An improved integer L-shaped method for the vehicle routing problem with stochastic demands

Y.N. Hoogendoorn, R. Spliet

Erasmus University Rotterdam, y.n.hoogendoorn@ese.eur.nl, spliet@ese.eur.nl

September 30, 2022

EI-2022-04

Abstract

We present an improved integer L-shaped method for the vehicle routing problem with stochastic demands. It exhibits speedups up to a factor of 325 compared to the current state-of-the-art, which allows us to solve previously unsolved benchmark instances to optimality. The algorithm builds on the state-of-the-art in a few ways. First, we rectify a few technical issues found in the current literature. Secondly, we improve valid inequalities known as partial route inequalities. Finally, we introduce three new types of valid inequalities. Additionally, we analyze two curious modeling choices which are common in the literature. First, we prove that imposing the use of a fixed number of routes can result in an arbitrarily large increase in the optimal objective value, and we prove the same result for additionally imposing that the demand of the customers on a route may not exceed the capacity in expectation. Secondly, our algorithm enables us to perform numerical experiments to illustrate the decrease in computation time, and increase of the optimal solution value which result from imposing these constraints for benchmark instances.

1 Introduction

The vehicle routing problem with stochastic demands (VRPSD) is the problem of designing routes for vehicles of limited capacity to satisfy random demand at customers, while the realization of demand is only learned upon arrival. A recourse action is taken to restock a vehicle when a realized demand exceeds the available load, or earlier to prevent this. The objective is to minimize the total expected costs. This problem has important applications such as the resupply of petrol stations, waste collection and humanitarian aid in disaster areas. In this paper, we present an exact algorithm for the VRPSD, which improves on state-of-the-art algorithms found in the current scientific literature. Furthermore, we continue the investigation started by Florio et al. (2020), to assess the effect of two peculiar modeling choices which are common in the scientific literature on exact algorithms for the VRPSD: i) fixing the number of routes, and ii) limiting the expected demand per route. We provide theoretical and numerical insights on this matter.

We are aware of eleven papers on exact algorithms for the VRPSD, which are listed in the first column of Table 1. This table includes additional information, which we discuss at the end of this introduction. The earliest paper is by Gendreau et al. (1995), while there seems to be an increased interest recently as evidenced by five out of eleven papers appearing since 2018. Different versions of the VRPSD have been considered over these years.

Before 2018, exact algorithms have only been applied in the case of what we call classical recourse, i.e., a corrective return trip to the depot is made when the capacity constraint of a vehicle is violated.

The notable exception is by Gendreau et al. (1995), where some customers do not place an order at all with a certain probability, which can be skipped. Depending on the probability distribution of demand, the expected costs of classical recourse can be evaluated efficiently using numerical methods, which is a computational benefit in the pursuit of exact algorithms.

Since 2018, the papers on exact algorithms for the VRPSD consider preventive recourse, in which it is allowed to make an early return trip to prevent a more expensive corrective return trip later. Note that optimal preventive recourse yields lower expected costs than classical recourse. Preventive recourse has only been applied for discrete demand distributions. In this case, the preventive recourse actions can be optimized using for example the dynamic programming algorithm of Yang et al. (2000), as applied by e.g. Louveaux and Salazar-González (2018). Instead of using optimal preventive recourse actions, Salavati-Khoshghalb et al. (2019b) and Salavati-Khoshghalb et al. (2019c) use a different policy to decide on taking preventive recourse. In this paper, we present an algorithm for both the VRPSD with classical and preventive recourse, capable of handling all of the demand distributions encountered in earlier work.

The exact algorithms currently found in the literature can be divided into two categories: i) integer L-shaped methods, appearing in eight out of eleven papers, and ii) branch-price-and-cut algorithms, appearing in the remaining three papers. Both types of algorithms employ branch-and-bound. The integer L-shaped methods make use of an edge-flow formulation, or arc-flow in the case of Louveaux and Salazar-González (2018), while the branch-price-and-cut algorithms make use of a set-partitioning formulation, or set-cover in the case of Christiansen and Lysgaard (2007). Essentially, the integer L-shaped methods deal with the expected costs of recourse in the separation of optimality cuts, while the branch-price-and-cut algorithms deal with this is in the pricing of new variables.

The branch-price-and-cut algorithms benefit from the strong LP-bound of the set-partitioning formulation. However, the pricing algorithms have pseudo-polynomial worst-case computation time, and particularly the dominance criterion seems to be negatively affected by the size of the vehicle capacity. This is perhaps the reason that for instance Florio et al. (2020) are able to report solving to optimality only those benchmark instances with a vehicle capacity that typically does not exceed 160 (with one notable exception of `P-n22-k8` with vehicle capacity 3000), while the vehicle capacity of unsolved benchmark instances range into the thousands.

An integer L-shaped method is an exact algorithm for a stochastic integer programming problem with complete recourse, introduced by Laporte and Louveaux (1993). By modelling the VRPSD as a two-stage stochastic programming problem, this algorithm can be applied. Lower bounds on the expected costs of recourse and valid inequalities specific to the VRPSD have been introduced to enhance the computational performance of the integer L-shaped method for the VRPSD. While current branch-price-and-cut algorithms for the VRPSD can reasonably be applied only to instances with discrete demand distributions with a small support and vehicle capacity, in which case they are usually the superior algorithm, the integer L-shaped methods are more widely applicable. In this paper, we present an improved integer L-shaped method for the VRPSD.

Our integer L-shaped method makes use of lower bounds on the expected costs of recourse of Laporte et al. (2002) and Louveaux and Salazar-González (2018), the latter of which we generalize for arbitrary demand distributions. We have also improved the lower bounds by Salavati-Khoshghalb et al. (2019a). However, even the improved bounds turn out to be of no computational benefit for common benchmark instances.

We also incorporate valid inequalities known as partial route inequalities. These were introduced by Hjorring and Holt (1999) for the VRPSD with classical recourse in case only a single vehicle is available, extended to the multi-vehicle case by Laporte et al. (2002), and generalized further by Jabali et al. (2014). We address a mistake in the partial route inequalities by Laporte et al. (2002) and those by Jabali et al. (2014). Moreover, we present new partial route inequalities which are stronger. Furthermore, Salavati-Khoshghalb et al. (2019a) and Louveaux and Salazar-González (2018) provide versions of the partial route inequalities for the VRPSD with preventive recourse, which we also improve.

Additionally, we present three new types of valid inequalities, which we refer to as route-split inequalities, partial route-split inequalities, and multi-route-split inequalities. Whereas integer L-shaped methods traditionally have treated the expected costs of recourse as a single value, we split this value up into parts which can be attributed to separate routes. We capitalize on this, resulting in three new types of valid inequalities. Note, that the route-split inequalities can in particular also be used as optimality cuts, which lie at the heart of any integer L-shaped method, potentially replacing the standard optimality cuts of Laporte et al. (2002).

We provide results of numerical experiments in which we apply our integer L-shaped method to benchmark instances from the literature. These results suggest that our algorithm outperforms the other integer L-shaped methods, and we are able to solve 152 previously unsolved benchmark instances to optimality within a runtime of one hour per instance. Current branch-price-and-cut algorithms are not suited to be applied on a wide variety of instances, such that our integer L-shaped method is by default superior on many benchmark instances from the literature. Furthermore, we also compare the performance of our integer L-shaped method to that of a branch-price-and-cut algorithm on those instances for which the latter is specifically well-suited. The results suggest that our algorithm even outperforms the branch-price-and cut algorithms on some, but not all, of these benchmark instances as well.

The VRPSD is a computationally difficult problem. It is NP-hard since the traveling salesman problem can be reduced to it in polynomial time. The recourse function as part of the objective value adds to the computational challenge. Nonetheless, instances of respectable size have been solved to optimality, e.g. Florio et al. (2020) report: "*Instances of moderate size (up to 76 nodes) could be solved in reasonable time (up to five hours), and larger instances (up to 148 nodes) could be solved in long runs of the algorithm.*" We further improve on these computational results with our integer L-shaped method. However, Florio et al. (2020) also provide evidence that these formidable results might be due to computational advantages resulting from two curious modeling choices in the VRPSD literature which we discuss next.

In the paper by Laporte et al. (2002), the expected capacity constraints are incorporated in the VRPSD. These constraints ensure that the expected total demand on a route may not exceed the vehicle capacity. All subsequent papers on exact algorithms for the VRPSD also include these constraints, until Florio et al. (2020) questioned their use. It is clear that the inclusion of the expected capacity constraints speeds up exact algorithms. But from a modeling perspective the solution quality may suffer unnecessarily from these constraints. Florio et al. (2020) report that their exact algorithm typically cannot solve instances in reasonable time if the expected capacity constraints are not imposed, while heuristic solutions to the VRPSD in which the vehicle capacity is increased provide numerical evidence on the loss of solution quality. In this paper, we prove that imposing the expected capacity constraints can result in an arbitrarily large increase of the objective value. Moreover, we present the results of numerical experiments, in which we use our exact algorithm to find the optimal solution of the VRPSD without these constraints. This helps indicate the size of the instances which can be solved by a state-of-the-art exact algorithm in the absence of the expected capacity constraints. Moreover, it gives insight into the magnitude by which solution quality suffers for benchmark instances.

A similar case is made for imposing that a fixed number of routes have to be used. We show that this constraint can also result in an arbitrarily large increase of the objective value. Interestingly, all papers in which an integer L-shaped method is used also impose a constraint on the number of used routes, while this is not done for branch-price-and-cut algorithms.

To facilitate our discussion, we refer to the VRPSD without expected capacity constraints as the fixed

Table 1: Overview of scientific literature on exact algorithms for the VRPSD.

Paper	Recourse	Distribution	Method	ECC	FRC
Gendreau et al. (1995)	Classical+	Discrete	L	No	Yes
Hjorring and Holt (1999)	Classical	Discrete	L	No	Yes
Laporte et al. (2002)	Classical	Normal, Poisson	L	Yes	Yes
Christiansen and Lysgaard (2007)	Classical	Poisson	BPC	Yes	No
Jabali et al. (2014)	Classical	Normal	L	Yes	Yes
Gauvin et al. (2014)	Classical	Poisson	BPC	Yes	No
Louveaux and Salazar-González (2018)	Preventive	Discrete triangular	L	Yes	Yes
Salavati-Khoshghalb et al. (2019a)	Preventive	Discrete triangular	L	Yes	Yes
Salavati-Khoshghalb et al. (2019b)	Preventive-	Discrete triangular	L	Yes	Yes
Salavati-Khoshghalb et al. (2019c)	Preventive-	Discrete triangular	L	Yes	Yes
Florio et al. (2020)	Preventive	Poisson	BPC	Yes+No	No

routes VRPSD (FR-VRPSD), and the FR-VRPSD without a fixed number of routes as the Basic-VRPSD. Our integer L-shaped method for the VRPSD uses valid inequalities and lower bounds which exploit the constraint on the fixed number of routes, but does not rely on the expected capacity constraints. Therefore, our integer L-shaped method for the VRPSD can immediately be applied to the FR-VRPSD, but not to the Basic-VRPSD. Throughout this paper, we also indicate how the integer L-shaped method is modified for the Basic-VRPSD. Note that Florio et al. (2021) also do not include these constraints, but impose a route duration constraint instead. We consider that constraint beyond the scope of this paper.

As an overview on the literature on exact algorithms for the VRPSD, we provide Table 1. In the column 'Paper', we list the papers. In the colum 'Recourse', the type of recourse of each paper is indicated, where 'Classical' refers to classical recourse, 'Classical+' to classical recourse where additionally the customers which have no demand can be skipped, 'Preventive' refers to optimized preventive recourse, and 'Preventive-' to preventive recourse where actions are determined according to a potentially suboptimal policy. In the column 'Distribution', the demand distribution under consideration in each paper is provided. In the column 'Method', we indicate the algorithm, where 'L' refers to integer L-shaped method, and 'BPC' refers to branch-price-and-cut. The columns 'ECC' and 'FRC' indicate whether expected capacity constraints and a fixed number of routes constraint is imposed, respectively.

This paper is structured as follows. First, in Section 2 we describe the Basic-VRPSD, FR-VRPSD and the VRPSD. Here we also discuss the expected recourse constraints and fixed number of routes constraint. The basics of the integer L-shaped method are explained in Section 3. The basic integer L-shaped method is improved with lower bounds on the expected costs of recourse in Section 4, partial route inequalities in Section 5, and route-split inequalities, partial route-split inequalities, and multi-route-split inequalities in Section 6. Finally, we provide the results of our numerical experiments in Section 7, and conclude in Section 8.

2 Problem description

In this section, we describe the VRPSD. For the sake of exposition, we introduce several versions in which we gradually include more constraints. We start with a basic version of the VRPSD in Section 2.1, which we refer to as the Basic-VRPSD. We then introduce the constraint that fixes the number of routes in Section 2.2, resulting in the version which we refer to as the FR-VRPSD. We prove that imposing this constraint can result in an arbitrarily large increase of the optimal solution value. In Section 2.3, we additionally introduce the expected capacity constraints. Because this results in the most common version of the VRPSD in the literature on exact algorithms, we refer to it as the VRPSD for the remainder of this paper. We prove that imposing the expected capacity constraints, can also result in an arbitrarily large increase of the optimal solution value. Each of these problems include a so-called recourse model. In Section 2.4, we discuss some computational aspects of the two different recourse models from the literature which are considered in this paper, classical and preventive. This provides six problems of interest in this paper: the Basic-VRPSD FR-VRPSD and VRPSD with classical and preventive recourse.

2.1 Basic vehicle routing problem with stochastic demands

Let $G = (V, E)$ be a complete undirected graph, with $V = \{0\} \cup V'$ the set of vertices such that 0 represents the depot and V' the customers. For each customer $v \in V'$, let $D_v \geq 0$ be the nonnegative random demand, such that all demands are independently distributed according to some known distribution, and the mean demand of customer v is μ_v. The demands are satisfied by visiting the customers using a homogeneous fleet of vehicles with capacity $C > 0$. A feasible solution consists of a collection of routes, which are simple cycles in G that start and end at the depot 0, such that each customer is included exactly once. With each edge $e \in E$, a nonnegative travel costs $c_e \geq 0$ is associated. We refer to the total travel costs corresponding with a collection of routes as the first stage travel costs.

Once the routes have been constructed, the vehicles execute these routes. Only when reaching a customer, the corresponding demand realization is observed. Therefore, it may occur that a route fails. That is, the current amount of goods in the vehicle is not enough to fully satisfy the demand of the current customer. When a route fails, a recourse action needs to be taken. We consider two recourse models: classical and preventive recourse. In both models, the vehicle leaves the depot with a full load C. Then, if the route fails, the vehicle empties its load at the customer, after which it travels back to the depot, is loaded to its capacity C and travels back to the customer to resume its route. We call such a return trip a corrective return trip. Additionally, in the case of preventive recourse, the vehicle may execute a preventive return trip. In this case, after serving a customer the vehicle travels back to the depot, is loaded to its capacity, and travels to the next customer on the route. The objective is to minimize the first stage travel costs plus the expected costs of recourse. We refer to this problem as the Basic-VRPSD.

Next, we present a two-stage stochastic programming formulation of the Basic-VRPSD. In this formulation, x_e is the decision variable corresponding to the number of times edge $e \in E$ is traversed. Furthermore, we denote by $Q(x)$ the expected costs of recourse given the routing solution $x \equiv (x_e)_{e \in E}$. Finally, $\delta(v)$ is the set of edges with one endpoint equal to $v \in V$ and $E(S)$ the set of edges with both endpoints in $S \subseteq V$. A two-stage stochastic programming formulation of the Basic-VRPSD is

$$\text{minimize} \quad \sum_{e \in E} c_e x_e + Q(x), \tag{1}$$

$$\text{subject to} \quad \sum_{e \in \delta(v)} x_e = 2 \qquad \forall v \in V', \tag{2}$$

$$\sum_{e \in E(S)} x_e \leq |S| - 1 \quad \forall S \subseteq V', S \neq \emptyset, \tag{3}$$

$$x_e \in \{0, 1\} \qquad \forall e \in E \setminus \delta(0), \tag{4}$$

$$x_e \in \{0, 1, 2\} \quad \forall e \in \delta(0). \tag{5}$$

The objective function (1) consists of the first stage travel costs and the expected costs of recourse.

Constraints (2) are the flow conservation constraints. Constraints (3) are the subtour elimination constraints and Constraints (4) and (5) are the integrality conditions.

Note that $Q(x)$ is only defined for feasible integer solutions. Nonetheless, throughout this paper we also consider non-integer solutions x such that $0 \leq x_e \leq 1$ for all $e \in E \setminus \delta(0)$, $0 \leq x_e \leq 2$ for all $e \in \delta(0)$, (2) and (3). For convenience, we refer to such solutions as solutions to the continuous relaxation of the Basic-VRPSD.

2.2 Fixed routes vehicle routing problem with stochastic demands

In the literature, the number of routes is often fixed to some integer m:

$$\sum_{e \in \delta(0)} x_e = 2m. \tag{6}$$

We refer to (1)-(6) as the fixed routes vehicle routing problem with stochastic demands (FR-VRPSD). The fixed number of routes constraint was already imposed by Stewart Jr and Golden (1983) and Dror et al. (1989). Fixing the number of routes is also common in the deterministic counterpart of the VRPSD, the CVRP. This constraint reflects the notion that a single route is executed by a unique vehicle. Although it is debatable whether such a requirement is reasonable for both the VRPSD and CVRP due to the absence of time or distance constraints, we put this issue aside here. However, while in the CVRP the number of routes corresponds to the number of departures from and returns to the depot, in the VRPSD only some of these are considered by (6). Indeed, the total number of returns consists of the deterministic returns, as well as stochastic returns due to recourse, and Constraint (6) imposes a limit on the number of deterministic returns only. Unless there is a good reason to distinguish between deterministic an stochastic return trips, the fixed number of routes constraint can be considered as rather arbitrary.

Another observation is that limiting the number of routes weakly increases the optimal objective value. In fact, the optimal solution value might get arbitrarily bad. This is already known for the CVRP, and carries over straightforwardly to the VRPSD. For convenience, we say that an instance of the FR-VRPSD is also an instance of the Basic-VRPSD, achieved by simply omitting m.

Theorem 1. *The worst-case difference between the optimal solution value of the FR-VRPSD and the Basic-VRPSD for the same instance, is arbitrarily large.*

Proof. Consider an instance of the FR-VRPSD with k customers at distance 1 from the depot and zero from each other. Demand is so small, that the expected recourse costs is negligible for all routes, irrespective of the recourse model. Furthermore, $m = k$. The optimal solution to the basic-VRPSD is to visit all customers on a single route at costs 2, while imposing that exactly k routes must be used, results in the optimal solution value of $2k$. Hence, by increasing k the difference in optimal solution values grows arbitrarily large. □

2.3 Vehicle routing problem with stochastic demands

Instead of the subtour elimination constraints (3), it is common in the scientific literature to use the expected capacity constraints:

$$\sum_{e \in E(S)} x_e \leq |S| - \left\lceil \frac{1}{C} \sum_{i \in S} \mu_i \right\rceil \qquad \forall S \subseteq V'. \tag{7}$$

The expected capacity constraints are more limiting than the subtour elimination constraints, and make sure that the total expected demand on a route does not exceed the vehicle capacity C. We refer to (1), (2), (4)-(7) as the vehicle routing problem with stochastic demands (VRPSD). Similar to before, we refer to any solution x such that $0 \leq x_e \leq 1$ for all $e \in E \setminus \delta(0)$, $0 \leq x_e \leq 2$ for all $e \in \delta(0)$, (2), (6) and (7) as a solution to the continuous relaxation of the VRPSD.

The expected capacity constraints (7) were introduced by Laporte et al. (2002). The practical value was argued by stating that "*otherwise some routes will systematically fail while on others vehicles will be highly underutilized*" (Laporte et al., 2002, p.416). Depending on the application this may, or may not be an issue. Moreover, the net effect is merely to potentially impose a deterministic return trip instead of stochastic return trips. In fact, imposing the expected capacity constraints can result in structurally more return trips in total, i.e., deterministic and expected number of stochastic return trips combined.

Furthermore, expected capacity constraints are more limiting than the subtour elimination constraints, and therefore weakly increase the optimal objective value. Similar to before, we say for convenience that an instance of the VRPSD is also an instance of the FR-VRPSD, and by extension of the Basic-VRPSD. We show, that the increase in the optimal solution value can grow arbitrarily large from imposing the expected capacity constraints. We compare in particular the optimal solution value of the VRPSD and the FR-VRPSD.

Theorem 2. *The worst-case difference between the optimal solution value of the VRPSD and the FR-VRPSD for the same instance, is arbitrarily large.*

Proof. Consider an instance with $2k$ customers, for k an arbitrary integer larger than 1. Let us name the first k customers $A_1, \ldots, A_k$. Demand is normally distributed with means $\mu_{A_i} = 1 - \frac{1}{2k}$ and standard deviations $\sigma_{A_i} = 0.1\mu_{A_i}$. The last k customers are $B_1, \ldots, B_k$, with normally distributed demand with means $\mu_{B_i} = \frac{1}{2k}$ and standard deviations $\sigma_{B_i} = 0.1\mu_{B_i}$. All customers A_i are located at the depot, whereas all B_i customers lie at a distance 1 from the depot. That is, $c_{(0,A_i)} = c_{(A_i,A_j)} = c_{(B_i,B_j)} = 0$ and $c_{(0,B_i)} = c_{(A_i,B_j)} = 1$, for all $i, j \in \{1, \ldots, k\}$. Furthermore, let $C = 1$ and $m = k$. An optimal solution is $\{(0, B_1, \ldots, B_k, 0), (0, A_1, A_2, 0), (0, A_3, 0), \ldots, (0, A_k, 0)\}$ with expected costs 2. When imposing the expected capacity constraints, an optimal solution is k routes $(0, B_i, A_i, 0)$ for $i \in \{1, \ldots, k\}$ with expected costs $2k$. It follows that by increasing k, the difference in optimal solution values grows arbitrarily large. □

2.4 Recourse models

The Basic-VRPSD, FR-VRPSD and the VRPSD make use of a function $Q(x)$, providing the expected costs of recourse given a feasible routing solution x. This function is different depending on the recourse model and domain of the demand distribution.

However, both classical and preventive recourse share the property that the expected costs of recourse can be decomposed over the routes and their orientation. Note that given one of the two orientations $o \in \{1, 2\}$ of some route r, the order of visited customers is fixed. Denoting Q_r^o as the expected costs of recourse for some route r with orientation o, we have

$$Q(x) = \sum_{r \in R} \min\{Q_r^1, Q_r^2\}, \tag{8}$$

with R the set of routes corresponding with solution x. For the remainder of this paper, we denote $s_v \equiv 2c_{(0,v)}$ as the costs of corrective recourse at vertex $v \in V'$ and $s_{(u,v)} \equiv c_{(0,u)} + c_{(0,v)} - c_{(u,v)}$ as the costs of preventive recourse between vertices $u, v \in V'$, $u < v$.

In the case of classical recourse and a continuous demand distribution, given that $r = (0 = v(0), v(1), \ldots, v(p), v(p+1) = 0)$ with orientation o, Dror et al. (1989) provide the following expression for Q_r^o:

$$Q_r^o = \sum_{i=1}^{p} s_{v(i)} \sum_{k=1}^{\infty} \mathbb{P}\left(\sum_{j=1}^{i-1} D_{v(j)} \leq kC, \sum_{j=1}^{i} D_{v(j)} > kC \right).$$

Note that in this case, a return trip is still not immediately made when the load is exactly zero after having visited a customer, but is postponed until the route fails at the next customer. This event has probability zero when the demand distribution is continuous, while for a discrete demand distribution this event might have nonzero probability. In that case, it is reasonable and tractable to perform a return also whenever the vehicle is fully empty as described by Hjorring and Holt (1999).

In the case of preventive recourse, it needs to be decided when to execute preventive return trips. When in the case of preventive recourse throughout this paper, we assume demand to be discrete, allowing the use of standard dynamic programming algorithms from the literature to optimize these decisions for each route separately, and to compute the expected costs of recourse. Similar to Yang et al. (2000), denoting $r = (0 = v(0), v(1), \ldots, v(p), v(p+1) = 0)$ for route r with fixed orientation o, we define $f_j^*(q)$ as the optimal expected costs of recourse incurred on route r after serving customer $v(j)$, given that the load is $q \in \{0, \ldots, C\}$. The initial condition is $f_{p+1}^*(q) = 0$ for all q, and the other $f_j^*(q)$ are defined recursively as:

$$f_j^*(q) = \min \left\{ \begin{array}{l} s_{(v(j),v(j+1))} + \sum_{\xi \in \Xi_{v(j+1)}} \mathbb{P}(D_{v(j+1)} = \xi) \left[\Gamma(\xi - C) s_{v(j+1)} + f_{j+1}^*(C\Gamma(\xi - C) + C - \xi) \right], \\ \sum_{\xi \in \Xi_{v(j+1)}} \mathbb{P}(D_{v(j+1)} = \xi) \left[\Gamma(\xi - q) s_{v(j+1)} + f_{j+1}^*(C\Gamma(\xi - q) + q - \xi) \right] \end{array} \right\}, \tag{9}$$

with Ξ_j the set of all demand realizations of D_j with nonzero probability, and $\Gamma(x) = \max\left\{ \left\lceil \frac{1}{C} x \right\rceil, 0 \right\}$ denotes the number of corrective return trips needed when demand exceeds the available load by x. Note that this dynamic program is adapted from Yang et al. (2000) similar to Louveaux and Salazar-González (2018), by subtracting the deterministic first stage travel costs, and accounting for the case when demand exceeds C with positive probability. Using this recursive formula, it follows that $Q_r^o = f_0^*(C)$, as we assume the vehicle leaves the depot with a full load C.

3 Integer L-shaped method

In this section, we describe an integer L-shaped method for the VRPSD. In subsequent sections we provide lower bounds and valid inequalities which enhance the performance of the algorithm. Throughout this paper, we also indicate how the integer L-shaped method is modified for the Basic-VRPSD. For example, the integer L-shaped methods for the VRPSD as found in the literature, typically make use of an optimality cut which is not valid for the Basic-VRPSD. Therefore, we introduce a new optimality cut for the latter case in this section. In this paper, we do not additionally discuss an integer L-shaped method for the FR-VRPSD. For the purpose of this paper, it is sufficient to apply the integer L-shaped method for the VRPSD to the FR-VRPSD, but replacing the expected capacity constraints with subtour elimination constraints. This is what we have done in our implementation.

The objective function (1) includes the function $Q(x)$ which might not be linear in x, and is only defined for integer values of x. Therefore, a variable θ is introduced in the integer L-shaped method, which will take the value of the expected costs of recourse of the optimal solution. The objective function (1) is replaced by

$$\min \sum_{e \in E} c_e x_e + \theta \tag{10}$$

and the constraint

$$\theta \geq Q(x) \tag{11}$$

is added, so that (10), (11), (2), (4)-(7) is a formulation of the VRPSD, and (10), (11), (2)-(5) is a formulation of the Basic-VRPSD.

In the integer L-shaped method, Constraint (11) is initially relaxed, and the constraint $\theta \geq L$ is added, where L is a lower bound on $Q(x)$ for all x. Note that for the VRPSD and Basic-VRPSD, it suffices to select $L = 0$, but stronger bounds are preferred.

The resulting mixed integer linear programming problem is solved using a branch-and-cut algorithm. Observe that the formulation includes an exponential number of capacity constraints (7) in case of the VRPSD, or subtour elimination constraints (3) in case of the Basic-VRPSD. These constraints are initially relaxed as well and added in a cutting planes fashion. As is common, we separate expected capacity constraints heuristically using the CVRPSEP package (Lysgaard, 2003), and separate violated subtour elimination constraints exactly in polynomial time by solving a maximum flow problem.

Whenever an integer solution $\hat{x}$ is found, $Q(\hat{x})$ is computed. If $\hat{x}$ is the solution with the lowest objective value found so far, it is stored. Next, it is checked whether the relaxed Constraint (11) is violated, i.e., whether $\theta < Q(\hat{x})$. In case of a violation, the integer solution $\hat{x}$ is removed from the feasible space by using an optimality cut. Subsequently, the current node of the branching tree is processed again, i.e., the corresponding LP relaxation which now includes the optimality cut is solved.

When applying the integer L-shaped method to the VRPSD, the following optimality cut by Laporte et al. (2002) is used:

$$\sum_{e \in E \setminus \delta(0): \hat{x}_e = 1} x_e \leq \sum_{e \in E \setminus \delta(0)} \hat{x}_e - 1. \tag{12}$$

This optimality cut forbids any integer solution that uses all the non-depot edges of $\hat{x}$. It is important to note that this cut is correct for the formulation of the VRPSD, but it is incorrect for the Basic-VRPSD which does not include Constraint (6) on the number of routes. To demonstrate that the optimality cut (12) is incorrect for the Basic-VRPSD, observe that cutting off the solution corresponding with the two routes $\{(0, 1, 2, 0), (0, 3, 4, 0)\}$ using (12) also cuts off the solution corresponding with the single route $\{(0, 1, 2, 3, 4, 0)\}$. Simply adding the depot edges $\delta(0)$ to both summations of (12) does not resolve the issue, as then the optimality cut also cuts off the solution corresponding with the three routes $\{(0, 1, 0), (0, 2, 0), (0, 3, 4, 0)\}$.

Instead, we introduce a new optimality cut. Given a route r, a function with as domain the feasible solutions x of the continuous relaxation of the Basic-VRPSD, is called a route activation function of r if it satisfies the following properties: i) it has the value 1 if and only if x is integer feasible and corresponds with a solution which includes the route r, and ii) if x is integer feasible and corresponds with a solution which does not include the route r, then it has a nonpositive value. Given a route

$r = (0 = v(0), v(1), \ldots, v(p), v(p+1) = 0)$, consider the function

$$W_r(x) = \begin{cases} x_{(0,v(1))} - 1 & \text{if } p = 1, \\ x_{(0,v(1))} + 3x_{(v(1),v(2))} + x_{(0,v(2))} - 4 & \text{if } p = 2, \\ x_{(0,v(1))} + 2x_{(v(1),v(2))} + \sum_{k=2}^{p-2} x_{(v(k),v(k+1))} + 2x_{(v(p-1),v(p))} + x_{(0,v(p))} - (p+2) & \text{otherwise.} \end{cases} \tag{13}$$

In Appendix A, we prove that the function $W_r(x)$ is a route activation function of r. Moreover, $W_r(x)$ is linear. Using $W_r(x)$, we define the following linear optimality cut for the Basic-VRPSD, which cuts of the integer solution x corresponding to the route set R:

$$\sum_{r \in R} W_r(x) \leq |R| - 1. \tag{14}$$

It is important to note that the integer L-shaped method in its current form may enumerate an enormous amount of integer solutions. Denoting by z^* the optimal solution value, all feasible solutions with first-stage costs less than $z^* - L$ will be enumerated. Roughly speaking, this means that the algorithm will perform poorly if the difference between the lower bound L and the expected costs of recourse of the optimal solution is large, compared to the differences in first stage travel costs of many good solutions. It will perform well if the optimal expected costs of recourse is slightly higher than its lower bound, and there are only few good solutions. But in the latter case, the value of the stochastic solution is probably low and the benefit of using an integer L-shaped method is limited to begin with. Therefore, besides choosing the lower bound L as strong as possible, it is important to improve the algorithm, which can be done by including valid inequalities.

4 Lower bounds on the expected costs of recourse

In this section, we discuss lower bounds on the expected costs of recourse, i.e., values of L such that $L \leq Q(x)$ for all x. We use these lower bounds in our implementation of the integer L-shaped method described in Section 3.

For the Basic-VRPSD, we simply use the trivial lower bound 0. Observe that if the probability of the nonnegative demand being greater than the vehicle capacity is zero for all customers, then the minimal expected costs of recourse is 0, obtained for a solution with one route per customer. Therefore, lower bounds for many benchmark instances from the literature will not deviate much from 0. Because we use such instances in our numerical experiments, we do not pursue stronger lower bounds for the Basic-VRPSD in this paper.

For the remainder of this section we consider the VRPSD. First, we merely mention two lower bounds defined by Laporte et al. (2002) for the case of VRPSD with classical recourse. One of the lower bounds applies when demand is normally distributed, the other when demand is Poisson distributed. We refer to Laporte et al. (2002) for details, and use these lower bounds in our numerical experiments on the VRPSD with normal or Poisson distributed demand.

Second, Louveaux and Salazar-González (2018) provide a lower bound that applies to both the VRPSD with classical and preventive recourse. They present it for the case that the probability of demand exceeding the vehicle capacity is 0, and mention that the lower bound could be adjusted when this is not the case. We present the lower bound in a form such that it also applies if this probability is nonzero.

If the support of demand is bounded, then $\max\{\Xi_v\}$ is the finite-valued largest realization of demand of customer $v \in V'$. In this case, denote by $\mathcal{A}$ the ordered set containing $\lceil \frac{1}{C} \max\{\Xi_v\} \rceil$ copies of the costs of recourse s_v, for all $v \in V'$, in increasing order. Additionally include one copy of each s_e, $e \in E$, in the case of preventive recourse. If the support of demand is unbounded for at least one customer, let $w \in V'$ be the customer with the smallest value s_w among all customers with unbounded support. Now $\mathcal{A}$ consists of $\lceil \frac{1}{C} \max\{\Xi_v\} \rceil$ copies of the costs of recourse s_v, for all $v \in V'$ such that $s_v < s_w$, and one copy of s_e, for all $e \in E$ such that $s_e < s_w$, all sorted in increasing order. This is followed by infinitely many copies of s_w. Denote by α_j the recourse costs at position j in the ordered set $\mathcal{A}$. Then,

$$\sum_{j=1}^{|\mathcal{A}|} \alpha_j \mathbb{P}\left(\sum_{v \in V'} D_v > C(j+m-1)\right) \tag{15}$$

is a lower bound on the expected costs of recourse for the following reason. Since we assume independently distributed demand, the number of preventive and corrective return trips would be smallest if all customers would be assigned to a single route. Incurring a lower bound α_j on the costs of recourse for every j-th return trip in excess of $m-1$ with the probability that $j+m-1$ return trips are required, gives a lower bound on the expected costs of recourse. Observe that if the largest demand realization does not exceed the vehicle capacity, i.e., the demand has bounded support and $\max\{\Xi_v\} \leq C$, for all customers $v \in V'$, this lower bound coincides with that of Louveaux and Salazar-González (2018).

Finally, there is a more recent lower bound on the expected costs of recourse that requires attention, introduced by Salavati-Khoshghalb et al. (2019a) for the VRPSD with preventive recourse. The presentation of this lower bound as found in their paper seems to contain a mistake, which is easily rectified. Moreover, we improve on their bound. Unfortunately, we conclude that even the improved version is weaker than the trivial lower bound 0 for all tested instances. Therefore, we do not use this bound in our implementation of the integer L-shaped method. The details of this discussion can be found in Appendix B. Note that the lower bound of Salavati-Khoshghalb et al. (2019a) relies on so-called partial routes, which we describe in Section 5.

We summarize the lower bounds on the expected costs of recourse which we use in our implementation of the integer L-shaped method. i) For the Basic-VRPSD, we use the trivial lower bound 0. ii) For the VRPSD with classical recourse and normally or Poisson distributed demand, we use the largest of the lower bounds by Laporte et al. (2002) and (15). iii) For the VRPSD with classical recourse and demand that is not normally or Poisson distributed, and for the VRPSD with preventive recourse, we use (15).

5 Partial route inequalities

Partial routes are a generalization of routes, and are used to construct valid inequalities for the VRPSD, referred to as partial route inequalities. They were first introduced in Hjorring and Holt (1999) for the single-vehicle VRPSD. Laporte et al. (2002) used them in the multi-vehicle VRPSD. Furthermore, Jabali et al. (2014) extended the definition of partial routes by introducing new types, resulting in so-called generalized partial routes.

In Section 5.1, for ease of writing we first introduce an alternative definition of the generalized partial routes of Jabali et al. (2014). Next, we present lower bounds on the expected costs of recourse for partial routes in Section 5.2. We present the lower bound of Laporte et al. (2002) for classical recourse, which we use in this paper, and introduce new lower bounds for preventive recourse which improve on those by Salavati-Khoshghalb et al. (2019a) and Louveaux and Salazar-González (2018). In Section 5.3, we present the partial route inequalities introduced by Laporte et al. (2002) for the VRPSD, and note how they can be used for the Basic-VRPSD. The partial route inequalities can be considered a framework for valid inequalities, which make use of lower bounds for partial routes as discussed in Section 5.2, and partial route activation functions which we describe in Section 5.3. We discuss a mistake in both the partial route activation functions introduced by Laporte et al. (2002) and introduced by Jabali et al. (2014), and suggest a new and improved partial route activation function. In Section 5.4, we describe our algorithm for separating partial route inequalities.

5.1 Partial route

A partial route reflects a partial ordering of customers on a route. A set of customers is grouped into disjoint subsets, and these subsets are ordered. The ordering of the customers within one subset is not fixed, while all customers in a preceding subset should be visited before all customers in subsequent subsets. To be precise, an ordered set $h = (U_0, \ldots, U_b)$ of subsets of V, with $b \geq 2$, is a partial route if and only if: i) the first and last subset consists only of the depot, i.e., $U_0 = U_b = \{0\}$, and the depot is not included in any other subset, ii) all subsets are disjoint except the first and last, i.e., $U_i \cap U_j = \emptyset$, for all $0 \leq i \neq j < b$, and iii) every subset containing strictly more than one customer is immediately preceded and succeeded by a singleton, i.e., $|U_i| > 1 \implies |U_{i-1}| = |U_{i+1}| = 1$ for all $i \in \{1, \ldots, b-1\}$. A subset with strictly more than one customer, which is part of a partial route, is referred to as an unstructured component.

We use the term partial route, even though our definition is equivalent to the definition of a generalized partial route by Jabali et al. (2014). It is a generalization of a partial route as defined by Hjorring and Holt (1999), which contains at most one unstructured component. We stick to the term partial route for ease of writing, and note that the generalized partial route could be generalized even further, by omitting condition iii), i.e., allowing unstructured components to succeed each other directly. We do not pursue the further generalized version in this paper.

Any route that visits all customers of the partial route, in the order prescribed by that partial route, is said to adhere to the partial route. Note that a single route adheres to a partial route, and not others, if and only if the partial route imposes a complete ordering, i.e. if $|U_k| = 1$ for all $k \in \{1, \ldots, b-1\}$. Otherwise, multiple routes adhere to a single partial route. For example, the partial route $(\{0\}, \{1, 2\}, \{3\}, \{4, 5\}, \{0\})$ has four routes adhering to it, namely $(0, 1, 2, 3, 4, 5, 0)$, $(0, 1, 2, 3, 5, 4, 0)$, $(0, 2, 1, 3, 4, 5, 0)$ and $(0, 2, 1, 3, 5, 4, 0)$.

5.2 Lower bounds for partial routes

The expected costs of recourse decomposes per route r, which we denoted by $\min\{Q_r^1, Q_r^2\}$ in (8). Let $\mathcal{R}_h$ be the set of all routes adhering to partial route h. The value P_h is a lower bound for partial route h, if $P_h \leq \min\{Q_r^1, Q_r^2\}$ for all $r \in \mathcal{R}_h$. These lower bounds will be of later use to define valid inequalities. Clearly, the best possible bound is $\min_{r \in \mathcal{R}_h}\{\min\{Q_r^1, Q_r^2\}\}$. However, it is not clear how to compute this bound efficiently, since there are $\sum_{k=1}^{b-1} |U_k|!$ routes in $\mathcal{R}_h$. Therefore, we rely on weaker bounds. In Section 5.2.1, we repeat the lower bound P_h^{class} for a partial route h found in Laporte et al. (2002) for the case of classical recourse. In Sections 5.2.2 and 5.2.3, we introduce the new lower bounds $P_h^{\text{prev}I}$ and $P_h^{\text{prev}II}$ for a partial route h in the case of preventive recourse, which are improved versions of the bounds by Salavati-Khoshghalb et al. (2019a) and Louveaux and Salazar-González (2018), respectively. These lower bounds are computed by modifying the dynamic program used to compute the expected costs of recourse, as found in Section 2.4, by replacing exact computations for a single route with various lower bounds for all routes adhering to partial route h. In Section 5.2.4, we demonstrate how the results of Sections 5.2.2 and 5.2.3 are merged into a single dynamic program, resulting in a third and strongest lower bound $P_h^{\text{prev}III}$ for a partial route h in the case of preventive recourse.

5.2.1 Lower bound for partial routes with classical recourse

The main idea of the lower bound for a partial route in the case of classical recourse by Laporte et al. (2002) is as follows. Given a partial route h, any set $U \in h$, and in particular an unstructured component, can be seen as a large customer with demand $\sum_{v\in U} D_v$, to which we associate the best-case corrective recourse costs $\min_{v\in U}\{s_v\}$. The expected costs of recourse of the route consisting of these large customers in the order of the partial route, provides the following lower bound on the expected costs of recourse for any route adhering to the partial route in one orientation:

$$P_{h1}^{\text{class}} = \sum_{k=1}^{b-1} \min_{v\in U_k}\{s_v\} \sum_{m=1}^{\infty} \mathbb{P}\left(\sum_{u\in\bigcup_{l=1}^{k-1} U_l} D_u \leq mC, \sum_{u\in\bigcup_{l=1}^{k} U_l} D_u > mC \right). \tag{16}$$

Observe that this is a lower bound because the costs $\min_{v\in U_k}\{s_v\}$ of each individual recourse action is smaller than or equal to the actual costs. Reversing the order of the customers in (16) provides a lower bound P_{h2}^{class} for the routes in the other orientation. Combining these values provides the lower bound $P_h^{\text{class}} = \min\{P_{h1}^{\text{class}}, P_{h2}^{\text{class}}\}$ for partial route h. Note that if a partial route has only a single route adhering to it, then the lower bound on the partial route is exactly the expected costs of recourse of that route.

5.2.2 First lower bound for partial routes with preventive recourse

Next, we consider the case of preventive recourse. In Section 2, we presented a dynamic programming algorithm to compute the expected costs of recourse corresponding to a route. We adjust this algorithm to compute a lower bound for a partial route. In particular, we first show how Salavati-Khoshghalb et al. (2019a) do this, and then show how this can be improved. For conciseness, we introduce some additional notation and use it rewrite the original dynamic program, as well as present the adjusted version. We capture part of the recursive relation (9) in the following function

$$F_v(f,q) = \sum_{\xi\in\Xi_v} \mathbb{P}(D_v = \xi)\left[\Gamma(\xi - q)s_v + f(C\Gamma(\xi - q) + q - \xi)\right],$$

for $v \in V$, $f : \{0,\dots,C\} \to \mathbb{R}_{\geq 0}$, $q \in \{0,\dots,C\}$. Here, we can interpret $F_v(f,q)$ as the expected costs of corrective recourse when arriving at customer v with q load remaining in the vehicle and f some function representing future expected recourse costs after serving customer v. Using this shorthand notation, the recursive definition of $f_j^*(q)$ in (9) corresponding with a route $(0 = v(0), v(1), \dots, v(p), v(p+1) = 0)$ can be written down more concisely for each $j \in \{0,\dots,p\}$ as

$$f_j^*(q) = \min\{s_{(v(j),v(j+1))} + F_{v(j+1)}(f_{j+1}^*, C),\ F_{v(j+1)}(f_{j+1}^*, q)\}.$$

Given a partial route $h = (U_0,\dots,U_b)$, we use our shorthand notation to represent the lower bound for the partial route h of Salavati-Khoshghalb et al. (2019a). Define $f_j^S(q)$ as a lower bound on the expected costs of recourse after serving a customer at the j-th position on a route adhering to h. It is crucial to observe that while in f_j^* the index $j \in V$ represents a customer, the ordering of the customers is in general not fixed on a partial route, and therefore the index j of f_j^S refers to a position on a route rather than an explicit customer. Let $U(j)$ be the set of customers that can appear on position j of a route adhering to partial route h. The recursive relation of Salavati-Khoshghalb et al. (2019a) to compute $f_j^S(q)$ can be written as

$$f_j^S(q) = \min_{u\in U(j)} \min_{v\in U(j+1): v\neq u} \min\{s_{(u,v)} + F_v(f_{j+1}^S, C),\ F_v(f_{j+1}^S, q)\}.$$

The lower bound on the expected costs of recourse of any route adhering to the partial route h in one orientation is given by $f_0^S(C)$. By also considering the reversed orientation and taking the smallest value of the two orientations, we obtain the lower bound for the partial route of Salavati-Khoshghalb et al. (2019a).

We improve on the lower bound by Salavati-Khoshghalb et al. (2019a) as follows. Define $f_{uj}^I(q)$ as a lower bound on the expected costs of recourse after serving customer $u \in V$ on the j-th position on a route adhering to h. The difference with f_j^S, is that we now specify which customer is served at position j. The recursive relation to compute $f_{uj}^I(q)$ is

$$f_{uj}^I(q) = \min_{v \in U(j+1): v \neq u} \min\{s_{(u,v)} + F_v(f_{v,j+1}^I, C),\ F_v(f_{v,j+1}^I, q)\}.$$

The lower bound $P_h^{\text{prev}I}$ for the partial route h is now obtained from $f_{00}^I(C)$, taking care of both orientations. Observe that any straightforward implementation of the dynamic programs results in the same number of computations for our new bound and the bound by Salavati-Khoshghalb et al. (2019a), while our new lower bound is stronger. Indeed, it immediately follows that our new lower bound is greater or equal to that of Salavati-Khoshghalb et al. (2019a). To demonstrate that it can be strictly greater, consider the following small example. Let $h = (\{0\}, \{1, 2\}, \{0\})$ be a partial route, the travel costs are $c_{(0,1)} = c_{(1,2)} = 1$, $c_{(0,2)} = 2$, the demands are $D_1 = 1$ and $D_2 = 2$ with probability one, and the vehicle capacity is $C = 2$. There are two routes adhering to partial route h, and the smallest expected costs of recourse among these routes is 2. The lower bound of Salavati-Khoshghalb et al. (2019a) is 0, and our new bound $P_h^{\text{prev}I}$ is 2. In this case the new bound $P_h^{\text{prev}I}$ is the best possible bound, while the nonnegative bound of Salavati-Khoshghalb et al. (2019a) is the worst possible. Note that this example is easily extended to larger partial routes, and to satisfy the expected capacity constraint.

5.2.3 Second lower bound for partial routes with preventive recourse

Next, we focus our attention to Louveaux and Salazar-González (2018), who present five different propositions for computing lower bounds for a partial route. The first four propositions all work for arbitrary independent demand distributions, whereas the last one is specialised for independent identically distributed random variables. Note that this last proposition coincides with the bound of Salavati-Khoshghalb et al. (2019a) when all demands are identically distributed. This was addressed in Section 5.2.2, so we do not discuss this proposition further. The results presented in the remaining four propositions of Louveaux and Salazar-González (2018) can be expressed as a dynamic program which uses a recursive function $f_k^L(q)$, for $k \in \{0, \ldots, b\}$ and $q \in \{0, \ldots, C\}$, to provide a lower bound for a partial route, which we show next. Then, we show how the corresponding lower bounds for a partial route is improved.

We use the shorthand notation $s(U) = \min_{v \in U}\{s_u\}$ for the cheapest corrective recourse action among the customers in set $U \subseteq V'$, and we similarly use $s(U, T) = \min_{u \in U, v \in T, u \neq v}\{s_{(u,v)}\}$ for the cheapest preventive return trip between a customer in $U \subseteq V'$ and $T \subseteq V'$. Here, we define $s(\{u\}, \{u\}) = +\infty$. Let $\bar{s}(U) = \min\{s(U), s(U, U)\}$.

Consider the partial route $h = (U_0, \ldots, U_b)$. We introduce the functions $f_k^{L1}(q)$, $f_k^{L2}(q)$, $f_k^{L3}(q)$ and $f_k^{L4}(q)$ which represent lower bounds on the expected future costs of recourse on a route that adheres to h, for $k \in \{0, \ldots, b\}$. In particular this future costs is that which follows after having served all customers in U_k, so not including the expected costs of recourse for serving customers in U_1 through U_k. Note that the index k of these functions now indicates a position of a set of customers on the partial route, which contrasts the dynamic program in Section 5.2.2 where the index represents a position of a customer on a route. Each of these functions corresponds to one of the four propositions of Louveaux and Salazar-González (2018) mentioned before.

$$f_k^{L1}(q) = \begin{cases} \bar{s}(U_1)\mathbb{P}\left(\sum_{v \in U_1} D_v > C\right) & \text{if } k = 0, q = C, |U_1| > 1, \\ 0 & \text{otherwise.} \end{cases}$$

$$f_k^{L2}(q) = \begin{cases} \min\left\{ s(U_k, U_{k+1}),\ \bar{s}(U_{k+1})\mathbb{P}\left(\sum_{v \in U_{k+1}} D_v > q\right)\right\} & \text{if } |U_{k+1}| > 1, \\ 0 & \text{otherwise.} \end{cases}$$

$$f_k^{L3}(q) = \begin{cases} \min\left\{ s(U_k, U_{k+1}),\ s(U_{k+1})\left(1 - \prod_{v \in U_{k+1}} \mathbb{P}(D_v \leq q)\right)\right\} \\ \qquad \text{if } |U_{k+1}| > 1, s(U_{k+1}, U_{k+1}) \geq s(U_{k+1})\left(1 - \frac{\prod_{v \in U_{k+1}} \mathbb{P}(D_v \leq q)}{\max_{v \in U_{k+1}} \{\mathbb{P}(D_v \leq q)\}}\right), \\ 0 \qquad \text{otherwise.} \end{cases}$$

$$f_k^{L4}(q) = \begin{cases} \min\{s(U_k, U_{k+1}),\ \bar{s}(U_{k+1})\} \\ \qquad \text{if } |U_{k+1}| > 1, U_{k+2} = \{u'\}, \\ \qquad \min\left\{\begin{matrix} s(U_{k+1}, U_{k+2}), \\ \sum_{\xi=0}^{q'} \mathbb{P}(D_{u'} = \xi) f_{k+2}^{L}(q' - \xi) + s_{u'}\mathbb{P}(D_{u'} > q') \end{matrix}\right\} \geq \bar{s}(U_{k+1}),\ \forall q' \leq q \\ 0 \qquad \text{otherwise.} \end{cases}$$

We note that the third requirement of $f_k^{L4}(q)$ slightly differs from that of Louveaux and Salazar-González (2018), where it is only needed that it holds for q itself, not for all $q' \leq q$. However, in their proof of this proposition, they use the fact that the optimal expected recourse function $f_j^*(q)$ is decreasing in q, which is only true when the triangle inequality is assumed on the travel costs (Yang et al., 2000). As both Louveaux and Salazar-González (2018) and our research do not explicitly assume this, we opted to modify the requirement to hold for all $q' \leq q$, which alleviates the need of assuming the triangle inequality.

In addition to the four functions introduced above, we introduce the function $f_k^{L5}(q)$ which recursively provides a lower bound on the future expected costs of recourse, for the specific case that $|U_k| = 1$ and $|U_{k+1}| = 1$, which is not covered by $f_k^{L1}(q)$, $f_k^{L2}(q)$, $f_k^{L3}(q)$ and $f_k^{L4}(q)$. The recursion is similar to that provided in Section 5.2.2.

$$f_k^{L5}(q) = \begin{cases} \min\{s_{(u,v)} + F_v(f_{k+1}^L, C), F_v(f_{k+1}^L, q)\} & \text{if } U_k = \{u\}, U_{k+1} = \{v\}, \\ 0 & \text{otherwise.} \end{cases}$$

Using these functions, we now define $f_k^L(q)$, which provides a lower bound that follows from the four mentioned propositions of Louveaux and Salazar-González (2018), as

$$f_k^L(q) = \max\left\{f_k^{L1}(q),\ f_k^{L2}(q),\ f_k^{L3}(q),\ f_k^{L4}(q),\ f_k^{L5}(q)\right\}.$$

The bound is initialized as $f_b^L(q) = 0$ for all $q \in \{0, \ldots, C\}$. A lower bound for the partial route h is obtained from $f_0^L(C)$, taking care of both orientations.

Recall that an unstructured component of a partial route h is a set $U \in h$ such that $|U| > 1$. A careful investigation of the above lower bound, reveals that, roughly stated, only the costs of recourse of the customers from the first until those in the first unstructured component are considered. Only with f_k^{L4} can additional customers be considered. This may result in a severe underestimation of the actual expected costs of recourse for routes with multiple unstructured components. Next, we present a new lower bound for partial routes with preventive recourse, which does not suffer from this.

Like before, we first introduce a shorthand notation. It is similar to $F_v(f,q)$ introduced in Section 5.2.2, but now for sets of customers U acting as an aggregated customer in the recursion. That is,

$$F_U(f,q) = \begin{cases} F_u(f,q) & \text{if } |U| = \{u\}, \\ \max\left\{ \begin{array}{c} \sum\limits_{\xi \in \Xi_U} \mathbb{P}\left(\sum\limits_{v \in U} D_v = \xi\right) [\Gamma(\xi - q)\bar{s}(U) + H(f, \bar{s}(U), C\Gamma(\xi - q) + q - \xi)], \\ \min\left\{ \begin{array}{c} s(U)\left(1 - \prod\limits_{v \in U} \mathbb{P}(D_v \le q)\right), \\ s(U)\left[1 - \max\limits_{v \in U} \mathbb{P}(D_v \le q)\right] + s(U,U) \max\limits_{v \in U} \mathbb{P}(D_v \le q) \end{array} \right\} + \min\limits_{q \in \{0,\dots,C\}} f(q) \end{array} \right\} & \text{if } |U| > 1, \end{cases}$$

for $U \subseteq V'$, $f : \{0,\dots,C\} \to \mathbb{R}_{\ge 0}$, $q \in \{0,\dots,C\}$. In the above expression, Ξ_U is defined as the set of all possible demand realizations of the random variable $\sum_{v \in U} D_v$ and $H(f,s,q) \equiv \min\{\min_{q' \in \{0,\dots,C\}} f(q') + s, \min_{q' \in \{0,\dots,q\}} f(q')\}$. The function $F_U(f,q)$ can be interpreted as follows. We distinguish between i) singletons U, in which case we use in some sense an exact value of the expected costs of future recourse, depending on f, and ii) unstructured components U, in which case we take a maximum of two values which provides a lower bound on the expected costs of future recourse. The first part of the maximum can be seen as the expected costs of corrective recourse when arriving at a large customer, represented by the set U, with demand equal to the sum over all customers in U and corrective recourse costs $\bar{s}(U)$, when q load remains in the vehicle, and f is some function representing future expected recourse costs after serving the large customer. The second part can be interpreted as the minimum of two cases in which only one corrective recourse is allowed: i) the expected costs of corrective recourse when at least one customer in U has a demand exceeding q, resulting in corrective recourse costs $s(U)$, and without additional preventive recourse, and ii) the expected costs when preventive recourse happens when the customer in U with the highest chance of not exceeding q, does not exceed q, and corrective recourse happens otherwise. Finally, $\min\limits_{q \in \{0,\dots,C\}} f(q)$ is added as a lower bound on the expected costs of future recourse.

With this shorthand notation, define $f_k^{II}(q)$ as a lower bound on the expected costs of recourse incurred after having served $U_0, \dots, U_k$ on a route that adheres to h with q load remaining. For $q \in \{0,\dots,C\}$, let $f_b^{II}(q) = 0$ and for $k \in \{0,\dots,b-1\}$ we recursively define

$$f_k^{II}(q) = \min\{s(U_k, U_{k+1}) + F_{U_{k+1}}(f_{k+1}^{II}, C),\ F_{U_{k+1}}(f_{k+1}^{II}, q)\}.$$

The lower bound $P_h^{\text{prev}II}$ is obtained from $f_0^{II}(C)$, taking care of both orientations. In Appendix C, we prove that this indeed is a lower bound. Furthermore, we prove that the bound obtained from f_k^{II} is stronger than that obtained from f_k^L which follows from the work of Louveaux and Salazar-González (2018).

5.2.4 Third lower bound for partial routes with preventive recourse

We can merge the computations presented in Sections 5.2.2 and 5.2.3, to obtain a third and stronger lower bound for partial route h in the case of preventive recourse. We apply both dynamic programs to recursively obtain $f_{uj}^I(q)$ and $f_k^{II}(q)$. However, for any k, j and u such that $U_k = U(j) = \{u\}$, after having computed f^I_{uj} and $f^{II}_k(q)$, we update their values as follows

$$f^I_{uj}(q) \leftarrow \max\{f^I_{uj}(q), f^{II}_k(q)\}$$
$$f^{II}_k(q) \leftarrow \max\{f^I_{uj}(q), f^{II}_k(q)\}.$$

In particular, since $U_0 = U(0) = \{0\}$, we obtain $f^I_{00}(C) = f^{II}_0(C)$, which results in a stronger bound $P^{\text{prev}III}_h$ than $P^{\text{prev}I}_h$ and $P^{\text{prev}II}_h$, after taking care of both orientations. For this reason, we make use of $P^{\text{prev}III}_h$ in our numerical experiments in case of preventive recourse. However, we prove in Appendix D that for instances with independent identically distributed demand $P^{\text{prev}III}_h = P^{\text{prev}I}_h$, in which case we use $P^{\text{prev}I}_h$ to avoid computational overhead.

5.3 Partial route inequalities

Next, we present the partial route inequalities of Laporte et al. (2002), which are valid inequalities for the VRPSD. They make use of partial route activation functions, which we define first. We discuss the partial route activation function of Laporte et al. (2002) and that of Jabali et al. (2014), and point out a mistake in both. We then suggest a new activation function, and show that this new activation function results in stronger partial route inequalities than using the partial route activation function of Jabali et al. (2014), for those cases that the latter is still valid.

A partial route activation function, extends the definition of a route activation function found in Section 3 to all routes adhering to a partial route. Given a partial route h, a function with as domain the feasible solutions x of the continuous relaxation of the Basic-VRPSD, is called a partial route activation function of h if: i) it has the value 1 if and only if x is integer feasible and corresponds with a solution which includes a route that adheres to h, and ii) if x is integer feasible and corresponds with a solution which does not include any route which adheres to h, then it has a nonpositive value. Note that the domain also includes all feasible solutions x to the continuous relaxation of the VRPSD. Therefore, this partial route activation function can be used both in case of the Basic-VRPSD, and the VRPSD.

A partial route inequality is defined for every set H of partial routes, such that no customer occurs on more than one partial route. In case of the VRPSD, H cannot contain more than m partial routes. We say that a solution adheres to H, if the solution contains a route that adheres to h for all $h \in H$. Let W_h be a linear partial route activation function for each partial route $h \in H$. Let L be a lower bound on the expected costs of recourse, as discussed in Section 4. Moreover, let $P(H)$ be a lower bound on the expected costs of recourse for all solutions that adhere to H. We require $P(H) > L$, otherwise nothing will be gained by imposing a partial route inequality. Similar to Laporte et al. (2002), we use $P(H) = \sum_{h \in H} P_h + L(H)$, where P_h is a lower bound for partial route h as discussed in Section 5.2 and $L(H)$ is defined as follows. In case of the VRPSD, $L(H)$ is a lower bound on the expected costs of recourse as described in Section 4, for an instance which includes all customers V' that are not included on any of the partial routes $h \in H$, and with $m - |H|$ vehicles. In case of the Basic-VRPSD, we choose $L(H) = 0$. The partial route inequality for an appropriate set of partial routes H is

$$\theta \geq L + (P(H) - L)\left(\sum_{h \in H} W_h(x) - (|H| - 1)\right). \tag{17}$$

Observe that if a feasible integer solution x adheres to H, the partial route inequality evaluates to $\theta \geq P(H)$. Otherwise, the partial route inequality is not stronger than $\theta \geq L$. The strength of the partial route inequality depends on the choice of L, $P(H)$ and W_h. We turn our attention to the partial route activation functions W_h.

In the paper by Laporte et al. (2002), only partial routes are considered which have at most one unstructured component, and the partial route activation function is defined accordingly. However, this partial route activation function is incorrect, which is observed by Jabali et al. (2014), who write: "*we show that contrary to what was presented in Proposition 2 of Laporte et al. ..., our definition ... is always valid.*" In Appendix E, we provide an explicit example to substantiate this claim. Next, we present the partial route activation function W_h^J used by Jabali et al. (2014) Denote by $\delta(S,T)$, for $S,T \subseteq V'$ such that $S \cap T = \emptyset$, the set of edges between the sets S and T. Consider the partial route $h = (U_0, \ldots, U_b)$. The partial route activation function can be written as

$$W_h^J(x) = \begin{cases} \sum_{k=1}^{b-1} 3\left(\sum_{e\in E(U_k)} x_e - (|U_k|-1)\right) + \left(\sum_{e\in\delta(U_0,U_1)} x_e - 1\right) & \text{, if } b=2. \\ \sum_{k=1}^{b-1} 3\left(\sum_{e\in E(U_k)} x_e - (|U_k|-1)\right) + \sum_{k=0}^{b-1}\beta_k^J\left(\sum_{e\in\delta(U_k,U_{k+1})} x_e - 1\right) + 1 & \text{, if } b\geq 3, \end{cases}$$

where β_k^J is 1 for $k=0$ and $k=b-1$, and it is 3 otherwise.

However, also W_h^J is not a partial route activation function for all partial routes h. Consider the partial route $h = (\{0\},\{1,2,3\},\{4\},\{0\})$. The partial route activation function is $W_h^L(x) = 3x_{(1,2)} + 3x_{(1,3)} + 3x_{(2,3)} + x_{(0,1)} + x_{(0,2)} + x_{(0,3)} + 3x_{(1,4)} + 3x_{(2,4)} + 3x_{(3,4)} + x_{(0,4)} - 10$. A solution x corresponding with the route $(0,1,2,4,3,0)$ which does not adhere to h, results in $W_h^L(x) = 1$. Note that in a private communication with Ola Jabali, she confirmed that the algorithm used in the paper by Jabali et al. (2014) would erroneously separate the partial route inequality corresponding to the above incorrect partial route activation function. We remark that the problem with W_h^J only arises when $b=3$, which corresponds to the partial route (U_0,U_1,U_2,U_3), where either U_1 or U_2, but not both, is an unstructured component. We believe the proof found in the paper by Jabali et al. (2014) is correct for all partial routes $h=(U_0,\ldots,U_b)$ such that $b\neq 3$, in which cases W_h^J is a valid partial route activation function.

Next we present a new partial route activation function, for all partial routes, which results in stronger partial route inequalities than those using the partial activation function of Jabali et al. (2014), for the cases that the latter is correct. Our partial route activation function W_h is

$$W_h(x) = \sum_{k=1}^{b-1}\alpha_k\left(\sum_{e\in E(U_k)} x_e - (|U_k|-1)\right) + \sum_{k=0}^{b-1}\beta_k\left(\sum_{e\in\delta(U_k,U_{k+1})} x_e - 1\right) + \gamma \tag{18}$$

where the coefficients are defined as follows

$$(\alpha_1,\ldots,\alpha_{b-1}) = \begin{cases} (3) & \text{if } b=2, \\ (4,4) & \text{if } b=3, \\ (3,2,3) & \text{if } b=4, \\ (3,2,1,\ldots,1,2,3) & \text{if } b\geq 5, \end{cases}$$

$$(\beta_0,\ldots,\beta_{b-1}) = \begin{cases} (1,0) & \text{if } b=2, \\ (1,3,1) & \text{if } b=3, \\ (1,2,1,\ldots,1,2,1) & \text{if } b\geq 4, \end{cases}$$

$$\gamma = \begin{cases} 0 & \text{if } b=2, \\ 1 & \text{if } b\geq 3. \end{cases}$$

In Appendix F, we prove that W_h is a partial route activation function. Observe that this partial route activation function W_h is identical to W_h^J by Jabali et al. (2014) for $b=2$, and recall that for $b=3$, W_h^J is not correct. For $b\geq 4$, using W_h instead of W_h^J yields a stronger partial route inequality. By stronger we mean that any solution to the continuous relaxation of the Basic-VRPSD which satisfies the partial route inequalities using W_h, also satisfies the partial route inequalities using $W_h^J(x)$. We prove this, by showing in Appendix G that $W_h(x) \geq W_h^J(x)$ for all such solutions x, and all partial routes h. Because $P(H) - L \geq 0$ it follows that $L + (P(H) - L) \left(\sum_{h\in H} W_h(x) - (|H| - 1)\right) \geq L + (P(H) - L) \left(\sum_{h\in H} W_h^J(x) - (|H| - 1)\right)$ for all solutions x to the continuous relaxation of the Basic-VRPSD. To demonstrate that $W_h(x)$ and $W_h^J(x)$ are not equal, consider the following example. Let $h = (\{0\}, \{1\}, \{2\}, \{3, 4\}, \{0\})$, and consider a solution $\hat{x}$ to the continuous relaxation of the Basic-VRPSD, such that $\hat{x}_{(0,1)} = \hat{x}_{(2,3)} = \hat{x}_{(3,4)} = 1$, $\hat{x}_{(1,3)} = \hat{x}_{(0,4)} = \frac{3}{4}$ and $\hat{x}_{(1,4)} = \hat{x}_{(0,2)} = \frac{1}{4}$. It follows that $W_h(\hat{x}) = \frac{3}{4}$ and $W_h^J(\hat{x}) = 0$, such that $W_h(\hat{x}) > W_h^J(\hat{x})$. Since the solutions to the continuous relaxation of the VRPSD are a subset of those of the Basic-VRPSD, this result immediately extends to the VRPSD. We conclude that using the improved partial route inequalities, results in stronger relaxations, as illustrated by the results of our numerical experiments in Section 7.

5.4 Separation of partial routes

In our implementation of the integer L-shaped method, we initially do not include any partial route inequalities. Whenever a fractional solution is found, we look for violated partial route inequalities using a separation algorithm. If such a violated inequality is found, we add it to the formulation. Next, we describe our separation algorithm.

We use the following heuristic. Let $G(x)$ be a subgraph of G corresponding with the solution x, including only those edges that have been selected with a positive value, and which do not connect to the depot. We use a depth-first procedure attributed to Hopcroft and Tarjan (1973) to identify all connected components of $G(x)$, which act as our candidates for partial routes. An appropriate implementation of this algorithm also immediately provides all articulation points. An articulation point is a vertex whose removal would split up a connected component in multiple connected components. They act as singletons in between unstructured components on the partial routes. For each connected component with a total flow to the depot of exactly two, we introduce a partial route. All other connected components are disregarded. If a connected component includes a vertex $v \in V'$ with integer flow to the depot in G, i.e., $x_{(0,v)}$ is integer, then $\{i\}$ is the first or last set in the corresponding partial route. In the appropriate order, a singleton corresponding to each articulation point of the connected component is added to the partial route, as well as the sets of vertices in between two articulation points which are added as an unstructured component. The result is a collection of partial routes, for which the partial route inequality is checked. The separation procedure runs in $\mathcal{O}(|E|)$.

6 Route-split inequalities

For both classical and preventive recourse, the recourse function $Q(x)$ can be split over the routes. This insight allows us to introduce new valid inequalities. We split the value of θ, distribute these values over multiple routes, and introduce lower bounds on these split values. We introduce route-split inequalities in Section 6.1, which are valid inequalities which could also be used as optimality cuts in the integer L-shaped method instead of those introduced in Section 3. Furthermore, we introduce partial route-split inequalities in Section 6.3, and multi-route-split inequalities in Section 6.3.

6.1 Route-split inequalities

Instead of only using a single variable θ in the integer L-shaped method, representing the expected costs of recourse, we introduce nonnegative variables θ_v, for $v \in V'$, which act as the expected costs of recourse per route. In the formulation we link these variables θ_v to routes. We introduce $|V'|$ of them, one for each customer, simply to have a sufficient amount of variables. The amount $|V'|$ is sufficient, because a solution consists of at most $|V'|$ routes. We add the constraint

$$\sum_{v \in V'} \theta_v = \theta, \tag{19}$$

to the formulation. The following are valid inequalities

$$\theta_{v(r)} \geq \min\{Q_r^1, Q_r^2\} W_r(x) \tag{20}$$

for all feasible routes r, which we refer to as route-split inequalities. Here, $v(r) \in V$ is the customer with the lowest index that is visited by r. For example, if $r = (0, 4, 5, 2, 7, 0)$ then $v(r) = 2$. The function $W_r(x)$ is a route activation function of r as defined in Section 3. Note that, technically, we had defined the domain of $W_r(x)$ as the set of feasible solutions to the continuous relaxation of the Basic-VRPSD. Because the feasible space of the continuous relaxation of the VRPSD is included in that of the Basic-VRPSD, the route-split inequalities (20) are also valid inequalities for the VRPSD.

We separate the route-split inequalities by direct verification, whenever the integer L-shaped method finds an integer solution. Note that when adding route-split inequalities until none are violated anymore, it is not necessary to add the optimality cuts (12), or (14). Indeed, for an integer solution $\hat{x}$ which does not violate the route-split inequalities (20) for any route, it follows from (19), the definition of the expected costs of recourse reflected in (8), and because W_r is a route activation function for r, that $\theta = Q(\hat{x})$. Nonetheless, it can be beneficial computationally to also add the optimality cuts, which are added simultaneously with violated routes-split inequalities. This can be seen in the results of our numerical experiments of Section 7.

6.2 Partial route-split inequalities

Just like the total expected costs of recourse can be split over routes, we can split a lower bound on the expected costs of recourse over partial routes. We impose a lower bound P_h for partial route h using the partial route activation function W_h. Observe that

$$\theta_{v(h)} \geq P_h W_h(x) \tag{21}$$

is a valid inequality for both the VRPSD and the Basic-VRPSD. Similar to before, we denote by $v(h) \in V$ the customer with the lowest index in the partial route h. We refer to (21) as partial route-split inequalities, and separate them with the same separation procedure as used for partial route inequalites, as described in Section 5.4.

Consider a partial route inequality (17) for an appropriate set of partial routes H. For the Basic-VRPSD, we have chosen the lower bounds $L = 0$ and $L(H) = 0$. Any solution to the continuous relaxation of the Basic-VRPSD which satisfies the partial route-split inequalities (21) for all $h \in H$, also satisfies the partial route inequality for H. Because the reverse is not true, it is computationally advantageous to include partial route-split inequalities instead of partial route inequalities. For the VRPSD, the same behavior can be expected for small values of L and $L(H)$. However, if L and $L(H)$ are sufficiently large then the addition of partial route inequalities could result in stronger bounds of the continuous relaxation, which we have observed for specific instances of the VRPSD. Hence, it could be beneficial to add both types of inequalities, as can be seen from the results of our numerical experiments of Section 7.

6.3 Multi-route-split inequalities

The valid inequalities specific to the Basic-VRPSD and VRPSD, discussed in this paper so far, use activation functions that are active for a single route at a time. Roughly stated, when a solution to the continuous relaxation is such that individual routes are not easily distinguished, the performance of these valid inequalities is limited. Consider for example an instance which includes customers 1, 2, 3 and 4, with a solution such that $x_{(0,1)} = x_{(0,2)} = x_{(0,3)} = x_{(0,4)} = 1$ and $x_{(1,2)} = x_{(2,3)} = x_{(3,4)} = x_{(1,4)} = 0.5$. For this case, it is easily verified, although laboriously, that for all routes r including only customers from $\{1, 2, 3, 4\}$ the route activation function is nonpositive, i.e., $W_r(x) \leq 0$, and that for all partial routes h including only customers from $\{1, 2, 3, 4\}$ the partial route activation function is nonpositive, i.e. $W_h(x) \leq 0$. Next, we introduce a new type of activation function, which is active for groups of customers for which multiple routes are required which are not easily distinguished. We use this activation function to introduce new valid inequalities for the Basic-VRPSD and VRPSD.

Given a subset of customers $S \subseteq V'$ and an integer k, a function with as domain the feasible solutions x of the continuous relaxation of the Basic-VRPSD, which includes the solutions of the continuous relaxation of the VRPSD, is called a k-routes activation function of S if for every feasible integer solution x in which the customers in S exclusively appear on $m' \leq k$ different routes, i.e., no other customers appear on these routes, the function has the value $k - m' + 1$, and it has a nonpositive value otherwise. Observe that a k-routes activation function of S has the value 1 if precisely k routes are used to exclusively visit all customers in S, 2 if $k - 1$ routes are used, etc. But the function is nonpositive if strictly more than k routes are used, or at least one other customer which is not in S appears on one of the routes.

Next, we provide a k-routes activation function. Let $m(S)$ be a lower bound on the number of routes which are needed to feasibly serve all customers in $S \subseteq V'$. In case of the Basic-VRPSD, we use $m(S) = 1$ for all $S \subseteq V'$, and for the VRPSD we use $\lceil \frac{1}{C} \sum_{i \in S} \mu_i \rceil$. For $S \subseteq V'$ and integer $m(S) \leq k \leq m$, consider the k-routes activation function

$$W_{(S,k)}(x) = (k - m(S) + 1) \sum_{e \in \delta(\{0\}, S)} x_e + (2(k - m(S)) + 3) \sum_{e \in E(S)} x_e - (2(k - m(S)) + 3)|S| + k + 1. \tag{22}$$

We prove in Appendix H that $W_{(S,k)}$ is a k-routes activation function. Note that for $m(S) = k = 1$, the function $W_{(S,k)}(x)$ reduces to the partial route activation function $W_h(x)$ for $h = (\{0\}, S, \{0\})$.

Before we are ready to define our new valid inequalities, we first also define additional lower bounds on the expected costs of recourse corresponding with S and k. Given an instance of the Basic-VRPSD, or VRPSD depending on the case, for $S \subseteq V'$ and integer j, we denote by $L(S, j)$ a lower bound on the expected costs of recourse of an induced instance corresponding with only customers S and in which precisely j routes must be used. Observe that in case of the Basic-VRPSD this induced instance is not an instance of the Basic-VRPSD but of the FR-VRPSD, because now a constraint on the number of routes is additionally imposed. If the induced instance is infeasible, then $L(S, j) = \infty$. Finally, note that we require that $L(S, j)$ is nonnegative.

We use these lower bounds $L(S, j)$ to define new valid inequalities. The lower bounds described in Section 4 can be used to obtain values of $L(S, j)$, by applying them to the induced instance. In particular, we use (15) of Louveaux and Salazar-González (2018) for the VRPSD. Preliminary experiments with our implementation of the lower bounds of Laporte et al. (2002), showed that they are not strong enough and computationally too demanding to be of benefit in many instances.

For $S \subseteq V'$ and integer k, define

$$\bar{L}(S,k) = \min_{j \in \{m(S),\ldots,k\}} \left\{ \frac{1}{k-j+1} L(S,j). \right\}$$

The multi-route-split inequality for $S \subseteq V'$ and integer k, is

$$\sum_{i \in S} \theta_i \geq \bar{L}(S,k) W_{(S,k)}(x). \tag{23}$$

The multi-route split inequalities are valid inequalities for the following reason. Whenever a feasible integer solution corresponds to the use of m' routes to visit all customers in S, for $m' \leq k$, the k-route activation function has the value $W_{(S,k)}(x) = k - m' + 1$. Observe that $\bar{L}(S,k) \leq \frac{1}{k-m'+1} L(S,m')$, which implies $\bar{L}(S,k) W_{(S,k)}(x) \leq L(S,m')$, and therefore $\bar{L}(S,k) W_{(S,k)}(x)$ is a lower bound on the expected costs of recourse to visit all customers in S. Otherwise, $W_{(S,k)}(x)$ is nonpositive, and because we require $\bar{L}(S,k)$ to be nonnegative, the value $\bar{L}(S,k) W_{(S,k)}(x)$ is nonpositive and is trivially a lower bound on the expected costs of recourse to visit all customers in S.

We also separate the multi-route-split inequalities with the same separation procedure as used for partial route inequalites, as described in Section 5.4. Every identified connected component, with vertices S, acts as a candidate for a violated multi-route-split inequality if the flow to the depot is strictly larger than 2, i.e., $\sum_{e \in \delta(\{0\},S)} x_e > 2$. In this case, we choose $k = \left\lceil \frac{1}{2} \sum_{e \in \delta(\{0\},S)} x_e \right\rceil$ and verify whether the multi-route split inequality for S and k is violated.

7 Numerical results

In this section, we provide the results of our numerical experiments. In Section 7.1, we illustrate the performance of our integer L-shaped method on instances of the VRPSD with classical recourse. We compare the performance of our integer L-shaped method with that of Jabali et al. (2014), which we consider the current best integer L-shaped method for the VRPSD with classical recourse. In Section 7.2, we discuss results for the VRPSD with preventive recourse. In this case, we compare with the algorithm of Louveaux and Salazar-González (2018), which we consider the current best integer L-shaped method for the VRPSD with preventive recourse. We also compare with the algorithm of Florio et al. (2020), which we consider the best branch-price-and-cut algorithm for the VRPSD with preventive recourse. Finally, in Section 7.3 we compare the solution values and computation times obtained when solving instances of the VRPSD, FR-VRPSD and the Basic-VRPSD. This provides empirical insight on the effect of using a fixed number of vehicles and using the expected capacity constraints.

There are various configurations of our integer L-shaped method: i) with or without partial route inequalities, ii) with or without partial route-split inequalities, iii) with or without multi-route-split inequalities, and iv) with the standard optimality cuts, the route-split inequalities, or both. Note that by standard optimality cuts, we refer to (12) in case of the VRPSD and FR-VRPSD and (14) in case of the Basic-VRPSD. This yields 24 configurations in total. We have tested all configurations in preliminary experiments on 270 benchmark instances from Jabali et al. (2014), and have selected three configurations for their effectiveness, which we refer to as `RS`, `PRS` and `P+MRS`. They all use the standard optimality cuts. Additionally `RS` includes the route-split and partial route inequalities, `PRS` includes the route-split and partial route-split inequalities, and `P+MRS` includes the route split, partial route-split and multi-route-split inequalities. For benchmarking purposes, we also include the configuration which we refer to as `BM`. This configuration only includes the standard optimality cuts and partial route inequalities, and resembles the integer L-shaped methods from the literature enhanced by stronger lower bounds on the expected costs of recourse and stronger lower bounds for partial routes. It allows us to illustrate the contribution of the novel inequalities of Section 6 to the performance of the integer L-shaped method.

In our experiments, we encounter the normal, discrete triangular and Poisson distributions for demand. In case of the normal distribution, we apply a numerical method to evaluate the expected costs of recourse with an accuracy of at least two decimal places. For the discrete triangular distribution, the computations are exact, up to machine precision. For the Poisson distribution, our computations resemble those of Florio et al. (2020). That is, we disregard demand realizations with a probability of 10^{-6} or less, and normalize the probabilities of all other realizations. This effectively results in a truncated Poisson distribution. Note that Florio et al. (2020) uses an accuracy parameter of 10^{-5} instead of 10^{-6}, and we comment on this difference in Section 7.2.

Our algorithm is coded in C++ and run on a Windows 10 computer with an Intel Xeon W-2123 3.6GHz processor and 16GB of RAM. The maximum allotted runtime is one hour per instance. All algorithms are run on a single thread and the branch-and-cut algorithm was built using CPLEX version 20.1 using generic callbacks.

7.1 Classical recourse

In this section, we compare the performance of our integer L-shaped method to the integer L-shaped method of Jabali et al. (2014) on the VRPSD with classical recourse. We perform this comparison on the instances used by Jabali et al. (2014). Details on these instances can be found in their paper, but note that demands follow a normal distribution with a coefficient of variation of 0.30. There are 27 instance classes, each containing 10 instances, making a total of 270 instances. Each class is identified with the number of customers n, the number of routes m and the average filling coefficient of $\bar{f} \equiv \frac{1}{mC} \sum_{i \in V'} \mu_i$.

Next, we present the results of applying our integer L-shaped method with the configurations `BM`, `RS`, `PRS` and `P+MRS` to these instances. We also report the results obtained by the integer L-shaped method of Jabali et al. (2014) for these instances. To account for a potential difference in computing power and a different quality of implementation, and to overcome any effects of the use of the incorrect partial route activation function, we additionally apply our own implementation of their integer L-shaped method which we refer to as `Jab`. This is a modified version of the `BM` configuration of our integer L-shaped method, in which we use the activation function $W_h^J(x)$ of Jabali et al. (2014) instead of our function $W_h(x)$ with improved coefficients, except for the case where $W_h^J(x)$ is incorrect as explained in Section 5.3. Note that `Jab` does include the stronger lower bounds on the expected costs of recourse which we present in this paper.

In Table 2, we show the number of instances that are solved to optimality per class of instance. In the first three columns, we report for each class of 10 instances, the number of customers n, the number of vehicles m, and the filling coefficient $\bar{f}$, respectively. In the column with header 'Jabali et al. (2014)', we provide the number of instances that were solved to optimality, as reported by Jabali et al. (2014), allowing a runtime of five hours per instance. In the remaining five columns, we report the number of instances solved to optimality within one hour for the configurations `Jab`, `BM`, `RS`, `PRS` and `P+MRS`, respectively. In Table 3, we similarly report in the columns 'Gap', the average optimality gaps in percentages of the instances that have not been solved by any method, where a dash '—' indicates that all instances were solved to optimality by at least one method, and in the columns 'Time' we report the average computation times in seconds of the instances that were solved by all methods.

Observe that our implementation of the algorithm of Jabali et al. (2014), `Jab`, is much faster than theirs, as it can solve more than twice the number of instances, even despite allowing one hour of computations instead of five. We find it likely that this is due to the increased computing power and a different quality of implementation. The `BM` configuration, which differs from `Jab` only by using an

Table 2: Number of instances of the VRPSD with classical recourse of Jabali et al. (2014) solved to optimality per class.

n	m	$\bar{f}$	Jabali et al. (2014)	Jab	BM	RS	PRS	P+MRS
60	2	0.90	10	10	10	10	10	10
60	2	0.925	9	10	10	10	10	10
60	2	0.95	5	10	10	10	10	10
70	2	0.90	10	10	10	10	10	10
70	2	0.925	5	10	10	10	10	10
70	2	0.95	2	8	9	10	10	10
80	2	0.90	9	10	10	10	10	10
80	2	0.925	4	10	10	10	10	10
80	2	0.95	0	8	8	8	8	8
50	3	0.85	8	10	10	10	10	10
50	3	0.875	6	10	10	10	10	10
50	3	0.90	2	9	9	9	9	9
60	3	0.85	5	10	10	10	10	10
60	3	0.875	1	8	8	9	9	9
60	3	0.90	0	7	8	8	8	8
70	3	0.85	7	9	9	9	9	9
70	3	0.875	2	9	9	9	9	9
70	3	0.90	0	5	5	6	6	6
40	4	0.80	4	10	10	10	10	10
40	4	0.825	4	10	10	10	10	10
40	4	0.85	1	8	8	8	8	8
50	4	0.80	4	7	8	8	7	8
50	4	0.825	1	10	10	10	10	10
50	4	0.85	0	6	6	7	7	7
60	4	0.80	1	7	7	8	7	9
60	4	0.825	2	8	8	9	9	9
60	4	0.85	0	6	5	6	7	7
	Total		102	235	237	244	243	246

improved activation function, is able to solve two more instances, has a lower average optimality gap on the instances that were not solved by any method, and lower average computation times on the instances that were solved by all methods. This suggests that the improved coefficients for the partial route inequalities provide a computational advantage.

The methods `RS`, `PRS` and `P+MRS` solve to optimality 7, 6 and 9 instances more than `BM` within the time limit of one hour. This suggest that incorporating the route-split, partial route-split and multi-route-split inequalities, results in a faster algorithm. Moreover, the methods `RS` and `P+MRS` have lower average computation times than `BM` over the instances that are solved by all methods, and `RS` also has a lower average optimality gap than `BM` over all instances that remain unsolved by all methods.

Finally, we note that in each row of Table 2, when for one configuration we report solving more instances than another, the former solves all instances of the latter, plus additional instances. The only exception is for instance `60_3_0.90_2`, which is solved to optimality only using the configuration `BM`, in 3591.06 seconds. It follows that we solved 145 previously unsolved instances to optimality, leaving 23 instances unsolved within a computation time on one hour.

Table 3: Average optimality gap in percentages over all instances that were not solved by any method, and average computation time in seconds over all instance that were solved by all methods, for instances of the VRPSD with classical recourse of Jabali et al. (2014).

			Jab		BM		RS		PRS		P+MRS	
n	m	$\bar{f}$	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time
60	2	0.90	—	15.43	—	14.50	—	18.37	—	12.66	—	10.47
60	2	0.925	—	16.18	—	16.23	—	17.23	—	14.80	—	14.70
60	2	0.95	—	217.96	—	243.34	—	248.56	—	155.85	—	192.49
70	2	0.90	—	26.36	—	21.57	—	12.27	—	9.98	—	10.30
70	2	0.925	—	160.86	—	128.20	—	129.25	—	93.39	—	107.23
70	2	0.95	—	257.14	—	237.63	—	227.30	—	176.52	—	180.16
80	2	0.90	—	50.16	—	35.93	—	45.51	—	30.97	—	30.95
80	2	0.925	—	177.97	—	169.47	—	148.55	—	175.13	—	179.88
80	2	0.95	1.09	445.02	1.03	562.97	0.97	281.74	0.96	368.25	0.98	414.50
50	3	0.85	—	23.55	—	25.79	—	23.02	—	27.76	—	23.91
50	3	0.875	—	43.65	—	40.47	—	38.89	—	45.77	—	38.38
50	3	0.90	2.99	310.13	3.18	251.10	2.95	215.96	3.02	265.72	2.87	257.70
60	3	0.85	—	262.59	—	233.26	—	154.93	—	281.54	—	150.38
60	3	0.875	1.42	103.51	1.59	106.21	1.48	91.30	1.06	72.90	1.51	73.60
60	3	0.90	0.87	254.80	0.91	309.15	0.82	252.02	0.91	391.44	0.97	375.07
70	3	0.85	1.17	36.65	0.84	33.02	0.41	36.28	0.59	48.62	0.95	33.02
70	3	0.875	1.65	593.31	0.84	736.22	1.23	576.91	2.35	774.81	1.68	607.04
70	3	0.90	1.38	1218.76	1.34	1013.07	1.35	1153.80	1.73	1469.94	1.58	1294.69
40	4	0.80	—	207.70	—	178.30	—	140.87	—	173.06	—	121.61
40	4	0.825	—	105.98	—	147.79	—	74.39	—	88.71	—	60.13
40	4	0.85	1.71	127.13	1.45	103.78	1.65	81.04	1.30	177.62	1.57	94.57
50	4	0.80	1.60	33.25	1.76	51.17	1.56	39.74	2.07	37.22	1.53	36.45
50	4	0.825	—	759.47	—	556.78	—	454.90	—	675.03	—	460.49
50	4	0.85	2.10	178.72	1.84	153.03	1.89	111.85	1.93	97.42	1.77	103.51
60	4	0.80	1.49	311.72	1.47	274.54	1.25	305.88	1.29	383.82	1.01	175.22
60	4	0.825	1.62	284.70	1.22	316.42	1.73	304.37	1.99	366.35	1.92	318.94
60	4	0.85	2.64	439.84	2.46	446.59	2.11	424.46	2.52	535.88	2.18	300.64
	Average		1.73	223.74	1.60	215.89	1.55	187.36	1.74	226.86	1.62	185.72

7.2 Preventive recourse

Next, we demonstrate the performance of our integer L-shaped method on the VRPSD with preventive recourse. We first compare our algorithm with the integer L-shaped method of Louveaux and Salazar-González (2018), and secondly compare with the branch-price-and-cut algorithm of Florio et al. (2020).

For our comparison with the integer L-shaped method of Louveaux and Salazar-González (2018), we use the instances of the VRPSD with preventive recourse that they used in their paper, excluding those with asymmetric travel costs. These are 32 instances divided over four series with the names `E031-09h`, `E051-05e`, `E076-07s` and `E101-08e`, which have 30, 50, 75 and 100 customers respectively. Demands follow independent identical discrete triangular distributions, centered around $\mu = 5$. More precisely, the set of realizations is $\Xi_i = \left\{\mu - \left\lfloor \frac{K}{2} \right\rfloor, \ldots, \mu + \left\lfloor \frac{K}{2} \right\rfloor\right\}$ and the probability distribution is given by $\mathbb{P}(\Xi_i = \xi) = \frac{\left\lceil \frac{K}{2} \right\rceil - |\xi - \mu|}{\left\lceil \frac{K}{2} \right\rceil^2}$ for all $i \in V'$, $\xi \in \Xi_i$. Here, K is a parameter defining the width of the triangular distribution. In the instances of Louveaux and Salazar-González (2018), the values 3 and 9 are used for K, the number of vehicles m is varied between 2 and 3 and the filling coefficient is set to 0.85 or 0.90 for $m = 2$ and 0.80 or 0.85 for $m = 3$, and the vehicle capacity C varies accordingly per instance.

In Table 4, we provide the results of our computational experiments. The first five columns provide for each instance the name of the series, the vehicle capacity C, the number of vehicles m, the width of the triangular distribution K and the filling coefficient $\bar{f}$. We include the results reported by Louveaux and Salazar-González (2018), and the results found by the configurations `BM`, `RS` and `PRS`. Note that we do not include `P+MRS`, because for these instances the lower bound $\bar{L}$ used in the multi-route-split inequalities are all 0. As a result, `PRS` and `P+MRS` are indistinguishable, and not separately reported on. Table 4 includes two columns with the header 'Obj'. The first occurrence in column six, corresponds with the objective value found per instance as reported by Louveaux and Salazar-González (2018). The second occurrence in column ten, corresponds with the solution value obtained by our integer L-shaped method. Note that the configurations `BM`, `RS` and `PRS` all provide the same solution values, which are not separately reported. Furthermore, for each instance we report in the columns 'Gap', the optimality gaps in percentages after termination of the algorithm, in the columns 'Time' the computation times in seconds, and in the columns 'Nodes', the number of nodes in the branching tree that have been processed upon termination. Note that Louveaux and Salazar-González (2018) allowed five hours of computation time, while we allow one hour. In Table 4, '5h' and '1h' indicates that the time limit has been exceeded. Also, Louveaux and Salazar-González (2018) report computation times in minutes, which we have converted into seconds by multiplying with 60.

Table 4: Computational results on instances of the VRPSD with preventive recourse of Louveaux and Salazar-González (2018).

					Louveaux and Salazar-González (2018)				BM				RS			PRS		
Series	C	m	K	f	Obj	Gap	Time	Nodes	Obj	Gap	Time	Nodes	Gap	Time	Nodes	Gap	Time	Nodes
E031-09h	84	2	3	0.90	332.75	0	0	325	332.75	0	0.10	8	0	0.10	6	0	0.10	6
E031-09h	79	2	3	0.95	335.30	0	1.2	2035	335.30	0	0.20	15	0	0.13	15	0	0.10	15
E031-09h	84	2	9	0.90	337.67	0	2.4	3632	337.67	0	1.11	323	0	0.66	203	0	0.71	195
E031-09h	79	2	9	0.95	344.53	0	78	36654	344.53	0	5.25	1723	0	3.19	1184	0	2.95	1100
E031-09h	59	3	3	0.85	358.95	0	16.8	17950	358.95	0	0.32	61	0	0.23	49	0	0.23	49
E031-09h	56	3	3	0.90	364.07	0	244.2	94518	364.07	0	1.24	385	0	0.99	308	0	0.93	313
E031-09h	59	3	9	0.85	367.16	0	1090.2	248044	367.16	0	5.40	1833	0	3.71	1386	0	3.35	1242
E031-09h	56	3	9	0.90	373.13	2.33	$5h$	604022	372.78	0	269.53	34414	0	73.90	15328	0	57.11	13374
E051-05e	139	2	3	0.90	441.00	0	3.6	3260	441.00	0	0.47	26	0	0.45	27	0	0.44	27
E051-05e	132	2	3	0.95	441.31	0	6	3889	441.31	0	3.26	318	0	1.51	134	0	1.51	134
E051-05e	139	2	9	0.90	443.01	0	18	10709	443.01	0	4.42	554	0	3.72	503	0	3.72	493
E051-05e	132	2	9	0.95	448.08	0.13	$5h$	314798	448.08	0	415.60	27957	0	196.03	15983	0	196.24	16969
E051-05e	99	3	3	0.85	459.00	0	9.6	6557	459.00	0	0.54	27	0	0.52	24	0	0.45	24
E051-05e	93	3	3	0.90	459.05	0	4.2	3449	459.05	0	1.08	95	0	1.12	90	0	1.07	89
E051-05e	99	3	9	0.85	460.55	0	37.2	22525	460.55	0	2.15	234	0	1.85	221	0	1.92	216
E051-05e	93	3	9	0.90	465.63	0	4346.4	303297	465.63	0	60.68	5994	0	51.72	5111	0	35.76	3469
E076-07s	209	2	3	0.90	549.00	0	1.8	757	549.00	0	0.93	9	0	0.93	9	0	0.86	9
E076-07s	198	2	3	0.95	550.16	0	20.4	7869	550.16	0	5.17	158	0	4.70	146	0	3.77	113
E076-07s	209	2	9	0.90	550.82	0	16.2	8522	550.82	0	9.76	258	0	7.83	210	0	6.52	159
E076-07s	198	2	9	0.95	554.80	0	15165.6	425613	554.80	0	601.67	16225	0	428.31	14369	0	335.49	11016
E076-07s	148	3	3	0.85	567.13	0	165	43343	567.13	0	25.18	896	0	12.26	527	0	17.13	746
E076-07s	139	3	3	0.90	569.27	0	2168.4	213546	569.27	0	19.21	678	0	41.65	1245	0	61.28	1669
E076-07s	148	3	9	0.85	569.95	0	6445.2	440721	569.95	0	122.87	4326	0	119.74	4098	0	124.82	4376
E076-07s	139	3	9	0.90	574.31	1.03	$5h$	579000	573.25	0	1495.33	24066	0	1190.24	22241	0	1450.73	24180
E101-08e	278	2	3	0.90	640.00	0	11.4	2819	640.00	0	2.85	31	0	3.17	31	0	3.08	31
E101-08e	264	2	3	0.95	641.73	0	1759.2	83765	641.73	0	54.18	1317	0	51.44	911	0	23.14	476
E101-08e	278	2	9	0.90	641.30	0	783	34436	641.30	0	68.82	910	0	91.66	752	0	304.82	3086
E101-08e	264	2	9	0.95	646.12	0.85	$5h$	172619	646.12	0.49	$1h$	20168	0.37	$1h$	26600	0.38	$1h$	26632
E101-08e	197	3	3	0.85	655.35	0	58.8	9025	655.35	0	19.11	415	0	7.50	128	0	7.48	141
E101-08e	186	3	3	0.90	658.30	0	544.2	40442	658.30	0	56.29	1242	0	106.79	2192	0	96.79	2184
E101-08e	197	3	9	0.85	658.98	0	6325.2	292928	658.98	0	889.91	14805	0	223.07	4148	0	168.78	3089
E101-08e	186	3	9	0.90	667.55	1.55	$5h$	267310	666.56	1.04	$1h$	21917	1.12	$1h$	21935	0.98	$1h$	23938

We can see in Table 4 that all our methods achieve a noteworthy decrease in computation time compared with Louveaux and Salazar-González (2018). Excluding the first instance, in which Louveaux and Salazar-González (2018) reports zero minutes of computation time, the speedups range from a factor 1.6 for `RS` in row 19, to 325 for `PRS` in row 7. Furthermore, while Louveaux and Salazar-González (2018) solved 27 out of 32 instances to optimality within five hours of computation time per instance, all configurations of our integer L-shaped method solve 30 out of 32 instances to optimality within one hour of computation time per instance, and have a smaller optimality gap for the remaining 2 unsolved instances. Furthermore, note that the average computation time over the 30 instances that were solved by our integer L-shaped method, are 138.09, 87.04 and 97.04 for the configurations `BM`, `RS` and `PRS`, respectively. We emphasize that the configuration `RS` has the lowest average computation time, which is 47% and 10% lower than that of `BM` and `PRS` respectively. This again indicates that including the route-split and partial route-split inequalities provides a computational advantage.

Note that unlike our comparison with the integer L-shaped method of Jabali et al. (2014), this comparison might be obscured since we have not made our own implementation of the algorithm of Louveaux and Salazar-González (2018). This is because their algorithm is actually designed for a directed graph, while our implementation is designed for an undirected graph and could not easily emulate their algorithm. Note that an algorithm suitable for a directed graph may be at a disadvantage due to the involved overhead. Nonetheless, observe that all configurations of our integer L-shaped method, process less branching nodes before identifying an optimal solution, which might be due to the use of stronger valid inequalities, resulting in a faster integer L-shaped method.

Next, we compare the performance of our integer L-shaped method to the branch-price-and-cut algorithm of Florio et al. (2020). First note that in their paper, Florio et al. (2020) also apply their algorithm on the instances of the VRPSD with preventive recourse of Louveaux and Salazar-González (2018). Only for instance `E031-09h`, with $m = 3$, $K = 9$, and $\bar{f} = 0.90$, do they report a better computation time than Louveaux and Salazar-González (2018), namely 55 minutes. In fact, they could not find the optimal solution within five hours of computation time for any of the instances of the `E051-05e` series, and do not report computation times for the other instances with more customers, but state that the branch-price-and-cut algorithm is not effective for those. Each of the configuration, BM, RS and PRS of our integer L-shaped method greatly outperforms their branch-price-and-cut algorithm on these instances.

Secondly, we investigate the remaining set of instances used by Florio et al. (2020). They applied their branch-price-and-cut algorithm on modified versions of the well-known CVRP benchmark instances, A, E, P and X with up to 200 customers. These instances are modified by imposing stochastic demand for each customer which is Poisson distributed with a rate equal to the deterministic demand of the original CVRP instance. We emphasize that we consider here the instances of the VRPSD, so with a fixed number of routes, as Florio et al. (2020) also reports on a version without.

In their paper, Florio et al. (2020) report solving 31 of these instances to optimality within a time limit of five hours. We have applied the configurations `BM`, `RS`, `PRS` and `M+PRS` of our integer L-shaped method to all instances. With one hour of computation time per instance, we could solve to optimality 9 instances that were also solved by Florio et al. (2020). Furthermore, the branch-price-and-cut algorithm was not applied by Florio et al. (2020) to 5 instances. In a private communication, Alexandre Florio said that these instances are troublesome due to the high demand and vehicle capacity. We could solve all these instances to optimality within one hour of computation time. In Table 5, we present computational results on all 14 instances that are solved by our integer L-shaped method. The interpretation of the values per column is the same as before, with the addition of a column that includes the vehicle capacity C per instance for illustrative purposes. With 'NA', we indicate that the instance was not attempted by Florio et al. (2020).

Table 5: Computational results on those instances of the VRPSD with preventive recourse of Florio et al. (2020) which were solved to optimality by our integer L-shaped method.

Instance	C	Florio et al. (2020) Obj	Time	BM Obj	Time	RS Time	PRS Time	P+MRS Time	NOP Time
E-n13-k4	6000	NA	NA	254.22	15.66	9.70	12.06	12.04	3.11
E-n22-k4	6000	NA	NA	377.38	10.49	0.50	0.50	0.49	0.46
E-n23-k3	4500	NA	NA	569.27	0.46	0.45	0.44	0.46	0.45
E-n30-k3	4500	NA	NA	541.17	1868.28	992.15	1114.35	1114.72	11.37
E-n33-k4	8000	NA	NA	844.23	1284.37	931.54	1192.53	675.46	34.76
A-n37-k5	100	710.07	2820	710.12	$1h$	1847.85	1460.20	1298.54	2642.25
E-n31-k7	140	407.97	0	408.00	719.58	315.80	272.74	272.88	272.26
P-n16-k8	35	514.64	0	514.65	2.60	1.82	1.76	1.56	1.28
P-n19-k2	160	229.24	120	229.26	1.45	1.30	1.25	1.28	0.97
P-n20-k2	160	234.34	2040	234.36	10.19	11.44	9.17	9.07	8.57
P-n21-k2	160	220.53	0	220.55	0.24	0.25	0.24	0.29	0.32
P-n22-k2	160	232.72	8760	232.74	7.06	7.47	8.71	8.71	14.44
P-n22-k8	3000	618.10	0	618.24	303.18	259.38	357.38	355.94	12.25
P-n40-k5	140	475.71	0	475.74	91.17	58.86	57.93	51.64	43.92

First, observe that the integer L-shaped method achieves lower computation times than those reported for the branch-price-and-cut algorithm by Florio et al. (2020), for at least 4 instances presented in Table 5. Next, observe that the solution values as reported by Florio et al. (2020) and those found with out integer L-shaped method, do not match for all instances. We believe that these differences are likely caused by numerical accuracy due to the following facts. In a private communication, Alexandre Florio shared with us the results of running the branch-price-and-cut algorithm of Florio et al. (2020), with an accuracy parameter of 10^{-6} instead of 10^{-5}. All of the instances for which the objective values previously did not match, now all objective values match up to at least two decimal places. We further remark that the optimal solutions that were shared with us by Alexandre Florio, are the same as the optimal solutions which we found.

Recall that we argued that the branch-price-and-cut algorithm is sensitive to the vehicle capacity. The instance P-n22-k8 with vehicle capacity 3000, is the only instance that is solved to optimally by the branch-price-and-cut algorithm of Florio et al. (2020), which has a vehicle capacity larger than 160. This does not just apply to the instances reported on in Table 5, but all instances of the A, E, P and X series. The results presented in Table 5 do not give evidence that our integer L-shaped method is sensitive to the vehicle capacity.

Given the results on the A, E, P and X instances of Florio et al. (2020), and on the instances of Louveaux and Salazar-González (2018), we conclude that neither our integer L-shaped method, nor the branch-price-and-cut algorithm of Florio et al. (2020), dominates the other in computational performance on the VRPSD with preventive recourse. We conjecture that the branch-price-and-cut algorithm benefits from strong LP bounds, while it is more sensitive to parameter values of the instance, such as the vehicle capacity.

7.3 Fixed number of routes and expected capacity constraints

In this section, we demonstrate the effect of imposing a fixed number of routes and expected capacity constraints on the optimal solution value of an instance, and on the computation time of our integer L-shaped method. We do this by comparing the results of related instances of the VRPSD, the FR-

VRPSD and the Basic-VRPSD. Recall that the FR-VRPSD is the VRPSD without expected capacity constraints, and in the Basic-VRPSD additionally the fixed number of routes constraint is relaxed.

We have selected some instances from the literature for this purpose, three from Jabali et al. (2014), two from Louveaux and Salazar-González (2018) and two from Florio et al. (2020). For each of these nine instances, we have created two additional versions, in which we include only the first n customers of the instance, for a small and medium value n, resulting in a total of 21 instances. We create these small instances, because the Basic-VRPSD can typically not be solved for large instances, and we want to illustrate where the boundary lies. For each of these instances, we modify the vehicle capacity C, such that the filling coefficient $\bar{f}$ is adhered to.

We presents the results obtained by applying the `M+PRS` configuration of the integer L-shaped method to each of these instances, because `M+PRS` is on average the most effective for these instances. Like before, for the instances of Louveaux and Salazar-González (2018) we apply the `PRS` configuration because it is indistinguishable from `M+PRS` in this case.

In Tables 6, 7 and 8 we provide our computational results for the instances of Jabali et al. (2014), Louveaux and Salazar-González (2018) and Florio et al. (2020), respectively. In the first columns of each table, we present the name of the original instance, number of vehicles m, filling coefficient $\bar{f}$ vehicle capacity C and number of customers n. In Table 7, we additionally include the parameter K of the triangular distribution. In the column with the header 'Type', we indicate per row whether it corresponds to an instance of the VRPSD, the FR-VRPSD or the Basic-VRPSD. Finally, in the column with the header 'Obj' we provide the best found solution value, in the column with the header 'Gap' we provide the optimality gap, and in the column with the header 'Time' we provide the computation time in seconds. Note that '1h' indicates that the time limit of one hour is reached.

Table 7: Computational results on the VRPSD, FR-VRPSD and Basic-VRPSD, for instances of Louveaux and Salazar-González (2018), using the MRS configuration of the integer L-shaped method.

Instance	m	K	$\bar{f}$	C	n	Type	Obj	Gap	Time
E031-09h	2	3	0.90	28	10	VRPSD	178.49	0.00	0.01
E031-09h	2	3	0.90	28	10	FR-VRPSD	178.49	0.00	0.02
E031-09h	2	3	0.90	28	10	Basic-VRPSD	178.41	0.00	0.10
E031-09h	2	3	0.90	56	20	VRPSD	266.04	0.00	0.04
E031-09h	2	3	0.90	56	20	FR-VRPSD	266.04	0.00	0.27
E031-09h	2	3	0.90	56	20	Basic-VRPSD	266.03	0.00	4.82
E031-09h	2	3	0.90	84	30	VRPSD	332.75	0.00	0.10
E031-09h	2	3	0.90	84	30	FR-VRPSD	332.75	0.00	3.82
E031-09h	2	3	0.90	84	30	Basic-VRPSD	332.48	0.00	227.26
E031-09h	3	3	0.85	20	10	VRPSD	208.09	0.00	0.02
E031-09h	3	3	0.85	20	10	FR-VRPSD	208.09	0.00	0.11
E031-09h	3	3	0.85	20	10	Basic-VRPSD	204.82	0.00	18.60
E031-09h	3	3	0.85	39	20	VRPSD	292.26	0.00	0.07
E031-09h	3	3	0.85	39	20	FR-VRPSD	292.26	0.00	79.78
E031-09h	3	3	0.85	39	20	Basic-VRPSD	291.66	7.01	1h
E031-09h	3	3	0.85	59	30	VRPSD	358.95	0.00	0.23
E031-09h	3	3	0.85	59	30	FR-VRPSD	365.79	4.96	1h
E031-09h	3	3	0.85	59	30	Basic-VRPSD	359.99	8.89	1h

Table 6: Computational results on the VRPSD, FR-VRPSD and Basic-VRPSD, for instances of Jabali et al. (2014), using the P+MRS configuration of the integer L-shaped method.

Instance	m	f	C	n	Type	Obj	Gap	Time
40_4_0.80_7	4	0.80	14	10	VRPSD	536.91	0.00	0.16
40_4_0.80_7	4	0.80	14	10	FR-VRPSD	526.34	0.00	15.79
40_4_0.80_7	4	0.80	14	10	Basic-VRPSD	511.25	0.00	2961.07
40_4_0.80_7	4	0.80	38	25	VRPSD	595.02	0.00	1081.48
40_4_0.80_7	4	0.80	38	25	FR-VRPSD	591.99	8.62	$1h$
40_4_0.80_7	4	0.80	38	25	Basic-VRPSD	599.50	17.38	$1h$
40_4_0.80_7	4	0.80	64	39	VRPSD	657.49	0.00	0.82
40_4_0.80_7	4	0.80	64	39	FR-VRPSD	682.85	11.45	$1h$
40_4_0.80_7	4	0.80	64	39	Basic-VRPSD	698.50	18.53	$1h$
50_3_0.85_5	3	0.85	26	10	VRPSD	521.17	0.00	0.10
50_3_0.85_5	3	0.85	26	10	FR-VRPSD	478.50	0.00	0.59
50_3_0.85_5	3	0.85	26	10	Basic-VRPSD	477.16	0.00	4.66
50_3_0.85_5	3	0.85	69	30	VRPSD	623.57	0.00	0.41
50_3_0.85_5	3	0.85	69	30	FR-VRPSD	623.57	0.00	2.47
50_3_0.85_5	3	0.85	69	30	Basic-VRPSD	623.57	0.00	249.34
50_3_0.85_5	3	0.85	108	49	VRPSD	732.13	0.00	0.89
50_3_0.85_5	3	0.85	108	49	FR-VRPSD	732.13	0.00	30.21
50_3_0.85_5	3	0.85	108	49	Basic-VRPSD	732.13	0.00	350.57
60_2_0.90_6	2	0.90	29	10	VRPSD	416.89	0.00	0.02
60_2_0.90_6	2	0.90	29	10	FR-VRPSD	416.89	0.00	0.33
60_2_0.90_6	2	0.90	29	10	Basic-VRPSD	416.89	0.00	0.55
60_2_0.90_6	2	0.90	111	35	VRPSD	533.39	0.00	11.17
60_2_0.90_6	2	0.90	111	35	FR-VRPSD	533.39	0.00	84.12
60_2_0.90_6	2	0.90	111	35	Basic-VRPSD	532.89	0.00	532.86
60_2_0.90_6	2	0.90	189	59	VRPSD	646.98	0.00	1.83
60_2_0.90_6	2	0.90	189	59	FR-VRPSD	646.98	0.00	23.79
60_2_0.90_6	2	0.90	189	59	Basic-VRPSD	646.98	0.00	60.51

Observe that for all instances the computation time is increasing in the order VRPSD, FR-VRPSD and Basic-VRPSD. The less constraints are included, the more computation time is required. The increase in computation times can be very large. For example, instance 40_4_0.80_7 of the VRPSD with $n = 39$ is solved in 0.82 seconds, while the FR-VRPSD and the Basic-VRPSD could not even be solved within the time limit of one hour. Out of the 21 instances, 5 instances of the FR-VRPSD could not be solved within one hour, and 8 instances of the Basic-VRPSD. In contrast, all instances of the VRPSD could be solved, 16 of which within one second, and the largest computation time is 55.45 seconds. Imposing a fixed number of routes and expected capacity constraints, have a substantial impact on the computation times of our integer L-shaped method.

The impact of these constraints on the solution value is less pronounced. Out of the 16 instances of the FR-VRPSD which could be solved, the optimal solution values are equal to those of the VRPSD for 13 instances, and they are lower only for 3 instances. Also note that those latter 3 instances all consist of only 10 customers. The inclusion of expected capacity constraints does not seem to impact the optimal solution value very often. Observe additionally, that out of the 13 instances of the Basic-VRPSD

Table 8: Computational results on the VRPSD, FR-VRPSD and Basic-VRPSD, for instances of Florio et al. (2020), using the P+MRS configuration of the integer L-shaped method.

Instance	m	f	C	n	Type	Obj	Gap	Time
P-n21-k2	2	0.93	89	10	VRPSD	176.70	0.00	0.11
P-n21-k2	2	0.93	89	10	FR-VRPSD	176.70	0.00	3.30
P-n21-k2	2	0.93	89	10	Basic-VRPSD	173.43	0.00	97.37
P-n21-k2	2	0.93	119	15	VRPSD	203.79	0.00	0.16
P-n21-k2	2	0.93	119	15	FR-VRPSD	203.79	0.00	21.78
P-n21-k2	2	0.93	119	15	Basic-VRPSD	198.84	0.00	791.68
P-n21-k2	2	0.93	160	20	VRPSD	220.55	0.00	0.29
P-n21-k2	2	0.93	160	20	FR-VRPSD	220.55	0.00	79.20
P-n21-k2	2	0.93	160	20	Basic-VRPSD	219.51	5.16	$1h$
P-n40-k5	5	0.88	35	10	VRPSD	334.46	0.00	0.06
P-n40-k5	5	0.88	35	10	FR-VRPSD	328.93	0.00	339.02
P-n40-k5	5	0.88	35	10	Basic-VRPSD	307.43	25.61	$1h$
P-n40-k5	5	0.88	96	25	VRPSD	399.78	0.00	5.14
P-n40-k5	5	0.88	96	25	FR-VRPSD	422.01	15.94	$1h$
P-n40-k5	5	0.88	96	25	Basic-VRPSD	391.12	24.83	$1h$
P-n40-k5	5	0.88	140	39	VRPSD	475.73	0.00	51.64
P-n40-k5	5	0.88	140	39	FR-VRPSD	525.20	20.32	$1h$
P-n40-k5	5	0.88	140	39	Basic-VRPSD	486.11	24.30	$1h$

which could be solved, the optimal solution values are equal to those of the FR-VRPSD for 4 instances, and they are lower for 8 instances. Note that the difference in solution value can be very small. For instance E031-09h with $n = 20$, the optimal solution values of the VRPSD and FR-VRPSD is 266.04, while that of the Basic-VRPSD is 266.03. We emphasize that in this case really a different solution is observed. For the VRPSD and FR-VRPSD the solution that we found consists of two routes, while for the Basic-VRPSD the solution consist of one route, which is the concatenation of the aforementioned two routes, with a small but positive probability of not doing preventive recourse at the point where the routes are concatenated.

We conclude from our experiments, that although the inclusion of the expected capacity constraints might seem awkward from a modeling perspective, it helps reduce computation times tremendously, while the optimal objective values do not seem to suffer much for these instances. Even though we think that caution should be used, we find it reasonable for computational purposes to impose these constraints. The same conclusion applies to imposing a fixed number of routes, although to a lesser extent.

8 Conclusion

In this paper, we present a state-of-the-art integer L-shaped method for the VRPSD. It incorporates all elements of other integer L-shaped methods from the literature, that we are aware of. Moreover, we rectify and improve on existing lower bounds and valid inequalities that are part of these methods, and introduce new valid inequalities. Our numerical experiments indicate that the resulting integer L-shaped method outperforms all others, and is competitive with the best branch-price-and-cut algorithm from the literature. We additionally argue based on our theoretical and numerical analysis, that it seems reasonable to impose a fixed number of routes and expected capacity constraints, although care should be taken that the optimal solution value does not deteriorate too much. As a next step, we think it is relevant to investigate imposing route duration constraints instead, as these constraints seem to provide a more accurate representation of limitations in practice. This investigation is already started by Florio et al. (2021) who present a branch-price-and-cut algorithm for the corresponding optimization problem. In future research, the integer L-shaped method might be modified to deal with those constraints as well, with the aim of creating a competitive algorithm that is successful for a wide range of instances for this problem.

Acknowledgements

We are grateful to Ola Jabali for having a look at an early draft of our work and for providing us with benchmark instances. We also thank Alexandre Florio for fruitful discussions on the numerical precision of the branch-price-and-cut algorithm, and for running his algorithm for us with a higher accuracy setting.

References

Christian H Christiansen and Jens Lysgaard. A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research Letters*, 35(6):773–781, 2007.

Moshe Dror, Gilbert Laporte, and Pierre Trudeau. Vehicle routing with stochastic demands: Properties and solution frameworks. *Transportation science*, 23(3):166–176, 1989.

Alexandre M Florio, Richard F Hartl, and Stefan Minner. New exact algorithm for the vehicle routing problem with stochastic demands. *Transportation Science*, 54(4):1073–1090, 2020.

Alexandre M Florio, Richard F Hartl, Stefan Minner, and Juan-José Salazar-González. A branch-and-price algorithm for the vehicle routing problem with stochastic demands and probabilistic duration constraints. *Transportation Science*, 55(1):122–138, 2021.

Charles Gauvin, Guy Desaulniers, and Michel Gendreau. A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 50:141–153, 2014.

Michel Gendreau, Gilbert Laporte, and René Séguin. An exact algorithm for the vehicle routing problem with stochastic demands and customers. *Transportation science*, 29(2):143–155, 1995.

Curt Hjorring and John Holt. New optimality cuts for a single-vehicle stochastic routing problem. *Annals of Operations Research*, 86:569–584, 1999.

John Hopcroft and Robert Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, 1973.

Ola Jabali, Walter Rei, Michel Gendreau, and Gilbert Laporte. Partial-route inequalities for the multi-vehicle routing problem with stochastic demands. *Discrete Applied Mathematics*, 177:121–136, 2014.

Gilbert Laporte and François V Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations research letters*, 13(3):133–142, 1993.

Gilbert Laporte, François V Louveaux, and Luc Van hamme. An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research*, 50(3):415–423, 2002.

François V Louveaux and Juan-José Salazar-González. Exact approach for the vehicle routing problem with stochastic demands and preventive returns. *Transportation Science*, 52(6):1463–1478, 2018.

Jens Lysgaard. Cvrpsep: A package of separation routines for the capacitated vehicle routing problem. *http://www. hha. dk/˜ lys/CVRPSEP. htm*, 2003.

Majid Salavati-Khoshghalb, Michel Gendreau, Ola Jabali, and Walter Rei. An exact algorithm to solve the vehicle routing problem with stochastic demands under an optimal restocking policy. *European Journal of Operational Research*, 273(1):175–189, 2019a.

Majid Salavati-Khoshghalb, Michel Gendreau, Ola Jabali, and Walter Rei. A hybrid recourse policy for the vehicle routing problem with stochastic demands. *EURO Journal on Transportation and Logistics*, 8(3):269–298, 2019b.

Majid Salavati-Khoshghalb, Michel Gendreau, Ola Jabali, and Walter Rei. A rule-based recourse for the vehicle routing problem with stochastic demands. *Transportation Science*, 53(5):1334–1353, 2019c.

William R Stewart Jr and Bruce L Golden. Stochastic vehicle routing: A comprehensive approach. *European Journal of Operational Research*, 14(4):371–385, 1983.

Wen-Huei Yang, Kamlesh Mathur, and Ronald H Ballou. Stochastic vehicle routing problem with restocking. *Transportation science*, 34(1):99–112, 2000.

A Route activation function

Consider the route $r = (0 = v(0), v(1), \ldots, v(p), v(p+1) = 0)$. We repeat the function found in (13).

$$W_r(x) = \begin{cases} x_{(0,v(1))} - 1 & \text{if } p = 1, \\ x_{(0,v(1))} + 3x_{(v(1),v(2))} + x_{(0,v(2))} - 4 & \text{if } p = 2, \\ x_{(0,v(1))} + 2x_{(v(1),v(2))} + \sum_{k=2}^{p-2} x_{(v(k),v(k+1))} + 2x_{(v(p-1),v(p))} + x_{(0,v(p))} - (p+2) & \text{otherwise.} \end{cases}$$

Theorem 3. *The function $W_r(x)$ is a route activation function of r.*

Proof. Suppose x corresponds with a route r, meaning if $p = 1$ then $x_{(0,v(1))} = 2$ and if $p \geq 2$ then $x_{(v(i),v(i+1))} = 1$ for all $i \in \{0, \ldots, p\}$. It is easily verified that $W_r(x) = 1$. Next, suppose x does not correspond to the route r. If $p = 1$, then $x_{(0,v(1))} \leq 1$ such that $W_r(x) \leq 0$. If $p = 2$, we distinguish two cases. If $x_{(v(1),v(2))} = 0$ then $W_r(x) \leq 0$. If $x_{(v(1),v(2))} = 1$, by the degree constraints (2), it follows that $x_{(0,v(1))} \leq 1$ and $x_{(0,v(1))} \leq 1$. Since x does not correspond to the route r, either $x_{(0,v(1))} = 0$ or $x_{(0,v(2))} = 0$ (or both). Therefore, $W_r(x) \leq 0$. Finally, if $p \geq 3$ we consider two similar cases. If $x_{(v(1),v(2))} = 0$ or $x_{(v(p-1),v(p))} = 0$ then $W_r(x) \leq 0$. If $x_{(v(1),v(2))} = x_{(v(p-1),v(p))} = 1$, by the degree constraints, $x_{(0,v(1))} \leq 1$ and $x_{(0,v(p))} \leq 1$. Since x does not correspond to the route r, at least one of the variables $x_{(0,v(1))}$, $x_{(0,v(2))}$ and $x_{(v(i),v(i+1))}$ for $i \in \{2, \ldots, p-2\}$ is zero. Therefore, $W_r(x) \leq 0$. □

B Lower bound by Salavati-Khoshghalb et al. (2019a)

Next, we comment on the lower bound on the expected costs of recourse for the VRPSD with preventive recourse by Salavati-Khoshghalb et al. (2019a). The presentation of this lower bound as found in their paper seems to contain a mistake, which we correct below. Moreover, we improve on their bound. Unfortunately, we conclude that even the improved version is often weaker than the trivial lower bound 0.

We present the lower bound described in Salavati-Khoshghalb et al. (2019a) in our notation. Furthermore, instead of the lower bound for partial route h of Salavati-Khoshghalb et al. (2019a) which is part of the description, we immediately use our stronger lower bound $P_h^{\text{prev}III}$ found in Section 5.2.4. Let s be the sum of the $m-1$ smallest preventive recourse costs s_e for $e \in E$. Let $h(v)$ be the partial route $\{\{0\}, V' \setminus \{v\}, \{v\}, \{0\}\}$, for all $v \in V'$. The lower bound on the expected costs of recourse of Salavati-Khoshghalb et al. (2019a) can be written as

$$\min_{v \in S}\{P_{h(v)}^{\text{prev}III}\} - s. \tag{24}$$

The idea behind this lower bound is as follows. A solution to the VRPSD, which consists of m routes, can be seen as a single route which visits all the customers, with $m-1$ mandatory preventive return trips. The value $\min_{v \in S}\{P_{h(v)}^{\text{prev}III}\}$ is a lower bound for the expected costs of recourse for a single route which visits all the customers. If we think of a solution to the VRPSD as a single route with $m-1$ mandatory preventive return trips, the costs of these mandatory preventive return trips should not be counted as recourse costs but as first stage travel costs. Therefore, $\min_{v \in S}\{P_{h(v)}^{\text{prev}III}\}$ is an overestimation and should be corrected.

The correction proposed in Salavati-Khoshghalb et al. (2019a) is to subtract the value s, i.e., the $m-1$ smallest preventive recourse costs s_e with $e \in E$. However, s is a lower bound on the costs of $m-1$ preventive return trips, which is clearly not sufficient, as also demonstrated in the following example. Let $n = 4$, $m = 2$ and $C = 3$. The costs are the Euclidean distances, where customers 1 and 2 lie at a distance one and two above the depot, respectively, and customers 3 and 4 lie at distance one and two to the right from the depot, respectively. The demands of customers 1, 2, 3 and 4 are 1 with probability $\frac{1}{4}$, $\frac{3}{4}$, $\frac{1}{4}$, and $\frac{3}{4}$, respectively, and 2 otherwise. The minimal recourse of visiting all customers with 2 vehicles is approximately 0.38, attained by the routes $(0, 1, 4, 0)$ and $(0, 3, 2, 0)$. The supposed lower bound of Salavati-Khoshghalb et al. (2019a), even particularly the lower valued original version which does not use our improved $P_h^{\text{prev}III}$ or even $P_h^{\text{prev}I}$, is approximately 0.59. This demonstrates that the suggested lower bound is invalid.

Instead, an upper bound on the costs of $m-1$ preventive return trips should be used, and it is sufficient to subtract s' defined as the $m-1$ largest preventive recourse costs s_e with $e \in E$. In fact, we can improve on this bound even further.

Note that each customer $v \in V'$ has only two edges incident to it in a solution to the VRPSD. Therefore, a stronger upper bound $s^* \leq s'$ is obtained by selecting the $m-1$ largest preventive recourse costs s_e with $e \in E$, with the additional condition that no customer $v \in V'$ occurs on more than two selected edges. This bound is the solution value of the following integer linear program, where z_e is the decision variable indicating whether edge $e \in E$, or rather the recourse costs s_e, is selected:

$$
\begin{aligned}
s^* = \max \sum_{e \in E} s_e z_e, & & \\
\text{s.t.} \sum_{e \in \delta(v) \cap E} z_e \leq 2 & & \forall v \in V', \\
\sum_{e \in E} z_e = k & & , \\
z_e \in \{0, 1\} & & \forall e \in E.
\end{aligned}
$$

The bound can be computed by solving the above integer linear program. When this is computationally too demanding, the solution value of the LP relaxation can be computed instead. Observe that this LP bound is also a valid bound, and it is also stronger than s'.

Summing up, a correct lower bound is $\min_{v \in S}\{P_{h(v)}^{\text{prev}III}\} - s^*$. However, this lower bound is negative for all benchmark instances which we consider in our numerical experiments. Therefore, we do not use this bound any further in this paper.

C Improved partial route lower bound

Next, we prove that the partial route lower bound $P_h^{\text{prev}II}$ as found in Section 5.2.3 is valid and stronger than the results of Louveaux and Salazar-González (2018). For convenience, we repeat the definitions of $F_U(f, q)$ and $f_k^{II}(q)$ from Section 5.2.3 here:

$$
F_U(f,q) = \begin{cases} F_u(f,q) & \text{if } |U| = \{u\}, \\ \max \left\{ \begin{array}{c} \sum\limits_{\xi \in \Xi_U} \mathbb{P}\left(\sum\limits_{v \in U} D_v = \xi\right) \left[\Gamma(\xi - q)\bar{s}(U) + H(f, \bar{s}(U), C\Gamma(\xi - q) + q - \xi)\right], \\ \min \left\{ \begin{array}{c} s(U)\left(1 - \prod\limits_{v \in U} \mathbb{P}(D_v \leq q)\right), \\ s(U)\left[1 - \max\limits_{v \in U} \mathbb{P}(D_v \leq q)\right] + s(U,U) \max\limits_{v \in U} \mathbb{P}(D_v \leq q) \end{array} \right\} + \min\limits_{q \in \{0,\ldots,C\}} f(q) \end{array} \right\} & \text{if } |U| > 1, \end{cases}
$$

and

$$
f_k^{II}(q) = \min\{s(U_k, U_{k+1}) + F_{U_{k+1}}(f_{k+1}^{II}, C),\ F_{U_{k+1}}(f_{k+1}^{II}, q)\}.
$$

Recall that $P_h^{\text{prev}II}$ is obtained from $f_0^{II}(C)$, taking care of both orientations.

To prove the validity of this lower bound, we first derive some properties of the separate components in $F_U(f,q)$. First, we focus on the term

$$\sum_{\xi\in\Xi_U}\mathbb{P}\left(\sum_{v\in U}D_v=\xi\right)[\Gamma(\xi-q)\bar{s}(U)+H(f,\bar{s}(U),C\Gamma(\xi-q)+q-\xi)] \tag{25}$$

of $F_U(f,q)$. For conciseness, we introduce the shorthand notation $I(U,f,s,q)$, with $U\subseteq V$, $f:\{0,\dots,C\}\to\mathbb{R}_{\geq 0}$, $s\geq 0$ and $q\in\{0,\dots,C\}$, defined as

$$I(U,f,s,q)=\sum_{\xi\in\Xi_U}\mathbb{P}\left(\sum_{v\in U}D_v=\xi\right)[\Gamma(\xi-q)s+f(C\Gamma(\xi-q)+q-\xi)],$$

such that we can express (25) as $I(U,H(f,\bar{s}(U),\cdot),\bar{s}(U),q)$. Observe that $I(U,f,s,q)$ can also be expressed as

$$I(U,f,s,q)=\sum_{\xi=0}^{q}\mathbb{P}\left(\sum_{v\in U}D_v=\xi\right)f(q-\xi)+$$
$$\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+q+1}^{kC+q}\mathbb{P}\left(\sum_{v\in U}D_v=\xi\right)[ks+f(kC+q-\xi)],$$

which is achieved by collecting the terms of ξ for which $\Gamma(\xi-q)=k$. We use these two expressions interchangeably. Next, we derive important properties of (25), which we use in the proof of the validity of the lower bound, by analysing $I(U,f,s,q)$ and $H(f,s,q)$.

First, we prove that $I(U,f,s,q)$ is decreasing in q and its range is bounded by s, if f is decreasing and its range is bounded by s. This seems sensible, when we interpret $I(U,f,s,q)$ as the expected costs of future recourse before serving the customers in U, f a function representing future recourse costs after U, with s the costs of corrective recourse and q the current load. Indeed, when we serve U with a higher starting load, we either leave U with a higher load, resulting in lower costs since f is decreasing, or we leave U with a lower load, as a result of executing one less corrective recourse action, resulting in lower costs because the range of the future recourse costs is bounded by the costs of a single corrective recourse action.

Lemma 1. *For a decreasing function $f:\{0,\dots,C\}\to\mathbb{R}_{\geq 0}$ and a value $s\geq 0$, if $f(0)-f(C)\leq s$ then $I(U,f,s,q)$ is decreasing in q and $I(U,f,s,0)-I(U,f,s,C)\leq s$.*

Proof. In this proof, we make use of the shorthand notation $\mathbb{P}(\xi)=\mathbb{P}\left(\sum_{v\in U}D_v=\xi\right)$. For any $q\in\{0,\dots,C-1\}$, we write

$$I(U,f,s,q+1)-I(U,f,s,q)=\sum_{\xi=0}^{q+1}\mathbb{P}(\xi)f(q+1-\xi)-\sum_{\xi=0}^{q}\mathbb{P}(\xi)f(q-\xi)+$$
$$\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+q+2}^{kC+q+1}\mathbb{P}(\xi)[ks+f(kC+q+1-\xi)]+$$
$$-\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+q+1}^{kC+q}\mathbb{P}(\xi)[ks+f(kC+q-\xi)]$$
$$=\sum_{\xi=0}^{q}\mathbb{P}(\xi)[f(q+1-\xi)-f(q-\xi)]+$$
$$\mathbb{P}(q+1)[f(0)-s-f(C-1)]+$$

$$\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+q+2}^{kC+q}\mathbb{P}(\xi)[f(kC+q+1-\xi)-f(kC+q-\xi)]+$$
$$\sum_{k=1}^{\infty}\mathbb{P}(kC+q+1)[f(0)-s-f(C-1)].$$

Because f is decreasing, it follows that $f(q+1-\xi)-f(q-\xi)\leq 0$ and $f(kC+q+1-\xi)-f(kC+q-\xi)\leq 0$. Because additionally $f(0)-f(C)\leq s$, it follows that $f(0)-f(C-1)-s\leq f(0)-f(C)-s\leq 0$. We conclude that $I(U,f,s,q+1)-I(U,f,s,q)\leq 0$, which means $I(U,f,s,q)$ is decreasing in q.

Next, we prove that $I(U,f,s,0)-I(U,f,s,C)\leq s$ as follows.

$$\begin{aligned}
I(U,f,s,C)+s &= \sum_{\xi=0}^{C}\mathbb{P}(\xi)f(C-\xi)+\sum_{k=1}^{\infty}\sum_{\xi=kC+1}^{(k+1)C}\mathbb{P}(\xi)[ks+f((k+1)C-\xi)]+s \\
&= \mathbb{P}(0)[s+f(C)]+\sum_{k=0}^{\infty}\sum_{\xi=kC+1}^{(k+1)C}\mathbb{P}(\xi)[(k+1)s+f((k+1)C-\xi)] \\
&= \mathbb{P}(0)[s+f(C)]+\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+1}^{kC}\mathbb{P}(\xi)[ks+f(kC-\xi)] \\
&= \mathbb{P}(0)[s+f(C)-f(0)]+\mathbb{P}(0)f(0)+\sum_{k=1}^{\infty}\sum_{\xi=(k-1)C+1}^{kC}\mathbb{P}(\xi)[ks+f(kC-\xi)] \\
&= \mathbb{P}(0)[s+f(C)-f(0)]+I(U,f,s,0) \\
&\geq I(U,f,s,0)
\end{aligned}$$

The inequality follows because $f(0)-f(C)\leq s$ by assumption. □

Next, we show that when consecutively visiting two groups of customers for which the costs of corrective recourse is the same, we can view these two groups as one large group of customers.

Lemma 2. *If $f:\{0,\dots,C\}\to\mathbb{R}_{\geq 0}$, $s\geq 0$, $U,T\subseteq V$ such that $U\cap T=\emptyset$, then $I(T,I(U,f,s,\cdot),s,q)=I(U\cup T,f,s,q)$ for all $q\in\{0,\dots,C\}$.*

Proof. Observe that

$$\begin{aligned}
I(T,I(U,f,s,\cdot),s,q) &= \sum_{\xi\in\Xi_T}\mathbb{P}\left(\sum_{v\in T}D_v=\xi\right)[s\Gamma(\xi-q)+I(U,f,s,C\Gamma(\xi-q)+q-\xi)] \\
&= \sum_{\xi\in\Xi_T}\mathbb{P}\left(\sum_{v\in T}D_v=\xi\right)\Bigg[s\Gamma(\xi-q)+ \\
&\qquad \sum_{\xi'\in\Xi_U}\mathbb{P}\left(\sum_{v\in U}D_v=\xi'\right)[s\Gamma(\xi'-(C\Gamma(\xi-q)+q-\xi))+ \\
&\qquad f(C\Gamma(\xi'-(C\Gamma(\xi-q)+q-\xi))+C\Gamma(\xi-q)+q-\xi-\xi')]\Bigg].
\end{aligned}$$

Note that

$$\begin{aligned}
\Gamma(\xi'-(C\Gamma(\xi-q)+q-\xi)) &= \Gamma(\xi'+\xi-q-C\Gamma(\xi-q)) \\
&= \max\left\{\left\lceil\frac{\xi'+\xi-q-C\Gamma(\xi-q)}{C}\right\rceil,0\right\}
\end{aligned}$$

$$
\begin{aligned}
&= \max\left\{\left\lceil \frac{\xi' + \xi - q}{C} - \Gamma(\xi - q)\right\rceil, 0\right\} \\
&= \max\left\{\left\lceil \frac{\xi' + \xi - q}{C}\right\rceil - \Gamma(\xi - q), 0\right\} \\
&= \Gamma(\xi' + \xi - q) - \Gamma(\xi - q),
\end{aligned}
$$

as $\xi' \geq 0$ and Γ is increasing. Continuing our derivation allows us to prove the lemma.

$$
\begin{aligned}
I(T, f', s, q) = &\sum_{\xi \in \Xi_T} \mathbb{P}\left(\sum_{v \in T} D_v = \xi\right) \left[s\Gamma(\xi - q) + \sum_{\xi' \in \Xi_U} \mathbb{P}\left(\sum_{v \in U} D_v = \xi'\right) [s(\Gamma(\xi' + \xi - q) + \right. \\
&\quad \left. - \Gamma(\xi - q)) + f(C\Gamma(\xi' + \xi - q) + q - (\xi + \xi')] \right] \\
&= \sum_{\xi \in \Xi_T} \sum_{\xi' \in \Xi_U} \mathbb{P}\left(\sum_{v \in T} D_v = \xi\right) \mathbb{P}\left(\sum_{v \in U} D_v = \xi'\right) [s\Gamma(\xi' + \xi - q) + \\
&\quad f(C\Gamma(\xi' + \xi - q) + q - (\xi + \xi')] \\
&= \sum_{\xi = 0}^{\infty} \sum_{\xi' = 0}^{\infty} \mathbb{P}\left(\sum_{v \in T} D_v = \xi\right) \mathbb{P}\left(\sum_{v \in U} D_v = \xi'\right) [s\Gamma(\xi' + \xi - q) + \\
&\quad f(C\Gamma(\xi' + \xi - q) + q - (\xi + \xi')] \\
&= \sum_{\xi'' = 0}^{\infty} \sum_{\xi = 0}^{\xi''} \mathbb{P}\left(\sum_{v \in T} D_v = \xi\right) \mathbb{P}\left(\sum_{v \in U} D_v = \xi'' - \xi\right) [s\Gamma(\xi'' - q) + \\
&\quad f(C\Gamma(\xi'' - q) + q - \xi''] \\
&= \sum_{\xi'' = 0}^{\infty} \mathbb{P}\left(\sum_{v \in U \cup T} D_v = \xi''\right) [s\Gamma(\xi'' - q) + f(C\Gamma(\xi'' - q) + q - \xi''] \\
&= I(U \cup T, f, s, q)
\end{aligned}
$$

□

To apply Lemmas 1 and 2 to (25), we next demonstrate that their requirements are satisfied by the function $H(f, s, q) = \min\{\min_{q' \in \{0,\ldots,C\}} f(q') + s, \min_{q' \in \{0,\ldots,q\}} f(q')\}$ as defined in Section 5.2.3 for $f : \{0 \ldots, C\} \to \mathbb{R}_{\geq 0}$, $s \geq 0$ and $q \in \{0, \ldots, C\}$. Furthermore, we also demonstrate that it underestimates the future recourse costs f.

Lemma 3. *The function $H(f, s, q)$ satisfies the following properties:*

1. *$H(f, s, q)$ is decreasing in q.*
2. *$H(f, s, 0) - H(f, s, C) \leq s$.*
3. *$H(f, s, q) \leq f(q)$.*

Proof. For $q \in \{0, \ldots, C - 1\}$, we derive

$$
\begin{aligned}
H(f, s, q + 1) &= \min\left\{\min_{q' \in \{0,\ldots,C\}} f(q') + s, \min_{q' \in \{0,\ldots,q+1\}} f(q')\right\} \\
&= \min\left\{\min_{q' \in \{0,\ldots,C\}} f(q') + s, \min_{q' \in \{0,\ldots,q\}} f(q'), f(q + 1)\right\} \\
&\leq \min\left\{\min_{q' \in \{0,\ldots,C\}} f(q') + s, \min_{q' \in \{0,\ldots,q\}} f(q')\right\}
\end{aligned}
$$

$$= H(f,s,q),$$

which proves property 1. Next, we derive

$$\begin{aligned} H(f,s,0) - H(f,s,C) &= \min\left\{\min_{q'\in\{0,\dots,C\}} f(q') + s,\ \min_{q'\in\{0\}} f(q')\right\} + \\ &\quad - \min\left\{\min_{q'\in\{0,\dots,C\}} f(q') + s,\ \min_{q'\in\{0,\dots,C\}} f(q')\right\} \\ &= \min\left\{\min_{q'\in\{0,\dots,C\}} f(q') + s, f(0)\right\} - \min\left\{\min_{q'\in\{0,\dots,C\}} f(q')\right\} \\ &= \min\{s, f(0) - \min_{q'\in\{0,\dots,C\}} f(q')\} \\ &\le s, \end{aligned}$$

which proves property 2. Finally, we note that

$$H(f,s,q) = \min\left\{\min_{q'\in\{0,\dots,C\}} f(q') + s,\ \min_{q'\in\{0,\dots,q\}} f(q')\right\} \le f(q),$$

which proves property 3. □

Now, we turn to the next component of $F_U(f,q)$, which is

$$\min\left\{\begin{array}{c} s(U)\left(1 - \prod_{v\in U}\mathbb{P}(D_v \le q)\right), \\ s(U)\left[1 - \max_{v\in U}\mathbb{P}(D_v \le q)\right] + s(U,U)\max_{v\in U}\mathbb{P}(D_v \le q) \end{array}\right\} + \min_{q\in\{0,\dots,C\}} f(q). \tag{26}$$

We analyse the recurrence relation f_k^{II} from which we obtain our lower bound $P_h^{\text{prev}II}$, and derive two intermediate results for the case in which the future costs of recourse are of the form (26).

Lemma 4. *For $U \subseteq V'$ and $s', s, a \ge 0$, if $f(q) = \min\left\{s', s\left[1 - \prod_{v\in U}\mathbb{P}(D_v \le q)\right]\right\} + a$ then, for $u \in V' \setminus U$, $q \in \{0,\dots,C\}$*

$$I(\{u\}, f, s, q) \ge \min\left\{s'\mathbb{P}(D_u \le q) + s\mathbb{P}(D_u > q), s\left[1 - \prod_{v\in U\cup\{u\}}\mathbb{P}(D_v \le q)\right]\right\} + a$$

Proof. We derive

$$\begin{aligned} I(\{u\}, f, s, q) &= \sum_{\xi=0}^{\infty}\mathbb{P}(D_u = \xi)[s\Gamma(\xi - q) + f(C\Gamma(\xi - q) + q - \xi)] \\ &\ge \sum_{\xi=0}^{q}\mathbb{P}(D_u = \xi)f(q - \xi) + \sum_{\xi=q+1}^{\infty}\mathbb{P}(D_u = \xi)[s + a] \\ &= \sum_{\xi=0}^{q}\mathbb{P}(D_u = \xi)\left[\min\left\{s', s\left[1 - \prod_{v\in U}\mathbb{P}(D_v \le q - \xi)\right]\right\} + a\right] + \mathbb{P}(D_u > q)[s + a] \\ &\ge \sum_{\xi=0}^{q}\mathbb{P}(D_u = \xi)\left[\min\left\{s', s\left[1 - \prod_{v\in U}\mathbb{P}(D_v \le q)\right]\right\} + a\right] + \mathbb{P}(D_u > q)[s + a] \\ &= \mathbb{P}(D_u \le q)\left[\min\left\{s', s\left[1 - \prod_{v\in U}\mathbb{P}(D_v \le q)\right]\right\} + a\right] + \mathbb{P}(D_u > q)[s + a] \end{aligned}$$

$$= \mathbb{P}(D_u \le q) \left[\min \left\{ s', s \left[1 - \prod_{v \in U} \mathbb{P}(D_v \le q) \right] \right\} \right] + s\mathbb{P}(D_u > q) + a$$

$$= \min \left\{ \begin{array}{c} s'\mathbb{P}(D_u \le q) + s\mathbb{P}(D_u > q), \\ s\mathbb{P}(D_u \le q) \left[1 - \prod_{v \in U} \mathbb{P}(D_v \le q) \right] + s\mathbb{P}(D_u > q) \end{array} \right\} + a$$

$$= \min \left\{ s'\mathbb{P}(D_u \le q) + s\mathbb{P}(D_u > q), s \left[1 - \prod_{v \in U \cup \{u\}} \mathbb{P}(D_v \le q) \right] \right\} + a,$$

which proves this lemma. □

Lemma 5. *For $U \subseteq V'$ and $s', s, a \ge 0$, if $f(q) = \min\left\{s', s\left[1 - \prod_{v \in U} \mathbb{P}(D_v \le q)\right]\right\} + a$ then, for $u \in V' \setminus U$, $q \in \{0, \dots, C\}$*

$$\min\{s' + I(\{u\}, f, s, C),\ I(\{u\}, f, s, q)\} \ge \min \left\{ s', s \left[1 - \prod_{v \in U \cup \{u\}} \mathbb{P}(D_v \le q) \right] \right\} + a$$

Proof. To prove the this lemma, by Lemma 4 it follows

$$\begin{aligned} \min\{s' + I(\{u\}, f, s, C),\ I(\{u\}, f, s, q)\} &\ge \min\{s' + a,\ I(\{u\}, f, s, q)\} \\ &\ge \min \left\{ \begin{array}{c} s', s'\mathbb{P}(D_u \le q) + s\mathbb{P}(D_u > q), \\ s \left[1 - \prod_{v \in U \cup \{u\}} \mathbb{P}(D_v \le q) \right] \end{array} \right\} + a. \end{aligned}$$

Inspecting the second term of the minimum, we note that it is a convex combination of s and s'. If $s \ge s'$, then $s' \le s\mathbb{P}(D_u > q) + s'\mathbb{P}(D_u \le q)$, so the second term is not the minimum. Otherwise, if $s < s'$, then $s\left[1 - \prod_{v \in U \cup \{u\}} \mathbb{P}(D_v \le q)\right] \le s \le s\mathbb{P}(D_u > q) + s'\mathbb{P}(D_u \le q)$, so the second term is again not the minimum. This means we can eliminate this term and obtain

$$\min\{s' + I(\{u\}, f, s, C),\ I(\{u\}, f, s, q)\} \ge \min \left\{ s', s \left[1 - \prod_{v \in U \cup \{u\}} \mathbb{P}(D_v \le q) \right] \right\} + a,$$

which completes the proof. □

Next, we use Lemmas 1-5 to prove that $P_h^{\text{prev}II}$ is a valid lower bound on the expected costs of recourse.

Theorem 4. *P_h^{prevII} provides a lower bound on the total optimal expected costs or recourse for any route adhering to partial route h.*

Proof. We show that the function $f_k^{II}(q)$ provides a lower bound on the future optimal expected costs of recourse incurred after having served all customers in U_0 through U_k, on any route adhering to partial route $h = (U_0, \dots, U_b)$, for all $q \in \{0, \dots, C\}$, $k \in \{0, \dots, b-1\}$ and fixed orientation. As this includes the case $f_0^{II}(C)$, it follows that $P_h^{\text{prev}II}$ provides a valid lower bound.

For $k \in \{0, \dots, b\}$ and $i \le |U_k|$, denote by $u(k, i) \in V$, the location in U_k which is visited as i-th location of U_k on r, so we may write $r = (0 = u(1, 0), u(1, 1), \dots, u(1, |U_1|), u(2, 1), \dots, u(b-1, |U_{b-1}|), u(b, 1) = 0)$. We define $f_{k,i}^*(q)$ like in (9) as the optimal expected costs of recourse on the remainder of route r, in the fixed orientation, after having served customer $u(k, i)$. That f_k^{II} should form an appropriate lower bound is equivalently stated as $f_{k,|U_k|}^*(q) \ge f_k^{II}(q)$, for all $k \in \{0, \dots, b\}$ and $q \in \{0, \dots, C\}$, which we will prove.

We show that $f^*_{k,|U_k|}(q) \geq f^{II}_k(q)$ by means of a nested backward induction, which we distinguish as outer and inner inductions. The outer backward induction is on k. The outer base case, $k = b$, is trivially satisfied, as $f^*_{b,1}(q) = 0 = f^{II}_b(q)$ by definition. The outer induction hypothesis is that $f^*_{k+1,|U_{k+1}|}(q) \geq f^{II}_{k+1}(q)$ for some $k \in \{0, \dots, b-1\}$, and the outer inductive step is to prove that also $f^*_{k,|U_k|}(q) \geq f^{II}_k(q)$. We consider two cases:

Case I: $|U_{k+1}| = 1$. In this case, we can write $U_{k+1} = \{u\}$ with $u(k+1, 1) = u$, and $F_{U_{k+1}}(f, q)$ is simply given by $F_u(f, q)$, which results in a relatively simple expression of $f^{II}_k(q)$. We derive

$$\begin{aligned} f^*_{k,|U_k|}(q) &= \min\{s_{u(k,|U_k|),u} + F_u(f^*_{k+1,1}, C),\ F_u(f^*_{k+1,1}, q)\} \\ &\geq \min\{s(U_k, U_{k+1}) + F_u(f^{II}_{k+1}, C),\ F_u(f^{II}_{k+1}, q)\} \\ &= f^{II}_k(q), \end{aligned}$$

where the inequality follows from the outer induction hypothesis, and because $f^*_{k+1,1}$ appears in F_u only with positive coefficients.

Case II: $|U_{k+1}| > 1$. In this case, the expression of $f^{II}_k(q)$ is more convoluted. A reason for this, is that $F_{U_{k+1}}(f^{II}_{k+1}, q)$ is now the largest of the expressions (25) and (26), particular to U_{k+1} and f^{II}_{k+1}. We shall consider these expressions separately, and show that they are both lower bounds on the optimal expected costs $F_{u(k+1,1)}(f^*_{k+1,1}, q)$, from which it follows that also in this case $f^*_{k,|U_k|}(q) = \min\{s_{u(k,|U_k|),u(k+1,1)} + F_{u(k+1,1)}(f^*_{k+1,1}, C), F_{u(k+1,1)}(f^*_{k+1,1}, q)\} \geq \min\{s(U_k, U_{k+1}) + F_{U_{k+1}}(f^{II}_{k+1}, C), F_{U_{k+1}}(f^{II}_{k+1}, q)\} = f^{II}_k(q)$. For notational convenience, denote $(u(k+1, 1), \dots, u(k+1, |U_k+1|)) = (1, \dots, u)$ and $U_k = \{z\}$.

First, we show that the value of expression (25) is less than or equal to the optimal expected costs $F_{u(k+1,1)}(f^*_{k+1,1}, q)$ for all $q \in \{0, \dots, C\}$. To prove this, we recursively construct a function f'_i with the property that it is less than or equal to the optimal expected costs function f^*_i, for all $i \in \{1, \dots, u\}$, which we use to arrive at our result. For $q \in \{0, \dots, C\}$, define $f'_u(q) = H(f^{II}_{k+1}, \bar{s}(U_{k+1}), q)$, and $f'_l(q) = I(\{l+1\}, f'_{l+1}, \bar{s}(U_{k+1}), q)$ for all $l \in \{1, \dots, u-1\}$. By backward induction on l, we show that $f^*_l(q) \geq f'_l(q)$ for all $q \in \{0, \dots, C\}$, that f'_l is decreasing in q, and that $f'_l(0) - f'_l(C) \leq \bar{s}(U_{k+1})$.

The inner base case, $l = u$, follows from the outer induction hypothesis $f^*_u(q) \geq f^{II}_{k+1}(q)$, and Lemma 3 which states that $f^{II}_{k+1}(q) \geq H(f^{II}_{k+1}, \bar{s}(U_{k+1}), q) = f'_u(q)$, f'_u is decreasing, and $f'_u(0) - f'_u(C) \leq \bar{s}(U_{k+1})$. The inner induction hypothesis is that $f^*_{l+1}(q) \geq f'_{l+1}(q)$ for all $q \in \{0, \dots, C\}$, f'_{l+1} is decreasing in q and that $f'_{l+1}(0) - f'_{l+1}(q) \leq \bar{s}(U_{k+1})$ for some $l \in \{1, \dots, u-1\}$ and we proceed with the inductive step.

Because we defined $f'_l(q)$ as $f'_l(q) = I(\{l+1\}, f'_{l+1}, \bar{s}(U_{k+1}), q)$, by Lemma 1 and the inner induction hypothesis $f'_l(q)$ is decreasing in q and $f'_l(0) - f'_l(C) \leq \bar{s}(U_{k+1})$. Furthermore, we derive

$$\begin{aligned} f^*_l(q) &= \min\{s_{(l,l+1)} + F_{l+1}(f^*_{l+1}, C), F_{l+1}(f^*_{l+1}, q)\} \\ &= \min\{s_{(l,l+1)} + I(\{l+1\}, f^*_{l+1}, s_{l+1}, C), I(\{l+1\}, f^*_{l+1}, s_{l+1}, q)\} \\ &\geq \min\{\bar{s}(U_{k+1}) + I(\{l+1\}, f'_{l+1}, \bar{s}(U_{k+1}), C), I(\{l+1\}, f'_{l+1}, \bar{s}(U_{k+1}), q)\} \\ &= \min\{\bar{s}(U_{k+1}) + f'_l(C), f'_l(q)\} \\ &= f'_l(q), \end{aligned}$$

where the inequality follows from the definition of I, using $\bar{s}(U_{k+1}) \leq s_{l+1}$ and $\bar{s}(U_{k+1}) \leq s_{(l,l+1)}$ and the inner induction hypothesis stating that $f^*_{l+1}(q) \geq f'_{l+1}(q)$ for all $q \in \{0, \dots, C\}$. This concludes the inner proof by induction.

Observe for $l \in \{0, \dots, u-1\}$ that by repeatedly applying Lemma 2 we find $f'_l(q) = I(\{l+$

$1,\dots,u\}, f'_u, \bar{s}(U_{k+1}), q)$. This implies

$$F_1(f_1^*, q) \geq F_1(f_1', q) = I(U_{k+1}, f_u', \bar{s}(U_{k+1}), q), \tag{27}$$

or equivalently that $F_{u(k+1,1)}(f^*_{k+1,1}, q) \geq I\left(U_{k+1}, H\left(f^{II}_{k+1}, \bar{s}(U_{k+1}), \cdot\right), \bar{s}(U_{k+1}), q\right) = F_{U_{k+1}}(f^{II}_{k+1}, q)$, which proves that the first expression is indeed a lower bound.

Next, we consider the second expression (26). We proceed similarly as with the first expression, by constructing a function f''_l for all $l \in \{1,\dots,u\}$ with the property that it is less than or equal to the optimal expected costs function f^*_l, for all $l \in \{1,\dots,u\}$. We define f''_l

$$f''_l(q) = \min\left\{ s(U_{k+1}, U_{k+1}), s(U_{k+1})\left[1 - \prod_{v\in\{l+1,\dots,u\}} \mathbb{P}(D_v \leq q)\right]\right\} + \min_{q'\in\{0,\dots,C\}} f^{II}_{k+1}(q')$$

Next, we show by backward induction on l that $f^*_l(q) \geq f''_l(q)$ for all $l \in \{1,\dots,u\}$. In the inner base case $l = u$, we have by the outer induction hypothesis $f^*_u(q) \geq f^{II}_{k+1}(q) \geq \min_{q'\in\{0,\dots,C\}} f^{II}_{k+1}(q') = f''_u(q)$. Next, we assume the inner induction hypothesis $f^*_{l+1}(q) \geq f''_{l+1}(q)$ for some $l \in \{1,\dots,u-1\}$. Then, by using the induction hypothesis and Lemma 5, we proceed with the inductive step and derive

$$\begin{aligned}
f^*_l(q) &= \min\{s_{(l,l+1)} + F_{l+1}(f^*_{l+1}, C), F_{l+1}(f^*_{l+1}, q)\} \\
&= \min\{s_{(l,l+1)} + I(\{l+1\}, f^*_{l+1}, s_{l+1}, C), I(\{l+1\}, f^*_{l+1}, s_{l+1}, q)\} \\
&\geq \min\{s(U_{k+1}, U_{k+1}) + I(\{l+1\}, f''_{l+1}, s(U_{k+1}), C), I(\{l+1\}, f''_{l+1}, s(U_{k+1}), q)\} \\
&\geq \min\left\{ s(U_{k+1}, U_{k+1}), s(U_{k+1})\left[1 - \prod_{v\in\{l+1,\dots,u\}} \mathbb{P}(D_v \leq q)\right]\right\} + \min_{q'\in\{0,\dots,C\}} f^{II}_{k+1}(q') \\
&= f''_l(q).
\end{aligned}$$

This concludes the inner proof by induction, which implies

$$F_1(f_1^*, q) \geq F_1(f_1'', q) = I(\{1\}, f_1'', s(U_{k+1}), q). \tag{28}$$

Combining (27) and (28) yields

$$\begin{aligned}
F_1(f_1^*, q) &\geq \max\{I(U_{k+1}, f'_u, \bar{s}(U_{k+1}), q),\ I(\{1\}, f''_1, s(U_{k+1}), q)\} \\
&\geq \max\left\{\begin{array}{c}
\sum_{\xi\in U_{k+1}} \mathbb{P}\left(\sum_{v\in U_{k+1}} D_v \leq \xi\right)[s\Gamma(\xi - q) + f'_u(C\Gamma(\xi - q) + q - \xi)], \\
\min\left\{\begin{array}{c} s(U_{k+1}, U_{k+1})\mathbb{P}(D_1 \leq q) + s(U_{k+1})\mathbb{P}(D_1 > q) \\ s(U_{k+1})\left[1 - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right]\end{array}\right\} + \min_{q'\in\{0,\dots,C\}} f^{II}_{k+1}(q')
\end{array}\right\},
\end{aligned}$$

where the last equality follows from Lemma 4. Finally, we note that $s(U_{k+1}, U_{k+1})\mathbb{P}(D_1 \leq q) + s(U_{k+1})\mathbb{P}(D_1 > q) = s(U_{k+1}) + [s(U_{k+1}, U_{k+1}) - s(U_{k+1})]\mathbb{P}(D_1 \leq q)$. If $s(U_{k+1}) \geq s(U_{k+1}, U_{k+1})$, we have that $s(U_{k+1}) + [s(U_{k+1}, U_{k+1}) - s(U_{k+1})]\mathbb{P}(D_1 \leq q) \geq s(U_{k+1}) + [s(U_{k+1}, U_{k+1}) - s(U_{k+1})]\max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)$. Otherwise $s(U_{k+1}) < s(U_{k+1}, U_{k+1})$, in which case it follows that $s(U_{k+1}) + [s(U_{k+1}, U_{k+1}) - s(U_{k+1})]\mathbb{P}(D_1 \leq q) \geq s(U_{k+1}) \geq s(U_{k+1})\left[1 - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right]$. As a result, we can replace

$\mathbb{P}(D_1 \le q)$ with $\max_{v \in U_{k+1}} \mathbb{P}(D_v \le q)$ to obtain

$$F_1^*(f_1^*, q) \ge \max \left\{ \begin{array}{c} \sum_{\xi \in U_{k+1}} \mathbb{P}\left(\sum_{v \in U_{k+1}} D_v \le \xi \right) [s\Gamma(\xi - q) + f_u'(C\Gamma(\xi - q) + q - \xi)], \\ \min \left\{ \begin{array}{c} s(U_{k+1}, U_{k+1}) \max_{v \in U_{k+1}} \mathbb{P}(D_v \le q) + s(U_{k+1}) \left[1 - \max_{v \in U_{k+1}} \mathbb{P}(D_v \le q) \right], \\ s(U_{k+1}) \left[1 - \prod_{v \in U_{k+1}} \mathbb{P}(D_v \le q) \right] \end{array} \right\} + \min_{q' \in \{0,\dots,C\}} f_{k+1}^{II}(q') \end{array} \right\}$$
$$= F_{U_{k+1}}(f_{k+1}^{II}, q).$$

We conclude Case II by deriving

$$\begin{aligned} f_u^*(q) &= \min\{s_{(u,1)} + F_1(f_1^*, C), F_1(f_1^*, q)\} \\ &\ge \min\{s(U_k, U_{k+1}) + F_{U_{k+1}}(f_{k+1}^{II}, C), F_{U_{k+1}}(f_{k+1}^{II}, q)\} \\ &= f_k^{II}(q). \end{aligned}$$

Combining the results of Cases I and II completes the outer inductive step, showing that f_k^{II}, and by extension $P_h^{\text{prev}II}$, forms a valid lower bound on the expected costs of recourse.

□

Now that we have proven that $P_h^{\text{prev}II}$ is a valid lower bound, we continue to show that it also dominates the bound obtained from Louveaux and Salazar-González (2018). In fact, we prove that $f_k^{II}(q) \ge f_k^L(q)$ for all $k \in \{0, \dots, b\}$ and $q \in \{0, \dots, C\}$.

Theorem 5. *For all $k \in \{0, \dots, b\}$, it holds that $f_k^{II}(q) \ge f_k^L(q)$.*

Proof. In this proof, we use backward induction on k. The base case is trivial, as $f_b^{II}(q) = 0 = f_b^L(q)$. The induction hypothesis is $f_{k'}^{II}(q) \ge f_{k'}^L(q)$ for all $k' > k$, with $k \in \{0, \dots, b-1\}$. Next, we proceed with the inductive step.

Recall that $f_k^L(q) = \max\{f_k^{L1}(q), f_k^{L2}(q), f_k^{L3}(q), f_k^{L4}(q), f_k^{L5}(q)\}$. To prove the inductive step, we show one by one that $f_k^{II}(q)$ is greater or equal to each of the five auxiliary functions constituting $f_k^L(q)$. Before considering these individual cases, note that $f_k^{II}(q) \ge 0$, which means we only have to check the cases for which $f_k^L(q)$, or equivalently any of its components, are strictly positive.

Case I. The function $f_k^{L1}(q)$ is only nonzero when $k = 0$, $q = C$ and $|U_1| > 1$. We derive

$$\begin{aligned} f_0^{II}(C) &= F_{U_1}(f_1^{II}, C) \\ &\ge \sum_{\xi \in \Xi_U} \mathbb{P}\left(\sum_{v \in U} D_v = \xi \right) [\Gamma(\xi - q)\bar{s}(U) + H(f, \bar{s}(U), C\Gamma(\xi - q) + q - \xi)] \\ &\ge \bar{s}(U_1) \mathbb{P}\left(\sum_{v \in U_1} D_v > C \right) \\ &= f_k^{LI}(C). \end{aligned}$$

Case II. The function $f_k^{L2}(q)$ is only nonzero when $|U_{k+1}| > 1$. We derive

$$\begin{aligned} f_k^{II}(q) &= \min\{s(U_k, U_{k+1}) + F_{U_{k+1}}(f_{k+1}^{II}, C),\ F_{U_{k+1}}(f_{k+1}^{II}, q)\} \\ &\ge \min\{s(U_k, U_{k+1}),\ F_{U_{k+1}}(0, q)\} \end{aligned}$$

$$\geq \min\left\{s(U_k, U_{k+1}),\ \bar{s}(U_{k+1})\mathbb{P}\left(\sum_{v\in U_{k+1}} D_v > q\right)\right\}$$
$$= f_k^{LII}(q).$$

Case III. The function $f_k^{L3}(q)$ is only nonzero when $|U_{k+1}| > 1$ and $s(U_{k+1}, U_{k+1}) \geq s(U_{k+1})\left(1 - \frac{\prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)}{\max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)}\right)$. We derive

$$\begin{aligned}
f_k^{II}(q) &\geq \min\{s(U_k, U_{k+1}),\ F_{U_{k+1}}(0, q)\}\\
&\geq \min\Bigg\{s(U_k, U_{k+1}), s(U_{k+1})\left(1 - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right),\\
&\qquad s(U_{k+1})(1 - \max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)) + s(U_{k+1}, U_{k+1}) \max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\Bigg\}\\
&\geq \min\Bigg\{s(U_k, U_{k+1}), s(U_{k+1})\left(1 - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right),\\
&\qquad s(U_{k+1})(1 - \max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)) +\\
&\qquad s(U_{k+1})\left[\max_{v\in U_{k+1}} \mathbb{P}(D_v \leq q) - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right]\Bigg\}\\
&\geq \min\left\{s(U_k, U_{k+1}), s(U_{k+1})\left(1 - \prod_{v\in U_{k+1}} \mathbb{P}(D_v \leq q)\right)\right\}\\
&= f_k^{LIII}(q).
\end{aligned}$$

Case IV. The function $f_k^{L4}(q)$ is only nonzero when $|U_{k+1}| > 1$, $U_{k+2} = \{u\}$ and $\min\left\{s(U_{k+1}, U_{k+2}),\ \sum_{\xi=0}^{q'} \mathbb{P}(D_u = \xi) f_{k+2}^{L}(q' - \xi) + s_u \mathbb{P}(D_u > q')\right\} \geq \bar{s}(U_{k+1})$ for all $q' \leq q$. We derive

$$\begin{aligned}
f_k^{II}(q) &\geq \min\{s(U_k, U_{k+1}),\ F_{U_{k+1}}(f_{k+1}^{II}, q)\}\\
&\geq \min\left\{\begin{array}{c} s(U_k, U_{k+1}),\\ \sum\limits_{\xi\in U_{k+1}} \mathbb{P}\left(\sum\limits_{v\in U_{k+1}} D_v = \xi\right)[\Gamma(\xi - q)\bar{s}(U_{k+1}) + H(f_{k+1}^{II}, \bar{s}(U_{k+1}), C\Gamma(\xi - q) + q - \xi)]\end{array}\right\}\\
&\geq \min\left\{\begin{array}{c} s(U_k, U_{k+1}),\\ \sum\limits_{\xi=0}^{q} \mathbb{P}\left(\sum\limits_{v\in U_{k+1}} D_v = \xi\right) H(f_{k+1}^{II}, \bar{s}(U_{k+1}), q - \xi) + \sum\limits_{\xi=q+1}^{\infty} \mathbb{P}\left(\sum\limits_{v\in U_{k+1}} D_v = \xi\right)\bar{s}(U_{k+1})\end{array}\right\}\\
&\geq \min\left\{\begin{array}{c} s(U_k, U_{k+1}),\\ \mathbb{P}\left(\sum\limits_{v\in U_{k+1}} D_v \leq q\right) H(f_{k+1}^{II}, \bar{s}(U_{k+1}), q) + \mathbb{P}\left(\sum\limits_{v\in U_{k+1}} D_v > q\right)\bar{s}(U_{k+1})\end{array}\right\},
\end{aligned}$$

where the last inequality follows from property 1 of Lemma 3. We note that, by using the induction hypothesis in this case, for all $q' \leq q$

$$\begin{aligned}
f_{k+1}^{II}(q') &= \min\{s(U_{k+1}, U_{k+2}) + F_{U_{k+2}}(f_{k+2}^{II}, C), F_{U_{k+2}}(f_{k+2}^{II}, q)\}\\
&\geq \min\{s(U_{k+1}, U_{k+2}), F_u(f_{k+2}^{II}, q)\}
\end{aligned}$$

$$\geq \min\{s(U_{k+1}, U_{k+2}), \sum_{\xi=0}^{q'} \mathbb{P}(D_u = \xi) f^{II}_{k+2}(\xi - q') + s_u \mathbb{P}(D_u > q')\}$$

$$\geq \min\{s(U_{k+1}, U_{k+2}), \sum_{\xi=0}^{q'} \mathbb{P}(D_u = \xi) f^{L}_{k+2}(\xi - q') + s_u \mathbb{P}(D_u > q')\}$$

$$\geq \bar{s}(U_{k+1}).$$

As a result, we obtain $H(f^{II}_{k+1}, \bar{s}(U_{k+1}), q) = \min\{\min_{q'} f^{II}_k(q') + \bar{s}(U_{k+1}), \min_{q' \leq q} f^{II}_{k+1}(q')\} \geq \bar{s}(U_{k+1})$. Combining the above results yields

$$\begin{aligned} f^{II}_k(q) &\geq \min\left\{ s(U_k, U_{k+1}),\ \mathbb{P}\left(\sum_{v \in U_{k+1}} D_v \leq q\right) \bar{s}(U_{k+1}) + \mathbb{P}\left(\sum_{v \in U_{k+1}} D_v > q\right) \bar{s}(U_{k+1}) \right\} \\ &= \min\left\{s(U_k, U_{k+1}),\ \bar{s}(U_{k+1})\right\} \\ &= f^{LIV}_k(q). \end{aligned}$$

Case V. The function $f^{L5}_k(q)$ is only nonzero when $U_k = \{u\}$ and $U_{k+1} = \{v\}$. By the induction hypothesis $f^{II}_{k+1}(q) \geq f^{L}_{k+1}(q)$, which yields

$$f^{II}_k(q) = \min\{s_{(u,v)} + F_v(f^{II}_{k+1}, C), F_v(f^{II}_{k+1}, q)\} \geq \min\{s_{(u,v)} + F_v(f^{L}_{k+1}, C), F_v(f^{L}_{k+1}, q)\} = f^{LV}_k(q).$$

Since $f^{II}_k(q)$ is greater than or equal to each of the five auxiliary functions of $f^L_k(q)$. It follows that $f^{II}_k(q) \geq f^L_k(q)$ for all $q \in \{0, \dots, C\}$, completing the inductive step, which concludes the proof. □

D Partial route lower bound under identically independently distributed demands

Theorem 6. *Under identically independently distributed demands $P^{prevIII}_h = P^{prevI}_h$.*

Proof. In the calculation of $P^{\text{prev}III}_h$, $f^I_{uj}(q)$ and $f^{II}_k(q)$ are updated whenever $U_k = U(j) = \{u\}$. Therefore, it is sufficient to show that $f^I_{uj}(q) \geq f^{II}_k(q)$ *before* the update. We do this by using backwards induction on k, for all k, j and u such that $U_k = U(j) = \{u\}$ and for all $q \in \{0, \dots, C\}$. The base case, $k = b$, $j = \sum_{i=0}^{b-1} |U_i|$ and $u = 0$, is trivial, as $f^I_{uj}(q) = 0 = f^{II}_k(q)$ for all $q \in \{0, \dots, C\}$ per definition.

The induction hypothesis is that $f^I_{u'j'}(q) \geq f^{II}_{k'}(q)$ for all $k' > k$, $j' > j$ and u' such that $U_{k'} = U(j') = \{u'\}$, $q \in \{0, \dots, C\}$ before the update rule, for some $k \in \{0, \dots, b-1\}$ and j such that $U_k = U(j) = \{u\}$. The inductive step is to prove that $f^I_{uj}(q) \geq f^{II}_k(q)$ for all $q \in \{0, \dots, C\}$ before the update rule. We consider two cases:

Case I: $U_{k+1} = U(j+1) = \{u\}$**.** In this case, we derive $f^I_{uj}(q) = \min\{s_{(u,v)} + F_u(f^I_{v,j+1}, C), F_v(f^I_{v,j+1}, q)\}$ and $f^{II}_k(q) = \min\{s_{(u,v)} + F_v(f^{II}_{k+1}, C), F_v(f^{II}_{k+1}, q)\}$. As $f^I_{v,j+1}(q) = f^{II}_{k+1}(q)$ due to the update rule, we know that $f^I_{uj}(q) = f^{II}_k(q)$.

Case II: $|U_{k+1}| > 1$**.** In this case, $U_{k+2} = U(j + |U_{k+1}| + 1) = \{v\}$, which implies $f^I_{v,j+|U_{k+1}|+1}(q) = f^{II}_{k+2}(q)$ due to the update rule. We construct a modified VRPSD instance, for which $f^I_{uj}(q)$ forms an upper bound on the expected costs of recourse and $f^{II}_k(q)$ a lower bound. The instance is defined identically as the original instance, except that the preventive and corrective recourse costs for customers in U_k, U_{k+1} and U_{k+2} are modified. The modified recourse costs are $s'_{(u,v')} = s(\{u\}, U_{k+1})$, $s'_{(v',v'')} = s(U_{k+1}, U_{k+1})$, $s'_{v'} = s(U_{k+1})$ and $s'_{(v',v)} = s(U_{k+1}, \{v\})$ for all $v', v'' \in U_{k+1}$.

We denote $f'_i(q)$ as the expected costs of future recourse on the modified instance after serving the ith customer on the route, with q load in the vehicle and $i \in \{j, \dots, j + |U_{k+1}| + 1\}$. We define $f'_{j+|U_{k+1}|+1}(q) = f^{II}_{k+2}(q)$ as the future expected costs of recourse after serving U_{k+2}. Note that the customer order in U_{k+1} is irrelevant, due to the identical distributions and the fact that the cost of both corrective and preventive recourse is identical for all customers in U_{k+1} in the modified instance.

We note that $f'_j(q)$ and $f'_l(q)$ are obtained for all $l \in \{j+1, \dots, j + |U_{k+1}|\}$, by replacing $s_{(u,v')}$, $s_{(v',v'')}$, $s_{v'}$ and $s_{(v',v)}$ in the calculation of $f^I_{uj}(q)$ and all $f^I_{v'l}(q)$ with the smaller values $s'_{(u,v')}$, $s'_{(v',v'')}$, $s'_{v'}$ and $s'_{(v',v)}$ respectively, for all $v', v'' \in U_{k+1}$. We conclude $f^I_{uj}(q) \geq f'_j(q)$ and $f^I_{v'l}(q) \geq f'_l(q)$ for all $v' \in U_{k+1}$, $l \in \{j+1, \dots, j + |U_{k+1}|\}$ and $q \in \{0, \dots, C\}$.

Next, we compute the bound $P^{\text{prev}II}_h$ described in Section 5.2.3 for the modified instance, which is done using the recursive function which we denote here by $f'^{II}_k(q)$. As it is shown in Appendix C that this function provides lower bounds on the expected cost of future recourse, it follows that $f'_j(q) \geq f'^{II}_k(q)$. However, inspection of $f'^{II}_k(q)$ reveals that it is equivalent to $f^{II}_k(q)$, due to the definition of the costs of recourse in this modified instance. As a result, we obtain $f^I_{uj}(q) \geq f'_j(q) \geq f'^{II}_k(q) = f^{II}_k(q)$, proving this case.

By induction, we have proven that $f^I_{uj}(q) \geq f^{II}_k(q)$ *before* the update rule for all k, j and u such that $U_k = U(j) = \{u\}$ and for all $q \in \{0, \dots, C\}$. This means that $f^I_{uj}(q)$ is not affected by the update rule, implying that $P^{\text{prev}I}_h = P^{\text{prev}III}_h$ in the case of identically independently distributed demands. □

E Activation function by Laporte et al. (2002)

Next, we discuss the partial route activation function of Laporte et al. (2002), and provide a counterexample which shows that it is incorrect. Denote by $\delta(S, T)$, for $S, T \subseteq V'$ such that $S \cap T = \emptyset$, the set of edges between the sets S and T. Consider the partial route $h = (U_0, \dots, U_b)$. If $b = 2$, the function W^L_h used as partial activation function by Laporte et al. (2002) can be written as

$$W^L_h(x) = \sum_{k=1}^{b-1} \left(\sum_{e \in E(U_k)} x_e - (|U_k| - 1) \right) + \left(\sum_{e \in \delta(U_0, U_1)} x_e - 1 \right), \text{ if } b = 2.$$

However, it turns out that this is not an an activation function for all partial routes h. Consider the partial route $h = (\{0\}, \{1, 2\}, \{0\})$. The function is $W_h(x) = x_{(1,2)} + x_{(0,1)} + x_{(0,2)} - 2$. A solution x corresponding with the routes $(0, 1, 0)$ and $(0, 2, 0)$, does not adhere to h, while $W^L_h(x) = 2$. For $b \geq 3$, the function W^L_h used as partial activation function by Laporte et al. (2002) can be written as

$$W^L_h(x) = \sum_{k=1}^{b-1} \left(\sum_{e \in E(U_k)} x_e - (|U_k| - 1) \right) + \sum_{k=0}^{b-1} \left(\sum_{e \in \delta(U_k, U_{k+1})} x_e - 1 \right) + 1, \text{ if } b \geq 3.$$

Again there is an issue. Consider the partial route $h = (\{0\}, \{1, 2\}, \{3\}, \{0\})$. The activation function is $W^L_h(x) = x_{(1,2)} + x_{(0,1)} + x_{(0,2)} + x_{(1,3)} + x_{(2,3)} + x_{(0,3)} - 3$. A solution x corresponding with the route $(0, 1, 3, 2, 0)$ which does not adhere to h, still results in $W^L_h(x) = 1$. This demonstrates that $W^L_h(x)$ is not a partial route activation function, which means that the corresponding partial route inequality is in general also not valid.

F Partial route activation function

In this section, we prove that the partial route activation function $W_h(x)$ as found in (18) is a valid partial route activation function. We use the shorthand notation $x(S) = \sum_{e \in E(S)} x_e$ for every $S \subseteq V$

and $x(S,T) = \sum_{e \in \delta(S,T)} x_e$ for $S \subseteq V$ and $T \subseteq V \setminus S$. For ease of writing, let $x(v,S) \equiv x(\{v\},S)$ for $v \in V$ and $S \subseteq V' \setminus \{v\}$, and even $x(u,v) \equiv x(\{u\},\{v\})$ for $u,v \in V$ such that $u \neq v$. This allows us to express $W_h(x)$ as follows.

$$W_h(x) = \sum_{k=1}^{b-1} \alpha_k \left(x(U_k) - (|U_k| - 1)\right) + \sum_{k=0}^{b-1} \beta_k \left(x(U_k, U_{k+1}) - 1\right) + \gamma \tag{29}$$

where the coefficients are defined as follows

$$(\alpha_1, \ldots, \alpha_{b-1}) = \begin{cases} (3) & \text{if } b = 2, \\ (4,4) & \text{if } b = 3, \\ (3,2,3) & \text{if } b = 4, \\ (3,2,1,\ldots,1,2,3) & \text{if } b \geq 5, \end{cases}$$

$$(\beta_0, \ldots, \beta_{b-1}) = \begin{cases} (1,0) & \text{if } b = 2, \\ (1,3,1) & \text{if } b = 3, \\ (1,2,1,\ldots,1,2,1) & \text{if } b \geq 4, \end{cases}$$

$$\gamma = \begin{cases} 0 & \text{if } b = 2, \\ 1 & \text{if } b \geq 3. \end{cases}$$

Let us first outline our proof of $W_h(x)$ being a partial route activation function. We decompose $W_h(x)$ into parts. The coefficient of each variable $x_{(u,v)}$ in $W_h(x)$, with $(u,v) \in E$, depends on b and the positions of u and v in the partial route. Therefore, we distinguish several types of parts, which are sufficient to decompose $W_h(x)$. For each of these parts we separately derive an upper bound on their value. This allows us to identify an upper bound on the value of $W_h(x)$ depending on x, and conditions on x for which this upper bound is attained. In particular, it allows us to conclude that $W_h(x)$ is a partial route activation function. Next, we proceed as outlined, by first showing intermediate results before arriving at the main theorem. We first provide two lemmas which characterize the values of a feasible integer solution x if it corresponds with a route that adheres to the partial route h or not.

Lemma 6. *For a partial route h with $b = 2$, a feasible integer solution x to the Basic-VRPSD corresponds with a solution which includes a route that adheres to h, if and only if $x(U_1) = |U_1| - 1$ and $x(0, U_1) = 2$.*

Proof. If $x(U_1) = |U_1| - 1$, by the degree constraints (2) and subtour elimination constraints (3), it follows that the edges selected by $x(U_1)$ correspond with a single path $P(U_1)$ visiting all customers in U_1. If additionally $x(0, U_1) = 2$ or, by the degree constraints, it follows that two edges are selected from the depot to each of the end points of the path $P(U_1)$. We conclude that x corresponds to a route which adheres to h. The reverse is immediate. □

Lemma 7. *For a partial route h with $b \geq 3$, a feasible integer solution x to the Basic-VRPSD corresponds with a solution which includes a route that adheres to h, if and only if $x(U_k) = |U_k| - 1$ and $x(U_k, U_{k+1}) = 1$ for all $k \in \{0, \ldots, b-1\}$.*

Proof. This proof is similar to that of Lemma 6. If $x(U_k) = |U_k| - 1$ for all $k \in \{0, \ldots, b-1\}$, by the degree and subtour elimination constraints, it follows that the edges selected by $x(U_k)$ correspond with a single path $P(U_k)$ visiting all customers in U_k. Note that if $|U_k| = 1$, we say that the path $P(U_k)$ consists of a single vertex but no edges. Suppose additionally that $x(U_k, U_k + 1) = 1$ for all $k \in \{0, \ldots, b-1\}$. Because $x(0, U_1) = x(0, U_{b-1}) = 1$, by the degree constraints, two edges are selected from the depot to one of the endpoints of the path in U_1 and in U_k. Furthermore, for each $k \in \{1, \ldots, b-1\}$ such that

$|U_k| = 1$, say $U_k = \{u\}$, by the degree constraints, two edges are selected which connect an endpoint of the path in U_{k-1} through u with U_{k+1}. Note that each endpoint of the path in U_k, for $k \in \{1, \dots, b-1\}$ such that $|U_k| > 1$, is not connected with an edge to both a vertex in U_{k-1} and U_{k+2} by the degree constraints. We conclude that x corresponds to a route which adheres to h. The reverse is immediate. □

The significance of Lemmas 6 and 7 is that we can express adherence to h in terms of the values of $x(U_k)$ and $x(U_k, U_{k+1})$ for $k \in \{0, \dots, b-1\}$. Appropriately decomposing $W_h(x)$ into parts consisting of these terms $x(U_k)$ and $x(U_k, U_{k+1})$, allows us to assess the value of $W_h(x)$ if x does and does not correspond with a route that adheres to h. Next, we first separately go over the parts which we will encounter in our decomposition.

Lemma 8. *For any $S \subseteq V'$, $v \in V'$ with $v \notin S$ and x a feasible integer solution to the Basic-VRPSD, it holds that $x(0, S) + 2x(S) + x(v, S) \leq 2|S|$.*

Proof. Observe that by the degree constraints it holds that $x(S, V' \setminus S) + 2x(S) = 2|S|$, such that $x(0, S) + 2x(S) + x(v, S) \leq 2|S|$. □

Lemma 9. *For any $S \subseteq V'$ and x an feasible integer solution to the Basic-VRPSD, it holds that $x(0, S) + 3x(S) \leq 3|S| - 1$, and equality holds if and only if $x(0, S) = 2$ and $x(S) = |S| - 1$.*

Proof. The subtour elimination constraint of S is $x(S) \leq |S| - 1$, and similar to Lemma 8 we have that $x(0, S) + 2x(S) \leq 2|S|$. Hence, $x(0, S) + 3x(S) \leq= 3|S| - 1$. If $x(0, S) = 2$ and $x(S) = |S| - 1$, then $x(0, S) + 3x(S) = 3|S| - 1$, i.e., equality holds. Conversely, suppose $x(0, S) + 3x(S) = 3|S| - 1$. Because x satisfies the subtour elimination constraint $x(S) \leq |S| - 1$, if $x(0, S) \leq 1$, the equation cannot hold. If $x(0, S) \geq 3$, because $x(0, S) + 2x(S) \leq 2|S|$ which implies $x(S) \leq |S| - \frac{1}{2}x(0, S)$, it follows that the equation also cannot hold since $x(0, S) + 3x(S) \leq 3|S| - \frac{3}{2} < 3|S| - 1$. Therefore, if $x(0, S) + 3x(S) = 3|S| - 1$ then $x(0, S) = 2$, and it follows that $x(S) = |S| - 1$. □

Lemma 10. *For any $S \subseteq V'$, $v \in V'$ with $v \notin S$ and x a feasible integer solution to the Basic-VRPSD, it holds that $x(0, S) + 3x(S) + 2x(v, S) \leq 3|S|$, and equality holds if and only if either $x(0, S) = 1$, $x(S) = |S| - 1$ and $x(v, S) = 1$, or $x(0, S) = 2$, $x(S) = |S| - 2$ and $x(v, S) = 2$.*

Proof. The subtour elimination constraint of $S \cup \{v\}$ is $x(S) + x(v, S) \leq |S|$, and by Lemma 8 it holds that $x(0, S) + 2x(S) + x(v, S) \leq 2|S|$. Hence, $x(0, S) + 3x(S) + 2x(v, S) \leq 3|S|$. If either $x(0, S) = 1$, $x(S) = |S| - 1$ and $x(v, S) = 1$, or $x(0, S) = 2$, $x(S) = |S| - 2$ and $x(v, S) = 2$, then equality holds. Conversely, suppose $x(0, S) + 3x(S) + 2x(v, S) = 3|S|$. By the degree constraints, $x(v, S)$ might only take the values 0, 1 and 2.

First, if $x(v, S) = 0$, because $x(0, S) + 2x(S) + x(v, S) \leq 2|S|$ by Lemma 8 and the subtour elimination constraint for S is $x(S) \leq |S| - 1$, it follows that $3|S| = x(0, S) + 3x(S) + 2x(v, S) \leq 3|S| - 1$ which is a contradiction. So $x(v, S)$ cannot be 0.

Next, suppose $x(v, S) = 1$. Lemma 8 implies $x(S) \leq |S| - \frac{1}{2}x(0, S) - \frac{1}{2}x(v, S)$. Therefore, $3|S| = x(0, S) + 3x(S) + 2x(v, S) \leq 3|S| - \frac{1}{2}x(0, S) + \frac{1}{2}x(v, S)$. Since $x(v, S) = 1$, $x(0, S) \leq 1$. If $x(0, S) = 0$, then by the subtour elimination constraint for S, it follows that $3|S| = x(0, S) + 3x(S) + 2x(v, S) = 3x(S) + 2 \leq 3|S| - 1$, which is a contradiction, so $x(0, S)$ cannot be 0 if $x(v, S) = 1$. If $x(0, S) = 1$, it follows that $x(S) = |S| - 1$.

Finally, suppose $x(v, S) = 2$. It similarly follows from Lemma 8 that $x(0, S) \leq 2$. If $x(0, S) = 0$ or $x(0, S) = 1$, it follows from $x(0, S) + 3x(S) + 2x(v, S) = 3|S|$ that $x(S) = |S| - \frac{4}{3}$ and $x(S) = |S| - \frac{5}{3}$, which violates integrality. If $x(0, S) = 2$ then $x(S) = |S| - 2$.

We conclude that equality holds if and only if either $x(0, S) = 1$, $x(S) = |S| - 1$ and $x(v, S) = 1$, or $x(0, S) = 2$, $x(S) = |S| - 2$ and $x(v, S) = 2$. □

Lemma 11. *For any $S \subseteq V'$, $v \in V'$ with $v \notin S$ and x a feasible integer solution to the Basic-VRPSD, it holds that $x(0,S) + x(0,v) + 4x(S) + 3x(v,S) \leq 4|S| + 1$, and equality holds if and only if $x(0,S) = 1$, $x(0,v) = 1$, $x(S) = |S| - 1$ and $x(v,S) = 1$.*

Proof. By the subtour elimination constraint for S, Lemma 10, and the degree constraint for v, it follows that $x(0,S) + x(0,v) + 4x(S) + 3x(v,S) = x(S) + (x(0,S) + 3x(S) + 2x(v,S)) + (x(0,v) + x(v,S)) \leq 4|S| + 1$, and equality holds if and only if $x(S) = |S| - 1$, $x(0,S) + 3x(S) + 2x(v,S) = 3|S|$ and $x(0,v) + x(v,S) = 2$. By Lemma 10, $x(0,S) + 3x(S) + 2x(v,S) = 3|S|$ and $x(S) = |S| - 1$ if and only if $x(0,S) = 1$, $x(S) = |S| - 1$ and $x(v,S) = 1$. Furthermore, $x(0,v) + x(v,S) = 2$ and $x(v,S) = 1$ if and only if $x(0,v) = 1$ and $x(v,S) = 1$. □

Lemma 12. *For any $S \subseteq V'$, $u, v \in V'$, such that $u, v \notin S$ and $u \neq v$, and x a feasible integer solution to the Basic-VRPSD, it holds that $x(u,S) + x(S) + x(v,S) \leq |S| + 1$, and equality holds if and only if $(x(u,S), x(S), x(v,S)) \in \{(1, |S| - 1, 1), (2, |S| - 2, 1), (1, |S| - 2, 2), (2, |S| - 3, 2)\}$.*

Proof. From the subtour elimination constraints for $S \cup \{u, v\}$, it follows that $x(u,S) + x(S) + x(v,S) \leq x(S \cup \{u, v\}) \leq |S| + 1$. If $x(u,S) + x(S) + x(v,S) = |S| + 1$ and $x(u,S) = 0$, it follows that $x(S) = |S| - 1$ and $x(v,S) = 2$. However, this violates the subtour elimination constraint for $S \cup \{v\}$ and we conclude that $x(u,S)$ cannot be 0. Similarly, $x(v,S)$ cannot be 0. By the degree constraints of u and v, it further follows that $x(u,S)$ and $x(v,S)$ can only take the values 1 or 2. As a result, equality holds if and only if $(x(u,S), x(S), x(v,S)) \in \{(1, |S| - 1, 1), (2, |S| - 2, 1), (1, |S| - 2, 2), (2, |S| - 3, 2)\}$. □

Lemma 13. *For any $S \subseteq V'$, $u, v \in V'$, such that $u, v \notin S$ and $u \neq v$, and x a feasible integer solution to the Basic-VRPSD, it holds that $x(0,u) + x(0,v) + 2x(S) + 2x(u,S) + 2x(v,S) \leq 2|S| + 4$, and equality holds if and only if $x(0,u) = 1$, $x(0,v) = 1$, $x(S) = |S| - 1$, $x(u,S) = 1$ and $x(v,S) = 1$.*

Proof. From the degree constraints for u and v, Lemma 12, and the subtour elimination constraint for S, it follows that $x(0,u) + x(0,v) + 2x(S) + 2x(u,S) + 2x(v,S) = (x(0,u) + x(u,S)) + (x(0,v) + x(v,S)) + (x(u,S) + x(S) + x(v,S)) + x(S) \leq 2|S| + 4$, and equality holds if and only if $x(0,u) + x(u,S) = 2$, $x(0,v) + x(v,S) = 2$, $x(u,S) + x(S) + x(v,S) = |S| + 1$ and $x(S) = |S| - 1$. Observe from Lemma 12, that if $x(u,S) + x(S) + x(v,S) = |S| + 1$ and $x(S) = |S| - 1$ if and only if $x(u,S) = x(v,S) = 1$. If additionally $x(0,u) + x(u,S) = 2$ and $x(0,v) + x(v,S) = 2$, then $x(0,u) = x(0,v) = 1$. □

Lemma 14. *For any $S \subseteq V'$, $u, v \in V'$, such that $u, v \notin S$ and $u \neq v$, and x a feasible integer solution to the Basic-VRPSD, it holds that $x(0,u) + 2x(u,S) + 2x(S) + x(v,S) \leq 2|S| + 2$, and equality holds if and only if $x(0,u) = 1$, $x(S) = |S| - 1$, $x(u,S) = 1$ and $x(v,S) = 1$.*

Proof. From the degree constraint for u, Lemma 12, and the subtour elimination constraint for S, it follows that $x(0,u) + 2x(u,S) + 2x(S) + x(v,S) \leq (x(0,u) + x(u,S)) + (x(u,S) + x(S) + x(v,S)) + x(S) \leq 2|S| + 2$, and equality holds if and only if $x(0,u) + x(u,S) = 2$, $x(u,S) + x(S) + x(v,S) = |S| + 1$ and $x(S) = |S| - 1$. From Lemma 12 it follows that if $x(u,S) + x(S) + x(v,S) = |S| + 1$ and $x(S) = |S| - 1$ then $x(u,S) = x(v,S) = 1$. If additionally $x(0,u) + x(u,S) = 2$ then $x(0,u) = 1$. □

Theorem 7. *The function $W_h(x)$ is a partial route activation function of h.*

Proof. Denote the partial route by $h = (U_0, \ldots, U_b)$. Below, we distinguish between the different cases of b. In each case, we indicate why $W_h(x) \leq 1$, and provide the necessary and sufficient conditions such that $W_h(x) = 1$. Because both x and the coefficients of $W_h(x)$ are integer, it follows that $W_h(x) \leq 0$ otherwise, which we do not repeat in each case for the sake of conciseness. Moreover, we shall repeatedly apply Lemma 7, but not specifically mention the condition $x(U) = |U| - 1$ for each singleton set $U \in h$, since it satisfied by default.

Case $b = 2$:
Let $h = (\{0\}, U, \{0\})$, then $W_h(x) = x(0,U) + 3x(U) - 3|U| + 2$. By Lemma 9 it holds that $W_h(x) \leq 1$. In particular $W_h(x) = 1$ if and only if $x(0,U) = 2$ and $x(U) = |U| - 1$. By Lemma 6, it follows that $W_h(x) = 1$ if and only if x corresponds with a solution which includes a route that adheres to h.
Case $b = 3$:
Let $h = (\{0\}, U_1, U_2, \{0\})$. A property of a partial route is that $|U_1| = 1$ or $|U_2| = 1$ (or both). If $|U_2| = 1$, say $U_2 = \{v\}$, then $W_h(x) = 4x(U_1) + x(0,U_1) + 3x(v,U_1) + x(0,v) - 4|U_1|$. By Lemma 11, it holds that $W_h(x) \leq 1$. In particular $W_h(x) = 1$ if and only if $x(U_1) = |U_1| - 1$ and $x(0,U_1) = x(v,U_1) = x(0,v) = 1$. By Lemma 7, it follows that $W_h(x) = 1$ if and only if x corresponds with a solution which includes a route that adheres to h. The case where $|U_1| = 1$ is identical.
Case $b = 4$:
Let $h = (\{0\}, U_1, U_2, U_3, \{0\})$. We distinguish two cases, $|U_2| = 1$, and $|U_2| > 1$. If $|U_2| = 1$, say $U_2 = \{v\}$, then $W_h(x) = 3x(U_1) + 3x(U_3) + x(0,U_1) + 2x(v,U_1) + 2x(v,U_3) + x(0,U_3) - 3|U_1| - 3|U_3| + 1$. We decompose $W_h(x)$ in three parts, indicated by parenthesis, as follows: $W_h(x) = (x(0,U_1) + 3x(U_1) + 2x(v,U_1)) + (3x(U_3) + 2x(v,U_3) + x(0,U_3)) - (3|U_1| + 3|U_3| - 1)$. By applying Lemma 10 to the first two parts, we find that $W_h(x) \leq 1$. Observe that by the degree constraint for v it cannot simultaneously hold that $x(v,U_1) = 2$ and $x(v,U_2) = 2$. Therefore, it further follows from Lemma 10 that $W_h(x) = 1$ if and only if $x(U_1) = |U_1| - 1$, $x(U_3) = |U_3| - 1$, and $x(0,U_1) = x(v,U_1) = x(v,U_3) = x(0,U_3) = 1$. By Lemma 7, it follows for that when $|U_2| = 1$ then $W_h(x) = 1$ if and only if x corresponds with a solution which includes a route that adheres to h.

Next, suppose $|U_2| > 1$. As a property of partial routes, it must hold that $|U_1| = |U_3| = 1$, say $U_1 = \{u\}$ and $U_3 = \{v\}$. In this case $W_h(x) = 2x(U_2) + x(0,u) + 2x(u,U_2) + 2x(v,U_2) + x(0,v) - 2|U_2| - 3$. By Lemma 13, it holds that $W_h(x) \leq 1$. In particular $W_h(x) = 1$ if and only if $x(U_2) = |U_2| - 1$ and $x(0,u) = x(u,U_2) = x(v,U_2) = x(0,v) = 1$. By Lemma 7, it follows that $W_h(x) = 1$ if and only if x corresponds with a solution which includes a route that adheres to h.
Case $b \geq 5$:
The function $W_h(x)$ can be constructed by the following procedure. Initialize a function as $-3|U_1| - 2|U_2| - \sum_{k=3}^{b-3} |U_k| - 2|U_{b-2}| - 3|U_{b-1}| + 4$, and proceed as follows

- If $|U_2| > 1$, denote $U_1 = \{u_1\}$ and $U_3 = \{u_3\}$, and add the part $x_{(0,u_1)} + 2x(u_1,U_2) + 2x(U_2) + x(u_3,U_2)$. Note that Lemma 14 applies to this part.
- Otherwise denote $U_2 = \{u_2\}$ and add the part $x(0,U_1) + 3x(U_1) + 2x(U_1,u_2)$. Note that Lemma 10 applies to this part.
- If $|U_{b-2}| > 1$, denote $U_{b-1} = \{u_{b-1}\}$ and $U_{b-3} = \{u_{b-3}\}$, and add the part $x_{(0,u_{b-1})} + 2x(u_{b-1},U_{b-2}) + 2x(U_{b-2}) + x(u_{b-3},U_{b-2})$. Note that Lemma 10 applies to this part.
- Otherwise denote $U_{b-2} = \{u_{b-2}\}$, and add the part $x(0,U_{b-1}) + 3x(U_{b-1}) + 2x(U_{b-1},u_{b-2})$. Note that Lemma 10 applies to this part.
- For all $k \in \{3,\ldots,b-3\}$, if $|U_k| > 1$ denote $U_{k-1} = \{u_{k-1}\}$ and $U_{k+1} = \{u_{k+1}\}$, and add the part $x(u_{k-1},U_k) + x(U_k) + x(u_{k+1},U_k)$. Note that Lemma 12 applies to this part.
- For all $k \in \{2,\ldots,b-3\}$ such that $|U_k| = |U_{k+1}| = 1$, denote $U_k = \{u_k\}$ and $U_{k+1} = \{u_{k+1}\}$, and add the part $x(u_k,u_{k+1})$.

By applying the indicated lemma for each part, we can conclude that $W_h(x) \leq 1$, and $W_h(x) = 1$ if and only if $x(U_k) = |U_k| - 1$ and $x(U_k,U_{k+1}) = 1$ for all $k \in \{0,\ldots,b-1\}$. Thus, $W_h(x)$ is maximal if and only if x adheres to h. By Lemma 7, it follows that $W_h(x) = 1$ if and only if x corresponds with a solution which includes a route that adheres to h. □

G Improved partial route activation function

Theorem 8. *For any partial route $h = (U_0, \dots, U_b)$, with $b \geq 4$, it holds that $W_h(x) \geq W_h^J(x)$ for all solutions x to the continuous relaxation of the Basic-VRPSD.*

Proof. First, let us prove that

$$\sum_{e \in \delta(U_{k-1}, U_k)} x_e + \sum_{e \in E(U_k)} x_e + \sum_{e \in \delta(U_k, U_{k+1})} x_e \leq |U_k| + 1, \forall k \in \{2, \dots, b-2\}. \tag{30}$$

If $|U_k| = 1$, then $U_k = \{u\}$ for some $u \in V'$. This means $|U_k| + 1 = 2 = \sum_{e \in \delta(u)} x_e \geq \sum_{e \in \delta(U_{k-1}, U_k)} x_e + \sum_{e \in E(U_k)} x_e + \sum_{e \in \delta(U_k, U_{k+1})} x_e$, where we use (2). Next, if $|U_k| > 1$, then $|U_{k-1}| = |U_{k+1}| = 1$ and $\sum_{e \in \delta(U_{k-1}, U_k)} x_e + \sum_{e \in E(U_k)} x_e + \sum_{e \in \delta(U_k, U_{k+1})} x_e \leq \sum_{e \in E(U_{k-1} \cup U_k \cup U_{k+1})} x_e \leq |U_{k-1} \cup U_k \cup U_{k+1}| - 1 = |U_k| + 1$ by (3), thus proving (30).

Now, let us consider $W_h(x) - W_h^J(x)$ for $b = 4$. We can write this difference as

$$\begin{aligned}
W_h(x) - W_h^J(x) &= -\left(\sum_{e \in E(U_2)} x_e - (|U_2| - 1)\right) - \left(\sum_{e \in \delta(U_1, U_2)} x_e - 1\right) - \left(\sum_{e \in \delta(U_2, U_3)} x_e - 1\right) \\
&= |U_2| + 1 - \left(\sum_{e \in \delta(U_1, U_2)} x_e + \sum_{e \in E(U_2)} x_e + \sum_{e \in \delta(U_2, U_3)} x_e\right) \\
&\geq 0,
\end{aligned}$$

where the equality follows immediately from (30). For $b \geq 5$, we have

$$\begin{aligned}
W_h(x) - W_h^J(x) = &-\left(\sum_{e \in E(U_2)} x_e - (|U_2| - 1)\right) - 2\sum_{k=3}^{b-3}\left(\sum_{e \in E(U_k)} x_e - (|U_k| - 1)\right) \\
&- \left(\sum_{e \in E(U_{b-2})} x_e - (|U_{b-2}| - 1)\right) - \left(\sum_{e \in \delta(U_1, U_2)} x_e - 1\right) \\
&- 2\sum_{k=2}^{b-3}\left(\sum_{e \in \delta(U_k, U_{k+1})} x_e - 1\right) - \left(\sum_{e \in \delta(U_{b-2}, U_{b-1})} x_e - 1\right) \\
= &\sum_{k=2}^{b-2}\left(|U_k| + 1 - \sum_{e \in \delta(U_{k-1}, U_k)} x_e + \sum_{e \in E(U_k)} x_e + \sum_{e \in \delta(U_k, U_{k+1})} x_e\right) \\
\geq &\ 0,
\end{aligned}$$

where the inequality follows from (30). Thus, we have that $W_h(x) \geq W_h^J(x)$ for all solutions x to the continuous relaxation of the Basic-VRPSD. □

H k-route activation function

For $S \subset V'$, let $m(S)$ be a lower bound on the number of routes required to visit all customers in S and consider an integer $k \geq m(S)$. We repeat the function found in (22).

$$\begin{aligned}
W_{(S,k)}(x) = (k - m(S) + 1) &\sum_{e \in \delta(\{0\}, S)} x_e + \\
&(2(k - m(S)) + 3) \sum_{e \in E(S)} x_e - (2(k - m(S)) + 3)|S| + k + 1.
\end{aligned}$$

Theorem 9. *The function $W_{(S,k)}(x)$ is a k-route activation function of S.*

Proof. Suppose that x corresponds to a solution in which all customers in S exclusively appear on m' different routes. Then, $\sum_{e\in\delta(\{0\},S)} x_e = 2m'$ and $\sum_{e\in E(S)} x_e = |S| - m'$ and it follows that $W_{(S,k)}(x) = k - m' + 1$. Observe that it is nonpositive if $m' > k$.

Now suppose that x corresponds to a solution in which all customers in S appear on m' different routes, but at least one other customer which is not included in S appears on at least one of these routes. It follows that $\sum_{e\in\delta(\{0\},S)} x_e \leq 2m'$ and from the subtour elimination constraints (3) that $\sum_{e\in E(S)} x_e \leq |S| - m'$. Suppose $\sum_{e\in E(S)} x_e \leq |S| - m' - 1$, then $W_{(S,k)}(x) \leq 2m(S) - k - m' - 2 \leq 0$, where the last inequality follows from $m(S) \leq k$ and $m(S) \leq m'$. Otherwise $\sum_{e\in E(S)} x_e = |S| - m' - 1$. In this case, because at least one other customer which is not included in S must appear on at least one of these routes, it must hold that $\sum_{e\in\delta(\{0\},S)} x_e \leq 2m' - 1$. It follows that $W_{(S,k)}(x) \leq m(S) - m \leq 0$. □