Exact Algorithms for the Chance-Constrained Vehicle Routing Problem*

Thai Dinh[1], Ricardo Fukasawa[2], and James Luedtke[1]

[1]Department of Industrial & Systems Engineering and Wisconsin Institute for Discovery , University of Wisconsin-Madison, Madison, WI 53706 ,
{tndinh,jim.luedtke}@wisc.edu
[2]Department of Combinatorics & Optimization, , University of Waterloo, Waterloo, ON, Canada N2L 3G1, , rfukasawa@uwaterloo.ca

May 19, 2016

Abstract

We study the chance-constrained vehicle routing problem (CCVRP), a version of the vehicle routing problem (VRP) with stochastic demands, where a limit is imposed on the probability that each vehicle's capacity is exceeded. A distinguishing feature of our proposed methodologies is that they allow correlation between random demands, whereas nearly all existing exact methods for the VRP with stochastic demands require independent demands. We first study an edge-based formulation for the CCVRP, in particular addressing the challenge of how to determine a lower bound on the number of vehicles required to serve a subset of customers. We then investigate the use of a branch-and-cut-and-price (BCP) algorithm. While BCP algorithms have been considered the state of the art in solving the deterministic VRP, few attempts have been made to extend this framework to the VRP with stochastic demands. In contrast to the deterministic VRP, we find that the pricing problem for the CCVRP problem is strongly $\mathcal{NP}$-hard, even when the routes being priced are allowed to have cycles. We therefore propose a further relaxation of the routes that enables pricing via dynamic programming. Numerical results indicate that the proposed methods can solve instances of CCVRP having up to 55 vertices.

1 Introduction

The deterministic vehicle routing problem (VRP) [12] is the problem of finding routes for a fleet of identical, fixed capacity vehicles that collect known amounts of goods from customers. When demands of customers are random variables, the problem is referred to as the vehicle routing problem with stochastic demands (VRPSD). In an optimization model for the VRPSD, one must determine how to handle the possibility that the demands on a planned route might exceed the capacity of a vehicle. One approach, taken, e.g., in [4, 15, 30, 32, 35], is to consider a recourse model, in which a recourse action must be taken if a vehicle's capacity is exceeded. This leads to a two-stage stochastic programming formulation, in which routes are determined in advance

*Fukasawa was supported by NSERC Discovery Grant RGPIN-05623. Luedtke was supported by NSF grants CMMI-0952907 and CMMI-1130266, and ONR award N00014-15-1-2268.

of knowing the random demands, and then, when the routes are implemented and demands are observed, recourse actions are taken if a vehicle's capacity is exceeded. The objective is to minimize the expected travel cost, including travel taken in the recourse stage. In order to make the evaluation of the expected recourse costs tractable, restrictive assumptions are usually placed on the form of the recourse taken (e.g., that it consists of a trip to/from the depot) and on the random demands. In particular, nearly all existing work assumes the random demands are independent of each other.

We study an alternative model, the chance-constrained VRP (CCVRP), which does not explicitly model the recourse actions to be taken when a vehicle's capacity is exceeded, and instead requires that such an event happens with low probability. This type of model leads to operational benefits like more consistent service and less need for complex recourse actions to be taken. The first attempt to solve the CCVRP was proposed in [32], where conditions are derived under which CCVRP can be reduced to a deterministic VRP. These conditions are restrictive, as they require customer demands to be independent and have identical coefficients of variation. In [24], the first exact solution technique for CCVRP was proposed using a branch-and-cut framework, but their implementation requires random demands to be independent and normally distributed. More recently, in [3] a formulation is proposed for a generalization of the CCVRP where service is required on both customers and the roads connecting them. This method is limited to (not necessarily independent) joint normally distributed customer demands, and the formulation has $O(n^2K)$ variables, where n is the number of customers and K is the number of vehicles, and exhibits symmetry in the case of identical vehicles. The largest instances reported to be solved optimally have about 10 customers, although promising heuristic methods are also proposed. In [20] the *robust* capacitated VRP is studied, where the constraint is that the capacity of the vehicles must be satisfied by all possible realizations of demands in an uncertainty set. When demands are joint normally distributed, the chance constraint is equivalent to a robust constraint with ellipsoidal uncertainty set (e.g., [5]), and thus the formulations used in [20] can be used to solve the CCVRP under the assumption of joint normal demands. Indeed, they mention this briefly in their work and give a more detailed discussion to it in the e-companion. Unfortunately, even though their approach can solve some instances of the robust capacitated VRP problem with around 100 customers under specific *polyhedral* uncertainty sets, the approach is not as successful for instances that were transformed from the CCVRP and the results in the e-companion only mention solving such instances with at most 23 customers.

In terms of methodology, the current best known algorithms for solving the deterministic VRP are based on a Dantzig-Wolfe reformulation, strengthened by valid inequalities, solved using a branch-and-cut-and-price (BCP) algorithm [1, 2, 11, 18, 28]. On the other hand, very little work has attempted to apply the BCP framework to solve a recourse-based or chance-constrained VRPSD model, or a robust VRP model. Most of the exact solution techniques for the recourse-based VRPSD models rely on variants of the integer L-Shaped method proposed by [26] or the branch-and-price algorithm proposed by [8], while most of the attempts to solve the CCVRP rely on variants of the branch-and-cut algorithms proposed in [24, 25]. To the best of our knowledge, the only work that considers solving a VRPSD model using BCP has been proposed by [19], but is again restricted to the assumption that random demands are independent.

In this work, we present branch-and-cut and BCP methods for the CCVRP which do not require the customer demands to be independent, and are able to solve to optimality, or near-optimality, instances with more than 50 customers. The only assumption we require on the customer demands is that we can compute a quantile of the random variable defined by the sum of customer demands in any subset of customers. This assumption holds for customer demands having joint normal distribution and for a scenario model of customer demands. We note that, using sample average approximation, the scenario model can be used to approximate a problem in which customer demands follow any distribution from which samples can be taken [27]. In the case of joint normally distributed random demands, our results complement those of [20] by demonstrating that the use of the edge-based formulation with capacity inequalities is viable for the robust capacitated VRP with ellipsoidal uncertainty set, and also by introducing a BCP approach for that problem class.

We begin in Section 2 by presenting the edge-based formulation of [24] and derive strong and computationally tractable bounds on the number of vehicles required to serve a subset of customers and remain chance constraint feasible, leading to improved capacity inequalities. This allows us to extend the formulation of [24] to more general cases. In Section 3, we explore the use of BCP for solving the CCVRP. We find that a direct extension of the pricing routine used in BCP for the deterministic VRP is challenging since the associated pricing problem is strongly $\mathcal{NP}$-hard (as opposed to pseudopolynomially solvable in the deterministic case) even when the distribution of random demands is a finite scenario model or consists of independent normal random variables. We thus propose a relaxed pricing scheme to overcome this challenge. In Section 4, we discuss how the proposed methodology can be adapted to solve a *distributionally robust* version of the problem, in which the distribution of the customer demands is assumed to be unknown, and the goal is to obtain routes that are feasible to the chance constraint for all distributions within a given set. Finally, in Section 5, we present results of a computational study which demonstrate that the proposed methods can solve instances of CCVRP with up to 55 customers. We find that the improved bounds we obtain on the number of vehicles required to serve a subset of customers are critical for solving these instances, and we also find that the BCP approach is beneficial for the largest of the test instances. We also empirically compare the solutions obtained with the CCVRP model to those obtained with a recourse-based model of the VRPSD, and find that the CCVRP solutions provide high quality solutions to the recourse-based model, whereas the reverse is often not true.

An extended abstract of this paper appeared in [14]. This full version of the paper contains all proofs that were omitted in the extended abstract, presents an additional strong $\mathcal{NP}$-hardness proof for pricing under independent normal demands, and discusses the extension of the proposed methodology to solve problems with a distributionally robust chance constraint. We also provide additional implementation details, especially about a heuristic that we used to obtain initial solutions to the CCVRP, which may be independently interesting. Finally, in the special case where demands are normally distributed, we take advantage of the results in [20] to obtain improved bounds (as compared to those in [14]) on the number of trucks required to serve a set of customers.

2 Problem definition and an edge-based formulation

Let $G = (V, E)$ be an undirected graph with vertices $V = \{0, 1, ..., n\}$. Vertex 0 represents the depot and the vertices $V_+ = \{1, ..., n\}$ represents the customers. Each customer $i \in V_+$ has a random demand D_i. The set of demands D is a random vector defined in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The expected value and variance of demand for customer $i \in V_+$ are denoted by d_i and σ_i^2, respectively. The length of edge $e \in E$ is denoted by $\ell_e \geq 0$. There are K available vehicles and each vehicle has a capacity of b. A route is a simple cycle C going through 0 (or an edge $0v$ twice, representing the route $0 - v - 0$ for $v \in V_+$). We say a route serves S if $V(C) \setminus \{0\} = S$. A *chance-constraint feasible route* is a route for which the set of customers $S \subseteq V_+$ that it serves satisfies $\mathbb{P}\{D(S) \leq b\} \geq 1 - \epsilon$, where $\epsilon \in (0, 1)$ is a given, typically small, parameter. Above, and throughout the rest of the paper, we use the following notation: given values w_t for a ground set T, for any $Q \subseteq T$ we denote $w(Q) := \sum_{t \in Q} w_t$.

The objective is to find a minimum length set of K chance constraint feasible routes such that every customer is visited exactly once.

2.1 Edge-based formulation

Let x_e represent the number of times edge e is used in a solution. For a subset of customers $S \subseteq V_+$, we let $\delta(S)$ be the cut-set defined by S and let $r_\epsilon(S)$ be the minimum number of vehicles needed to serve S with chance-constraint feasible routes. We call $r_\epsilon(S)$ the *minimum vehicle requirements*. The edge-based formulation is then [24]:

$$\begin{aligned}
\min_x \quad & \sum_{e \in E} \ell_e x_e && && \text{(1a)}\\
\text{s.t.} \quad & \sum_{e \in \delta(\{i\})} x_e = 2 && i \in V_+ && \text{(1b)}\\
& \sum_{e \in \delta(\{0\})} x_e = 2K && && \text{(1c)}\\
& \sum_{e \in \delta(S)} x_e \geq 2 r_\epsilon(S) && S \subseteq V_+ && \text{(1d)}\\
& x_e \leq 1 && e \in E \setminus \delta(\{0\}) && \text{(1e)}\\
& x_e \in \mathbb{Z}_+ && e \in E. && \text{(1f)}
\end{aligned}$$

Constraints (1b) require that each customer is visited exactly once by some vehicle, whereas (1c) states that K vehicles must leave and enter at the depot. Constraints (1d) are the capacity inequalities, which enforce that enough vehicles are assigned to any subset of customers.

A similar model has been used for the deterministic VRP with customer demands $d_i, i \in V_+$ where $r_\epsilon(S)$ in (1d) is replaced with the minimum number of vehicles $r_0(S)$ required to serve the customers in the set S. Calculating this quantity exactly requires solving the strongly $\mathcal{NP}$-hard bin-packing problem. Fortunately, for the deterministic VRP, the easily computed lower bound $k(S) := \lceil d(S)/b \rceil$ yields a valid formulation, and the resulting cuts have been shown empirically to be effective. A key challenge for the CCVRP is to determine how to compute a lower bound for $r_\epsilon(S)$ that is at least sufficient to provide a valid formulation, and that is as close to $r_\epsilon(S)$ as possible in order to yield strong inequalities. When the formulation (1) was studied in [24], they proposed to use the value

$$\tau_\epsilon^I(S) = \left\lceil \left(d(S) + \Phi^{-1}(1-\epsilon)\sqrt{\sigma^2(S)} \right) / b \right\rceil, \tag{2}$$

as an approximation of $r_\epsilon(S)$. This is a valid lower bound when demands are independent normal, but is not necessarily valid in other cases. In (2), Φ^{-1} is the inverse of the cumulative density function of the standard normal distribution. The next subsection is devoted to deriving valid lower bounds on $r_\epsilon(S)$ that are strong but cheap to compute.

2.2 Vehicle requirements in the capacity inequalities

We now discuss how to obtain $k_\epsilon(S) \leq r_\epsilon(S)$, such that formulation (1) is still valid for the CCVRP if we replace $r_\epsilon(S)$ with $k_\epsilon(S)$. We refer to such lower bounds on the minimum vehicle requirements as *valid* lower bounds. We begin with a simple valid lower bound, $k_\epsilon(S)$, defined as:

$$k_\epsilon(S) = \begin{cases} 1, & \text{if } \mathbb{P}\{D(S) \leq b\} \geq 1-\epsilon \\ 2, & \text{otherwise.} \end{cases}$$

which just states that at least two vehicles are needed to serve the set of customers S if the probability that the sum of customer demands in the set S exceeding a single vehicle's capacity is too high.

Theorem 1. *$k_\epsilon(S)$ is a valid lower bound for $r_\epsilon(S)$.*

Proof. Let F^r be the set of $x \in \{0,1\}^E$ that satisfy (1), and F^k be the set of $x \in \{0,1\}^E$ that satisfy (1) with $r_\epsilon(S)$ in (1d) replaced by $k_\epsilon(S)$. It is immediate that $r_\epsilon(S) \geq k_\epsilon(S)$, so $F^r \subseteq F^k$. Hence, it remains to show $F^r \supseteq F^k$.

Suppose, for the sake of contradiction, that there exists an integral solution $\hat{x} \in F^k \setminus F^r$. As these sets only differ in the constraints (1d), this implies there exists $S' \subseteq V_+$ such that $\sum_{e\in\delta(S')} \hat{x}_e < 2r_\epsilon(S')$. Next, note that since $k_\epsilon(S) \geq 1$, constraints (1d) using $k_\epsilon(S)$ in place of $r_\epsilon(S)$, together with (1b) and (1c) imply that $\hat{x}$ defines a set of K tours, each of which begins and ends at the depot, and which visit each node exactly once. Let $P_1, P_2, \ldots, P_q$ be the set of disjoint simple paths defined by $\hat{x}$ which enter the set S', visit one or more nodes in S', then depart it. (Note that it is possible that $q > K$ as a tour may enter and leave the set S' more than once.) Let S_t be the set of nodes in S' visited by path P_t, $t = 1, 2, ..., q$, and observe that these sets form a partition of S', and $\sum_{e\in\delta(S_t)} \hat{x}_e = 2$ for each t. Now note that any edge in $\delta(S')$ is an edge in $\delta(S_t)$ for some t and, likewise, any edge in $\delta(S_t)$ for which $\hat{x}_e > 0$ must be in $\delta(S')$. So it holds that $\sum_{e\in\delta(S')} \hat{x}_e = \sum_{t=1}^{q} \sum_{e\in\delta(S_t)} \hat{x}_e = 2q$. Thus, $2r_\epsilon(S') > \sum_{e\in\delta(S')} \hat{x}_e = 2q$ and hence $q < r_\epsilon(S')$. Therefore, there exists $j \in \{1, 2, ..., q\}$ such that $\mathbb{P}\{D(S_j) \leq b\} < 1-\epsilon$, because otherwise, we can use q vehicles to pick up demands from S and the minimum vehicle requirement $r_\epsilon(S)$ would be no larger than q. However, since $\mathbb{P}\{D(S_j) \leq b\} < 1-\epsilon$, $k_\epsilon(S_j) = 2$ by definition and $\sum_{e\in\delta(S_j)} \hat{x}_e \geq 2k_\epsilon(S_j) = 4$, which is a contradiction. □ □

Note that computing $k_\epsilon(S)$ only requires computing $\mathbb{P}\{D(S) \leq b\}$. Moreover, any value that is between $k_\epsilon(S)$ and $r_\epsilon(S)$ yields a valid lower bound. To improve on $k_\epsilon(S)$, given a random variable X, we use $Q_p(X)$ to denote the p-th quantile of X, that is, $Q_p(X) = \inf\{\alpha : \mathbb{P}\{X \leq \alpha\} \geq p\}$. [1] An improved lower bound on $r_\epsilon(S)$ can be obtained using the following lemma.

Lemma 1. *Let $p := 1 - r_\epsilon(S)\epsilon$. For all $S \subseteq V_+$, we have $r_\epsilon(S) \geq \lceil Q_p(D(S))/b \rceil$.*

Proof. First, note that if $r_\epsilon(S)\epsilon \geq 1$, then $Q_{1-r_\epsilon(S)\epsilon} = -\infty$ and the result is trivial, so we may assume $r_\epsilon(S)\epsilon < 1$. Since $r_\epsilon(S)$ is an integer, it suffices to show that $r_\epsilon(S)b \geq Q_p(D(S))$, which is equivalent to showing $\mathbb{P}\{D(S) \leq r_\epsilon(S)b\} \geq 1 - r_\epsilon(S)\epsilon$.

By definition of $r_\epsilon(S)$, there is an assignment of the customers in S to $r_\epsilon(S)$ vehicles such that the probability of each vehicle failing its capacity constraint is at most ϵ. Let S_k be the customers in S assigned to vehicle $k \in \{1, 2, ..., r_\epsilon(S)\}$. It follows that $\mathbb{P}\{D(S_k) > b\} \leq \epsilon$ for $k = 1, 2, ..., r_\epsilon(S)$ and that $\sum_{k=1}^{r_\epsilon(S)} D(S_k) = D(S)$. Let E_k be the event that $D(S_k) > b$ for $k = 1, 2, ..., r_\epsilon(S)$. For an event E, we denote its complement E^c. Let A be the event that $D(S) \leq r_\epsilon(S)b$. Thus, we need to show that $P(A) \geq 1 - r_\epsilon(S)\epsilon$, given that $P(E_k) \leq \epsilon$ for $k = 1, 2, ..., r_\epsilon(S)$.

We next derive a connection between event A and events E_k for $k = 1, 2, ..., r_\epsilon(S)$. Since $\sum_{k=1}^{r_\epsilon(S)} D(S_k) = D(S)$, it follows that $D(S_k) \leq b$ for *all* $k = 1, 2, ..., r_\epsilon(S)$ implies that $D(S) \leq$

[1] Note that $\mathbb{P}\{D(S) \leq b\} \geq 1-\epsilon \iff Q_{1-\epsilon}(D(S)) \leq b$

$r_\epsilon(S)b$. In other words, $\bigcap_{k=1}^{r_\epsilon(S)} E_k^c \implies A$, from which it follows that $\left(\bigcup_{k=1}^{r_\epsilon(S)} E_k\right)^c \implies A$. Therefore,

$$\mathbb{P}(A) \geq \mathbb{P}\Big(\big(\bigcup_{k=1}^{r_\epsilon(S)} E_k\big)^c\Big) = 1 - \mathbb{P}\big(\bigcup_{k=1}^{r_\epsilon(S)} E_k\big) \geq 1 - \sum_{k=1}^{r_\epsilon(S)} P(E_k) \geq 1 - r_\epsilon(S)\epsilon$$

where the second inequality follows from the union bound and the final inequality follows since $\mathbb{P}(E_k) \leq \epsilon$ for $k = 1, 2, ..., r_\epsilon(S)$. □ □

The lower bound given by Lemma 1 cannot be directly used because its calculation uses the value of $r_\epsilon(S)$ itself. However, one can use Lemma 1 to derive a computable lower bound. For $S \subseteq V_+$, define $a(S, 1) = 1$ and for $k = 2, \dots, K$ define

$$a(S, k) := \min\Big\{k, \lceil Q_{1-(k-1)\epsilon}(D(S))/b \rceil\Big\}.$$

Using $a(S, k)$ we obtain the following valid lower bound:

$$\rho_\epsilon(S) = \max\big\{a(S, k) : k = 1, 2, ..., K\big\}.$$

Theorem 2. *For all $S \subseteq V_+$, $k_\epsilon(S) \leq \rho_\epsilon(S) \leq r_\epsilon(S)$.*

Proof. Given $S \subseteq V_+$, we first argue that $k_\epsilon(S) \leq \rho_\epsilon(S)$. By definition, $\rho_\epsilon(S) \geq a(S, 2)$. We, next argue that $a(S, 2) \geq k_\epsilon(S)$ by considering two cases. If $\mathbb{P}\{D(S) \leq b\} \geq 1 - \epsilon$, then $k_\epsilon(S) = 1$. Since $a(S, 2)$ is a positive integer, it follows trivially in this case that $a(S, 2) \geq 1 = k_\epsilon(S)$. Thus, suppose $\mathbb{P}\{D(S) \leq b\} < 1 - \epsilon$. It follows that $Q_{1-\epsilon}(D(S)) > b$. Thus, $Q_{1-\epsilon}(D(S))/b > 1$ and $\lceil Q_{1-\epsilon}(D(S))/b \rceil \geq 2$. Hence, $a(S, 2) = \min\{2, \lceil Q_{1-\epsilon}\Big(D(S)\Big)/b \rceil\} \geq 2 = k_\epsilon(S)$, by definition of $k_\epsilon(S)$ because $\mathbb{P}\{D(S) \leq b\} < 1 - \epsilon$.

We now show that for any k, $a(S, k)$ is a lower bound on $r_\epsilon(S)$. Indeed, this is trivially true if $r_\epsilon(S) \geq k$ or if $k = 1$. Otherwise, $r_\epsilon(S) \leq k - 1$, in which case Lemma 1 shows that $r_\epsilon(S) \geq \lceil Q_p(D(S))/b \rceil$, where $p = 1 - r_\epsilon(S)\epsilon$. Then $a(S, k) \leq r_\epsilon(S)$ since $Q_p(D(S))$ is nondecreasing in p. Therefore, the maximum of these lower bounds over $k = 1, \dots, K$ is also a lower bound. □ □

We next discuss how alternative valid lower bounds can be obtained in the special case when the customer random demands are joint normally distributed.

2.3 Joint normal random demands

In this section we assume customer demands follow a joint normal distribution with covariance matrix $\Sigma \succeq 0$. In this case, if $S \subseteq V_+$ denotes the set of customers visited by a route, and y^S is the binary vector having $y_i^S = 1, i \in S$ and $y_i^S = 0, i \in V_+ \setminus S$, then the route is chance-constraint feasible if and only if y^S satisfies the following constraint:

$$d^\top y + \kappa(\epsilon)\sqrt{y^\top \Sigma y} \leq b \tag{3}$$

where $\kappa(\epsilon) := \Phi^{-1}(1 - \epsilon)$. In the robust capacitated VRP studied in [20], a route visiting customer set S is said to be *robust feasible with respect to uncertainty set $\mathcal{A}(\epsilon)$* if y^S satisfies the following robust constraint:

$$\alpha^\top y \leq b \qquad \forall \alpha \in \mathcal{A}(\epsilon), \tag{4}$$

where $\mathcal{A}(\epsilon)$ is a tractable convex set. It is well-known (e.g., [5]) that, provided $\epsilon \leq 0.5$, an individual joint-chance constraint of the form (3) is equivalent to the robust constraint (4) with $\mathcal{A}(\epsilon) = \{q = Uv : \|v\| \leq \kappa(\epsilon)\}$ and $UU^T = \Sigma$.

In [20], the following bound on the number of vehicles required to serve a given subset of customers S was derived for the robust capacitated VRP.

Theorem 3 (Proposition 3 of [20])**.** *Let $S \subseteq V_+$. The number of vehicles required to serve the set of customers S with robust feasible routes with respect to uncertainty set $\mathcal{A}(\epsilon)$ is at least*

$$\left\lceil \max\left\{\alpha^\top y^S : \alpha \in \mathcal{A}(\epsilon)\right\}/b \right\rceil.$$

Therefore, we define

$$\tau^J_\epsilon(S) := \left\lceil \left(d^\top y^S + \kappa(\epsilon)\sqrt{(y^S)^\top \Sigma y^S}\right)/b \right\rceil,$$

and since $d^\top y^S + \kappa(\epsilon)\sqrt{(y^S)^\top \Sigma y^S} = \max\left\{\alpha^\top y^S : \alpha \in \mathcal{A}(\epsilon)\right\}$ it follows from Theorem 3 that, if the demands follow a joint normal distribution with mean vector d and covariance matrix Σ and $\epsilon \leq 0.5$, then

$$\rho_\epsilon(S) \leq \tau^J_\epsilon(S) \leq r_\epsilon(S), \forall S \subseteq V_+.$$

We note that when demands follow an independent normal distribution, then $\tau^J_\epsilon(S) = \tau^I_\epsilon(S)$, so this result generalizes the bound from [24].

3 Dantzig-Wolfe formulation and branch-and-cut-and-price

Set partitioning formulations for routing problems are based on enumerating *elementary routes* or relaxations of them. We start by describing such an approach for the deterministic VRP. For that case, an *elementary route* is a closed walk $v_0, v_1, \ldots, v_k, v_{k+1} = v_0$, for some $k \geq 1$ such that (i) $v_0 = 0$, $v_i \in V_+, \forall i = 1, \ldots, k$ and $v_{i-1}, v_i \in E, \forall i = 1, \ldots, k+1$; (ii) $v_i \neq v_j, \forall 0 < i < j < k+1$ and (iii) $d^\top y \leq b$, where $y_v := \sum_{i=1}^k \mathbb{1}_{\{v=v_i\}}$ is the number of times v appears in the route and d is the vector of deterministic customer demands.

Let $\mathcal{Q}$ be the set of elementary routes and $\lambda_j \in \{0,1\}$ represent if elementary route j is used. Let $q^e_j := \sum_{i=0}^k \mathbb{1}_{\{e=v_i v_{i+1}\}}$, that is, the number of times edge e appears in route j. By using the relationship $x_e = \sum_{j\in\mathcal{Q}} q^e_j \lambda_j$, $\forall e \in E$, we obtain from (1) a set-partitioning based formulation for the deterministic VRP [18]:

$$\min_{\lambda} \quad \sum_{j\in\mathcal{Q}}\sum_{e\in E} \ell_e q^e_j \lambda_j \tag{5a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{Q}}\sum_{e\in\delta(\{i\})} q^e_j \lambda_j = 2 \qquad i \in V_+ \tag{5b}$$

$$\sum_{j\in\mathcal{Q}}\sum_{e\in\delta(\{0\})} q^e_j \lambda_j = 2K \tag{5c}$$

$$\sum_{j\in\mathcal{Q}}\sum_{e\in\delta(S)} q^e_j \lambda_j \geq 2r_0(S) \qquad S \subseteq V_+ \tag{5d}$$

$$\sum_{j\in\mathcal{Q}} q^e_j \lambda_j \leq 1 \qquad e \in E \setminus \delta(\{0\}) \tag{5e}$$

$$\lambda_j \in \{0,1\} \qquad j \in \mathcal{Q}. \tag{5f}$$

In order to use (5) in a BCP approach it must be possible to solve the pricing subproblem of λ variables efficiently. The pricing subproblem consists of finding elementary routes of minimum reduced cost, which is strongly $\mathcal{NP}$-hard. In [18] condition (ii) was relaxed, leading to what is called a q-route [9]. Pricing q-routes is still $\mathcal{NP}$-hard due to the knapsack-type condition (iii), but it can be solved in pseudo-polynomial time [9] if the demands are integer. We note that more complex column-generation schemes have also been proposed (see for instance [2]), strengthening (5) by forbidding some (or all) cycles in q-routes. Our approach can be adapted to those cases, but for simplicity we choose to only present it based on q-routes.

To adapt (5) for the CCVRP, all we need to do is replace $r_0(S)$ in (5d) by $r_\epsilon(S)$ or any of its valid lower bounds derived in Section 2 and consider $\mathcal{Q}$ as the set of *chance-constraint feasible q-routes (CCq-routes)*, where a CCq-route is a closed walk satisfying (i) and replacing condition (iii) by

$$\mathbb{P}\{D^\top y \leq b\} \geq 1 - \epsilon \tag{6}$$

We note that chance-constraint feasible routes are CCq-routes satisfying (ii).

Unfortunately, in contrast to the deterministic VRP, the pricing of CCq-routes is *strongly* $\mathcal{NP}$-hard, even in two different special cases on the distribution of random demands: a scenario model and independent normal.

Theorem 4. *Suppose the distribution of demands is specified by S joint demand scenarios $d^s \in \mathbb{Z}^n_+$, $s = 1, \dots, S$, where $\mathbb{P}\{D = d^s\} = 1/S$ for $s = 1, \dots, S$. Then finding the minimum cost CCq-route is strongly $\mathcal{NP}$-hard.*

Proof. Let $\mathcal{S} := \{1, \dots, S\}$. For any depot-round walk w (i.e., a walk satisfying (i)) and its associated vector $y \in \mathbb{Z}^{V_+}$ representing the number of times v is visited in w, we let $S(w) := \left\{s \in \mathcal{S} : y^\top d^s \leq b\right\}$. Then, (6) is equivalent to $|\mathcal{S} \setminus S(w)| \leq \epsilon S$.

We prove strong $\mathcal{NP}$-hardness by a reduction from the strongly $\mathcal{NP}$-hard Hamiltonian cycle problem. Suppose we have a graph $G = (V_+ \cup \{0\}, E)$ and we want to determine if it has a Hamiltonian cycle.

Construct the following instance of the minimum cost CCq-route problem. The graph is the same, with 0 as the depot. All edge costs are -1. Set $b = 2n - 1$. Let $S = n + 1$ and pick $\epsilon > 0$ such that $\epsilon/2 < \frac{1}{S} < \epsilon$. Now construct one scenario s_v per vertex $v \in V_+$ as follows: all vertices except v have demand 1, whereas vertex v has demand n. Finally, we construct a scenario s' with all vertices having demand b.

Now we show that any depot-round walk w is a CCq-route if and only if it is elementary, that is, each vertex is visited at most once. Suppose w is elementary. Then it satisfies scenarios s_v for every $v \in V_+$. Therefore, $\mathcal{S} \setminus S(w) \subseteq \{s'\}$ and hence w is a CCq-route.

Conversely, suppose that w is a CCq-route. If w consists of just $0, v_1, 0$, then it is clearly elementary. So now let us consider the case when w is is a walk of $k > 1$ vertices. Note that w satisfies scenario s_v if and only if v is visited at most once in w. Also, note that since $k > 1$, w violates scenario s' and hence it cannot violate any other scenario. So w is elementary.

Therefore, finding a minimum cost CCq-route is equivalent to finding a minimum cost elementary route and the graph has a Hamiltonian cycle if and only if there exists an elementary route of cost $-n$. □ □

Theorem 5. *Suppose the random demands are independent and normally distributed random variables, with expected value $d_i \in \mathbb{Z}_+$ and variance $\sigma_i^2 \in \mathbb{Z}_+$ for $i \in V_+$. Then finding the minimum cost CCq-route is strongly $\mathcal{NP}$-hard.*

Proof. For a q-route, let y_i represent the number of times customer i is visited on the route. As it is a q-route, y_i are nonnegative integers (not necessarily $0-1$). Thus, the chance constraint can be represented by:

$$d^\top y + \kappa(\epsilon)\Big(\sum_{i \in V_+} \sigma_i^2 y_i^2\Big)^{1/2} \leq b. \tag{7}$$

Then, the pricing problem is to find a minimum cost CCq-route that satisfies (7). We show this problem is strongly $\mathcal{NP}$-hard, again by reduction from the Hamiltonian cycle problem. Suppose we have a graph $G = (V_+ \cup \{0\}, E)$ and we want to determine if it has a Hamiltonian cycle.

Construct the following instance of the minimum cost CCq-route problem. The graph is the same, with 0 as the depot. All edge costs are -1, $d_i = 0$ and $\sigma_i^2 = 1$ for all $i \in V_+$, and choose ϵ and b such that $b/\kappa(\epsilon) = \sqrt{n}$. We consider the decision version of the problem in which we wish to decide if there exists a CCq-route having cost less than or equal to $-n$. Given this data, the constraint (7) reduces to $\sum_{i \in V_+} y_i^2 \leq n$.

Suppose the graph has a Hamiltonian cycle. This defines a depot-round walk of cost equal to $-n$, and because each node is visited exactly once, it holds that $\sum_{i \in V_+} y_i^2 = n$, and so it defines a CCq-route.

Conversely, suppose there exists a CCq-route having cost less than or equal to $-n$. Let y_i represent the number of times node $i \in V_+$ is visited on the walk, so that $\sum_{i \in V_+} y_i^2 \leq n$ and also $\sum_{i \in V_+} y_i = n$. We claim that this implies $y_i = 1$, $i \in V_+$, and so the CCq-route defines a Hamiltonian cycle. Indeed, we argue that the unique minimum of $\sum_{i \in V_+} y_i^2$ over y satisfying $\sum_{i \in V_+} y_i = n$ and $y \in \mathbb{Z}_+^n$ is attained by setting $y_i = 1$ for all $i \in V_+$, so that any other y satisfying $\sum_{i \in V_+} y_i = n$ violates $\sum_{i \in V_+} y_i^2 \leq n$. Consider any $y \in \mathbb{Z}_+^n$ satisfying $\sum_{i \in V_+} y_i = n$ but not having $y_i = 1$ for all i. There are two nodes, say i and j, with $y_i = 0$ and $y_j \geq 2$. We argue that the solution with $y'_i = 1$ and $y'_j = y_j - 1$ (and all other $y'_k = y_k$) has $\sum_{k \in V_+} (y'_k)^2 < \sum_{k \in V_+} y_k^2$. Indeed, $\sum_{k \in V_+} (y_k^2 - (y'_k)^2) \geq y_j^2 - (y_j - 1)^2 - 1 = 2y_j - 2 > 0$. □ □

Theorem 5 also implies that pricing of q-routes is strongly $\mathcal{NP}$-hard if the customer demands are joint normally distributed. Using the relationship between chance constraints with normal random demands and robust constraints discussed in Section 2.3, this in turn implies that in the robust capacitated VRP model [20], pricing of q-routes that are *robust feasible* with respect to an ellipsoidal uncertainty set is strongly $\mathcal{NP}$-hard. We note that a similar result was shown in [29], however they assume that the uncertainty set is a finite set of integer points in a polyhedron, which can be seen as the case when the uncertainty set is a bounded integral polyhedron.

To overcome the difficulty in pricing CCq-routes, we propose to further relax the capacity constraints defining CCq-routes used in the set partitioning formulation. We present two approaches for doing this: one that applies to any distribution for which we can evaluate (6) and the other that uses distribution-specific arguments when the demands are normally distributed.

3.1 Relaxed pricing

The key advantage of using q-routes instead of elementary routes in (5) is to enable pricing via dynamic programming. The approach is valid since the set of q-routes contains the set of elementary routes and, in any $\{0,1\}$ solution to (5), constraints (5b) ensure that only elementary routes are chosen. We build upon that idea to further relax constraint (6) so that any chance constraint feasible route is still feasible, while making use of the constraints in (5) to ensure that in a $\{0,1\}$ solution, only chance constraint feasible routes are chosen.

Since the original condition (iii) can be handled by dynamic programming, we choose to relax the condition (6) in the definition of a CCq-route to a similar knapsack-type constraint $\pi^\top y \le b_\pi$. To make sure that chance constraint feasible routes are still feasible, we must have that

$$b_\pi \ge b_\pi^* := \max\Big\{\pi^\top y : \mathbb{P}\{D^\top y \le b\} \ge 1-\epsilon, y_i \in \{0,1\}, i \in V_+\Big\}. \tag{8}$$

It is then clear that the following proposition holds.

Proposition 1. *Let y be a binary vector satisfying* (6)*. If* (8) *holds, then y satisfies:*

$$\pi^\top y \le b_\pi. \tag{9}$$

We call *relaxed chance constraint feasible q-routes (rCCq-routes)* the closed walks satisfying conditions (i) in the definition of elementary routes and replacing (iii) in that definition by (9). For any integral and nonnegative choice of the π coefficients, once b_π^* (or an upper bound) is determined, we can proceed with the pricing exactly the same way as is done in the deterministic VRP, using (9) as the knapsack constraint.

In the deterministic VRP, the constraints (5b) impose that, in a $\{0,1\}$ solution to (5) only elementary routes may have their corresponding variable equal to 1. Since each q-route satisfies the capacity constraints, the constraints (5d) are not required for validity of the formulation (5), and so a pure branch-and-price algorithm could be applied. In contrast, in our proposed BCP approach for the CCVRP, even if a $rCCq$-route is an elementary route, it may not be a chance constraint feasible route. The formulation thus requires the constraints (5d), which impose that in a $\{0,1\}$ solution to (5) only chance constraint feasible routes may have their corresponding variable equal to 1.

We next discuss the choice of the coefficients π and calculation of the value b_π^*. Any integer values of π can be used, but it is natural to choose values that correlate with the size of the items, and so we choose to use $\pi_i = d_i$ (these may be scaled and rounded to obtain acceptably small integers). Calculating b_π^* requires solving the chance-constrained knapsack problem (8). This can be a computationally challenging problem, although it only needs to be solved once as a preprocessing step. When the demands assume a discrete distribution having finitely many demand scenarios, the preprocessing problem (8) can be solved using specialized techniques [31]. With joint normal random demands with mean vector d and covariance matrix $\Sigma \succeq 0$, the capacity chance constraint in (8) can be modeled as

$$d^\top y + \kappa(\epsilon)\sqrt{y^\top \Sigma y} \le b. \tag{10}$$

which can be reformulated as a second-order cone constraint when $\epsilon \le 0.5$. In this case, (8) becomes a binary second-order cone programming program. In addition, if desired, any upper bound on b_π^* that is computationally cheaper to compute can be used. In the next section, we discuss an alternative relaxation scheme that can be used when customer demands are joint normally distributed that avoids solving (8) altogether.

3.2 Relaxed pricing for joint normal demands

In the case that demands are joint normally distributed with covariance matrix $\Sigma \succeq 0$, we proceed as follows. Define η^* as the optimal objective value of the following semidefinite program

$$\eta^* = \max_{\eta,p,Q} \eta \tag{11a}$$

$$\text{s.t.} \quad d_i\eta \le p_i \qquad i \in V_+ \tag{11b}$$

$$\Sigma = \text{diag}(p_1, ..., p_n) + Q \tag{11c}$$

$$Q \succeq 0, \tag{11d}$$

and derive the following constraint based on η^*

$$d^\top y + \kappa(\epsilon)\sqrt{\eta^* d^\top y} \le b. \tag{12}$$

Proposition 2. *Let y be a binary vector satisfying* (10)*, then y satisfies* (12)*.*

Proof. Let (η^*, p^*, Q^*) be an optimal solution of (11) and let P^* be the diagonal matrix having $P^*_{ii} = p^*_i, i \in V_+$. It suffices to show that $\eta^* d^\top y \le y^\top \Sigma y$. It follows that $y^\top \Sigma y = y^\top P^* y + y^\top Q^* y \ge y^\top P^* y$, since $Q^* \succeq 0$. Since y is a binary vector, $y^\top P^* y = \sum_{i \in V_+} p^*_i y_i^2 = (p^*)^\top y$. Since p^* is feasible to (11), $p^*_i \ge d_i \eta^*$, $\forall i \in V_+$. Therefore, it follows that $y^\top \Sigma y \ge (p^*)^\top y \ge \eta^* d^\top y$. □ □

Proposition 2 shows that (12) is a relaxation of (6). The relaxation (12) is less computationally expensive to obtain as it only requires solving a semidefinite program. However, (12) is not linear in terms of y. Nonetheless, observing that the left-hand-side of (12) is a monotone increasing function of $d^\top y$, it is possible to represent this constraint as

$$d^\top y \le \hat{b}_d(\eta^*) \tag{13}$$

where

$$\hat{b}_d(\eta^*) = \left(\sqrt{b + I_\epsilon \eta^*} - \sqrt{I_\epsilon \eta^*}\right)^2 \text{and } I_\epsilon = \left(\frac{1}{2}\kappa(\epsilon)\right)^2.$$

Thus, the linear constraint (13) is a relaxation of the exact capacity chance constraint (10) and so it can be used to generate $rCCq$-routes. We conclude by noting that for *independent* normal random variables, the value of η^* has the closed-form solution $\eta^* = \min\{\sigma_i^2/d_i : i \in V_+\}$.

3.3 Improved relaxed pricing for independent normal demands.

When the random demands are independent and are normally distributed with mean vector $d \in \mathbb{Z}^n_+$ and variance vector $\sigma^2 \in \mathbb{Z}^n_+$, an improved relaxation of the pricing subproblem with the q-route relaxation can be solved by a dynamic program with a larger state space. As in (7), if y_i represents the number of times customer $i \in V_+$ is visited in the q-route, then the chance constraint in the pricing subproblem has the form: $d^\top y + \kappa(\epsilon)\sqrt{\sum_{i \in V_+} \sigma_i^2 y_i^2} \le b$. By using the fact that $y_i \le y_i^2$ for $y_i \in \mathbb{Z}_+$, we further relax this constraint as follows:

$$d^\top y + \kappa(\epsilon)\Big(\sum_{i \in V_+} \sigma_i^2 y_i\Big)^{1/2} \le b. \tag{14}$$

Any q-route that satisfies (7) also satisfies (14), and thus if we enforce (14) in the pricing subproblem, no chance-constraint feasible route is excluded.

Using (14), a dynamic program with pseudo-polynomial state space can be used to solve the pricing problem, similar to [22], by considering $d^\top y$ and $\sum_{i\in V_+}\sigma_i^2 y_i$ as two different resources in the corresponding resource-constrained shortest path problem. We briefly summarize the dynamic program as follows. We define the value function $\mathcal{V}(m,s,v)$ to represent the minimum cost q-route that reaches node v, having visited customers with total sum of means equal to m and total sum of variances equal to s. We let $\bar{c}_e$ be the cost for visiting edge e in the pricing subproblem (determined by values of dual variables in (5b) - (5e)). Then, the optimality equation of the dynamic program is as follows

$$\mathcal{V}(m,s,v) = \begin{cases} +\infty, & \text{if } m+\kappa(\epsilon)\sqrt{s} > b, m < 0, \text{ or } s < 0 \\ \min\{\mathcal{V}(m-d_i, s-\sigma_i^2, i) + \bar{c}_e : e = iv \in \delta(v)\}, & \text{otherwise,} \end{cases}$$

with $\mathcal{V}(0,0,0) := 0$.

4 Extension to distributionally robust chance constraint

We now consider an adaptation of the CCVRP model in which the distribution, $\mathbb{P}$, of customer demands is not assumed to be known. Instead, we use a *distributionally robust chance constraint* (DRCC) [5, 13, 16, 17, 23] in which we assume the distribution is known to lie within a given ambiguity set $\mathcal{P}$ of possible distributions. An assignment of a set S of customers to a vehicle is then considered to be feasible if it satisfies the chance constraint for every distribution in $\mathcal{P}$. Specifically, if we let y^S be the defined by $y_i^S = 1$ if $i \in S$ and $y^S = 0$ otherwise, then a set S of customers may be assigned to a vehicle only if y^S satisfies the following DRCC:

$$\inf_{\mathbb{P}\in\mathcal{P}} \Big\{ \mathbb{P}\{D^\top y \le b\} \Big\} \ge 1-\epsilon. \tag{15}$$

The constraint (15) is often referred to as an "individual" DRCC as it involves only a single inequality. A route that visits a set of customers S such that y^S satisfies (15) is called *DR feasible*. Thus, the distributionally robust VRP (DRVRP) is to find a minimum length set of K DR feasible routes such that every customer is visited exactly once.

To extend the edge-based formulation (1) to the DRVRP, we define, for a subset of customers S, the distributionally robust minimum vehicle requirements $r_\epsilon^{\mathcal{P}}(S)$ to be the minimum number of vehicles needed to serve S with DR feasible routes. Then the edge-based formulation for the DRVRP is identical to (1), except that $r_\epsilon^{\mathcal{P}}(S)$ are used as the vehicle requirements in (1d).

Our goal is to leverage existing results on reformulations of distributionally robust chance constraints. We make the following assumptions on the ambiguity set $\mathcal{P}$.

A1. The ambiguity set $\mathcal{P}$ is non-empty.

A2. $y = 0$ satisfies (15).

A3. There exists a closed convex cone $\mathcal{F}_\epsilon^{\mathcal{P}}$ such that the DRCC (15) is satisfied by $y \in \mathbb{R}^n$ if and only if $(y,b) \in \mathcal{F}_\epsilon^{\mathcal{P}}$.

Assumptions A1 and A2 are mild technical requirements that avoid trivial cases. The focus in the DRCC literature has been on deriving formulations of the form in Assumption A3 in which the set $\mathcal{F}_\epsilon^{\mathcal{P}}$ is convex and tractable for a fixed b. However, we make use of the observation that the reformulations derived in the literature often also define convex cones over the variable space

(y, b). For example, suppose $\mathcal{P} = \{\mathbb{P} : \mathbb{E}_{\mathbb{P}}(D) = \mu, \mathrm{Cov}_{\mathbb{P}}(D) = \Sigma\}$ where $d \in \mathbb{R}^n$ is the vector of known means and $\Sigma \in \mathbb{R}^{n\times n}$ is the known positive definite covariance matrix. In this case, the set $\mathcal{F}^{\mathcal{P}}_{\epsilon}$ is of the form:

$$\mathcal{F}^{\mathcal{P}}_{\epsilon} = \{(y, b) \in \mathbb{R}^{n+1} : d^{\top} y + \kappa(\epsilon)\sqrt{y^{\top}\Sigma y} \leq b\} \tag{16}$$

where $\kappa(\epsilon) = \sqrt{(1-\epsilon)/\epsilon}$ [16, 5]. Calafiore and El Ghaoui [5] present several other examples of definitions for the set $\mathcal{P}$ which yield a formulation having a form identical or similar to (16). A general framework for defining ambiguity sets based on projection from a higher-dimensional set is provided in [21], which in many cases lead to tractable formulations in which $\mathcal{F}^{\mathcal{P}}_{\epsilon}$ is defined as a projection from a higher-dimensional cone. One example of an ambiguity set that does *not* in general satisfy Assumption A3 is the set defined based on ϕ-divergence from a reference distribution studied by Jiang and Guan [23]. However, in this case, Jiang and Guan show that the associated DRCC problem can be reformulated as a standard chance-constrained problem on the reference distribution, with a perturbed risk level. Thus, when using such an ambiguity set, the results from the previous sections can be applied directly using by using the perturbed risk level. We refer the reader to [6, 7, 34, 36] for further examples of DRCC reformulations.

4.1 Vehicle requirements in the capacity inequalities

As in Section 2.2, we derive a computable lower bound on the vehicle requirements $r^{\mathcal{P}}_{\epsilon}(S)$. The main idea is that when the DRCC can be represented as in Assumption A3, the DRCC is equivalent to a robust linear constraint, and hence the bound used in [20] for the robust VRP can be applied. We define the following function $g_{\epsilon} : \mathbb{R}^n \to \mathbb{R}$:

$$g_{\epsilon}(y) = \min\{u : (y, u) \in \mathcal{F}^{\mathcal{P}}_{\epsilon}\}.$$

Under Assumption A3, the DRCC (15) is satisfied if and only if $g_{\epsilon}(y) \leq b$.

Theorem 6. *Suppose $\mathcal{P}$ satisfies Assumptions A1-A3. Then, for all $S \subseteq V_+$, we have*

$$r^{\mathcal{P}}_{\epsilon}(S) \geq \lceil g_{\epsilon}(y^S)/b \rceil$$

where y^S is the binary vector having $y^S_i = 1, i \in S$ and $y^S_i = 0, i \in V_+ \setminus S$.

Proof. We first observe that because $\mathcal{F}^{\mathcal{P}}_{\epsilon}$ is a closed convex cone, the function g_{ϵ} is positively homogeneous and convex. In addition g_{ϵ} is a proper convex function. Indeed $\mathcal{P}$ is nonempty by A1, and hence using A3, for any $y \in \mathbb{R}^n$ and some fixed $\mathbb{P}' \in \mathcal{P}$,

$$\begin{aligned} g_{\epsilon}(y) &= \min\left\{u : \inf_{\mathbb{P}\in\mathcal{P}}\left\{\mathbb{P}\{D^{\top} y \leq u\}\right\} \geq 1-\epsilon\right\} \\ &\geq \min\left\{u : \mathbb{P}'\{D^{\top} y \leq u\} \geq 1-\epsilon\right\} = Q^{\mathbb{P}'}(1-\epsilon) > -\infty. \end{aligned}$$

In addition, A2 implies that $g_{\epsilon}(0) < +\infty$. Let g^*_{ϵ} be the conjugate function of g_{ϵ}. As g_{ϵ} is positively homogeneous, we have

$$g^*_{\epsilon}(\alpha) = \begin{cases} 0 & \alpha \in \mathcal{A}_{\epsilon} \\ +\infty & \text{otherwise,} \end{cases}$$

where $\mathcal{A}_\epsilon = \{\alpha \in \mathbb{R}^n : \alpha^\top y \leq g_\epsilon(y) \; \forall y \in \mathbb{R}^n\}$. Then, as g_ϵ is a proper convex function, conjugate duality implies $g_\epsilon(y) = \max\{\alpha^\top y : \alpha \in \mathcal{A}_\epsilon\}$. But since the DRCC (15) is equivalent to $g_\epsilon(y) \leq b$, this implies the DRCC is in turn equivalent to the robust constraint

$$\alpha^\top y \leq b \quad \forall \alpha \in \mathcal{A}_\epsilon.$$

Then, we apply Theorem 3 (Proposition 3 of [20]) to conclude that

$$r_\epsilon^{\mathcal{P}}(S) \geq \lceil \max\{\alpha^\top y^S : \alpha \in \mathcal{A}_\epsilon\}/b \rceil = \lceil g_\epsilon(y^S)/b \rceil.$$

□ □

Evaluating $g_\epsilon(y^S)$ for a fixed set $S \subseteq V_+$ in general requires solving the convex program $\min\{u : (y^S, u) \in \mathcal{F}_\epsilon^{\mathcal{P}}\}$. However, as y^S is fixed, this problem may be significantly simpler than a problem in which the constraints $(y, u) \in \mathcal{F}_\epsilon^{\mathcal{P}}$ are used in a formulation where y are decision variables. For example, in the case of a DRCC of the form (16) or similar, evaluating $g_\epsilon(y^S)$ reduces to evaluating the left-hand side of (16).

4.2 Dantzig-Wolfe formulation and relaxed pricing

The development of the Dantzig-Wolfe formulation follows closely that of Section 3. The only difference is that instead of using CCq-routes, we now use *distributionally robust chance constraint feasible q routes* (DRq-routes), where a DRq-route is a closed walk satisfying (i) $v_0 = 0$, $v_i \in V_+, \forall i = 1, \ldots, k$ and $v_{i-1}, v_i \in E, \forall i = 1, \ldots, k+1$; and (15) with $y_v := \sum_{i=1}^k \mathbb{1}_{\{v=v_i\}}$ as the number of times v appears in the route.

As in Section 3, we further relax the pricing problem to allow routes which violate (15), but instead satisfy a knapsack constraint of the form $\pi^\top y \leq b$, where π is a given integral vector, e.g., $\pi = d$. Thus, for $\pi \in \mathbb{Z}_+^n$, we define

$$b_\pi^{\mathcal{P}} := \max\Big\{\pi^\top y : \inf_{\mathbb{P} \in \mathcal{P}} \mathbb{P}\{D^\top y \leq b\} \geq 1 - \epsilon, y_i \in \{0,1\}, i \in V_+\Big\}.$$

By construction, relaxed pricing may be accomplished by replacing the constraint (15) with the constraint $\pi^\top y \leq b_\pi^{\mathcal{P}}$.

Under Assumption A3, calculating $b_\pi^{\mathcal{P}}$ amounts to solving

$$\max\Big\{\pi^\top y : (y, b) \in \mathcal{F}_\epsilon^{\mathcal{P}}, y_i \in \{0,1\}, i \in V_+\Big\}.$$

The difficulty of this problem depends on the structure of the set $\mathcal{F}_\epsilon^{\mathcal{P}}$, but again it is solved just once as a pre-processing step of the algorithm. When $\mathcal{F}_\epsilon^{\mathcal{P}}$ has the form (16) with $\kappa(\epsilon) > 0$, the problem can be formulated as a binary second-order cone programming problem. Also in this case, we may follow the development of Section 3.2 to obtain the relaxed constraint

$$d^\top y + \kappa(\epsilon)\sqrt{\eta^* d^\top y} \leq b. \tag{17}$$

where η^* is calculated by solving the semidefinite program (11).

5 Computational Study

In this section we describe results from a computational study using the methods developed in the previous sections to solve CCVRP. We describe details of our implementation, including a heuristic for generating primal feasible solutions, in Section 5.1. Section 5.2 describes our test instances, and Section 5.3 presents the computational results and their analysis.

In Appendix A we present results of an experiment comparing the solutions obtained from the CCVRP model to those obtained from a recourse model of VRPSD. These results indicate that, when evaluated in the recourse model, the CCVRP model yields solutions that are on average only about 1% more costly than those of the recourse model, whereas the solutions of the recourse model may have high probability of vehicle capacity violation.

5.1 Implementation details

Our BCP implementation is based on the code of [18]. We refer the reader to [18] for details of the BCP implementation in general, and focus our discussion on differences in our implementation from that. As suggested in [18], we branch on sets S such that $2 < x^*(\delta(S)) < 4$, giving preference to sets that minimize $|x^*(\delta(S)) - 2.7|/d(S)$ where x^* is the solution to the LP relaxation. However, we disabled strong branching, since we found it to be too time-consuming in our test instances. In addition, the code in [18] automatically determines if branch-and-cut or BCP should be used to solve a particular instance and also what length of cycles should be eliminated from q-routes in the pricing problem of BCP. We did not use that feature since one of our goals was to test the performance of BCP against branch-and-cut. We instead choose in advance whether to use branch-and-cut or BCP and, in the latter case, we chose to eliminate 3-cycles from the $rCCq$-routes in the pricing problem.

We also developed a heuristic algorithm for CCVRP. The heuristic adapts and extends the simple idea proposed by Clarke and Wright [10] for the deterministic VRP. The idea behind this method is that if a route visiting customers $v_1, v_2, ..., v_k$ is merged with another route visiting customers $w_1, w_2, ..., w_l$ to obtain a larger route that visits all those $k + l$ customers in that same order, the total savings from this merger can be easily calculated. For example, if the resulting route visits customers $v_1, v_2, ..., v_k, w_1, w_2, ..., w_l$, the total savings is $\ell_{0v_k} + \ell_{0w_1} - \ell_{v_k w_1}$. Clarke and Wright's algorithm starts with n simple routes, which go from the depot to a customer and go straight back to the depot. At each iteration, it looks at all the possible pairs of routes, and then merges the pair that yields a feasible route with the maximum savings. The algorithm terminates once exactly K routes are obtained.

A direct adaptation of this heuristic from the deterministic VRP to CCVRP is straightforward. The only change that is needed is that when considering merging two routes, feasibility of the chance constraint must be checked. We also introduce several enhancements. First, we use the linear programming (LP) relaxation solution at the root node to obtain an improved set of starting routes. After solving the LP relaxation, we perform a graph search to find all connected components, where two nodes are connected if the edge variable value is greater than a threshold. We use a threshold of 0.8, which results in a relatively large number of smaller connected components, which are then used to form starting routes. Second, we use a look-ahead strategy to make decisions that are less greedy. In particular, we use the following process for choosing which routes to merge in each step. For each possible pair of routes, we tentatively merge the routes and then run the standard greedy merging procedure until completion. We choose to merge the pair of routes which yields the best solution after the greedy process. This process is repeated until the number of routes reaches K, or until there are no pairs of routes that can be feasibly merged (in the latter case, the heuristic fails to find a solution). Finally, for each route, we check if swapping the order of any three consecutive customers in the route would yield a shorter route, and update the route if so.

5.2 Test Instances

We test our methods on CCVRP instances created by adapting the deterministic VRP instances available at `http://vrp.atd-lab.inf.puc-rio.br/`. Ten benchmark deterministic instances having 32–55 vertices were chosen from this library. For each deterministic instance we created six different CCVRP instances, corresponding to three choices for the distribution (independent normal, joint normal, and scenario) and two cases for the level of variance in the distribution (low and high). The scenario distribution with high variance is based on sampling from a model of random demands that includes the possibility that customers are "no shows" (i.e., have zero demand) with positive probability. For this model of random demands, exactly evaluating the probability that a vehicle's capacity is exceeded is computationally challenging, motivating the use of a sample average scenario approximation. The graph and edge lengths for each CCVRP instance are used directly from the corresponding deterministic instance. We use $\epsilon = 0.05$ in all of our test instances. The number of vehicles, K, used for an instance was usually increased slightly from the deterministic instance (and more for the instances with high variances) as the chance constraint effectively reduces the capacity of an individual vehicle compared to the deterministic case. In such cases, we change the name of the original deterministic instance to reflect the increase in number of vehicles. Details of how the distributions of customer demands were generated in each of the three distribution cases are given in Appendix B.

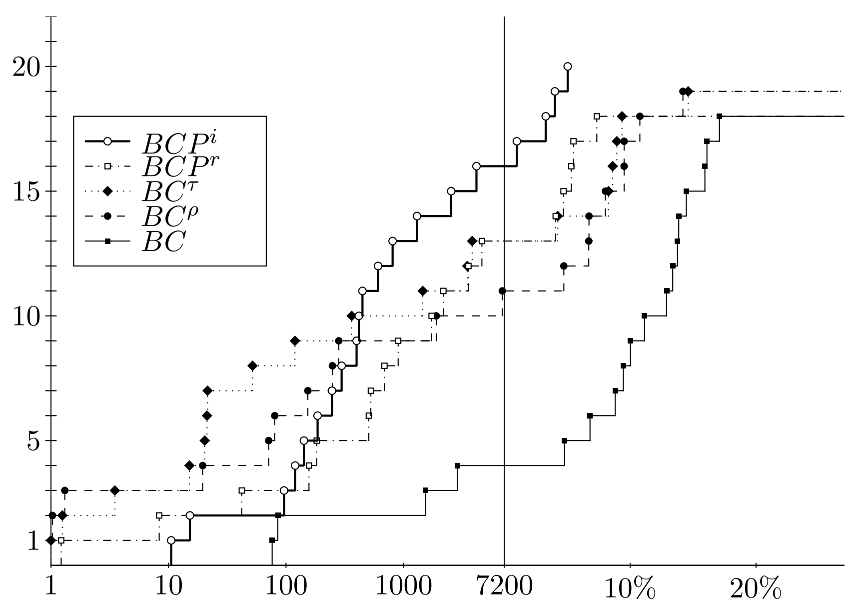

Figure 1: Summary of results for instances with independent normal distribution.

5.3 Results

We tested five solution methods for the instances with independent normal random demands. Three branch-and-cut (BC) methods based on formulation (1) were tested, using valid lower bounds $k_\epsilon(S)$, $\rho_\epsilon(S)$ and $\tau_\epsilon^J(S)$ for $r_\epsilon(S)$, named BC, BC^ρ, BC^τ, respectively. Two branch-and-cut-and-price (BCP) methods based on formulation (5) were tested, one using the relaxed pricing strategy proposed in Section 3.2 (denoted BCP^r) and the other using the improved relaxed pricing discussed in Section 3.3 (denoted BCP^i). For the relaxed pricing in BCP^r, we use $\eta^* = \min\{\sigma_i^2/d_i : i \in V_+\}$ in (13). Both BCP methods use $\tau_\epsilon^J(S)$ as the bound on $r_\epsilon(S)$ in the capacity inequalities. The same methods were tested for the instances with joint normal random demands, except for BCP^i, which only applies for instances with independent normal random demands. In the BCP^r method, the value η^* for use in (13) is calculated by solving the semidefinite program (11) using SeDuMi 1.3 [33]. For the instances with a scenario model of random demands, the two applicable branch-and-cut methods (BC and BC^ρ) and the one applicable branch-and-price method (BCP^r) were tested. For BCP^r, $\rho_\epsilon(S)$ was used as the bound on $r_\epsilon(S)$ in the capacity inequalities, and relaxed pricing was done using the method in Section 3.1, with b_π computed by solving the basic "big-M" integer programming equivalent to (8) [31], solved with GuRoBi 5.6.2.

All experiments were run on a Dell R510 machine with 128G memory, and two 2.66G X5650 Xeon Chips, having 12 cores each. Our implementation is serial so we use only one core. A time limit of 7200 seconds was imposed. When solving the preprocessing problem (8) for the scenario instances, a time limit of 2000 seconds was imposed on the preprocessing time. In case the preprocessing problem is terminated due to the time limit, the best upper bound obtained on the optimal value is used for b_π in (9). In all cases, the preprocessing time is included in the reported solution time.

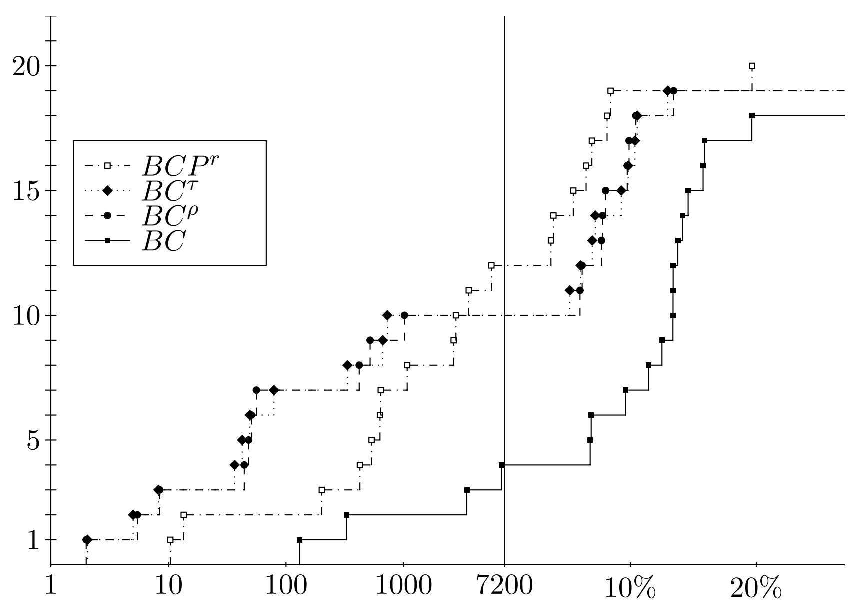

Figure 2: Summary of results for instances with joint normal distribution.

Figures 1 – 3 show the aggregate results of the methods for the 20 instances having independent normal, joint normal, and scenario distributions, respectively. The x-axis in these plots is broken

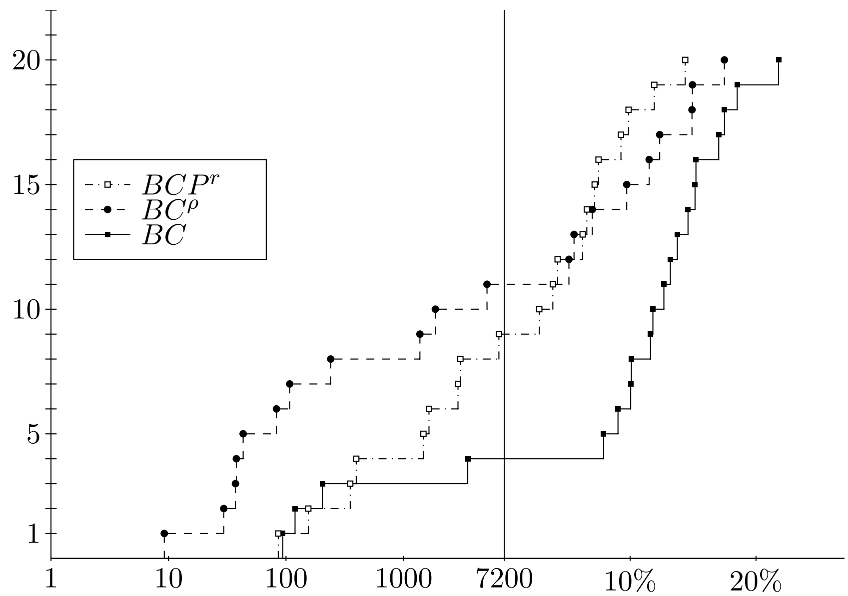

Figure 3: Summary of results for instances with scenario distribution.

into two intervals. The units in the left interval are seconds, and a point (x, y) that has x in the left-interval represents that y instances were solved in at most x seconds by the corresponding algorithm. For instances not solved within the time limit, we recorded the final optimality gap, calculated as $\frac{UB-LB}{UB}$, where UB and LB are the upper and lower bounds, respectively, obtained by the method on that instance within the time limit. The units in the right interval of the x-axis is percentage optimality gap, and a point (x, y) with x in the right interval then represents that the method achieved final optimality gap of at most x in y test instances. Note that the left interval of the x-axis (for times) uses a log-scale and the right-interval (for percentage gaps) uses a linear scale. In addition, note that the heuristic proposed in Section 5.1 may fail to find a feasible solution if it cannot reduce the number of vehicles to the desired value. In that case (and if the branch-and-bound search did not find a feasible solution), we declare that instance to have a final gap of 100%. This is the reason why some of the curves shown in the figures never reach the 20 instance mark.

The figures show that the improved valid lower bounds on $r_\epsilon(S)$ have a very significant impact in all instances. The formulation BC, which uses $k_\epsilon(S)$ is clearly the weakest of all. Interestingly, for instances with normally distributed demands, BC^τ and BC^ρ have similar results, although BC^τ has a more clear advantage in the independent normal case. For that reason, we consider that BC^τ is the best branch-and-cut formulation, when applicable, but also find that the more general BC^ρ also performs well.

We next compare the BCP methods based on (5) to the BC methods based on (1). We find that for easier instances, the best BC methods perform better than the BCP methods, in that they are able to solve many instances more quickly. However, for more difficult instances, the better bounds provided by (5) are more beneficial. Indeed, although the BCP curves start below the BC curves, the BCP curves eventually exceed the BC curve, indicating that using BCP enables solving some instances within the time limit that could not be solved by just BC, and also that the final optimality gaps on the unsolved instances are smaller when using BCP. This behavior is consistent with other BCP approaches, see, e.g., [18]. Finally, note that in the independent normal case, BCP^i outperforms BCP^r, and hence the extra time spent pricing with the better relaxation appears to be worth it.

Inst.	BC^τ				BCP^i			
	UB	G/T	RG	RT	UB	G/T	RG	RT
A-n32-k5-L	**801**	**1s**	**0.9%**	**0s**	**801**	**15s**	**0.0%**	**15s**
A-n34-k6-L	**789**	**4s**	**1.5%**	**0s**	**789**	**11s**	**0.0%**	**11s**
A-n36-k5-L	**835**	**363s**	**4.8%**	**0s**	**835**	**809s**	**2.3%**	**18s**
A-n37-k5-L	**687**	**15s**	**3.2%**	**1s**	**687**	**96s**	**0.8%**	**19s**
B-n39-k5-L	**563**	**20s**	**1.1%**	**0s**	**563**	**1302s**	**1.4%**	**27s**
A-n44-k7-L	**971**	**3495s**	**4.6%**	**1s**	**971**	**120s**	**0.4%**	**13s**
A-n45-k7-L	**981**	**119s**	**4.5%**	**0s**	**981**	**246s**	**1.2%**	**17s**
P-n50-k12-L	741	8.9%	8.9%	1s	**734**	**298s**	**0.4%**	**12s**
P-n51-k12-L	783	8.6%	8.6%	1s	**777**	**399s**	**0.8%**	**10s**
A-n55-k10-L	1132	4.3%	5.2%	1s	**1132**	**142s**	**0.4%**	**36s**
A-n32-k5-H	**868**	**52s**	**7.0%**	**0s**	880	5.0%	5.0%	34s
A-n34-k6-H	**831**	**21s**	**5.5%**	**0s**	**831**	**417s**	**1.2%**	**28s**
A-n36-k6-H	**875**	**1460s**	**5.2%**	**0s**	**875**	**2548s**	**2.8%**	**43s**
A-n37-k5-H	**711**	**21s**	**5.2%**	**1s**	**711**	**608s**	**1.8%**	**62s**
B-n39-k6-H	**586**	**1s**	**0.4%**	**0s**	**586**	**186s**	**0.4%**	**70s**
A-n44-k7-H	1025	9.3%	9.3%	0s	1022	1.0%	1.6%	45s
A-n45-k7-H	**1024**	**3838s**	**7.6%**	**0s**	1050	4.0%	4.0%	71s
P-n50-k13-H	813	14.6%	14.6%	1s	**782**	**4176s**	**1.0%**	**30s**
P-n51-k12-H	-	100.0%	100.0%	1s	837	3.3%	3.3%	35s
A-n55-k11-H	1206	8.3%	8.3%	1s	**1188**	**448s**	**0.3%**	**144s**

Table 1: Detailed computational results for instances with independent normal distribution.

Tables 1 – 3 show the detailed statistics for the independent normal, joint normal, and scenario demand instances, respectively. For brevity, we present only statistics for the best BC and BCP approaches, as concluded from Figures 1 – 3. The columns of those tables are as follows. **Inst.** gives the instance name, where in the instance name, the last character indicates if it is a low (L) or high (H) variance instance and the number following `-k` indicates the number of vehicles used. For each method (BC or BCP), column **UB** shows the best upper bound found at the end of the execution. A dash represents that no feasible solution was found. Column **G/T** gives the total time (in seconds) it took to solve the corresponding instance. If the instance was not solved within the time limit of 7200 seconds, then the number in that column represents the final optimality gap (in %) at that time. Finally, columns **RG**, **RT** show, respectively, the gap (in %) obtained at the root node and the time it took to solve that node (note that **RT** does not include the time spent solving the preprocessing problems (11) and (8)). The detailed statistics are highlighted in bold if the instance was solved to optimality with the corresponding method.

We find that the detailed experiments corroborate our previous conclusions, though it can be seen from the tables that even in some larger instances, the branch-and-cut method still outperforms branch-cut-and-price. We also see that the time spent solving the root node using BCP^i on the independent normal instances is significantly larger than the BCP methods on the other instance classes, indicating that the improved pricing of Section 3.3 is indeed significantly more time-consuming. On the other hand, we also observe that BCP^i yields significantly smaller

Inst.	BC^r				BCP^r			
	UB	G/T	RG	RT	UB	G/T	RG	RT
A-n32-k5-L	**802**	**2s**	**1.6%**	**0s**	**802**	**14s**	**1.3%**	**2s**
A-n34-k6-L	**789**	**8s**	**1.5%**	**0s**	**789**	**10s**	**0.0%**	**1s**
A-n36-k5-L	**838**	**727s**	**5.2%**	**0s**	**838**	**3589s**	**3.8%**	**2s**
A-n37-k5-L	**701**	**37s**	**5.1%**	**1s**	**701**	**535s**	**4.1%**	**3s**
B-n39-k5-L	**563**	**79s**	**1.1%**	**0s**	**563**	**1073s**	**2.0%**	**3s**
A-n44-k7-L	998	7.3%	7.3%	0s	**982**	**5585s**	**2.4%**	**2s**
A-n45-k7-L	**989**	**666s**	**5.6%**	**1s**	**989**	**630s**	**2.7%**	**2s**
P-n50-k12-L	752	10.5%	10.5%	2s	745	4.0%	4.0%	1s
P-n51-k12-L	787	9.3%	9.3%	1s	784	3.7%	3.7%	1s
A-n55-k11-L	1158	6.1%	6.1%	1s	**1144**	**2666s**	**2.1%**	**2s**
A-n32-k5-H	**870**	**333s**	**7.7%**	**0s**	**870**	**2787s**	**7.1%**	**1s**
A-n34-k6-H	**832**	**42s**	**5.9%**	**0s**	**832**	**425s**	**3.5%**	**1s**
A-n36-k6-H	884	5.2%	8.2%	0s	892	7.0%	7.0%	2s
A-n37-k5-H	**711**	**49s**	**5.5%**	**1s**	**711**	**640s**	**5.3%**	**3s**
B-n39-k6-H	**591**	**5s**	**1.3%**	**0s**	**591**	**202s**	**2.3%**	**3s**
A-n44-k7-H	1028	9.9%	9.9%	1s	1024	5.5%	5.5%	2s
A-n45-k8-H	1061	7.0%	10.2%	0s	1063	6.6%	6.6%	2s
P-n50-k13-H	794	13.1%	13.1%	1s	806	8.5%	8.5%	1s
P-n51-k12-H	-	100.0%	100.0%	1s	966	19.7%	19.7%	1s
A-n55-k11-H	1223	10.6%	10.6%	1s	1238	8.2%	8.3%	2s

Table 2: Detailed computational results for instances with joint normal distribution.

root gaps.

Finally, we comment on the time spent solving the preprocessing problems (11) and (8) in the BCP^r method for the joint normal and scenario instances, respectively. For the joint normal instances, the average time spent solving the SDP (11) over the 20 instances was 19.5 seconds, with the maximum time being 64.3 seconds. Solving the MIP formulation of (8) for the scenario instances was significantly more time-consuming. Five of the 20 instances were not solved to optimality within the 2000 second time limit, and the average solution time of the remaining 15 instances was 576.3 seconds. We note, however, that our implementation used only the most basic MIP formulation from [31], and so these preprocessing times could potentially be reduced substantially using specialized methods from [31], and in the case of instances in which the preprocessing problem was terminated suboptimally, this may also lead to an improved bound on b_π for use in (9).

6 Conclusion

We propose branch-and-cut and branch-and-cut-and-price approaches for the CCVRP with very mild assumptions on the distribution of customer demands. In particular, we allow for correlations between random customer demands, a condition which has been often overlooked by previous works. The key challenges that were addressed were the derivation of strong, but easy to compute lower bounds on the minimum number of vehicles required to serve a subset of customers and

Inst.	BC^ρ				BCP^r			
	UB	G/T	RG	RT	UB	G/T	RG	RT
A-n32-k5-L	**802**	**9s**	**1.9%**	**0s**	**802**	**86s**	**1.3%**	**2s**
A-n34-k6-L	**789**	**37s**	**1.5%**	**0s**	**789**	**352s**	**0.0%**	**1s**
A-n36-k5-L	**838**	**1379s**	**5.3%**	**0s**	**838**	**6490s**	**3.8%**	**3s**
A-n37-k5-L	**687**	**83s**	**3.2%**	**1s**	**687**	**155s**	**2.4%**	**3s**
B-n39-k5-L	**561**	**43s**	**0.7%**	**0s**	**561**	**1477s**	**1.6%**	**5s**
A-n44-k7-L	980	5.5%	5.5%	1s	**971**	**2909s**	**1.5%**	**2s**
A-n45-k7-L	**981**	**1866s**	**5.0%**	**0s**	**981**	**1650s**	**1.9%**	**2s**
P-n50-k12-L	768	12.3%	12.3%	2s	764	6.6%	6.6%	2s
P-n51-k12-L	791	9.7%	9.7%	1s	785	3.9%	3.9%	1s
A-n55-k11-L	1146	5.1%	5.1%	1s	1162	4.2%	4.2%	3s
A-n32-k6-H	**892**	**30s**	**6.2%**	**0s**	**892**	**396s**	**5.9%**	**1s**
A-n34-k6-H	**848**	**240s**	**7.7%**	**0s**	896	9.9%	9.9%	2s
A-n36-k6-H	896	7.0%	9.7%	0s	898	7.5%	7.5%	2s
A-n37-k6-H	**733**	**108s**	**7.1%**	**0s**	**733**	**3046s**	**6.4%**	**3s**
B-n39-k6-H	**606**	**38s**	**3.3%**	**0s**	606	2.8%	4.4%	4s
A-n44-k8-H	1088	11.5%	11.5%	1s	1094	9.3%	9.3%	2s
A-n45-k8-H	**1047**	**5138s**	**7.9%**	**0s**	1060	6.2%	6.5%	2s
P-n50-k13-H	814	15.0%	15.0%	1s	828	11.9%	11.9%	1s
P-n51-k13-H	898	17.5%	17.5%	1s	841	7.2%	7.5%	1s
A-n55-k11-H	1290	14.9%	14.9%	1s	1314	14.4%	14.4%	3s

Table 3: Detailed computational results for instances with scenario distribution.

the derivation of pseudo-polynomial time pricing routines. The latter challenge arises from the strong $\mathcal{NP}$-hardness of a generalization of the pricing for the deterministic VRP.

Our approach represents the first successful BCP approach for the CCVRP. The formulations proposed are promising and can be used to solve instances with up to 55 vertices. Nonetheless, several improvements can still be made to the implementation, in particular of the pricing routines, as well as the investigation of further valid inequalities for the problem. These remain the topic of further research.

References

[1] Baldacci, R., Mingozzi, A.: A unified exact method for solving different classes of vehicle routing problems. Math. Program. **120**(2), 347–380 (2009)

[2] Baldacci, R., Mingozzi, A., Roberti, R.: New route relaxation and pricing strategies for the vehicle routing problem. Oper. Res. **59**(5), 1269–1283 (2011)

[3] Beraldi, P., Bruni, M.E., Laganà, D., Musmanno, R.: The mixed capacitated general routing problem under uncertainty. European Journal of Operational Research **240**, 382–392 (2015)

[4] Bertsimas, D.J.: A vehicle routing problem with stochastic demand. Oper. Res. **40**(3), 574–585 (1992)

[5] Calafiore, G.C., El Ghaoui, L.: On distributionally robust chance-constrained linear programs. Journal of Optimization Theory and Applications **130**(1), 1–22 (2006)

[6] Chen, W., Sim, M., Sun, J., Teo, C.P.: From CVaR to uncertainty set: Implications in joint chance-constrained optimization. Oper. Res. **58**(2), 470–485 (2010)

[7] Cheng, J., Delage, E., Lisser, A.: Distributionally robust stochastic knapsack problem. SIAM Journal on Optimization **24**(3), 1485–1506 (2014)

[8] Christiansen, C.H., Lysgaard, J., Wøhlk, S.: A branch-and-price algorithm for the capacitated arc routing problem with stochastic demands. Oper. Res. Lett. **37**(6), 392–398 (2009)

[9] Christofides, N., Mingozzi, A., Toth, P.: Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. Math. Program. **20**(1), 255–282 (1981)

[10] Clarke, G., Wright, J.W.: Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. **12**(4), 568–581 (1964)

[11] Contardo, C., Martinelli, R.: A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. Discr. Optim. **12**, 129–146 (2014)

[12] Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. Man. Sci. **6**(1), 80–91 (1959)

[13] Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. **58**(3), 595–612 (2010)

[14] Dinh, T., Fukasawa, R., Luedtke, J.: Exact algorithms for the chance-constrained vehicle routing problem. In: Q. Louveaux, M. Skutella (eds.) Integer Programming and Combintatorial Optimization. Springer-Verlag (2016)

[15] Dror, M., Laporte, G., Louveaux, F.V.: Vehicle routing with stochastic demands and restricted failures. Zeitschrift für Oper. Res. **37**(3), 273–283 (1993)

[16] El Ghaoui, L., Oks, M., Oustry, F.: Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. Oper. Res. **51**(4), 543–556 (2003)

[17] Erdoğan, E., Iyengar, G.: Ambiguous chance constrained problems and robust optimization. Math. Program. **107**(1), 37–61 (2005)

[18] Fukasawa, R., Longo, H., Lysgaard, J., de Aragão, M.P., Reis, M., Uchoa, E., Werneck, R.F.: Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Math. Program. **106**(3), 491–511 (2006)

[19] Gauvin, C., Desaulniers, G., Gendreau, M.: A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands. Comput. & Oper. Res. **50**, 141–153 (2014)

[20] Gounaris, C.E., Wiesemann, W., Floudas, C.A.: The robust capacitated vehicle routing problem under demand uncertainty. Oper. Res. **61**(3), 677–693 (2013)

[21] Hanasusanto, G.A., Roitch, V., Kuhn, D., Wiesemann, W.: A distributionally robust perspective on uncertainty quantification and chance constrained programming. Math. Program. **151**(1), 35–62 (2015)

[22] Irnich, S., Desaulniers, G.: Shortest path problems with resource constraints. In: G. Desaulniers, J. Desrosiers, M. Solomon (eds.) Column Gen. Springer (2005)

[23] Jiang, R., Guan, Y.: Data-driven chance constrained stochastic program. Math. Program. pp. 1–37 (2015)

[24] Laporte, G., Louveaux, F., Mercure, H.: Models and exact solutions for a class of stochastic location-routing problems. European J. Oper. Res. **39**(1), 71–78 (1989)

[25] Laporte, G., Louveaux, F., Mercure, H.: The vehicle routing problem with stochastic travel times. Trans. Sci. **26**(3), 161–170 (1992)

[26] Laporte, G., Louveaux, F.V., Van Hamme, L.: An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. Oper. Res. **50**(3), 415–423 (2002)

[27] Luedtke, J., Ahmed, S.: A sample approximation approach for optimization with probabilistic constraints. SIAM J. Optim. **19**, 674–699 (2008)

[28] Pecin, D., Pessoa, A., Poggi, M., Uchoa, E.: Improved branch-cut-and-price for capacitated vehicle routing. In: Lecture Notes in Computer Science, vol. 8494 – Integer Programming and Combinatorial Optimization, pp. 393–403. Springer (2014)

[29] Pessoa, A.A., Pugliese, L.D.P., Guerriero, F., Poss, M.: Robust constrained shortest path problems under budgeted uncertainty (2014). URL `http://www.optimization-online.org/DB_FILE/2014/10/4601.pdf`

[30] Secomandi, N., Margot, F.: Reoptimization approaches for the vehicle-routing problem with stochastic demands. Oper. Res. **57**(1), 214–230 (2009)

[31] Song, Y., Luedtke, J.R., Küçükyavuz, S.: Chance-constrained binary packing problems. INFORMS J. Comput. (2014)

[32] Stewart, W.R., Golden, B.L.: Stochastic vehicle routing: A comprehensive approach. Euro. J. Oper. Res. **14**(4), 371–385 (1983)

[33] Sturm, J.F.: Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. Optimization Methods and Software **11**(1–4), 625–653 (1999)

[34] Yang, W., Xu, H.: Distributionally robust chance constraints for non-linear uncertainties. Math. Program. **155**(1), 231–265 (2014)

[35] Yang, W.H., Mathur, K., Ballou, R.H.: Stochastic vehicle routing problem with restocking. Trans. Sci. **34**(1), 99–112 (2000)

[36] Zymler, S., Kuhn, D., Rustem, B.: Distributionally robust joint chance constraints with second-order moment information. Math. Program. **137**, 167–198 (2011)

Appendix A : Comparison of the CCVRP with recourse models

We conducted one additional experiment to compare the solution obtained by the CCVRP with the one obtained by a recourse model for VRPSD, where the assumed recourse is that a return trip must be made to the depot whenever a vehicle's capacity is exceeded. This experiment was done for instances having independent normal customer demands, since these are the only instances that can be solved for the recourse model. For such instances, the optimal solution for the recourse-based model and for the CCVRP were obtained, denoted as x^r and x^{cc}, respectively. Both solutions were then compared according to two quantities: $z^r(x)$, the objective value of the recourse-based two-stage stochastic program, and $\eta(x)$, which represents the largest probability that a vehicle will have its capacity violated. Note that by design, $z^r(x^r) \leq z^r(x^{cc})$ and that $\eta(x^{cc}) \leq 0.05$ – though it is possible that $\eta(x^r) < \eta(x^{cc})$. The instances selected for this experiment are smaller than those in the previous sections due to difficulty in solving the recourse-based model for larger instances.

The results are presented in Table 4. We find that when evaluating the CCVRP solution in the recourse model, the value $z^r(x^{cc})$ was, on average, only about 1% more than the optimal value $z^r(x^r)$, and the largest increase was 3.4%. On the other hand, while $\eta(x^{cc})$ was always (by design) less than 0.05, $\eta(x^r)$ was greater than 0.15 in three of the instances, and as high as as 0.5, meaning that in the solution there was a vehicle whose capacity would be exceeded on average 50% of the time. We thus conclude that the CCVRP model tends to yield solutions that are high quality for the recourse model, whereas the reverse is not true. In addition, the CCVRP model is not dependent on a particular assumption of the recourse taken, and can be solved also when customer demands are not independent.

Inst.	Var.	x^{cc}		x^r	
		η	z^r	η	z^r
E-n13-k14	Low	0.017	277.6	0.500	271.4
E-n22-k4	Low	0.001	373.0	0.001	373.0
E-n22-k5	Low	0.048	402.9	0.076	399.2
P-n22-k2	Low	0.024	213.5	0.024	213.5
P-n22-k3	Low	0.031	238.6	0.064	237.3
E-n13-k14	High	0.040	291.2	0.083	281.6
E-n22-k4	High	0.041	375.5	0.041	375.5
E-n22-k5	High	0.036	426.4	0.237	414.2
P-n22-k2	High	0.010	215.2	0.010	215.2
P-n22-k3	High	0.007	240.4	0.169	239.7

Table 4: Comparison of solutions from chance-constrained and recourse models for VRPSD.

Appendix B : Instance Generation Details

Instances with independent normal random demands were generated by letting the mean customer demand d_i be equal to the demand of the customer i in the deterministic instance. Low variance instances were generated by choosing the standard deviations σ_i uniformly at random in the interval $[0.07 * d_i, 0.13 * d_i]$. For high variance instances, σ_i was selected uniformly at random in the range $[0.14 * d_i, 0.26 * d_i]$.

Instances with joint normal distributions were generated by first choosing the means and standard deviations using the same procedure as for the independent normal instances. To determine correlation between two customers $i, j, i \neq j$, we first let $\gamma_{ij} = 1/(\ell_{ij} * U(0.4, 1.6))$, where $U(0.4, 1.6)$ is a number chosen uniformly at random in $[0.4, 1.6]$. We then set the correlation between customers $i \neq j$ as $\rho_{ij} = 0.2\gamma_{ij}/(\overline{\gamma} + \underline{\gamma})$, where $\overline{\gamma}$ and $\underline{\gamma}$ are the largest and smallest values of γ_{ij} over all $i \neq j$. Correlations are determined in this way so that customers that are closer together tend to have higher correlation, and the scaling ensures that all correlations are significantly less than 1.0. The covariance between customers i and j is then set as $\Sigma_{ij} = \rho_{ij}\sigma_i\sigma_j$. This procedure successfully yielded a positive definite matrix in each of our test instances.

Each of the scenario distribution consists of 200 equally likely scenarios. The low variance instances were created by generating a sample of 200 scenarios from the joint normal distribution used in the low variance joint normal instances. The high variance instances were generated similarly, except that each customer i also had a probability, p_i, of having zero demand. These probability values were first generated randomly in such a way that about half of the customers have $p_i = 0$, and the rest have p_i between 0 and 0.4. Thus, to generate each scenario, the demands for all customers were first generated according to the joint normal distribution. Then, for each customer i, its demand in that scenario was set to zero with probability p_i. This distribution was used to provide a test with a distribution in which it is difficult to exactly calculate $\mathbb{P}\{D(S) \leq b\}$ for a subset of customers S, motivating the use of the scenario approximation. For both of the high and low variance instances, the sampled demands were rounded to the nearest integer.