



WP

Christian H. Christiansen & Jens Lysgaard

A Column Generation Approach to the Capacitated
Vehicle Routing Problem with Stochastic Demands

Logistics/SCM
Research Group

A Column Generation Approach to the Capacitated Vehicle Routing Problem with Stochastic Demands

Christian H. Christiansen* and Jens Lysgaard

Logistics/SCM Research Group

Department of Business Studies

Aarhus School of Business

Fuglesangs Allé 4

DK-8210 Aarhus V

Denmark

E-mail: {chc,lys}@asb.dk

*Corresponding author

March 2006

Abstract

In this article we introduce a new exact solution approach to the Capacitated Vehicle Routing Problem with Stochastic Demands (CVRPSD). In particular, we consider the case where all customer demands are distributed independently and where each customer's demand follows a Poisson distribution.

The CVRPSD can be formulated as a Set Partitioning Problem. We show that, under the above assumptions on demands, the associated column generation subproblem can be solved using a dynamic programming scheme which is similar to that used in the case of deterministic demands.

To evaluate the potential of our approach we have embedded this column generation scheme in a branch-and-price algorithm. Computational experiments on a large set of test instances show promising results.

Keywords: Routing, Stochastic programming, Logistics, Branch and Bound, Dynamic programming.

1 Introduction

The Capacitated Vehicle Routing Problem (CVRP) in a deterministic environment has been widely studied throughout the literature, and can be described as follows. A set of customers must be provided with known quantities of a common commodity from a single depot. To make the deliveries a fleet of identical vehicles, each with a given capacity, is available. The objective is to find a collection of routes of minimum total travel cost under the restrictions that i) each route begins and ends at the depot, ii) each customer is serviced exactly once, and iii) the total demand on any route does not exceed the vehicle capacity.

The CVRP has been extended in numerous directions for instance by incorporating time windows, multiple depots or maximum route duration. For thorough reviews of the CVRP with various extensions see (Ball et al., 1995; Toth and Vigo, 2002). In this article we consider the Capacitated Vehicle Routing Problem with Stochastic Demands (CVRPSD), which may be viewed as a stochastic counterpart of the CVRP. The CVRPSD differs from the CVRP with respect to the following points:

1. In the CVRPSD, the customers' demands are stochastic variables of which only the probability distribution for each customer is assumed known at the time of planning.
2. In the CVRPSD, it is the *expected* total travel cost that must be minimized.
3. In the CVRPSD, the total actual demand on a route may exceed the vehicle capacity. In such cases a *failure* is said to occur. A strategy is required for updating the routes in case of such an event. The actual action resulting from this strategy is called a *recourse* action. The particular strategy affects the expected cost of a given route, so the strategy must be known at the time of planning.

The CVRPSD has not received nearly the same level of attention as the CVRP. In the literature there are given several reasons for the limited attention paid to the CVRPSD. Of these perhaps the most important reason is that the CVRP problem in itself is very hard to solve, and adding a stochastic dimension to the problem only makes it even more intractable.

Nonetheless, neglecting the stochastic nature of demands during the planning of the routes can incur substantially higher expected costs, than what would have been the result if the stochastic demands had been explicitly included in the route planning. For the TSP this has been thoroughly illustrated in (Dror et al., 1989). However, the effects causing the increased expected cost do not only relate to the customers' sequence on each route. To see this, consider the undirected network in Figure 1.

Each letter denotes a customer, whereas 0 denotes the depot. For each customer, the parentheses contain the possible actual demands associated with the customer. For customers A and C , the probability for each of the two possible actual demands is 0.5. Further, for each customer the value in square brackets is the expected value of the stochastic demand. If a vehicle is depleted it must return to the depot to reload and then continue its route from the point of failure. The vehicle capacity is 10 units.

Considering only the expected values of the demands as input to a CVRP model would yield the collection of routes with minimum distance traveled, not including the expected failure costs, namely the routes $0 - B - A - 0$ and $0 - C - 0$. However, considering the stochastic demands as input to a CVRPSD model would yield the routes $0 - B - C - 0$ and $0 - A - 0$.

The expected travel cost for each solution is 33 and 29.5, respectively. As this example illustrates, neglecting the stochastic nature of demands can cause a suboptimal solution, not only as a consequence of the sequence on each route, but also as a consequence of wrongful allocation of customers

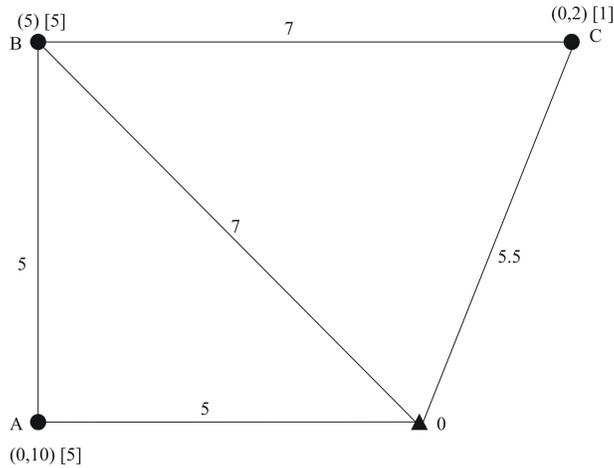


Figure 1: Routing with stochastic demands

to the routes. Several other reasons have been given for incorporating stochastic demands into the route planning, for an overview of these see e.g. (Bertsimas, 1992; Savelsbergh and Goetschalckx, 1995; Haughton, 2002).

The article is organized as follows. Section 2 gives a brief literature review regarding research on a priori routing regarding CVRPSD. Section 3 focuses on the CVRPSD modeling. Section 4 presents our new solution method, and Section 5 provides computational results obtained with the proposed algorithm. Concluding remarks are given in section 6.

2 Literature review

To the best of our knowledge the first considering of the CVRPSD was in (Tillman, 1969). He considered a multi depot variant of the CVRP with Poisson distributed demands. The model considered a cost trade off between exceeding the vehicle capacity and finishing the route with excess capacity. The solution approach was a modification of the savings algorithm originally introduced in (Clarke and Wright, 1964). For further review of savings based approaches see (Beasley, 1984; Dror and Trudeau, 1986).

Several modeling approaches have been explored regarding the CVRPSD, see, e.g. (Gendreau et al., 1996). Two frequently used approaches are chance constrained programming and two stage stochastic programming, respectively.

Chance constrained models implicitly incorporate the cost of failure. This is done by introducing a threshold value, limiting the maximum probability of failure for each route in the final route collection.

Two stage stochastic programming models, however, incorporate the cost of failure explicitly. The

first stage contains the planning of routes, taking into account the expected failure costs incurred by the execution of the routes. The second stage contains the execution of the planned routes according to the chosen strategy for updating the routes in case of failure.

Yet another approach to the CVRPSD has been Markov decision process modeling. However, due to the large number of states this approach has received limited attention (Dror et al., 1989).

For a comparison of the chance constrained approach and the two stage stochastic programming approach see (Bastian and Kan, 1992; Dror et al., 1989; Stewart and Golden, 1982).

Regarding the two stage stochastic programming approach, (Bertsimas, 1992) formulated two widely accepted recourse actions (A) and (B), respectively. These are based on two different assumptions regarding the time at which a customer's actual demand becomes known. Strategy (A) assumes that a customer's actual demand becomes known only upon arrival at the customer. Strategy (B), however, assumes that actual demands become known early enough to enable the vehicle to skip customers with zero actual demand. The recourse action under both strategies is to deplete the vehicle at the point of failure, return to the depot to reload and continue the originally planned route from the point of failure. In the particular case that a customer's demand equals the remaining load of the vehicle, the vehicle returns to the depot to reload before visiting the next customer.

In their article from 1993, Laporte and Louveaux developed an integer L-shaped method for stochastic programs with recourse (Laporte and Louveaux, 1993). Their approach is based on adding feasibility cuts to a relaxed flow formulation of a CVRPSD until an integer feasible solution is found. If the discrepancy between a lower bound on the expected cost of failure and the current value of the failure cost is above some threshold, an optimality cut is added to the formulation. However, in their article no computational results were presented. The method has been applied to the CVRPSD in 1995 (Gendreau et al., 1995), 1998 (Laporte and Louveaux, 1998) and 2002 (Laporte et al., 2002). In the latter, instances with up to 100 customers were solved.

From the computational results it can be seen that the L-shaped method seems to perform best on problems with small expected demands relative to the vehicle capacity.

In this article we develop a new solution approach to the CVRPSD. Specifically, we formulate the CVRPSD as a Set Partitioning Problem and develop a dynamic programming algorithm for solving the associated column generating subproblem.

This approach is motivated by the tendency in deterministic vehicle routing that column generation approaches are particularly competitive in the case of tight restrictions (e.g. tight capacity restrictions). Therefore, column generation seems to be a promising approach to solving those instances that have proven most difficult to existing algorithms.

Indeed, we expect our approach to be competitive for solving CVRPSD instances in which it is optimal to have only a few customers per route. It is worth emphasizing that this is exactly the instance type for which the L-shaped approach in (Laporte et al., 2002) seems relatively less effective. By the introduction of our approach, a wider range of problem instances is expected to be solvable, hence strengthening the practical ability of stochastic models.

3 Notation and model formulation

To formally describe the model let $G = (V, E)$ be an undirected graph, with vertex set $V = \{0, \dots, n\}$ and edge set E . Vertex 0 represents a depot, and each of the vertices in $V_c = \{1, \dots, n\}$ represents a customer. With each edge $\{i, j\}$ is associated a travel cost d_{ij} . Each customer i has a Poisson distributed demand with an expected value of $\lambda_i > 0$. We make the assumption that each λ_i is integer. The customers' demands are assumed to be independent. The vehicle capacity is denoted Q . For each customer i , we let q_i denote the stochastic variable describing the demand at customer i .

We define a feasible *elementary* route as a path $(0, v_1, \dots, v_k, 0)$ where v_1, \dots, v_k are k different customers whose total expected demand does not exceed Q . (As in (Laporte et al., 2002), we do not permit routes whose total expected demand exceeds Q , as such routes would systematically fail.) For any feasible elementary route r , let c_r denote its expected cost. Further, let \mathfrak{R}_e denote the set of all feasible elementary routes.

Let α_{ir} be a binary parameter describing the number of times route r visits customer i , and let x_r be a binary variable of value 1 if route r is chosen and 0 otherwise. This leads to the following Set Partitioning formulation:

$$(P_{org})$$

$$\min: \sum_{r \in \mathfrak{R}_e} c_r x_r \tag{1}$$

$$\text{s.t.: } \sum_{r \in \mathfrak{R}_e} \alpha_{ir} x_r = 1 \quad \forall i \in V_c \tag{2}$$

$$x_r \in \{0, 1\} \quad \forall r \in \mathfrak{R}_e \tag{3}$$

The objective (1) minimizes the total expected distribution cost. Constraints (2) ensure that each customer is contained in exactly one route, whereas the constraints (3) are the binary constraints on the decision variables.

The recourse that we have chosen is identical to strategy (A), with one exception. If the vehicle, on a route $(0, 1, \dots, k, 0)$, at some customer $i < k$ is exactly depleted, we assume that it continues, without an intermediate return to the depot, along the route until it at some customer $j \in \{i + 1, \dots, k\}$ encounters a positive demand (or reaches the depot). The failure cost corresponding to this recourse is $2d_{0j}$ (with $d_{00} = 0$).

We now consider in detail how the expected cost c_r of a route $r \in \mathfrak{R}_e$ can be calculated. For notational convenience, we assume that the route is $(0, 1, \dots, k, 0)$.

The total expected cost c_r can be decomposed into two elements. The first element (C_D) is the deterministic cost of following the path $(0, 1, \dots, k, 0)$, which must be done irrespective of actual demands. This cost element is simply $d_{01} + d_{k0} + \sum_{i=1}^{k-1} d_{i,i+1}$.

The second element (C_S) is the cost incurred by the stochastic nature of demands. Generally, C_S is the expected cost of travel to/from the depot as a result of failures, for the entire route as a whole.

We note that the extra traveling represented by C_S must be done *in addition* to traveling the path represented by C_D , so that C_D and C_S are additive. Indeed, we obtain $c_r = C_D + C_S$.

The complicating part of c_r is the calculation of C_S . As it turns out, however, we are able to separate C_S into k additive terms, one for each customer $1, \dots, k$. This is shown in the following.

Let $u > 0$ be an integer parameter describing the accumulated number of failures. The probability $F(i|u, Q)$ that the total actual demand on the path $(0, 1, \dots, i)$ does not exceed uQ can be calculated as follows:

$$F(i|u, Q) = Pr\left(\sum_{h=1}^i q_h \leq uQ\right). \quad (4)$$

Proposition 1 *The probability in (4) depends only on the total expected demand along the path to i , not on how this total expected demand is divided among the customers on the path.*

Proof: Our assumption that the demands are independent Poisson distributions implies that the expression $\sum_{h=1}^i q_h$ is in itself a Poisson distributed variable with an expected value of $\sum_{h=1}^i \lambda_h$. As such, the distribution of $\sum_{h=1}^i q_h$ does not depend on the individual expected demands, but only on $\sum_{h=1}^i \lambda_h$. \square

Proposition 1 is the key to our algorithm for solving the column generation subproblem in Subsection 4.2.

As a consequence of Proposition 1 we can for the remainder of this Section leave the assumption of the path $(0, \dots, i)$, and simply consider any elementary path from 0 to i on which the total expected demand is a given value, say, Λ . We let $Po(\Lambda)$ represent any variable which follows a Poisson distribution with an expected value of Λ . In addition, we define $F(\Lambda, U)$ as the probability that the total actual demand on a path, whose total expected demand is Λ , does not exceed U , where U is a positive integer:

$$F(\Lambda, U) = Pr(Po(\Lambda) \leq U). \quad (5)$$

We now turn to consider *failures* in more detail.

Definition 1 *For a given integer $u \geq 1$ and any elementary path $(0, \dots, j, i)$ we say that the u 'th failure occurs at customer i if and only if the total actual demand on the path $(0, \dots, j, i)$ exceeds uQ and the total actual demand on the path $(0, \dots, j)$ does not exceed uQ .*

Let Λ denote the total expected demand on the path $(0, \dots, i)$. The probability that the u 'th failure occurs at customer i is then $F(\Lambda - \lambda_i, uQ) - F(\Lambda, uQ)$.

For any elementary path $(0, \dots, i)$ with total expected demand Λ , the expected number of failures $\text{FAIL}(\Lambda, i)$ at customer i can then be calculated by summing over all possible failures:

$$\text{FAIL}(\Lambda, i) = \sum_{u=1}^{\infty} F(\Lambda - \lambda_i, uQ) - F(\Lambda, uQ), \quad (6)$$

which in practical computations is approximated by replacing ∞ with some sufficiently large number.

Since the failure cost $2d_{0i}$ is incurred for every failure at customer i , the *expected failure cost* $\text{EFC}(\Lambda, i)$ at customer i , for any elementary path $(0, \dots, i)$ with total expected demand Λ , can be calculated as follows:

$$\text{EFC}(\Lambda, i) = 2d_{0i}\text{FAIL}(\Lambda, i). \quad (7)$$

This result is originally obtained in (Dror et al., 1989). We note that our assumption of independent Poisson demands permits a tractable calculation of expected failure costs, in the light of Proposition 1.

4 Solution procedure

Our solution procedure is based on Dantzig-Wolfe decomposition. As is usual in Set Partitioning based approaches to vehicle routing, we make a few modifications to the formulation in order to obtain a more tractable problem.

4.1 The master problem

To obtain the master problem denoted P_M , we i) relax the integrality constraints (3), ii) change the partitioning constraints to covering constraints in order to obtain a smaller dual solution space, and iii) enlarge the set of feasible routes by permitting non-elementary paths. This leads to the following master problem:

$$\begin{aligned} (P_M) \\ \min: & \sum_{r \in \mathfrak{R}} c_r x_r \end{aligned} \quad (8)$$

$$\text{s.t.}: \sum_{r \in \mathfrak{R}} a_{ir} x_r \geq 1 \quad \forall i \in V_c \quad (9)$$

$$x_r \geq 0 \quad \forall r \in \mathfrak{R} \quad (10)$$

In P_M , the set \mathfrak{R} contains all feasible elementary routes as well as all non-elementary routes without 2-cycles (i - j - i) on which the total expected demand does not exceed Q . A customer i contributes λ_i

to the total expected demand on every arrival at i on the route. The coefficient a_{ir} equals the number of times that customer i is visited on route r .

We initialize P_M by n single-customer routes and solve this LP. By solving P_M a vector of dual prices π_1, \dots, π_n is obtained related to the constraint set (9), so that the dual price associated with customer i is π_i . The dual prices are used in the subproblem in the search for one or more columns with negative reduced cost. If such columns are identified, they are added to the LP, which is then reoptimized. The steps of column generation and LP reoptimization are repeated until no further columns with negative reduced cost exist. The current solution is then optimal for P_M .

If the LP solution is integer, it is optimal. (If not all inequalities (9) are satisfied with equality in an integer solution, we change the inequalities to equations, resolve the LP, and continue the iterative procedure.) If the LP solution is fractional we resort to branching in order to eventually obtain an integer solution. The overall *branch-and-price* algorithm is a variant of a branch-and-bound algorithm in which column generation is performed at each node in the branch-and-bound tree.

The two main ingredients: Column generation and Branching, respectively, are described in the following two subsections.

4.2 Column generation

In the case of deterministic demands, the column generation subproblem has frequently been solved by a dynamic programming algorithm which effectively solves a shortest path problem on a particular acyclic network. This applies to, e.g., the approaches in (Christofides et al., 1981; Hadjiconstantinou et al., 1995). The generated paths are invariably restricted to those without 2-cycles, which can be done without increasing the computational complexity using the idea in (Houck et al., 1980).

Our column generation approach is quite similar to this, but with the modification that expected failure costs must be taken into account. We note that the calculation of expected failure costs is not affected by permitting non-elementary routes. In the following we describe our construction of the network.

We let $G_S = (V_S, A_S)$ denote the graph that we construct for the purpose of solving the column generation subproblem. V_S contains $(n + 1)Q + 1$ vertices. Vertex $v(0, 0)$ is the beginning of any generated path. Each vertex $v(\Lambda, i)$, for $\Lambda = 1, \dots, Q$ and $i = 0, \dots, n$, represents all paths without 2-cycles from 0 to i on each of which the total expected demand equals Λ .

Beginning with an empty set A_S , we add arcs to A_S as follows:

1. For $i = 1, \dots, n$, add an arc from $v(0, 0)$ to $v(\lambda_i, i)$ and set its cost to $d_{0i} + \text{EFC}(\lambda_i, i) - \pi_i$.
2. For each ordered pair $i, j \in V_c$, $i \neq j$ and each $\Lambda = 1, \dots, Q - 1$, add an arc from $v(\Lambda, j)$ to $v(\Lambda + \lambda_i, i)$ (provided that $\Lambda + \lambda_i \leq Q$) and set its cost to $d_{ji} + \text{EFC}(\Lambda + \lambda_i, i) - \pi_i$.

3. For each $\Lambda = 1, \dots, Q$ and each $j = 1, \dots, n$, add an arc from $v(\Lambda, j)$ to $v(\Lambda, 0)$ and set its cost to d_{j0} .

The shortest path in G_S from $v(0, 0)$ to $v(\Lambda, 0)$, for some $\Lambda \in \{1, \dots, Q\}$, represents the route of minimum reduced cost on which the total expected demand equals Λ . As such, computing the shortest path from $v(0, 0)$ to $v(\Lambda, 0)$ for all $\Lambda = 1, \dots, Q$ effectively solves the column generation subproblem. This can be done in $O(n^2Q)$ time, also in the case that 2-cycles are prohibited.

4.3 The branching strategies

A traditional branching rule is to branch on single flow variables (as in (Hadjiconstantinou et al., 1995)). However, the restriction of forcing two customers to be adjacent is typically more restrictive than the complementary restriction of forcing two customers not to be adjacent. This often leads to unbalanced branch and bound trees.

In order to obtain branch-and-bound trees that are more balanced, we adopt the branching strategy proposed by G elinas et al. for the Vehicle Routing Problem with Time Windows and Backhauls (G elinas et al., 1995). They introduced a branching procedure based on the time windows. In each branching the parent problem is split into two restricted problems each containing a restricted time window for some customer.

In a similar way, we branch on the capacity resource. If a solution to the master problem is fractional, then either a customer is visited more than once on the same route or a customer is visited on several routes. This means that at least one customer i will be visited on two paths of the form $(0, \dots, i)$ with two different accumulated expected demands.

To illustrate this, consider a customer i which is visited twice, possibly on two different fractional routes, with different total expected demands on the path from the depot. These visits correspond to two vertices $v(\Lambda_1, i)$ and $v(\Lambda_2, i)$, respectively, in G_S (see Subsection 4.2). As $\Lambda_1 \neq \Lambda_2$, this fractional solution can be eliminated by choosing a threshold δ between Λ_1 and Λ_2 , and creating two restricted subproblems as follows. In the first problem, we permit only vertices $v(\Lambda, i)$ with $\Lambda \leq \delta$ to be visited on any path in G_S , and in the second problem, only vertices $v(\Lambda, i)$ with $\Lambda > \delta$ are permitted to be visited. In the remainder of the article this strategy is referred to as A , whereas the original flow variable branching strategy is referred to as B . Both strategies have been included in the computational testing in order to compare their relative performance.

5 Computational results

The algorithm was tested on several instances originally developed for the CVRP. The data for these CVRP instances are available at www.branchandcut.org. To convert each CVRP instance into a stochastic instance, the original deterministic demand values were regarded as *expected* values of the stochastic (Poisson distributed) demands in the corresponding CVRPSD instance. For all instances

the capacity and demands are divided by their largest common denominator. This decreases the time consumed by the algorithm. However, this must be taken into account when calculating the expected cost. The instances chosen include all instances of the Augerat test sets A and P, and the Christofides and Eilon test set with up to 60 customers. For the latter; instances E-23-3 and E-30-3 are omitted due to their high vehicle capacity, which leads to computational inaccuracies regarding the calculation of the penalty costs. No test results for instances with more than 60 customers are included since preliminary testing showed that none of these were solvable.

Each instance was run on a Pentium Centrino 1500Mhz computer with 480MB of RAM. For each instance we set a time limit of 1200 seconds. For each instance, data were recorded when the root node was solved, and when the algorithm terminated, either due to timeout or optimality. These data are given in Table 1. Columns 2 – 5 show the data collected after solving the root node, whereas columns 6 – 14 show the data collected after the algorithm had terminated. Each column (1-15) will now be discussed in detail.

Column 1 shows the name of the instance. Column 2 shows the objective value after solving the root node. Column 3 shows the number of columns generated at the root node, and column 4 shows the number of times the master problem has been solved. Column 5 shows the sum of the decision variables after having solved the root node. We note that this sum may be integer also for a fractional solution. Indeed, an integer solution is only obtained at the root for the instance P-23-8 in Table 1.

Columns 6 and 7 show the total time spent by the algorithm in seconds, when using branching strategy *A* and *B*, respectively (the sign "##" indicates that the algorithm has reached the time limit). Columns 8 and 9 show the number of nodes solved in the branching tree for each of the two branching strategies. Columns 10 and 11 present lower bounds on the optimal objective value, for each of the alternative branching strategies. If a lower bound is marked with an upper case (*) the solution is optimal. Column 12 shows an upper bound on the optimal objective value. This bound can originate from three different sources: 1) The optimal solution 2) The best integer solution found during branching 3) The best obtainable expected cost given the solution of the CVRP Counterpart (column 14). Column 13 shows the sum of the decision variables, hence also the number of routes in the optimal solution. Column 14 shows the lowest expected cost if the stochastic demand is not included in the route planning; this is calculated on the basis of the optimal solution of its CVRP counterpart. For each route in the optimal solution of the CVRP instance, the minimum expected route cost is calculated, given the customers and their sequence. The expected solution cost is then the sum of the minimum expected route costs for all routes in the CVRP solution. (If alternative optimal CVRP solutions are known, the alternative resulting in the minimum expected cost is used.) Finally column 15 shows the ratio between the objective value found at the root node and the best known upper bound ($\frac{(LB_{root})}{(UB)}$).

From Table 1 it can be seen that our algorithm (with a few exceptions) solves all problem instances with up to 40 customers either to optimality or within 1 percent of the optimal solution. On several instances, the number of routes in the optimal solution exceeds the minimum possible for serving all customers (the minimum possible number of routes is the number after the second '-' in the name).

For the instance P-55-15 this is particular evident in that three extra routes are formed.

Comparing the best expected solution cost given a deterministic route planning (Column 14) to the optimal expected solution cost, it is clear that neglecting the stochastic nature of demands during the route planning can incur large cost increases. This is especially evident for the instance E-33-4. For this particular instance the extra expected cost is more than 15 percent of the optimal solution cost. This proves that deterministic CVRP models applied to a situation containing stochastic demands could incur larger actual costs than a stochastic model applied to the same situation.

The largest instance successfully solved by the algorithm was an instance with 60 customers and 16 routes. This is considerable progress when comparing to the results obtained in (Laporte et al., 2002) where only instances involving up to 4 vehicles were solved to optimality. Furthermore, the solved problems with more than 50 customers are all characterized by having a small average number of customers on each route. In general the algorithm seems to perform better when the number of required vehicles is large relative to the number of customers. This is contrary to the results in (Laporte et al., 2002) where the computational effort required increases sharply with the number of routes. This leads to the conclusion that our algorithm broadens the range of VRPSD instances solvable, hence strengthens the applicability of VRPSD models.

The lower bound obtained at the root is very close to the optimal solution for all instances solved. Despite this fact the number of nodes in the branch-and-bound tree can be very large. This is particularly clear for the instance P-55-15. For this instance the solution at the root was within half a percent of the optimal solution, even so the number of nodes in the tree exceeded twenty thousand, when using strategy A and thirty thousand when using strategy B. This can be partly explained by the nature of the failure costs. Regardless of the direction in which a route is traveled the failure cost tends to be very low for the customers at the beginning of the route until the accumulated expected demand approaches Q . From here the failure cost increases rapidly. The route's total failure cost may change only slightly when changing the direction of the route. This element of near-symmetry seems to make the branching less effective. However, when considering the number of problems solved by the each of the two branching strategies, strategy A solves more problems than does strategy B. Furthermore, when comparing the instances solved by both branching strategies, strategy A tends to search fewer nodes before reaching optimum than does strategy B. These two facts indicate that strategy A in general performs better than strategy B.

6 Concluding remarks

In this article we have introduced a new branch-and-price algorithm for solving the Capacitated Vehicle Routing Problem with Stochastic Demands (CVRPSD), where the objective is to minimize the expected solution cost. We show that under the assumption of independently Poisson distributed demands the column generation problem can be formulated as a shortest path problem on an acyclic network and solved by dynamic programming.

The algorithm was tested on a large number of CVRP instances that were converted into stochastic

instances. The algorithm showed good results by solving almost all instances with up to around forty customers, and by solving a few instances with over fifty customers and more than 10 vehicles. This is a significant progress compared to previous work done on the CVRPSD. Moreover, we proposed a new branching strategy for the VRPSD based on accumulated demand, which shows some potential, when compared to a well known flow-variable based branching strategy.

7 References

- Ball, M., Magnanti, T., Monma, C., and Nemhauser, G. Network Routing. Elsevier Science Publisher B.V., North-Holland; 1995.
- Bastian, C. and Kan, A. R. The stochastic vehicle routing problem revisited. European Journal of Operational Research; 1992;56;407–411.
- Beasley, J. Fixed routes. Journal of the Operational Research Society; 1984;35;49–55.
- Bertsimas, D. A vehicle routing problem with stochastic demand. Operations Research; 1992;40;574–585.
- Christofides, N., Mingozzi, A., and Toth, P. Exact algorithm for the vehicle routing problem, based on spanning tree and shortest path relaxations. Mathematical Programming; 1981;20;255–282.
- Clarke, G. and Wright, J. W. Scheduling of vehicles from a central depot to a number of delivery points. Operations Research; 1964;12;568–581.
- Dror, M., Laporte, G., and Trudeau, P. Vehicle routing with stochastic demands: properties and solution frameworks. Transportation Science; 1989;23;166–176.
- Dror, M. and Trudeau, P. Stochastic vehicle routing with modified savings algorithm. European Journal of Operational Research; 1986;23;228–235.
- Gélinas, S., Desrochers, M., Desrosiers, J., and Solomon, M. A new branching strategy for time constrained routing problems with application to backhauling. Annals of Operations Research; 1995;61;91–109.
- Gendreau, M., Laporte, G., and Séguin, R. An exact algorithm for the vehicle routing problem with stochastic demands and customers. Transportation Science; 1995;29;143–155.
- Gendreau, M., Laporte, G., and Séguin, R. Stochastic vehicle routing. European Journal of Opera-

tional Research; 1996;88;3–12.

- Hadjiconstantinou, E., Christofides, N., and Mingozzi, A. Exact algorithms for the vehicle routing problem based on q -paths and k -shortest paths relaxation. *Annals of Operations Research*; 1995;61;21–43.

- Haughton, M. Route reoptimization's impact on delivery efficiency. *Transportation Research Part e*; 2002;38;53–63.

- Houck, D., Picard, J., Queyranne, M., and Vemuganti, R. The travelling salesman problem as a constrained shortest path problem: theory and computational experience. *Opsearch*; 1980;17;93–109.

- Laporte, G. and Louveaux, F. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*; 1993;13;133–142.

- Laporte, G. and Louveaux, F. Solving stochastic routing problems with the integer L-shaped method. In Crainic, T. and Laporte, G. (Eds), *Fleet management and logistics*, Kluwer Academic Publishers, United states; 1998. p. 159–167.

- Laporte, G., Louveaux, F., and Hamme, L. An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research*; 2002;50;415–423.

- Savelsbergh, M. and Goetschalckx, M. A comparison of the efficiency of fixed versus variable vehicle routes. *Journal of Business Logistics*; 1995;16;163–187.

- Stewart, W. and Golden, B. Stochastic vehicle routing: a comprehensive approach. *European Journal of Operational Research*; 1982;14;371–385.

- Tillman, F. The multiple terminal delivery problem with probabilistic demands. *Transportation Science*; 1969;3;192–204.

- Toth, P. and Vigo, D., editors. *The vehicle routing problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia; 2002.

Table 1: Computational results

Inst.	LB Root	Cols.	PM	Routes Root	Time A	Time B	Nodes A	Nodes B	LB A	LB B	UB	Routes Opt	Best Det	$\frac{LB_{Root}}{UB}$
A-32-5	817.31	1221	67	5	282	##	2467	13799	853.6*	842.32	853.6	5	890.13	0.96
A-33-5	700.01	962	44	5	8	9	117	107	704.2*	704.2*	704.2	5	722.99	0.99
A-33-6	775	832	43	6.05	49	##	909	26195	793.9*	789.65	793.9	6	816.58	0.98
A-34-5	803.26	1212	62	5.31	##	##	16059	15155	825.26	823.17	827.87	-	839.95	0.97
A-36-5	838.83	1641	113	5	##	##	8035	8093	852.09	851.01	907.55	-	907.55	0.92
A-37-5	687.4	1758	95	5	##	##	9191	8773	707.54	702.99	708.34	-	709.83	0.97
A-37-6	1007.98	1237	67	6.48	##	##	16195	14965	1030.44	1021.83	1030.75	-	1069.32	0.98
A-38-5	739.19	1207	55	5.47	##	##	14499	13865	761.12	755.05	778.09	-	831.99	0.95
A-39-5	866.92	1722	73	6.07	3	4	9	23	869.18*	869.18*	869.18	6	903.26	1
A-39-6	850.09	1462	84	6.04	279	##	2431	11289	876.6*	872.28	876.6	6	960.81	0.97
A-44-7	1007.55	1871	59	6.43	##	##	11077	10053	1021.29	1017.27	1025.48	-	1047.18	0.98
A-45-6	984.38	1861	95	6.75	##	##	9313	10285	1006.88	1001.01	1096.19	-	1096.19	0.9
A-45-7	1254.23	1954	84	7.13	882	##	5365	9553	1264.83*	1263.09	1264.83	7	1302.2	0.99
A-46-7	986.39	1951	84	7	##	##	8149	9257	999.87	997.25	1002.41	-	1069.66	0.98
A-48-7	1160.52	2482	91	7.13	##	##	7729	8221	1180.22	1177.8	1248.27	-	1248.27	0.93
A-53-7	1093.64	3273	128	7.69	##	##	5385	5321	1109.34	1106.67	1180.1	-	1180.1	0.93
A-54-7	1262.49	2636	98	7.44	##	##	5925	5741	1279.93	1272.65	1342.87	-	1342.87	0.94
A-55-9	1148.4	1881	55	9.76	##	##	9979	9597	1173.56	1159.82	1264.18	-	1264.18	0.91
A-60-9	1489.82	3043	108	9.2	##	##	6889	5737	1503.65	1498.23	1608.4	-	1608.4	0.93
E-22-4	409.86	460	34	4.2	1	1	9	13	411.57*	411.57*	411.57	4	411.73	1
E-33-4	844.35	3030	148	4	86	##	73	1161	850.27*	850.07	850.27	4	850.27	0.99
E-51-5	538.75	3290	138	5.48	##	##	2771	2643	544.63	543.67	553.26	-	553.26	0.97
P-16-8	511.27	79	8	8.5	0	0	17	29	512.82*	512.82*	512.82	8	512.82	1
P-19-2	210.9	620	63	2.17	153	154	1815	1529	224.06*	224.06*	224.06	3	229.68	0.94
P-20-2	221.11	871	88	2.18	352	496	3191	3819	233.05*	233.05*	233.05	2	233.05	0.95
P-21-2	217.75	1037	100	2.14	5	7	27	45	218.96*	218.96*	218.96	2	218.96	0.99
P-22-2	223.67	1129	103	2.21	219	329	1335	1903	231.26*	231.26*	231.26	2	231.26	0.97
P-22-8	677.97	205	18	8.92	0	0	65	53	681.06*	681.06*	681.06	9	707.8	1
P-23-8	619.52	22	19	9	1	1	0	0	619.52*	619.52*	619.52	9	662.31	1
P-40-5	471.47	1905	80	5	9	6	35	29	472.5*	472.5*	472.5	5	475.45	1
P-45-5	519.03	2153	101	5.14	##	##	4561	4265	527.77	526.63	546.05	-	546.05	0.95
P-50-7	573.66	2016	67	7.13	##	##	5311	5273	581.19	579.97	606.41	-	606.41	0.95
P-50-8	659.67	1522	55	8.78	##	##	10409	10359	666.9	666.86	724.69	-	724.69	0.91
P-50-10	750.27	1320	50	11	##	##	18901	18529	756.52	757.44	792.2	-	792.2	0.95
P-51-10	802.58	1436	57	10.99	430	995	5431	11177	809.7*	809.7*	809.7	11	859.24	0.99
P-55-7	582.12	2782	102	7	##	##	2441	3303	585.47	585.46	616.44	-	616.44	0.94
P-55-8	599.67	2457	87	7.31	##	##	2365	3945	603.31	603.69	2438†	-		0.25
P-55-10	734.69	1747	56	10.17	##	##	11257	11525	740.02	741.25	797.21	-	797.21	0.92
P-55-15	1062.67	867	21	17.18	792	1756	20027	40873	1068.05*	1068.05*	1068.05	18	1191.34	0.99
P-60-10	793.71	2102	59	10.58	##	##	4969	6961	798.63	799.58	831.24	-	831.24	0.95
P-60-15	1080.85	1175	29	16.08	##	1348	18347	19029	1085.12	1085.49*	1085.49	16	1133.3	1

†upper bound = total expected cost if all customers were serviced on separate routes

Working Papers from Logistics/SCM Research Group

- L-2006-04 Christian H. Christiansen & Jens Lysgaard: A column generation approach to the capacitated vehicle routing problem with stochastic demands.
- L-2006-03 Christian Larsen: Computation of order and volume fill rates for a base stock inventory control system with heterogeneous demand to investigate which customer class gets the best service.
- L-2006-02 Søren Glud Johansen & Anders Thorstenson: Note: Optimal base-stock policy for the inventory system with periodic review, backorders and sequential lead times.
- L-2006-01 Christian Larsen & Anders Thorstenson: A comparison between the order and the volume fill rates for a base-stock inventory control system under a compound renewal demand process.
- L-2005-02 Michael M. Sørensen: Polyhedral computations for the simple graph partitioning problem.
- L-2005-01 Ole Mortensen: Transportkoncepter og IT-støtte: et undersøgelsesoplæg og nogle foreløbige resultater.
- L-2004-05 Lars Relund Nielsen, Daniele Pretolani & Kim Allan Andersen: K shortest paths in stochastic time-dependent networks.
- L-2004-04 Lars Relund Nielsen, Daniele Pretolani & Kim Allan Andersen: Finding the K shortest hyperpaths using reoptimization.
- L-2004-03 Søren Glud Johansen & Anders Thorstenson: The (r,q) policy for the lost-sales inventory system when more than one order may be outstanding.
- L-2004-02 Erland Hejn Nielsen: Streams of events and performance of queuing systems: The basic anatomy of arrival/departure processes, when focus is set on autocorrelation.
- L-2004-01 Jens Lysgaard: Reachability cuts for the vehicle routing problem with time windows.



Handelshøjskolen i Århus

Aarhus
School of Business

ISBN 87-7882-121-5

Department of Business Studies

Aarhus School of Business
Fuglesangs Allé 4
DK-8210 Aarhus V - Denmark

Tel. +45 89 48 66 88
Fax +45 86 15 01 88

www.asb.dk