



# On the vehicle routing problem with stochastic demands and duration constraints: formulations and a hybrid metaheuristic

Jorge E. Mendoza, Louis-Martin Rousseau, Juan Villegas

## ► To cite this version:

Jorge E. Mendoza, Louis-Martin Rousseau, Juan Villegas. On the vehicle routing problem with stochastic demands and duration constraints: formulations and a hybrid metaheuristic. 10th Metaheuristics International Conference (MIC), Aug 2013, Singapore, Singapore. pp.1 - 1. hal-00874293

HAL Id: hal-00874293

<https://hal.archives-ouvertes.fr/hal-00874293>

Submitted on 16 Oct 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# On the vehicle routing problem with stochastic demands and duration constraints: formulations and a hybrid metaheuristic approach

Jorge E. Mendoza<sup>1</sup>, Louis-Martin Rousseau<sup>2</sup>, Juan G. Villegas<sup>3</sup>

<sup>1</sup> LUNAM Université - LISA (EA 4094 CNRS)  
Université Catholique de l'Ouest  
jorge.mendoza@uco.fr

<sup>2</sup> CIRRELT - École Polytechnique de Montréal  
louis-martin.rousseau@cirrelt.ca

<sup>3</sup> Departamento de Ingeniería Industrial  
Universidad de Antioquia  
jvillega@udea.edu.co

## Abstract

The vehicle routing problem with stochastic demands (VRPSD) consists in designing transportation routes of minimal expected cost to satisfy a set of customers with random demands of known probability distributions. In this research we present two strategies to deal with route duration constraints in the VRPSD. To solve the resulting problem, we proposed a greedy randomized adaptive search procedure (GRASP) with a post optimization procedure. The GRASP component uses a set of randomized route-first, cluster-second heuristics to generate starting solutions and a variable neighborhood descent (VND) procedure to carry on the local search phase. The post optimizer selects the best possible routes to assemble the final solution from the set of all routes found in the local optima reached by the GRASP. We discuss extensive computational experiments analysing the cost of considering route duration constraints on the VRPSD. In addition, we report state-of-the-art solutions for a established set of benchmarks for the classical VRPSD.

## 1 Introduction

The vehicle routing problem with stochastic demands (VRPSD) can be defined on a complete and undirected graph  $G = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = \{0, \dots, n\}$  is the vertex set and  $\mathcal{E} = \{(v, u) : v, u \in \mathcal{V}, v \neq u\}$  is the edge set. Vertices  $v = 1, \dots, n$  represent the customers and vertex  $v = 0$  represents the depot. A weight  $t_e$  is associated with edge  $e = (v, u) = (u, v) \in \mathcal{E}$ , and it represents the travel time between vertices  $v$  and  $u$ . Each customer  $v$  has a random demand  $\xi_v$  for a given product. Customer demands are met using an unlimited fleet of homogeneous vehicles with capacity  $Q$  located at the depot. The exact quantity demanded by each customer is not known until the vehicle arrives at the customer location. It is assumed, however, that each customer's demand follows an independent and known probability distribution and that all demand realizations (actual quantities) are nonnegative and less than the capacity of the vehicle.

The VRPSD is classically formulated as a two-stage stochastic program. In the first stage, a set  $\mathcal{R}$  of *planned* routes is designed. Each route  $r \in \mathcal{R}$  is a sequence of vertices  $r = (0, v_1, \dots, v_i, \dots, v_{n_r}, 0)$ , where  $v_i \in \mathcal{V} \setminus \{0\}$  and  $n_r$  is the number of customers serviced by the route. During the planning phase, each route is designed so that the total expected load does not exceed the capacity of the vehicle (i.e.,  $\sum_{v \in r \setminus 0} E[\xi_v] \leq Q \forall r \in \mathcal{R}$ ) and every customer is visited by exactly one route. In the second stage, each route is executed until a *route failure* occurs, that is, the capacity of the vehicle is exceeded. A *recourse action* is then applied to recover the feasibility of the failing route. The recourse action is defined as a return to the depot to reload the vehicle, followed by a trip back to the customer location to complete the service. The route is then resumed from that point as originally planned. The second-stage solution is then the actual set of routes traveled by the vehicles. The problem consists in determining during the first stage the set of planned routes  $\mathcal{R}$  that minimizes the total expected travel time of the

routes in the second-stage solution  $E[\tilde{T}(\mathcal{R})] = \sum_{r \in \mathcal{R}} E[\tilde{T}_r]$ , where  $\tilde{T}_r$  is the (random) duration of route  $r$ , and  $E[\cdot]$  denotes the expected value.

The VRPSD with duration constraints (VRPSDDC) extends the problem to include a maximum route duration  $T$  for each planned route  $r \in \mathcal{R}$ . Different strategies can be applied to deal with duration constraints. For instance, [5] enforces the duration constraint on the expected duration of each planned route (i.e.,  $E[\tilde{T}_r] \leq T \forall r \in \mathcal{R}$ ); [1] imposes a hard constraint on the second-stage duration of the route, meaning that planned routes should verify the constraint for any possible vector of demand realizations; and [4] penalizes violations of the duration constraint in an additional objective function, driver remuneration, and solve the problem as a multiobjective optimization problem with posterior articulation of preferences. In this research we introduce two alternative strategies.

Note that since all demand realizations are less than the capacity of the vehicle, the maximum number of failures in a route is  $n_r - 1$ . Consequently, the total duration of a route  $\tilde{T}_r$  follows a discrete distribution with  $2^{n_r - 1}$  possible outcomes. Let us refer to each possible outcome for  $\tilde{T}_r$  as a *duration profile*. Let us also denote as  $\mathcal{P}_r$  the set of all duration profiles for route  $r$  and as  $T_r^p$  the total duration of the route if profile  $p \in \mathcal{P}_r$  is observed. Note also that if demands follow additive probability functions, as it is classically assumed in the VRPSD literature, the probability of having a route failure while visiting each customer of a given route can be analytically computed (see for instance [3] for details on how to calculate such probability). Knowing these *failure probabilities*, allows us to compute the probability  $Pr(p)$  of observing each duration profile  $p \in \mathcal{P}_r$ . We propose a tractable algorithm to compute  $\mathcal{P}_r$  and  $Pr(p) \forall p \in \mathcal{P}_r$ .

Our first strategy, hereafter referred to as CC, consists in enforcing the duration constraint as a chance constraint, meaning that  $Pr(\tilde{T}_r \leq T) \geq 1 - \beta \forall r \in \mathcal{R}$ , where  $\beta \in [0, 1]$  is a preset threshold. Our second strategy, henceforth DR, consists on implementing a recourse action to recover from violations of the duration constraint. The considered recourse action is simply to pay the drivers overtime. Therefore, during the first stage, we minimize  $E[C] = \sum_{r \in \mathcal{R}} \left[ E[\tilde{T}_r] + \sum_{p \in \mathcal{P}_r | T_r^p > T} \phi((T_r^p - T) \times Pr(p)) \right]$ , where  $\phi(\cdot)$  is a function to compute the cost of the expected overtime. Note that in our approaches  $\beta$  and  $\phi(\cdot)$  may be adapted to capture the decision maker's aversion towards violations of the duration constraint.

## 2 GRASP with heuristic concentration

---

### Algorithm 1 GRASP+HC: General structure

---

```

1: function GRASPHC( $G, \mathcal{H}, K$ )
2:    $\Omega \leftarrow \emptyset, k \leftarrow 1$ 
3:   while  $k \leq K$  do
4:     for  $h \in \mathcal{H}$  do
5:        $tsp^k \leftarrow \text{startSolution}(h, G)$ 
6:        $s^k \leftarrow \text{split}(G, tsp^k)$ 
7:        $s^k \leftarrow \text{vnd}(G, s^k)$ 
8:        $s^* \leftarrow \text{update}(s^k, s^*)$ 
9:       for  $r \in s^k$  do
10:         $\Omega \leftarrow \Omega \cup r$ 
11:       end for
12:        $k \leftarrow k + 1$ 
13:     end for
14:   end while
15:    $\mathcal{R} \leftarrow \text{setPartitioning}(G, \Omega, s^*)$ 
16:   return  $\mathcal{R}$ 
17: end function

```

---

To solve our two formulations for the VRPSDDC, namely CC and DR, we developed a greedy randomized adaptive search procedure (GRASP) enhanced with heuristic concentration. Algorithm 1 describes the proposed approach. At each GRASP iteration  $k$  (lines 3–14) the algorithm selects a randomized TSP heuristic  $h$  from a set  $\mathcal{H}$  and uses it to build a giant TSP tour  $tsp^k$  visiting all customers (line 5). Then, the algorithm uses an adaptation of the s-split procedure for the VRPSD [3] to optimally partition  $tsp^k$  into a set of feasible routes that make up a starting solution  $s^k$  (line 6). Next, the algorithm uses the

first-improvement versions of the re-locate and 2-opt neighborhoods to perform a variable neighborhood descent (VND) from the starting solution  $s^k$  (line 7). At the end of iteration  $k$ , the algorithm updates the best solution  $s^*$  (line 8) and adds the routes of the local optimum (i.e.,  $s^k$ ) to a set  $\Omega$  (lines 9–11). After  $K$  iterations the GRASP stops and the heuristic concentration (HC) takes place. In this phase, our method uses a commercial optimizer to solve a set-partitioning formulation over the set of routes  $\Omega$  (line 15). To speed up the HC phase, the algorithm uses the objective function of the best solution found by the GRASP as an initial upper bound for the set-partitioning problem. Note that the concrete implementations of `split()`, `vnd()`, and `select()` slightly vary depending on the formulation (i.e., CC or DR) being solved while the implementations of `startSolution()` and `setPartitioning()` remain unchanged.

### 3 Computational experiments

To the best of our knowledge, there are not publicly-available benchmarks for the VRPSDDC. To build a set of instances, we first ran 10 times our GRASP+HC on each of the 40 Christiansen-and-Lysgaard VRPSD benchmarks. We obtained solutions with a maximum gap of 0.03% with respect to the best known solutions (BKSs) which are 38 optimal solutions reported in [2] and 2 heuristic solutions reported in [3]. From each of the 40 VRPSD instances we built a VRPSDDC instance by setting  $T = \max\{E[\tilde{T}_r] | r \in \mathcal{R}\}$ , where  $\mathcal{R}$  is the best solution found for the original VRPSD instance. Note that the solutions we found for the original VRPSD instances are the same we would have found if we solve the new VRPSDDC instances dealing with duration constraints as proposed in [5], that is, enforcing  $E[\tilde{T}] \leq T \forall r \in \mathcal{R}$ . For the sake of brevity we will henceforth refer to the latter formulation as ED.

In our first experiment, we ran GRASP+HC on CC setting  $\beta = 0.05$ . The results show that, as expected, CC leads to solutions that are more robust to violation of the duration constraint than those obtained by ED. For instance, on CC solutions, the maximum probability of failing the duration constraint for a single route is 4.94% while the same figure is 44.63% on ED solutions. Our data also suggest that this improvement in the robustness is obtained with only small increments on the expected cost of the solutions (an average 3.6%). In our second experiment we analyse the behavior of ED solutions when there are costs associated to violations of the duration constraints. We evaluated a posteriori our ED solutions using objective function  $E[C]$  with three different  $\phi(\cdot)$  functions (linear, piece-wise linear, and quadratic) and compared the results with those delivered by GRASP+HC running on DR (with the same  $\phi(\cdot)$  functions). The results show that ED leads to solutions that not only have high probabilities of violating the duration constraints (as concluded in our first experiment) but that also become expensive when there are cost associated to the magnitude of those violations.

### References

- [1] A. Erera, J. C. Morales, and M. Savelsbergh. The vehicle routing problem with stochastic demands and duration constraints. *Transportation Science*, 44(4):474–492, 2010.
- [2] C. Gauvin. Un algorithme de génération de colonnes pour le problème de tournées de véhicules avec demandes stochastiques. Master’s thesis, École Polytechnique de Montréal, 2012.
- [3] J. E. Mendoza and J. G. Villegas. A multi-space sampling heuristic for the vehicle routing problem with stochastic demands. *Optimization Letters*, 2012. DOI 10.1007/s11590-012-0555-8.
- [4] K. C. Tan, C. Y. Cheong, and C. K. Goh. Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. *European Journal of Operational Research*, 177(2):813–839, 2007.
- [5] W. H. Yang, K. Mathur, and R. Ballou. Stochastic vehicle routing with restocking. *Transportation Science*, 34(1):99–112, 2000.