i,j\in S_{lk}} x_{ijh} + \sum_{j\in S_{lk}} x_{0jk} - |S_{lk}| - 1 = \sum_{j\in S_{lk}} x_{0jk} \geq 0$$

Here, the last inequality is valid since $x_{0ih} \in \{0, 1, 2\}$ for all $i \in \mathcal{L}$ and for all $h \in K$.

It may be noted that a similar argument can be generated for $\alpha'_k(S_{lk})$ defined below:

$$\alpha'_k(S_{lk}) = \sum_{h\in K}\sum_{i,j\in S_{lk}} x_{ijh} + \sum_{j\in S_{lk}}\sum_{h\in K} x_{0jh} - |S_{lk}| - \frac{1}{Q_k}\sum_{j\in S_{lk}} q_j$$

However, if multiple vehicles of different types enter S_{lk}, S_{lk} is less likely to contribute to a violated constraint or cutting plane. Therefore, $\alpha_k(S_{lk})$ is preferred to define the order of sets added in search 3.

Appendix D: Gomory cut validity

Suppose that a linear program is given as follows:

$$\underset{x}{\text{minimize}} \quad c^T x \tag{39a}$$

$$\text{subject to} \quad Ax = b \tag{39b}$$

$$x \geq 0 \tag{39c}$$

Note that this program is written in a standard form, in which all variables are required to be non-negative and in which artificial and slack variables are added to constraints to generate constraints that must hold at equality. Note that $[A|b]$ represents a simplex tableau in standard form, for which $A = [N \;\; I]$ is the initial coefficient matrix and b represents the column vector of right hand side coefficients. Selecting the columns from A corresponding to the basic variables in the optimal LP-solution and merging those into an invertible matrix B, one can now write $B^{-1}A = B^{-1}b$ to derive an optimal simplex tableau for the linear program, in which $A^* = B^{-1}A$ represents the coefficient matrix and $b^* = B^{-1}b$ is a column vector of right hand side coefficients. Even if the original coefficients were integers, the coefficients in the final simplex tableau are often fractional.

Take a row A_i^* in which fractional coefficients appear such that b_i^* is fractional too. Since the constraints hold at equality, for the optimal LP-solution x^*, it must hold that

$$\sum_j \lfloor A_{ij}^* \rfloor x_j + \sum_j (A_{ij}^* - \lfloor A_{ij}^* \rfloor) x_j = \sum_j A_{ij}^* x_j^* = b_i^* = \lfloor b_i^* \rfloor + (b_i^* - \lfloor b_i^* \rfloor) \tag{40}$$

If one now enforces that $x_j^* \in \mathbb{N}, \ \forall j$, this means that $\sum_j \lfloor A_{ij}^* \rfloor x_j$ is integer valued. Therefore, $\sum_j (A_{ij}^* - \lfloor A_{ij}^* \rfloor) x_j$ is the fractional part of the left hand side, which must be at least as large as the fractional part of the right hand side: $b_i^* - \lfloor b_i^* \rfloor$ (Gomory, 1958). These need not be equal, since $\sum_j (A_{ij}^* - \lfloor A_{ij}^* \rfloor) x_j$ may be larger than 1 and exceed $b_i^* - \lfloor b_i^* \rfloor$, which is always smaller than 1.

Since the ultimate goal is to derive an optimal solution $x^* \in \mathbb{N}$, the Constraints (41) can be added to the linear program, which eliminate some fractional solutions, but which does not eliminate any feasible solution from consideration.

$$\sum_j (A_{ij}^* - \lfloor A_{ij}^* \rfloor) x_j \geq (b_i^* - \lfloor b_i^* \rfloor) \tag{41}$$

This procedure can be repeated if that if perceived to be beneficial.

Appendix E: Edge weight interpretation

A graphical representation of the edge weight function is provided in Figure 1. Here, $d(u)$ represents to the distance any point u to the center of gravity and $d(0)$ represents the distance of the depot to the center of gravity. It shows that driving a large vehicle near the city center is relatively expensive and decreasing gradually until the costs of 'merely' driving the actual distance, represented by a multiplication factor equal to 1.

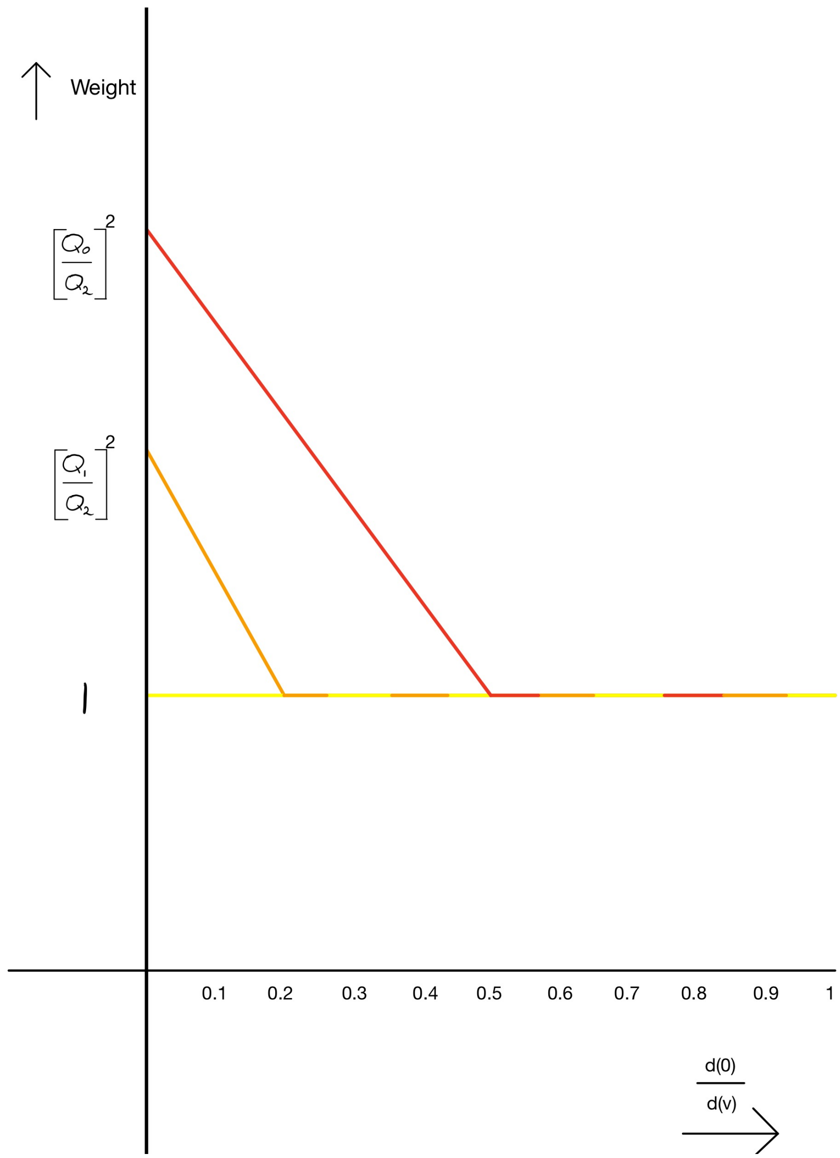

Figure 1: The relative weight of driving any vehicle for a given point v.

Appendix F: Specific results Heterogeneous Fleet Vehicle Routing Problem

In Table 6, the results for the randomly generated HVRP problems are presented. These HVRPs are solved using the exact HVRP branch-and-cut algorithm, which was terminated after 4 hours of computation time. Furthermore, they are solved heuristically, using the sweep heuristic presented in this thesis. Finally, Table 7 contains the success rates of the search procedures.

In both tables, an asterisk (*) is placed in the column 'Number of customers' when a problem was not solved to optimality within these 4 hours. Furthermore, α is rounded to two decimals and the remaining values are rounded to one decimal.

Table 6: Specific results for the HVRP

Number of customers	α	Computation time branch-and-cut (s)	Objective value branch-and-cut	Computation time heuristic (s)	Objective value heuristic	Optimality gap (%)	Vehicle types branch-and-cut	Vehicle types heuristic
13*	0.58	NA	7045.3	0.0	7938.6	12.7	2	2
15*	0.30	NA	8817.2	0.0	9761.7	10.7	3	2
16*	0.47	NA	8381.5	0.0	9301.4	11.0	3	3
19*	0.73	NA	8693.4	0.0	12318.3	41.7	2	3
26*	0.12	NA	25355.7	1.3	22502.3	-11.3	3	3
26*	0.13	NA	18399.1	0.1	25196.4	36.9	3	3
28*	0.79	NA	12318.6	0.1	13743.8	11.6	3	3
32*	0.68	NA	10537.5	0.0	16098.9	52.8	3	3
35*	0.11	NA	21170.4	355.8	30049.8	41.9	3	3
35*	0.61	NA	11050.1	0.1	14684.8	32.9	2	2
35*	0.70	NA	12206.9	13.1	12077.7	-1.1	1	3
40*	0.37	NA	14541.4	401.7	22170.9	52.5	3	3
50*	0.37	NA	16716.6	0.8	24295.7	45.3	2	3

Table 7: Success rates search procedures HVRP

Number of customers	α	Success rate search 1-3 (%)	Success rate search 4 (%)	Success rate search 5 (%)	Success rate search 6 (%)
13*	0.58	0.2	0.0	0.5	0.0
15*	0.30	0.2	0.0	0.5	0.0
16*	0.47	0.1	0.0	4.1	0.0
19*	0.73	0.1	0.1	0.2	0.0
26*	0.12	0.1	0.0	0.4	0.0
26*	0.13	0.0	0.0	1.0	0.0
28*	0.79	0.0	0.0	0.1	0.0
32*	0.68	0.1	0.1	0.4	0.0
35*	0.11	0.1	0.0	1.2	0.0
35*	0.61	0.0	0.0	0.4	0.0
35*	0.70	0.2	0.1	1.9	0.0
40*	0.37	0.0	0.1	1.0	0.0
50*	0.37	0.1	0.3	6.1	0.0