Revista Educación en Ingeniería
ISSN 1900-8260
Enero a Junio de 2014, Vol. 9, N°. 17, pp. 108-117 • © 2014 ACOFI • http://www.educacioneningenieria.org
Recibido: 22/08/2013 • Aprobado: 29/01/2014

COMPARACIÓN ENTRE LOS ENFOQUES DE PROGRAMACIÓN DINÁMICA Y OPTIMIZACIÓN POR SIMULACIÓN PARA LA SOLUCIÓN DEL VRPSD CON DESCARGA PREVENTIVA

COMPARING DYNAMIC PROGRAMMING AND SIMULATION-OPTIMIZATION APPROACHES FOR SOLVING THE SINGLE VRPSD WITH RESTOCKING

Silvia Adriana Galván Núñez, Carlos Eduardo Díaz Bohórquez y Henry Lamos Díaz
Universidad Industrial de Santander, Bucaramanga (Colombia)

Resumen

El problema de ruteo de vehículos con demandas estocásticas (VRPSD) se propone encontrar la mejor ruta vehicular con el mínimo costo esperado. Este trabajo presenta dos enfoques para evaluar la función objetivo del VRPSD con descarga preventiva usando algoritmos genéticos (GA). El primer enfoque se basa en programación dinámica (DP) usando una recursión en reversa. El segundo enfoque está asociado a un modelo de simulación en el que se utilizó simulación Monte Carlo. Los enfoques presentados se compararon con el propósito de establecer cuál ofrece mejores estimaciones de los valores de la función objetivo. Los resultados computacionales muestran que aunque el enfoque DP provee mejores estimaciones en términos de valores de función objetivo que la simulación Monte Carlo, éste brinda resultados cercanos a los de DP con una reducción significativa del tiempo computacional en comparación de DP.

Palabras claves: VRPSD; programación dinámica; simulación Monte Carlo; algoritmos genéticos

Abstract

The single Vehicle Routing Problem with Stochastic Demands (VRPSD) looks to find the best vehicle route with the minimum expected cost. This paper presents two approaches to evaluate the objective function of the VRPSD with preventive restocking using Genetic Algorithms (GA). The first approach is dynamic programming (DP) using a recursion that moves backward from the last node of the sequence. The second approach is based on a simulation model in which Monte Carlo simulation is implemented for this purpose. The presented approaches were compared in order to establish which one offers a better estimation of the objective function values. The computational results show that although the DP approach provides better estimations in terms of objective function values than Monte Carlo simulation, the second approach gives results close to the DP and with a significant reduction of the computational time with regard to DP.

Keywords: VRPSD; Dynamic Programming; Monte Carlo simulation; Genetic Algorithms

Introduction

The deterministic Vehicle Routing Problem (VRP) is a combinatorial optimization problem classified in the class NP-Hard (Toth & Vigo, 2002). Formally, the VRP is defined as a directed graph $G = (V; A)$ where $V = \{v_0, v_1, \dots, v_n\}$ is the vertex set (v_0 denotes the depot, vertex $v_1, \dots, v_n$ denote the customers), and $A = \{(v_i, v_j): v_i, v_j \in V, i < j\}$ is an arcs set representing the connection between the depot and the customers. In addition, there exists a homogeneous vehicle fleet of size m with known capacity Q that have to serve the customers that are located in a certain geographical area. The associated cost of visiting the customers is represented by a symmetric matrix cost $C = (c_{ij})$. The VRP consists on design m routes with the lowest cost satisfying: (i) each route starts and ends at the depot, (ii) each customer is visited once by a single vehicle, and (iii) the total demand of a route does not exceed Q. The VRP has been used widely real-life situations in which the customers' demands are known. However, there are several specific situations where the vehicle routing problem turns away from the idealized model to incorporate more flexible models that respond to the dynamic and complex environment of the existing transportation systems, considering then the route design under uncertainty of some or all parameters, these models are called Stochastic Vehicle Routing Problem (SVRP) (Bianchi, Manfrin, et al., 2005).

The VRPSD real-life situations arise due to delivery or collection of goods in which the company responsible for the distribution has customers with uncertain demand. Under this consideration, the demand is only revealed at the moment when the vehicle visits the customer. In the deterministic case, the routes are planned so that vehicles have enough capacity to meet the demands of the customers given some pre-established routes. If the demands are stochastic, the concept of "pre-established routes" has a different interpretation and it is necessary to use decision rules or routing policies that reinterpret the above concept (Manfrin, 2004). In literature, three main approaches have been studied regarding the type of routing policy that takes place (N Secomandi, 2000): a) A priori (static policy, online). b) Dynamic (re-optimization policy, online). c) Mixed (preventive restocking policy).

The first policy is called static or a priori which is part of the two-stage stochastic processes. The first stage determines a sequence of customers (called a priori route) to be visited in that order for a vehicle and in the second stage the route runs as defined. If the route fails, a recursive action is taken. (Bertsimas, 1992) proposed cyclic heuristics based on the perspective of the worst case when the probability distribution is identical customers. (Gendreau, Laporte, Séguin, & Seguin, 1995) proposed an exact solution for the VRPSD using the integer L-Shaped method, in (Tan, Cheong, & Goh, 2007) a multi-objective evolutionary algorithm and simulation method for calculating the cost of the route was used. (Rei, Gendreau, & Soriano, 2010) proposed an exact method for solving the VRPSD where the objective function evaluation is done using Monte Carlo simulation. (Laporte, Louveaux, & Van Hamme, 2001) and (Hjorring & Holt, 1999) the L-Shaped method was implemented. (Mendoza, Castanier, Guéret, Medaglia, & Velasco, 2010) and (Mendoza, Castanier, & Guéret, 2011) the VRPSD with multiple compartments was solved using memetic algorithms and evolutionary and constructive heuristics respectively.

In second place there are dynamic policies which are formulated mathematically as a multi-stage stochastic problem. (N Secomandi, 2000) developed a neuro dynamic programming methodology. (Christiansen & Lysgaard, 2007) solved the Vehicle Routing Problem with Capacity and Stochastic Demands (CVRPSD) through the exact branch-and-price method. (Novoa &

Storer, 2009) and (Nicola Secomandi, 2001) developed a rollout algorithm for solving the VRPSD with a single vehicle. (Novoa & Storer, 2009) implemented a rollout algorithm and Monte Carlo simulation are used. (Ak & Erera, 2007) presented the paired-vehicle strategy for VRPSD using Tabu Search.

The last policy is the preventive restocking policy. This policy combines a priori and dynamic framing itself a two-stage stochastic process, in which the vehicle follows a route in the first stage priori yet enabled with state dependent rules that allow replenishment anticipated in the second stage. (Yang, Mathur, & Ballou, 2000) solved the VRPSD with preventive restocking for the case of one and multiple vehicles, developing the heuristics route first-cluster next and cluster first-cluster next. (Bianchi, Birattari, et al., 2005) tested the impact on the development of 18 simulated annealing metaheuristics, Tabu search, ant colony and evolutionary algorithms, using the distance of the route a priori as a quick approximation of the objective function. (Ismail & Irhamah, 2008) the hybrid Genetic algorithms (GA) and Tabu search were implemented for a real application related to garbage collection in a residential area of Malaysia. (Tripathi, Kuriger, & Wan, 2009) incorporated an element for calculating the objective function based on the optimization technique by simulation. (Shanmugam, Ganesan, & Vanathi, 2011) and (Galván, Arias, & Lamos, 2013) solved the VRPSD using the metaheuristics PSO and PSO with evolutionary operators respectively.

The rest of paper is organized as follows: Section 2 defines the two approaches for the evaluation of the objective function values for the VRPSD; Section 3 explains the metaheuristic GA; Section 4 discusses the computational results and finally, Section 5 shows some concluding remarks of this research.

Approaches for the computation of the objective function value for the VRPSD

The VRPSD under the preventive restocking policy is defined as follows:

Let $G = (V; A)$ be a complete graph where:

- $V = \{0, 1, 2, \dots, n\}$ is the vertex set. The vertex 0 denotes the depot and vertex $1, \dots, n$ represent the customers.
- $A = \{(i, j): i, j \in V, i < j\}$ is an arcs set that connects the vertex.
- $C = (c_{ij})$. is the cost matrix that denotes the travel cost between the vertex.

It is considered that C is symmetric and satisfies the triangular inequality $c(i, j) \leq c(i, k) + c(k, j)$.

Moreover, the customers have stochastic demands $\xi_i, i = 1, \dots, n$ with known probability distributions. It is assumed that the demand of the customer i does not exceed the vehicle capacity Q and it has a discrete probability distribution $p_{ik} = prob(\xi_i = k), k = 0,1,2, \dots, K \leq Q.\ i = 1, 2, \dots, n$. The random variables ξ_i are independents. The objective of the VRPSD considering the preventive restocking policy is to find an a priori tour and a preventive restocking policy at each vertex that minimize total expected cost. The costs under consideration in this problem are the following:

- Cost of travelling from a customer as other as planned.
- Restocking cost: cost of go back to the depot before to visit the next planned customer.
- Failure cost: cost of go back to the depot for replenish due to insufficiency on the vehicle capacity to serve a customer upon arrival.

The VRPSD can be formulated as a two-index stochastic program as proposed in (Laporte et al., 2001). Let x_{ij} be an integer variable equal to the number of times that the arc (v_i, v_j) appears in the first stage solution, that is, in the a priori tour. x_{ij} can take the value of 0 or 1 if $j; j > 1$. x_{ij} can also be equal 2 if the vehicle goes back to the depot between vertex v_i and (v_i, v_j). In addition, let $R(x)$ be the policy cost.

Subject to:

$$Minimize \sum_{i<j} c_{ij}x_{ij} + R(x) \quad (5)$$

$$\sum_{j=1}^{n} x_{0j} = 2m, \quad (6)$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \quad (k = 2, \dots, n), \quad (7)$$

$$\sum_{v_i,v_j \in S} x_{ij} \leq |S| - \left[\sum_{v_i \in S} \frac{E(\xi_i)}{D}\right] \quad (S \subset V \setminus \{v_1\}; 2 \leq |S| \leq n-2), \quad (8)$$

$$0 \leq x_{ij} \leq 1 \quad (2 \leq i < n), \quad (9)$$

$$0 \leq x_{1j} \leq 2 \quad (j = 2, \dots, n), \quad (10)$$

$$x = (x_{ij}) \ integer. \quad (11)$$

Constraints (6) and (7) specify the degree of each vertex; constraint (8) ensures that no sub-route can be created and the total route demand does not exceed the vehicle capacity.

The literature presents two approaches for the evaluation of the expected cost from the objective function for the VRPSD with preventive restocking. The first approach is based on dynamic programming (Yang et al., 2000) and the second one relies on a simulation approach, e.g. using Monte Carlo simulation for this purpose. The next subsections will discuss both approaches for the evaluation of the objective function for the VRPSD.

Dynamic Programming approach for the VRPSD

An a priori tour is denoted as $1 \rightarrow 2 \rightarrow \cdots \rightarrow n$ for a particular vehicle. After serving the customer j, it is assumed that vehicle has a residual capacity l and let $f_j(l)$ denote the total expected cost from the customer j onwards. If S_j represents the set of all possible loads that a vehicle can have after serving the customer j, then $f_j(l)$ for $l \in S_j$ satisfies (Manfrin, 2004):

$$f_j(l) = \min\left\{f_j^p(l), f_j^r(l)\right\} \ (1)$$

Where:

$$f_j^p(l) = c_{i,j+1} + \sum_{k:k\leq l} f_{j+1}(l-k)p_{j+1,k} + \sum_{k:k>l}\left[b + 2c_{j+1,0} + f_{j+1}(l+L-k)\right]p_{j+1,k} \ (2)$$

and

$$f_j^r(l) = c_{j,0} + c_{0,j+1} + \sum_{k=1}^{K} f_{j+1}(L-k)p_{j+1,k}, \ (3)$$

with the boundary condition

$$f_n(l) = c_{n,0}, \quad l \in S_n f_n(l) = c_{n,0}, \quad l \in S_n \ (4)$$

In equations (1)-(3), $f_j^p(l)$ represents the cost of going directly to customer j and $f_j^r(l)$ is the preventive restocking cost. The above equations are used recursively in order to determine the value of the objective function of the planned route and the optimal sequence of decisions after attending to customers.

Equation (4) represents the boundary condition which means that the expected cost after the last customer visited onwards, independent of any residual charging the vehicle at the time, is equal to the fixed cost of going from the customer n to the depot. Algorithm 1 represents the computation of the total expected cost.

Algorithm 1. Computation of the expected value of the objective function

$\boldsymbol{for}\ l = L, L-1,\ \dots,\ 0\ \boldsymbol{do}$

$f_n(l) = c_{n,0}$

$\boldsymbol{for}\ j = n-1, n-2,\ \dots,\ 1\ \boldsymbol{do}$

$Compute\ f_j^r(l)\ using\ f_{j+1}(\cdot)$

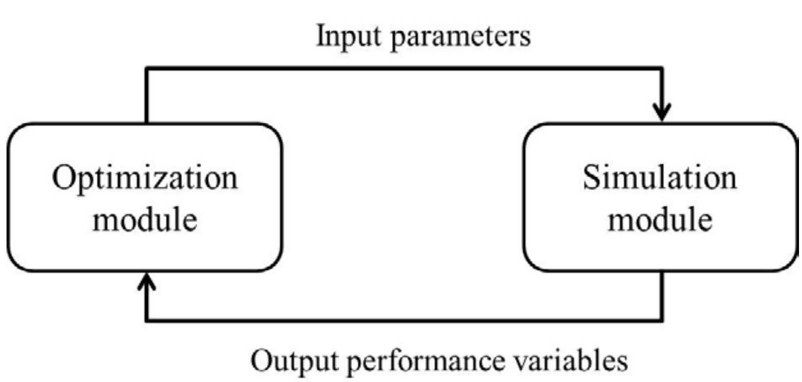
$\boldsymbol{for}\ l = L, L-1,\ \dots,\ 0\ \boldsymbol{do}$

$Compute\ f_j^p(l)$

Simulation-Optimization approach for the VRPSD

The simulation optimization approach seeks to determine the best configuration of input parameters to feed the simulation model produced results close to the optimum. The approach consists of two modules: one module where optimization is carried out exploration in the search space through a method of optimization and simulation module which is responsible for measuring the optimality using a simulation model (Ólafsson & Kim, 2002) (Figure 1).

Figure 1. General scheme of the simulation-optimization approach



Genetic algorithms for the VRPSD

In this research the metaheuristic Genetic Algorithms (GA) was used for the exploration and exploitation of the solutions in the search space. Genetic algorithms are adaptive methods of artificial intelligence, based on the natural phenomenon of reproduction and natural selection of the fittest individual. Such algorithms are used for solving problems where the solution space tends to be infinite. Broadly speaking, the genetic algorithm randomly searches and evaluates possible solutions in an optimization function. If the solution makes this function grows, the algorithm starts searching in areas close to the solution, if instead the proposed solution makes the optimization function decreases, the algorithm discards the solution.

In GA, the population consists of a set of chromosomes with correspond to the solutions. A crossover operator plays the role of reproduction and a mutation operator is assigned to make random changes in the solutions.

Initial population

The initial population corresponds to a set of individual. An individual is represented by an a priori route which is encoded as the integer permutation $X = (x_0, x_1, \ldots, x_n)$, where x_0 denotes the depot, and $x_1, \ldots, x_n$ are the customers (chromosomes). This population could be generated randomly or using a heuristic method. For this research, the Nearest Neighbor Heuristic (NNH) was implemented for this purpose.

Fitness value

For each individual or a priori tour, the fitness value is measured in order to establish the objective function values. This value can be obtained using a Dynamic Programming approach or a simulation approach.

Selection

The selection process consists in choosing two individuals (parent solutions) within the population. The selection procedure is stochastic and biased toward the best solutions using a roulette-wheel scheme (Berger & Barkaoui, 2003).

Crossover

This evolutionary operator can be executed, for instance, by using tournament. In this research, we implemented the crossover method proposed by (Pereira, Tavares, Machado, & Costa, 2002). This operator, each individual receives genetic information expressed in terms of sub-route from another individual and inserts it in one of its own routes.

Mutation

We used a swap mutation based operator adapted by the work of (Geetha, Ganesan, & Vanathi, 2010).

The algorithm 2 presents the general scheme for the solution of the VRPSD using GA.

Algorithm 2. Genetic algorithm for the VRPSD

1: Initialize the population of size m using the NNH

2: Evaluate fitness value for each individual

3: **while** stop criteria has not met **do**

4: Select two parents from the current population via roulette wheel selection

5: Apply crossover operator using the GVR algorithm

6: Improve the offspring using a mutation operator based on gene exchange

7: Evaluate the fitness value for the new individuals

8: Create a new generation with the new individuals

9: **end while**

10: Return the best individual

Experimental setup and computational results

In order to measure the performance for the dynamic programming and simulation-optimization approaches, we execute a series of experiments over a carefully designed test bed. The test bed is constructed taking into account the factors which are defined in Table 1.

Table 1. Definition of factors and levels for the Designs of experiments.

FACTOR	LEVEL 1	LEVEL 2
Number of customers (n)	100	200
Customers location (P)	Uniform	Normal distribution with two clusters
Average demand D_i	[50, 70]	[80, 100]
Spread S_i	[1, 5]	[10, 15]
Customers served before the preventive restocking (r)	4	8

In addition, for the generation of customer demands, 100 samples were considered for the simulation module and the customer demands follows a discrete uniform probability distribution. On the other hand, the GA parameters set for this research are the following: population size $m = 20$ and the mutation percentage $mut = 10\%$.

All algorithms were implemented in Matlab 7.1 and executed on a computer with a processor Intel Core i3 and 4GB of RAM. Since the dynamic programming algorithm requires considerable time for execution, time limits were established taking into account the number of customers defined, so that for $n = 100$, the time limit is 600 seconds and for $n = 200$, the time is equal to 1200 seconds. The results of the objective function values for the VRPSD under de dynamic programming and simulation-optimization approach are shown in Table 2 and Table 3, respectably. We used a design of experiments 2^{5-2} (Table 1) where each experiment was replicated five times, and the conclusions are made with respect to the average value of the objective function.

Table 2. Computational results for the VRPSD using dynamic programming approach

EXP	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5	AVERAGE	STANDARD DEVIATION
1	4530	4581,3	4528,6	4528,7	4535	4531	2,34
2	2613	2618,85	2628,73	2610,2	2620,24	2618,2	5,87
3	1987,31	1994,06	1984,29	1990,38	1995,17	1990,2	3,72
4	3540,26	3545,18	3543,02	3540,21	3532,01	3540,1	4,08
5	8422,32	8424,22	8417,43	8419,64	8418,11	8420,34	2,34
6	4689	4693,41	4684,53	4694,48	4689,29	4690.10	3,24
7	3749,27	3743,09	3742,03	3744,2	3748,24	3745,4	2,62
8	6986,29	6989,46	6983,22	6983,66	6983,28	6985,2	2,21

Table 3. Computational results for the VRPSD using simulation-optimization approach.

EXP	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5	AVERAGE	STANDARD DEVIATION
1	5103,92	5098,42	5121,99	5094,48	5098,49	5103,46	9,74
2	2882,82	2889,65	2854,73	2880,2	2880,24	2877,53	11,91
3	2257,31	2245,06	2244,49	2246,38	2245,87	2247,82	4,79
4	4058,76	4096,88	4067,02	4049,21	4102,71	4074,92	21,16
5	9276,72	9285,62	9268,53	9299,94	9243,31	9274,82	18,89
6	5085	5087,41	5074,53	5107,58	5110,89	5093,08	13,92
7	4121,47	4143,09	4122,03	4126,2	4130,46	4128,65	7,91
8	8016,96	7990,46	8013,87	8007,66	7997,86	8005,36	9,91

As shown in Table 2 and Table 3, the objective function values for the VRPSD using the dynamic programming approach are better than using Monte Carlo simulation, which at first sight it could be conclude that the best objective function estimator is dynamic programming approach. However, in terms of computational time (see Table 4), Monte Carlo simulation offers results close to the first approach in a reasonable computational time.

Table 4. Comparison of the computational time (seconds)

INSTANCE	DYNAMIC PROGRAMMING	MONTE CARLO SIMULATION
1	600	281,5
2	600	303,05
3	600	287,84
4	600	295,83
5	1200	592,42
6	1200	586,75
7	1200	605,41
8	1200	586,55

Conclusions

This paper presented formally the single VRPSD with preventive restocking using two different approaches to the calculation of the objective function, the first is a dynamic programming approach (DP) and the second is simulation optimization approach using specifically Monte Carlo simulation. For the exploration and exploitation of the solutions in the search space, the metaheuristic genetic algorithm was developed.

In order to determine the best approach for the evaluation of the objective function for the VRPSD, both approaches were executed over a carefully design test bed. In this research, we used a design of experiments in order to measure the performance of the GA, the experiments take the main effects of the factors that influence over the objective function. Results show that although the DP approach provides better estimations in terms of objective function values than Monte Carlo simulation, the second approach gives results close to the DP and with a significant reduction of the computational time with regard to DP. This results are suitable for real-life situations of companies dedicated to transportation logistics with high demand uncertainty.

References

Ak, A., & Erera, L. (2007). A Paired-Vehicle Recourse Strategy for the Vehicle-Routing Problem with Stochastic Demands. *Transportation Science*, 41(2), pp. 222–237. doi:10.1287/trsc.1060.0180.

Berger, J., & Barkaoui, M. (2003). A new hybrid genetic algorithm for the capacitated vehicle routing problem. *Journal of the Operational Research Society*, 54(12), pp. 1254–1262. doi:10.1057/palgrave.jors.2601635.

Bertsimas, D. J. (1992). A Vehicle Routing Problem with Stochastic Demand. *Operations Research*, 40(3), pp. 574–585. doi:10.1287/opre.40.3.574

Bianchi, L., Birattari, M., Chiarandini, M., Manfrin, M., Mastrolilli, M., Paquete, L., … Schiavinotto, T. (2005). Metaheuristics for the Vehicle Routing Problem with Stochastic Demands. *Journal of Mathematical Modelling and Algorithms*, 5(1), pp. 91–110. doi:10.1007/s10852-005-9033-y.

Bianchi, L., Manfrin, M., Rossi-doria, O., Birattari, M. et al. (2005). *Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands Technical. Technical Report No. IDISIA-06-05.*

Christiansen, C., & Lysgaard, J. (2007). A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research Letters*, 35(6), pp. 773–781. doi:10.1016/j.orl.2006.12.009.

Galván, S., Arias, J., & Lamos, H. (2013). Optimización por simulación basado en EPSO para el problema de ruteo de vehículos con demandas estocásticas. *DYNA*, (179), pp. 60–69.

Geetha, S., Ganesan, P., & Vanathi, P. T. (2010). A hybrid particle swarm optimization with genetic operators for vehicle routing problem. *Journal of Advances in Information Technology*, 1(4), pp. 181–188. doi:10.4304/jait.1.4.181-188.

Gendreau, M., Laporte, G., Séguin, R., & Seguin, R. (1995). An exact algorithm for the vehicle routing problem with stochastic demands and customers. *Transportation Science*, 29(2), pp. 143–155.

Hjorring, C., & Holt, J. (1999). New optimality cuts for a single-vehicle stochastic routing problem. *Annals of Operations Research*, (86), pp. 569-584.

Ismail, Z. & Irhamah. (2008). Solving the vehicle routing problem with stochastic demands via hybrid genetic algorithm-tabu search. *Journal of Mathematics and Statistics*, 4(3), pp. 161-167.

Laporte, G., Louveaux, F. & Van Hamme, L. (2001). An integer L-Shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research*, 50(3), pp. 415-423.

Manfrin, M. (2004). *Ant colony optimization for the vehicle routing problem. Computer.* Université Libre de Bruxelles.

Mendoza, J. E., Castanier, B., & Guéret, C. (2011). Constructive heuristics for the multicompartment vehicle routing problem with stochastic demands. *Transportation Science*, pp. 1–18. doi:10.1287/trsc.1100.0353

Mendoza, J. E., Castanier, B., Guéret, C., Medaglia, A. L., & Velasco, N. (2010). A memetic algorithm for the multi-compartment vehicle routing problem with stochastic demands. *Computers & Operations Research*, 37(11), pp. 1886–1898. doi:10.1016/j.cor.2009.06.015.

Novoa, C., & Storer, R. (2009). An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 196(2), pp. 509–515. doi:10.1016/j.ejor.2008.03.023.

Ólafsson, S., & Kim, J. (2002). Simulation Optimization. In *Proceedings of the 2002 Winter Simulation Conference E. Yiicesan, C.-H. Chen, J. L. Snowdon, and J. M . Chames, eds.* (pp. 79–84). IEEE.

Pereira, F. B., Tavares, J., Machado, P., & Costa, E. (2002). GVR : a new genetic representation for the vehicle routing problem. In *Proceedings of AICS 2002—13th Irish conference on artificial Intelligence and cognitive science.*

Rei, W., Gendreau, M., & Soriano, P. (2010). A hybrid monte carlo local branching algorithm for the single vehicle routing problem with stochastic demands. *Transportation Science*, 44(1), pp. 136–146. doi:10.1287/trsc.1090.0295.

Secomandi, N. (2000). Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 27(11-12), pp. 1201–1225. doi:10.1016/S0305-0548(99)00146-X.

Secomandi, Nicola. (2001). A rollout policy for the vehicle routing problem with stochastic demands. *Operations Research*, 49(5), pp. 796–802.

Shanmugam, G., Ganesan, P., & Vanathi, P. T. (2011). Meta heuristic algorithms for vehicle routing problem with stochastic demands. *Journal of Computer Science*, 7(4), pp. 533–542.

Tan, K., Cheong, C., & Goh, C. (2007). Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. *European Journal of Operational Research*, 177(2), pp. 813–839. doi:10.1016/j.ejor.2005.12.029.

Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. (P. Toth & D. Vigo, Eds.)*Optimization* (Vol. 9, p. 367). Society for Industrial and Applied Mathematics. doi:10.1137/1.9780898718515

Tripathi, M., Kuriger, G., & Wan, H. (2009). An ant based simulation optimization for the vehicle routing problem with stochastic demands. In *Proceedings of the 2009 Winter Simulation Conference, IEEE,* pp. 2476–2487.

Yang, W.-H., Mathur, K., & Ballou, R. (2000). Stochastic Vehicle Routing Problem with Restocking. *Transportation Science*, 34(1), pp. 99–112. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/20422950.