SORT 43 (1) January-June 2019, 3-24 DOI: 10.2436/20.8080.02.77

A simheuristic for routing electric vehicles with limited driving ranges and stochastic travel times

Lorena Reyes-Rubiano[1,*], Daniele Ferone[2], Angel A. Juan[3,4]
and Javier Faulin[1]

Abstract

Green transportation is becoming relevant in the context of smart cities, where the use of electric vehicles represents a promising strategy to support sustainability policies. However the use of electric vehicles shows some drawbacks as well, such as their limited driving-range capacity. This paper analyses a realistic vehicle routing problem in which both driving-range constraints and stochastic travel times are considered. Thus, the main goal is to minimize the expected time-based cost required to complete the freight distribution plan. In order to design reliable routing plans, a simheuristic algorithm is proposed. It combines Monte Carlo simulation with a multi-start metaheuristic, which also employs biased-randomization techniques. By including simulation, simheuristics extend the capabilities of metaheuristics to deal with stochastic problems. A series of computational experiments are performed to test our solving approach as well as to analyse the effect of uncertainty on the routing plans.

MSC: 90B08.

Keywords: Vehicle routing problem, electric vehicles, green transport and logistics, smart cities, simheuristics, biased-randomized heuristics.

1. Introduction

The growing public concern about living conditions and environmental preservation, specially in the context of modern cities, leads to the emergence and consolidation of the sustainable city concept, which integrates social, environmental, and economic dimensions (McKinnon et al., 2015). Smart sustainable cities call for an intelligent management of resources considering the social welfare in order to achieve a sustainable growth (Bibri and Krogstie, 2017). On the one hand, companies need to satisfy an in-

* *Corresponding author:* lorena.reyes@unavarra.es
[1] Institute of Smart Cities, Public University of Navarra, Pamplona, Spain
[2] Dept. of Informatics, Systems and Communication (DISCo), University of Milano-Bicocca, Milano, Italy
[3] IN3 – Computer Science Dept., Open University of Catalonia, Barcelona, Spain
[4] Euncet Business School, Terrassa, Spain
Received: March 2018
Accepted: November 2018

creasing consumers' demand that requires an intense freight transportation activity. This activity has to be carried out without generating economic inefficiencies. On the other hand, the welfare and environmental deterioration brings to light the need for smarter distribution systems that guarantee sustainability of this transportation activity. Nowadays, the main initiatives and policies in the area of transportation and logistics consider pollution targets. Typically, governmental urban guidelines focus on improving passenger and freight transportation due to their noticeable impact on the citizens' quality of life. Indeed, freight transportation generates around 10% of the greenhouse gases and ozone precursors released in the atmosphere (Eurostat, 2016). Since these emissions affect people's health, their reduction does not only generate an environmental benefit, but also social and economic gains. According to The World Bank (2018), 60% of the operation costs in developing countries are referred to energy bills. This reinforces the idea of a strong correlation between the different sustainability dimensions.

Consequently, the sustainability concept promotes the use of vehicles running on alternative fuel technologies. In particular, electric vehicles (EVs) represent a promising option to mitigate the negative impacts caused by transport activities in city logistics. Governments in many countries promote initiatives and regulations that aim at increasing the use of EVs, specially in city logistics and transportation activities. As a result, urban mobility is evolving to incorporate EVs. These incentives are motivated by the potential of zero-emission vehicles to reduce externalities on the citizens and the environment (Eurostat, 2016). As a way of responding to the aforementioned challenges related to sustainable logistics, a number of relevant initiatives have been released, e.g.: *(i)* Lean and Green Europe (www.lean-green.eu); *(ii)* US / Canada Smartway Transport Partnership (www.nrcan.gc.ca); or *(iii)* UNCTAD Sustainable Freight Transport and Finance (www.unctad.org). Some of these initiatives are supported and sponsored by private companies that acknowledge the importance of an environmentally sustainable growth. Responding to social and business needs, a large number of enterprises are incorporating both EVs and hybrid vehicles in their supply chain activities. In summary, the traditional paradigm of freight distribution in modern cities is changing with the introduction of these new technologies and the adoption of distribution concepts based on horizontal cooperation (Pérez-Bernabeu et al., 2015, Serrano-Hernández et al., 2017).

However, despite these technological advances there are still barriers to the full development of sustainable freight transportation. Examples of these barriers are: inefficient operations, poor infrastructures, or lack of sustainable policies. EVs require extra operational efforts due to the limited life of their batteries, the amount of time required to refill them, and the lack of recharging stations in modern cities. These technical limitations introduce driving-range constraints that do not exist in the case of traditional internal combustion vehicles (Juan et al., 2016). In addition to the previously described barriers, one has to take into account that the battery consumption rate depends on a wide range of random or difficult to predict factors, such as traffic congestion, road characteristics affecting the energy consumption, weather conditions, driving style, etc. In other words, real-life is full of uncertainty that has to be taken into account when considering travel times. Accordingly, this paper analyses the electric vehicle routing problem with stochastic travel times (EVRPST), which also considers time-based driving-range constraints (Figure 1). Being a rich extension of the classical vehicle routing problem (VRP), the EVRPST is also an *NP-hard* optimization problem, which justifies the use of heuristic-based solving approaches. Our main goal is to design an 'efficient' routing plan that satisfies a set of customers' demands using a homogeneous fleet of electric vehicles, each of them characterized by a limited loading capacity and driving range. Furthermore, we consider a more realistic VRP in which transport times are not deterministic but random variables instead. Efficiency will be measured in terms of total transport time. In other words, our main goal is to minimize the total expected time necessary to complete the delivery. Notice that random travel times could cause the exhaustion of the vehicle battery before completing its assigned route. Such a route failure will require a costly corrective action, which will be also measured in time units (Eshtehadi, Fathian and Demir, 2017).

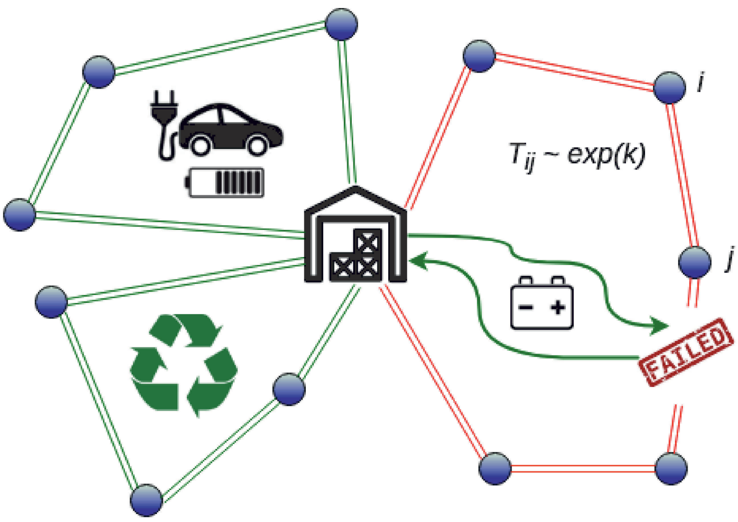


Figure 1: *A simple representation of the EVRPST with driving-range constraints.*

To solve the EVRPST, a novel simheuristic approach integrating Monte Carlo simulation within a multi-start framework is proposed. A review on basic concepts of simheuristic algorithms can be found in Juan et al. (2015). Also, the generation of solutions inside the multi-start framework is based on the use of biased-randomized techniques, which allow to extend deterministic heuristic into enhanced probabilistic algorithms. Grasas et al. (2017) provide an updated review of biased-randomized algorithms. Our solving approach considers the use of energy safety stocks: that is, during the design of the routing plan, a certain percentage of the battery is reserved for covering emergency situations with higher-than-expected travel times. Notice that using higher levels of safety stock leads to shorter routes and a higher number of required vehicles.

In contrast, using lower levels of safety stock will increase the probability of suffering a route failure. Whenever this occurs, we assume that the failing battery has to be replaced by a new one. In our computational experiments, this corrective action has a time-based penalty cost equivalent to a round-trip from the depot to the current position of the battery that needs to be replaced. All in all, the main contributions of this paper are: (i) to mitigate the lack of works on vehicle routing problems considering both driving-range limitations and uncertainty conditions; (ii) to develop and test a simheuristic approach for the EVRPST; and (iii) to analyse the effect of random travel times and the use of energy safety stocks on the routing plans.

The remaining of the paper is organized as follows: Section 2 reviews related work in the transportation literature; Section 3 provides some additional details on the problem under study; our simheuristic solving approach is explained in Section 4; Section 5 describes a series of computational experiments, while the associated results are discussed in Section 6; finally, Section 7 concludes the paper and identifies potential lines for future research.

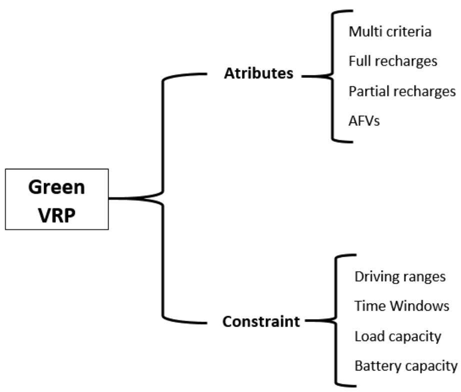


Figure 2: *Frequent attributes and constraints in the G-VRP.*

2. Literature review

The use of EVs in transport activities is related to several urban changes in terms of infrastructure and distribution strategies. On the one hand, some of these challenges relate to infrastructure and fleet configurations (Juan, Goentzel and Bektaş, 2014b, Shao, Guan and Bi, 2018). On the other hand, EVs have started to replace conventional vehicles in city logistics, redefining transport operations (Hof, Schneider and Goeke, 2017). Many logistics and transportation problems in smart cities can be modeled as rich VRP variants (Cáceres-Cruz et al., 2014). The rich VRP has been a very active research line in combinatorial optimization problems. This is partly due to the difficulty of managing multiple attributes and constraints, such as the different sustainability dimensions: economic, social, and environmental (McKinnon et al., 2015). In particular, the 'green'

VRP (G-VRP) is a rich VRP which considers routing problems using alternative fuel vehicles (AFVs) (Erdoğan and Miller-Hooks, 2012). One popular G-VRP variant is the so-called pollution routing problem or PRP (Bektaş and Laporte, 2011). In the PRP, the main objective is to minimize the energy consumption. It also includes time windows as a realistic constraint. Figure 2 provides a scheme that summarizes different attributes and constraints frequently associated with the G-VRP (Lin et al., 2014).

2.1. The deterministic G-VRP

A key restriction in VRPs with EVs is the limited capacity of their batteries, which might require multiple recharging stops. Hence, (Erdoğan and Miller-Hooks, 2012) solve a G-VRP allowing intermediate stops by implementing procedures based on the well-known savings heuristics (Clarke and Wright, 1964) and the popular density-based clustering algorithm. Demir, Bektaş and Laporte (2012) solves a PRP with time windows, where customer sequences are first defined and, afterwards, the travel speeds are optimized by means of an adaptive large neighbourhood search (ALNS) metaheuristic. Juan et al. (2014b) address the G-VRP with multiple driving ranges. The goal of this work is to define alternative fleet configurations based on EVs and hybrid-electric vehicles. The authors describe an integer programming formulation and a multi-round heuristic algorithm that iteratively constructs a solution. Schneider, Stenger and Goeke (2014) propose an ALNS metaheuristic with some local searches with the aim of minimizing the total distribution cost, which includes the cost of using a fleet of vehicles plus the actual routing cost. Additionally, these authors considered intermediate stops in recharging stations. Similarly, the ALNS metaheuristic is hybridized with the adaptive variable neighbourhood search framework by Schneider, Stenger and Hof (2015), who deal with a routing problem with EVs-related constraints and also consider intermediate stops. Koç and Karaoglan (2016) design a simulated annealing metaheuristic, based on an exact method, to solve the G-VRP for the small-scale instances proposed by Erdoğan and Miller-Hooks (2012). Hiermann et al. (2016) study the VRP with EVs, time windows, and recharging stations. Hof et al. (2017) consider EVs to solve a location-routing problem where the objective is to determine whether the battery swap stations should be defined from candidate locations or closer to the set of customers. Finally, the G-VRP with multiple objectives – including both monetary and environmental costs – is discussed by Sawik, Faulin and Pérez-Bernabeu (2017a, b, c).

2.2. The stochastic G-VRP

Stochastic combinatorial optimization has received increasing interest during the last decades (Bianchi et al., 2009, Ritzinger, Puchinger and Hartl, 2015). Solving a stochastic VRP requires a methodology able to deal with the random components of the problem, which is not straightforward, as discussed in Juan et al. (2011a, 2013). The most frequent random variables are: customers' demands, service and travel times, and frequency of order placing (Bozorgi, Farasat and Mahmoud, 2017). The previous articles highlight the importance of dealing with uncertainty, and study realistic characteristics such as urban transport dynamics. In most existing works, travel times are assumed to be constant, but this is not a realistic assumption. Hence, Ritzinger et al. (2015) propose to deal with uncertain travel times by modelling them as stochastic and time-dependent variables.

Uncertainty conditions are sometimes addressed by means of stochastic programming. This approach provides high quality solutions for small instances (Bozorgi-Amiri, Jabalameli and Al-e Hashem, 2013). Erdoğan and Miller-Hooks (2012) present an exact model to solve the VRP with stochastic travel times. These authors assess the influence of route duration on environmental indicators, such as energy consumption. Another relevant problem is the time-dependent VRP, where the travel times are different depending on the specific period. Gendreau, Ghiani and Guerriero (2015) provides a literature review on these topics. Travel times may vary by exogenous variables, such as traffic congestion, weather conditions, moving targets, or mobile obstacles. They might also be influenced by endogenous variables: for example, by varying the vehicles' speeds or by choosing highways over standard roads.

Recently, Eshtehadi et al. (2017) address a VRP with stochastic demands and travel times. These authors develop a solving approach based on an exact method that is able to solve instances with up to 20 nodes considering multiple scenarios. The authors tackle the stochasticity describing two scenarios that represent the best and the worst conditions for demand and travel times. To conclude this literature review, Table 1

Table 1: *An illustrative set of works covering the most popular G-VRP variants.*

Papers	Atributes	Constraints	Solution Approach
Shao et al. (2018)		Driving range	GA
Eshtehadi et al. (2017)	Stochastic demands Stochastic travel times	Driving range	SP
Sawik et al. (2017a,b,c)	Multi criteria	Driving range	EM
Koç and Karaoglan (2016)		Driving range	SAM and EM
Hiermann et al. (2016)	Full recharges	Time windows	EM
Desaulniers et al. (2016)	Full recharges Partial recharges	Driving range Time windows	EM
Schneider et al. (2015)	Full recharges	Driving range	ALNS
Felipe et al. (2014)	Full recharges	Driving range	GSA
Juan et al. (2014a)	Heterogeneous fleet	Driving range	RMS
Schneider et al. (2014)	Full recharges	Driving range	ALNS
Erdoğan and Miller-Hooks (2012)	Stochastic travel time Full recharges	Driving range	SP

ALNS: Adaptive large neighbourhood search. EM: Exact method. SP: Stochastic programming.
RMS: A randomized multi-start algorithm. GA: Genetic algorithm.GSA: Greedy algorithm. SH: Savings heuristic.
SAM: Simulated annealing metaheuristic. DC: Density-based clustering algorithm

summarizes some illustrative works providing evidence about the most studied G-VRP variants.

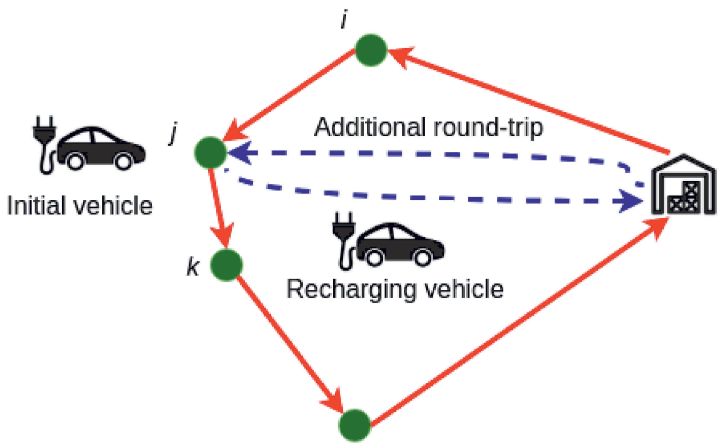
3. Additional details on the EVRPST

The EVRPST is defined on an undirected graph $G = (N, A)$. Here, N contains the depot (node 0) and a set of customers $N^* = \{1, 2, \ldots, n\}$. Also, $A = \{(i, j) \mid i, j \in N, i \neq j\}$ is the set of edges connecting any two nodes in N. Each customer $i \in N^*$ has a demand $d_i > 0$. There is a set V of homogeneous vehicles, each of them with a loading capacity of $q >> max\{d_i\}$. As it is usual in most VRPs (Toth and Vigo, 2014), the following assumptions hold: (i) all customers' demands must be satisfied; (ii) each vehicle route starts and ends at the depot; (iii) each customer is visited exactly once; and (iv) the demand to be served in each route does not exceed the vehicle loading capacity. Moreover, the time-based cost of traversing each edge (i, j) is given by an independent random variable $T_{ij} = T_{ji} > 0$, which follows a known probability distribution with mean $E(T_{ij}) = t_{ij}$. Thus, the additional constraint is considered as well: the expected travel time employed by a vehicle to complete its route is limited by the battery duration, $t_{max} > \{\sum E[T_{ij}]\}$.

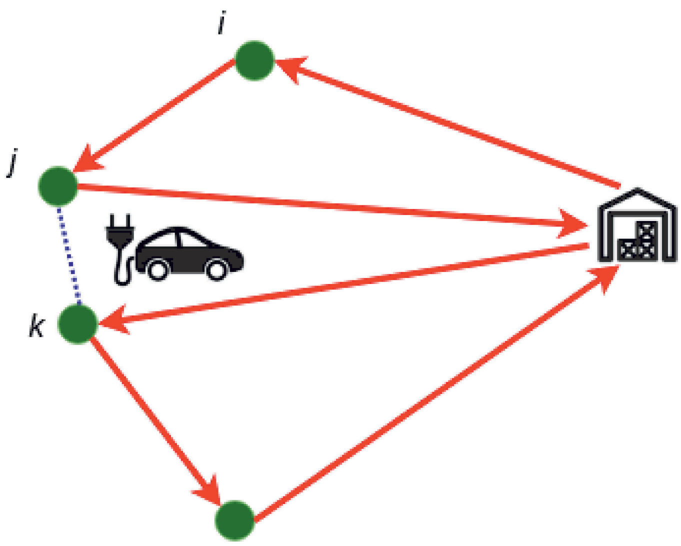
However, considering stochastic travel times implies introducing uncertainty about how much energy will be required to complete a route. Energy consumption and travel times depend on multiple factors, such as current load of the vehicle, road type, vehicle speed, driving skills, etc. This uncertainty makes it hard to guarantee feasible solutions when hard time-related constraints on batteries duration are considered. In particular, electric vehicles have a risk of batteries exhaustion during the trip, which is considered as a *route failure*. Decision makers may define corrective actions to properly address these failures when they happen. They might also define preventive actions to be applied before the vehicle runs out of battery. Figure 3 illustrates some examples of these types of actions.

On the one hand, a corrective action to resume the routing plan is required when a vehicle A runs out of energy after visiting a customer j (failure type I). In our computational experiments, we will assume that the cost of this corrective action is the time needed for a new vehicle B to complete a round-trip from the depot to the current location of A to supply a new battery. On the other hand, a preventive action could also be applied: if there is a high risk of running out of battery after serving a customer j, vehicle A might decide to return from j to the depot for recharging or swapping batteries (failure type II); after that, it might resume its planned route from the next customer, k. The time-based cost of such a preventive action could be estimated as the time requested to visit the depot for recharging batteries plus the time employed in moving from the depot to the next customer in the original route, k.

Although the simheuristic methodology introduced in this paper is quite flexible and could be easily extended to consider preventive actions, in our computational experiments we have only considered corrective actions (i.e., type I failures). Accordingly,



(a) Preventive action for type II failure



(b) Corrective action for type I failure

Figure 3: *Different actions to deal with route failures while using electric vehicles.*

the objective function minimizes the expected time-based cost required to complete the delivery process. Notice that this time-based cost is a non-smooth function, since it includes the 'penalty' cost associated with applying these corrective actions whenever route failures occur. Hence, if T_v represents the total time employed by vehicle v in completing its route, the objective function can be expressed as:

$$\min E\left(\sum_{v\in V} T_v\right) \tag{1}$$

with:

$$T_v = \begin{cases} \sum\limits_{\substack{i,j \in N \\ i \neq j}} T_{ij} \cdot z_{ijv} & \text{if } \sum\limits_{\substack{i,j \in N \\ i \neq j}} T_{ij} \cdot z_{ijv} \leq \ t_{max} \\ \sum\limits_{\substack{i,j \in N \\ i \neq j}} T_{ij} \cdot z_{ijv} + 2 \cdot T_{j0} & \text{otherwise} \end{cases} \tag{2}$$

where the decision variable z_{ijv} takes the value 1 if vehicle v covers the edge (i, j), while it takes the value 0 otherwise.

4. A simulation-optimization approach

Some of the first works employing simulation-optimization methods to deal with the VRP are due to Faulin and Juan (2008) and Faulin et al. (2008). Our solving methodology relies on a simheuristic approach, which proposes the integration of simulation techniques within a heuristic framework to address stochastic optimization problems in a natural way (Juan et al., 2015). In a simheuristic approach, the metaheuristic component is responsible for searching and filtering out promising solutions, while the simulation component is responsible for estimating different statistics associated with these promising solutions when considered in a stochastic environment. When properly designed, the simulation component can also provide feedback that is then used by the heuristic framework to better guide the search process De Armas et al. (2017). In this paper, we propose to integrate Monte Carlo simulation (MCS) into a biased-randomized multi-start framework. Biased-randomized versions of a constructive heuristic allow for fast generation of high-quality solutions (Grasas et al., 2017). These techniques have been successfully applied in solving different combinatorial optimization problems in areas such as vehicle routing (Dominguez, Juan and Faulin, 2014, Dominguez et al., 2016b, a, c), scheduling (Juan et al., 2014c, Ferone et al., 2018, Gonzalez-Neira et al., 2017) and facility location (Alvarez Fernandez et al., 2018). When complemented with some local search and encapsulated inside a multi-start (or similar) framework, they constitute a strong basis that can be easily extended into a simheuristic framework (Grasas, Juan and Lourenço, 2016). Our biased-randomized multi-start (BR-MS) simheuristic approach builds upon the biased-randomized version of Clarke and Right Savings (BRCWS) procedure proposed by Juan et al. (2011b). The complete algorithm is summarized in Pseudo-code 1 and described next in more detail.

First, the stochastic instance is transformed into a deterministic one by using expected travel times as initial estimates for the real stochastic values. Then, following the Clarke and Wright (1964) heuristics, a dummy solution is created and the savings associated with traversing each edge are computed. This initial solution (*initSol*) is improved by the classical *2-Opt* local search operator, and its expected travel time (stochastic cost) is estimated by using a fast MCS with just *sSim* runs – typically in the order of a few hundreds. Notice that, as any other solution we will generate, *initSol* will have

Algorithm 1: *BR-MS simheuristic for the EVRPST.*

```
procedure SIMHEURISTIC solve(test, nodes, edges)
                                        ▷ test: maxTime, β, sSim, lSim, s
                                        ▷ nodes: coordinates, demand
                                        ▷ edges: travel time
    savings ← computeSavings (nodes, edges)
    initSol ← savingsHeuristic (nodes, savings)      ▷ Clarke and Wright (1964)
    initSol ← localSearch (initSol)                   ▷ 2-Opt
    stochCost(initSol) ← simulation (initSol, sSim)
    baseSol ← initSol
    bestStochSolList ← add (initSol)                  ▷ elite solutions
    while (elapsedTime < maxTime) do
        newSol ← BRCWS (nodes, savings, β, s)         ▷ Juan et al. (2011b)
        newSol ← localSearch (newSol)                 ▷ 2-Opt
        if (detCost(newSol) < detCost(baseSol)) then
            stochCost(newSol) ← simulation (newSol, sSim)
            if (stochCost(newSol) < detCost(baseSol)) then
                baseSol ← newSol
            end if
            update (bestStochSolList)
        end if
    end while
    for (each sol in bestStochSolList) do
        stochCost(sol) ← simulation (sol, lSim)
    end for
    return bestSol in bestStochSolList
end procedure
```

two time-based costs: the one associated with the deterministic version of the problem (*detCost*) and the one associated with the stochastic one (*stochCost*). At this stage, *initSol* is stored as our temporary reference or 'base' solution (*baseSol*) and included in a list of 'elite' stochastic solutions (*bestStochSolList*). Afterwards, a multi-start process is repeated until a termination criterion (*maxTime*) is met. In each iteration, a new deterministic solution (*newSol*) is generated by using the BRCWS procedure. Once a fast local search is applied, this solution is labeled as 'promising' if its deterministic time-based cost is lower than that of *baseSol*. If it is not promising, *newSol* is discarded and a new iteration starts. If it is promising, a new fast MCS is applied to estimate the stochastic cost (expected time) associated with *newSol*. Whenever appropriate, *baseSol* is replaced by *newSol* and the *bestStochSolList* is updated. Once the ending criterion is met, the expected time associated with each elite solution in *bestStochSolList* is assessed again, this time using a more intensive MCS with *lSim* runs, typically in the order of a few thousands. Notice that while the assessments in the main loop are required to be fast, because the number of solutions to assess may be extraordinarily high, those applied to a reduced list of elite solutions can employ more computing time.

The computational time of the algorithm is bounded by *maxTime*. Regarding its computational complexity, each iteration has three stages: the construction with BR-CWS, the local search, and the simulation phase. The computational complexity of BRCWS is bounded by the number of the edges m, since the merging can be done in constant time but it is necessary to examine all savings. Since each client is served exactly once, the local search swapping moves are bounded to $O(n^2) = O(m)$. Finally, the complexity of the simulation stage is $O(m \cdot sSim)$. Therefore, the complexity of each iteration is dominated by the simulation phase, and it is $O(m \cdot sSim)$.

As usually done in the related literature (Grasas et al., 2017), the biased-randomized procedure is based on the use of a geometric probability distribution, which makes use of a parameter β $(0 < \beta < 1)$. The BRCWS heuristic is adapted from the one proposed by Juan et al. (2011b) to ensure the feasibility of the generated solutions. In particular, it is guaranteed that the expected travel time of each vehicle will not exceed the duration of the batteries. However, as discussed before, under stochastic conditions it is not possible to guarantee that a route is failure-free. Accordingly, the reliability of each solution (i.e., the probability that a solution does not suffer any route failure) is also estimated from the data obtained in the previous simulation runs. As a way to increase these reliability levels, different levels of safety stock are considered for each vehicle. In other words, during the route-design stage, a given percentage of the vehicle driving-range capacity ($s\%$) is reserved as a safety stock to be used in case of higher-than-expected travel times. The specific value of s is a decision variable to be determined during the simulation-optimization process, since it will depend on the specific instance being analysed as well as on the probability distribution used to model travel times.

Notice that a relatively high value of s leads to short and reliable routes, i.e., routes employing short travel times and with a low probability of experiencing a failure due to the existence of a noticeable safety stock. Unfortunately, this also requires the use of more vehicles to cover all customers. On the contrary, a relatively value of s produces longer routes with a higher probability of suffering a failure (low reliability), but it requires a lower number of routes to cover all customers.

Regarding the MCS module, the steps followed to assess the stochastic performance (expected travel time) of a given solution are: (i) using random sampling from the assigned probability distributions, we run different executions of the routing plan in order to obtain random observations of the total travel time associated with it; (ii) from these random observations, different statistics can be computed for each routing plan, e.g.:

average time, variability of these times, etc.; (iii) using the same simulation outcomes, we estimate the reliability of each routing plan as the quotient between the number of route failures and the number of simulation runs. These experiments are repeated for different percentages of the safety stock level, s.

5. Computational experiments

This section presents a set of extensive computational experiments carried out to test our simheuristic approach for the EVRPST. Firstly, we introduce the instances that will be used to test our approach. Secondly, the algorithm parameters are discussed. Finally, the computational results are provided – they will be fully analysed in the next section. The algorithm has been implemented as a Java application. A standard personal computer with an Intel Core i5 CPU at 3.2 GHz and 4 GB RAM has been employed to perform all the experiments.

5.1. Benchmark instances

As a benchmark for our test, a set of 27 instances originally proposed by Uchoa et al. (2017) are selected. The original instances already included a maximum distance per route. They have been adapted so they use time-based costs instead of distance-based ones; i.e., Euclidean distances are considered to be travel times and the maximum distance per route is transformed into a maximum time per route. These instances are derived from the ones proposed by Christofides (1976), Golden et al. (1998), and Li, Golden and Wasil (2005). Table 2 shows the main characteristics of these instances.

In order to perform numerical experiments under uncertainty conditions, the aforementioned deterministic instances have been extended to consider stochastic travel times as follows: if the original instance shows a determinisitic travel time $t_{ij} = t_{ji} > 0$ when moving from node i to node j (with $i \neq j$), then we consider that the stochastic travel time T_{ij} is a random variable following an exponential probability distribution with $E[T_{ij}] = t_{ij}$ and $Var[T_{ij}] = t_{ij}^2$. In a real-life scenario, the specific probability distributions associated with each stochastic travel time would need to be fitted from historical observations, but our solving approach would still be valid. Furthermore, different levels of safety stock – as a percentage of the battery capacity (i.e., vehicle driving range) – have been considered in our experiments: $s \in \{0\%, 5\%, 10\%, \ldots, 35\%\}$.

5.2. Parameter settings

One of the advantages of our algorithm is that it does not require a complex fine-tuning process. In fact, after some quick trial-and-error experiments, the following values were set for each parameter:

Table 2: Characteristics of the benchmark instances.

Instance	n	$\|V\|$	q	t_{max}
Golden_1	240	9	550	650
Golden_2	320	10	700	900
Golden_3	400	10	900	1200
Golden_4	480	10	1000	1600
Golden_5	200	5	900	1800
Golden_6	280	7	900	1500
Golden_7	260	9	900	1300
Golden_8	440	10	900	1200
CMT6	50	6	160	200
CMT7	75	11	140	160
CMT8	100	9	200	230
CMT9	150	14	200	200
CMT10	199	18	200	200
CMT13	120	11	200	720
CMT14	100	11	200	1040
Li_21	560	10	1200	1800
Li_22	600	15	900	1000
Li_23	640	10	1400	2200
Li_24	720	10	1500	2400
Li_25	760	19	900	900
Li_26	800	10	1700	2500
Li_27	840	20	900	900
Li_28	880	10	1800	2800
Li_29	960	10	2000	3000
Li_30	1040	10	2100	3200
Li_31	1120	10	2300	3500
Li_32	1200	11	2500	3600

n = number of customers; $|V|$ = number of vehicles
q = capacity of each vehicle
t_{max} = maximum time allowed per route

- The biased-randomized selection during the construction process was generated by using a geometric probability distribution with parameter $\beta \in (0.23, 0.30)$ – i.e., at each iteration a random value inside the previous interval was assigned to β.
- The number of simulation runs was set to $sSim = 400$ for fast simulations (on each promising solution) and to $lSim = 10,000$ for intensive simulations (on each elite solution).
- For each instance, the algorithm was run 20 times, each time employing a different seed for the pseudo-random number generator.
- For each instance and seed, the algorithm was executed for $maxTime = 90$ seconds. Notice that this time does not include the time employed in computing the intensive simulations – however, since the number of elite solutions is reduced, this final step takes just a few additional seconds.

5.3. Computational results

Table 3 summarizes the results obtained both using the BRCWS procedure – a deterministic component inside the simheuristic – and the complete MS-BR simheuristic algorithm. Both approaches were run using the same parameters setting as described in Section 5.2. Also, in this comparison, no safety stock is considered, i.e., $s = 0\%$.

Table 3: *Performance of best deterministic and stochastic solutions.*

	BRCWS (deterministic component)			MS-BR Simheuristic	
Instance	BDS-Det	BDS-Stoch (a)	Reliability	BSS-Stoch	Reliability
CMT6	546.59	586.75	0.97	586.75	0.97
CMT7	856.26	1060.14	0.86	1040.29	0.88
CMT8	870.60	911.39	0.97	911.14	0.97
CMT9	1118.03	1189.43	0.95	1183.26	0.96
CMT10	1375.31	1439.11	0.95	1431.04	0.96
CMT13	1537.88	1544.24	0.99	1539.03	0.99
CMT14	823.11	823.24	0.99	823.24	0.99
Golden_1	5786.96	9939.65	0.02	9298.79	0.05
Golden_2	8646.93	13376.35	0.01	12754.47	0.03
Golden_3	12828.23	17757.94	0.01	16416.42	0.06
Golden_4	17963.58	23019.70	0.02	21764.50	0.06
Golden_5	7334.24	7679.08	0.78	7602.17	0.83
Golden_6	9829.11	12119.12	0.14	11371.87	0.30
Golden_7	12270.11	15998.37	0.04	15274.38	0.08
Golden_8	13753.22	18831.50	0.01	17869.64	0.03
Li_21	20465.47	24826.35	0.03	23939.78	0.08
Li_22	16612.02	23985.19	0.00	23330.96	0.00
Li_23	23192.07	27986.58	0.02	27176.38	0.07
Li_24	26160.76	30327.41	0.04	30086.13	0.06
Li_25	17618.46	27426.64	0.00	26942.85	0.00
Li_26	28728.31	34534.97	0.01	32076.98	0.09
Li_27	18460.02	28341.25	0.00	28160.91	0.00
Li_28	32654.00	35986.88	0.08	35547.75	0.20
Li_29	35230.52	38188.93	0.10	36485.80	0.87
Li_30	40363.61	44088.03	0.07	42891.96	0.48
Li_31	44248.09	47195.81	0.13	46263.44	0.58
Li_32	45959.99	50720.75	0.04	49407.09	0.15
Average	16490.84	19996.24	0.31	19340.31	0.40

BDS-Det: Best deterministic solution in a deterministic scenario.
BDS-Stoch: Best deterministic solution in a stochastic scenario.
BSS-Stoch: Best stochastic solution in a stochastic scenario.

Hence, column *BDS-Det* shows the cost (in total travel time) associated with the best-found solution obtained for the deterministic version of the problem when it is applied in a deterministic scenario (without uncertainty); column *BDS-Stoch* provides the expected cost of the same solution when it is employed in a stochastic scenario; the reliability column gives an estimate of the probability that the best deterministic solution can be used in a stochastic scenario without suffering any route failure – notice that reliabilities can be low in some cases since no safety stock is considered. Similarly, column *BSS-Stoch* shows the expected cost of the best-found solution for the stochastic version of the problem when applied in a stochastic scenario. Finally, the reliability column provides an estimate of the probability that this solution can be completed as designed – without route failures. As depicted in Figure 4, *BDS-Det* and *BDS-Stoch* act as a lower bound and an upper bound, respectively, for *BSS-Stoch*. Thus, in general, it is not a good idea to apply the best-found solution for the deterministic version of the problem to a scenario under uncertainty, since it might often result in a sub-optimal plan. Instead, it is better to use a simulation-optimization approach to generate solutions with a better performance under stochastic conditions (usually by offering a higher reliability level and thus avoiding expensive corrective actions).

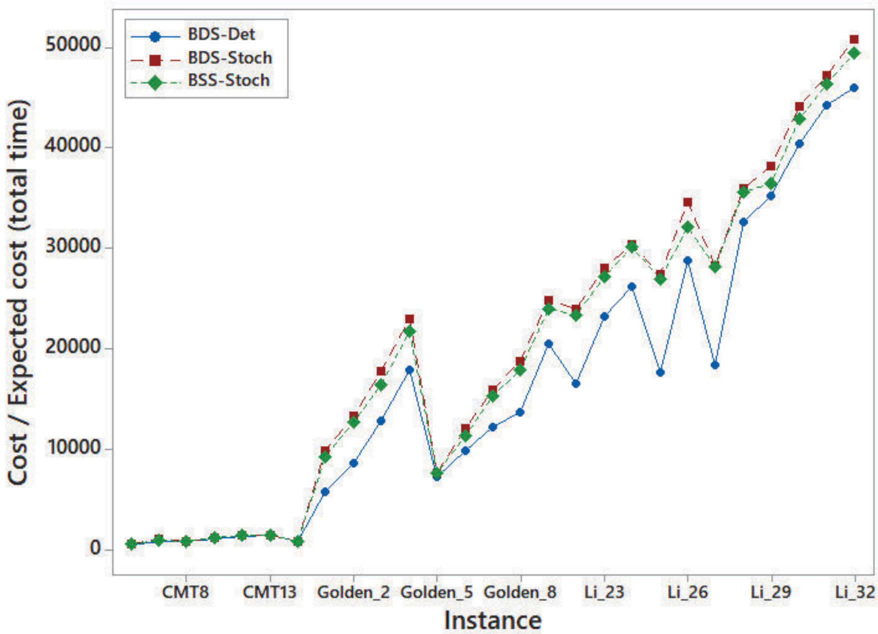


Figure 4: *Visual comparison among BDS-Det, BDS-Stoch, and BSS-Stoch.*

For each instance and safety stock level s, Table 4 shows the expected cost (in total travel time) provided by our simheuristic algorithm in a stochastic scenario. The table also shows the reliability associated with each solution – which tends to increase with the safety stock level –, as well as the gap with respect to the solution obtained without using any safety stock.

Table 4: *Solution performance considering different safety stock levels.*

	BDS-Stoch	BSS-Stoch with s = 5%			BSS-Stoch with s = 15%			BSS-Stoch with s = 25%			BSS-Stoch with s = 35%		
Instance	E[Cost] (a)	E[Cost] (c)	Reliab.	Gap (c)-(a)	E[Cost] (e)	Reliab.	Gap (e)-(a)	E[Cost] (g)	Reliab.	Gap (g)-(a)	E[Cost] (i)	Reliab.	Gap (i)-(a)
CMT6	586.75	584.80	0.97	−0.33%	586.94	0.97	0.03%	580.22	0.97	−1.11%	577.34	0.98	−1.60%
CMT7	1060.14	1044.42	0.87	−1.48%	1040.23	0.87	−1.88%	1041.44	0.88	−1.76%	1021.55	0.89	−3.64%
CMT8	911.39	910.92	0.97	−0.05%	909.31	0.97	−0.23%	895.55	0.99	−1.74%	890.08	0.98	−2.34%
CMT9	1189.43	1185.18	0.96	−0.36%	1180.13	0.96	−0.78%	1165.24	0.96	−2.03%	1150.64	0.97	−3.26%
CMT10	1439.11	1433.89	0.96	−0.36%	1429.24	0.96	−0.69%	1440.93	0.95	0.13%	1423.00	0.98	−1.12%
CMT13	1544.24	1554.70	1.00	0.68%	1552.56	1.00	0.54%	1554.78	1.00	0.68%	1555.34	1.00	0.72%
CMT14	823.24	820.79	1.00	−0.30%	819.92	1.00	−0.40%	822.17	1.00	−0.13%	820.66	1.00	−0.31%
Golden_1	9939.65	8814.54	0.09	−11.32%	7911.15	0.27	−20.41%	7698.07	0.56	−22.55%	8094.93	0.81	−18.56%
Golden_2	13376.35	12159.70	0.07	−9.10%	10669.63	0.32	−20.24%	10373.62	0.59	−22.45%	10883.65	0.87	−18.64%
Golden_3	17757.94	16135.89	0.09	−9.13%	14569.99	0.34	−17.95%	13520.73	0.75	−23.86%	13127.24	0.95	−26.08%
Golden_4	23019.70	20693.91	0.15	−10.10%	19188.46	0.48	−16.64%	18460.61	0.82	−19.81%	18498.44	0.97	−19.64%
Golden_5	7679.08	7584.29	0.95	−1.23%	7585.41	0.95	−1.22%	7335.11	0.97	−4.48%	7231.07	0.99	−5.83%
Golden_6	12119.12	10963.40	0.45	−9.54%	10206.53	0.76	−15.78%	9812.63	0.91	−19.03%	9657.49	0.97	−20.31%
Golden_7	15998.37	14250.60	0.22	−10.92%	13333.32	0.44	−16.66%	12568.18	0.75	−21.44%	12115.77	0.96	−24.27%
Golden_8	18831.50	16971.04	0.08	−9.88%	15499.76	0.32	−17.69%	14579.05	0.69	−22.58%	14314.21	0.93	−23.99%
Li_21	24826.35	22865.97	0.19	−7.90%	21736.92	0.47	−12.44%	20683.79	0.85	−16.69%	20724.91	0.97	−16.52%
Li_22	23985.19	21542.09	0.02	−10.19%	20772.52	0.11	−13.39%	19038.31	0.49	−20.62%	20424.35	0.78	−14.85%
Li_23	27986.58	25514.42	0.24	−8.83%	24106.71	0.59	−13.86%	23381.52	0.90	−16.45%	23126.42	0.99	−17.37%
Li_24	30327.41	29102.20	0.11	−4.04%	26439.78	0.87	−12.82%	26200.40	0.93	−13.61%	26121.15	1.00	−13.87%
Li_25	27426.64	26020.12	0.00	−5.13%	24084.62	0.06	−12.19%	24501.08	0.32	−10.67%	N/A	N/A	N/A
Li_26	34534.97	29439.93	0.75	−14.75%	29316.25	0.80	−15.11%	28893.00	0.96	−16.34%	29010.87	0.99	−16.00%
Li_27	28341.25	27203.26	0.00	−4.02%	25756.63	0.04	−9.12%	26592.44	0.23	−6.17%	N/A	N/A	N/A
Li_28	35986.88	33494.04	0.76	−6.93%	32982.25	0.76	−8.35%	31505.10	0.98	−12.45%	31559.27	1.00	−12.30%
Li_29	38188.93	36379.32	0.92	−4.74%	35586.57	0.71	−6.81%	33713.97	0.99	−11.72%	34091.61	1.00	−10.73%
Li_30	44088.03	42149.09	0.31	−4.40%	40709.81	0.90	−7.66%	40839.56	1.00	−7.37%	40826.65	1.00	−7.40%
Li_31	47195.81	45952.96	0.45	−2.63%	44212.40	0.88	−6.32%	44151.37	0.99	−6.45%	44985.68	1.00	−4.68%
Li_32	50720.75	48539.65	0.16	−4.30%	46575.65	1.00	−8.17%	46595.38	0.99	−8.13%	46018.94	1.00	−9.27%
Averages			0.47	−5.60%		0.66	−9.49%		0.83	−11.44%		0.96	−11.67%

One should notice that, in most cases, using a safety stock during the design stage might be a good strategy to reduce the impact of route failures whenever travel times are higher than expected. This concept is further discussed in the next section. Also, note that for a safety stock level of 35% (or higher), there are some instances that cannot be solved during the design stage; i.e., assuming such a high safety stock level, some customers in instances *Li_25* and *Li_27* cannot be reached from the depot in the reduced 'standard' time of the batteries (i.e., without considering the extra time that can be provided by the energy safety stock). That justifies that we focus on safety stock levels between 0% and 35% of the original battery capacity.

6. Analysis of results

For each considered safety stock level, $s \in \{0\%, 5\%, 10\%, \ldots, 35\%\}$, Figure 5 uses boxplots to illustrate the distribution of the reliability indices associated with the best-found stochastic solutions for each instance.

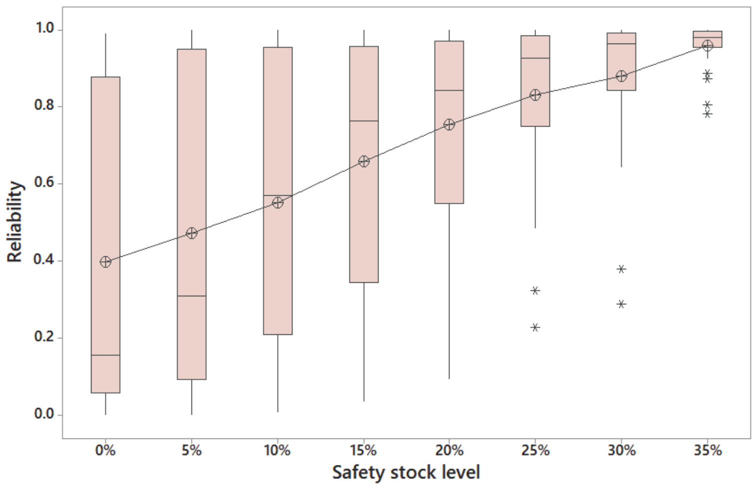


Figure 5: *Reliability values for different safety stock levels.*

Notice that the higher the safety stock level, the higher the average reliability index is. Moreover, increasing the safety stock level also contributes to reduce the variability in these reliability indices – i.e., increasing the safety stock has the expected effect of reducing the number of route failures, which in turn reduces the extra costs generated by corrective actions. Of course, increasing the safety stock level makes the solution more 'robust' against uncertainty (thus reducing the cost due to corrective actions), but it also requires the use of additional routes in the solution, which raises the cost (total time employed) of the final distribution plan. Therefore, this trade-off must be taken into account when finding the right level of safety stock for each individual instance.

Finally, Figure 6 shows the expected travel times, across all instances, for each safety stock level. The most relevant observation here, is that the expected cost (total travel time) can be reduced, on average, by using safety stock levels between 20% and 25% of the original capacity. Of course, the specific safety stock level to use will depend upon the actual instance as well as on the probability distribution employed to model the travel times. Still, the point here is that the use of safety stocks can contribute to reduce the total expected cost of the distribution plan by making this plan less sensitive to the risk of route failures.

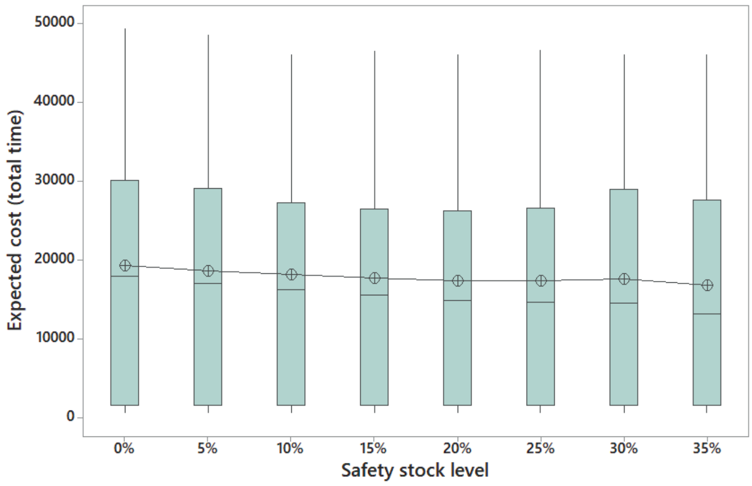


Figure 6: *Expected travel times for different safety stock levels.*

7. Conclusions and future research

The transportation sector is one of the most pollutant ones in modern societies. As a consequence, a number of government regulations have been set to promote the use of electric vehicles in order to reduce the air pollution. However, the current infrastructure of cities makes it difficult to fully develop green logistics and transportation practices. For instance, the use of electric vehicles for freight distribution has to deal with multiple obstacles, such as scattered network configuration and the technical limitations of those vehicles. So far, only a reduced number of works have studied the electric vehicle routing problem with stochastic travel times. Aiming at reducing this gap in the literature, the paper analyses the aforementioned problem considering also driving-range limitations, which might cause route failures when the vehicle runs out of battery.

Our methodology combines Monte Carlo simulation with a multi-start framework, which also integrates a biased-randomized constructive heuristic. Our simheuristic algorithm also makes use of safety stocks during the routing design stage, thus decreasing the risk of suffering route failures. In other words, we focus on constructing reliable solutions with a low risk of requesting corrective actions. Our results prove that using deterministic solutions in stochastic scenarios might lead to sub-optimal distribution plans that can be easily improved by using a simulation-optimization technique such as the one proposed here. They also illustrate how the use of the suitable energy safety stock levels during the routing design stage can increase the reliability of the distribution plans, thus reducing the total expected costs.

Some future lines can extend this work. In particular, we are interested in: (i) analysing the effect of preventive strategies – such as the ones already described in this paper – on the expected cost of the considered instances; (ii) extending our methodology (e.g., by hybridizing it with Petri nets) so it can also take into account possible correlations among travel times associated with different edges; (iii) extending our results to the heterogeneous fleet scenario, where vehicles might have different driving ranges and batteries; and (iv) including different sustainability dimensions related to environmental and social costs of these distribution activities, specially in the context of smart cities.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness (TRA2015-71883-REDT), and the Ibero-American Program for Science and Technology for Development (CYTED2014-515RT0489). Moreover, we appreciate the financial support of the Erasmus+ Program (2018-1-ES01-KA103-049767) as well as the support of the UPNA doctoral program.

References

Alvarez Fernandez, S., Ferone, D., Juan, A.A., Silva, D.G. and de Armas, J. (2018). A 2-stage biased-randomized iterated local search for the uncapacitated single allocation p-hub median problem. *Transactions on Emerging Telecommunications Technologies*, 29, e3418.

Bektaş, T. and Laporte, G. (2011). The pollution-routing problem. *Transportation Research Part B: Methodological*, 45, 1232–50.

Bianchi, L., Dorigo, M., Gambardella, L.M. and Gutjahr, W.J. (2009). A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing*, 8, 239–87.

Bibri, S.E. and Krogstie, J. (2017). Smart sustainable cities of the future: An extensive interdisciplinary literature review. *Sustainable Cities and Society*, 31, 183–212.

Bozorgi, A.M., Farasat, M. and Mahmoud. (2017). A. A time and energy efficient routing algorithm for electric vehicles based on historical driving data. *IEEE Transactions on Intelligent Vehicles*, 2, 308–20.

Bozorgi-Amiri, A., Jabalameli, M. and Al-e Hashem, S.M. (2013). A multi-objective robust stochastic programming model for disaster relief logistics under uncertainty. *OR spectrum*, 35, 905–33.

Cáceres-Cruz, J., Arias, P., Guimarans, D., Riera, D. and Juan, A.A. (2014). Rich vehicle routing problem: A survey. *ACM Computing Surveys*, 47, 1-28.

Christofides, N. (1976). The vehicle routing problem. *Revue française d'automatique, informatique, recherche opérationnelle. Recherche opérationnelle*, 10, 55–70.

Clarke, G. and Wright, J.W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12, 568–81.

De Armas, J., Juan, A.A., Marquès, J.M. and Pedroso, J.P. (2017). Solving the deterministic and stochastic uncapacitated facility location problem: from a heuristic to a simheuristic. *Journal of the Operational Research Society*, 68, 161–76.

Demir, E., Bektaş, T. and Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223, 346–59.

Desaulniers, G., Errico, F., Irnich, S. and Schneider, M. (2016). Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, 64, 1388–1405.

Dominguez, O., Guimarans, D., Juan, A.A. and de la Nuez, I. (2016a). A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls. *European Journal of Operational Research*, 255, 442–62.

Dominguez, O., Juan, A.A., Barrios, B., Faulin, J. and Agustin, A. (2016b). Using biased randomization for solving the two-dimensional loading vehicle routing problem with heterogeneous fleet. *Annals of Operations Research*, 236, 383–404.

Dominguez, O., Juan, A.A. and Faulin, J. (2014). A biased-randomized algorithm for the two-dimensional vehicle routing problem with and without item rotations. *International Transactions in Operational Research*, 21, 375–98.

Dominguez, O., Juan, A.A., de la Nuez Pestana, I.A. and Ouelhadj, D. (2016c). An ils-biased randomization algorithm for the two-dimensional loading hfvrp with sequential loading and items rotation. Journal of the Operational Research Society, 67, 37–53.

Erdoğan, S. and Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48, 100–14.

Eshtehadi, R., Fathian, M. and Demir, E. (2017). Robust solutions to the pollution-routing problem with demand and travel time uncertainty. *Transportation Research Part D: Transport and Environment*, 51, 351–63.

Eurostat. Energy, transport and environment indicators (2016). Available at: http://ec.europa.eu (accessed August, 2018).

Faulin, J., Gilibert, M., Juan, A.A., Vilajosana, X. and Ruiz, R. (2008). Sr-1: A simulation-based algorithm for the capacitated vehicle routing problem. In: *Simulation Conference, 2008. WSC 2008. Winter. IEEE*, 2708–16.

Faulin, J. and Juan, A.A. (2008). The algacea-1 method for the capacitated vehicle routing problem. *International Transactions in Operational Research*, 15, 599–621.

Felipe, A., Ortuño, M.T., Righini, G. and Tirado, G. (2014). A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E: Logistics and Transportation Review*, 71, 111–28.

Ferone, D., Gruler, A., Festa, P. and Juan, A.A. (2018). Enhancing and extending the classical GRASP framework with biased randomization and simulation. *Journal of the Operational Research Society*.

Gendreau, M., Ghiani, G. and Guerriero, E. (2015). Time-dependent routing problems: A review. *Computers & Operations Research*, 64, 189–97.

Golden, B.L., Wasil, E.A., Kelly, J.P. and Chao, I.M. (1998). The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results. In: *Fleet management and logistics*. Springer, 33–56.

Gonzalez-Neira, E.M., Ferone, D., Hatami, S. and Juan, A.A. (2017). A biased-randomized simheuristic for the distributed assembly permutation flowshop problem with stochastic process-ing times. *Simulation Modelling Practice and Theory*, 79(Supplement C), 23–36.

Grasas, A., Juan, A.A., Faulin, J., de Armas, J. and Ramalhinho, H. (2017). Biased randomization of heuristics using skewed probability distributions: a survey and some applications. *Computers & Industrial Engineering*, 110, 216–28.

Grasas, A., Juan, A.A. and Lourenço, HR. (2016). SimILS: a simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization. *Journal of Simulation*, 10, 69–77.

Hiermann, G., Puchinger, J., Ropke, S. and Hartl, R.F. (2016). The electric fleet size and mix vehicle routing problem with time windows and recharging stations. *European Journal of Operational Research*, 252, 995–1018.

Hof, J., Schneider, M. and Goeke, D. (2017). Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops. *Transportation Research Part B*, 97, 102–12.

Juan, A.A., Faulin, J., Grasman, S., Riera, D., Marull, J. and Mendez, C. (2011a). Using safety stocks and simulation to solve the vehicle routing problem with stochastic demands. *Transportation Research Part C: Emerging Technologies*, 19, 751–65.

Juan, A.A., Faulin, J., Jorba, J., Caceres, J. and Marquès, J.M. (2013). Using parallel & distributed computing for real-time solving of vehicle routing problems with stochastic demands. *Annals of Operations Research*, 207, 43–65.

Juan, A.A., Faulin, J., Jorba, J., Riera, D., Masip, D. and Barrios, B. (2011b). On the use of monte carlo simulation, cache and splitting techniques to improve the clarke and wright savings heuristics. *Journal of the Operational Research Society*, 62, 1085–97.

Juan, A.A., Faulin, J., Pérez-Bernabeu, E. and Jozefowiez, N. (2014a). Horizontal cooperation in vehicle routing problems with backhauling and environmental criteria. *Procedia - Social and Behavioral Sciences*, 111, 1133–41.

Juan, A.A., Goentzel, J. and Bektaş, T. (2014b). Routing fleets with multiple driving ranges: Is it possible to use greener fleet configurations? *Applied Soft Computing Journal*, 21, 84–94.

Juan, A.A., Lourenço, H.R., Mateo, M., Luo, R. and Castella, Q. (2014c). Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues. *International Transactions in Operational Research*, 21, 103–26.

Juan, A.A., Mendez, C.A., Faulin, J., De Armas, J. and Grasman, S.E. (2016). Electric vehicles in logistics and transportation: A survey on emerging environmental, strategic, and operational challenges. *Energies*, 9, 1–21.

Juan, A.A., Pascual, I., Guimarans, D. and Barrios, B. (2015). Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem. *International Transactions in Operational Research*, 22, 647–67.

Koç, C. and Karaoglan, I. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing Journal*, 39, 154–64.

Li, F., Golden, B. and Wasil, E. (2005). Very large-scale vehicle routing: New test problems, algorithms, and results. *Computers and Operations Research*, 32, 1165–79.

Lin, C., Choy, K.L., Ho, G.T.S., Chung, S.H. and Lam, H.Y. (2014). Survey of green vehicle routing problem: past and future trends. *Expert Systems with Applications*, 41(4 part 1), 1118–38.

McKinnon, A., Cullinane, S., Browne, M. and Whiteing, A. (2015). Green Logistics: Improving the Environmental Sustaintability of Logistics. 2nd ed. London: Kogan Page. ISBN: 978-0-7494-6625-1.

Pérez-Bernabeu, E., Juan, A.A., Faulin, J. and Barrios, B.B. (2015). Horizontal cooperation in road transportation: a case illustrating savings in distances and greenhouse gas emissions. *International Transactions in Operational Research*, 22, 585–606.

Ritzinger, U., Puchinger, J. and Hartl, R.F. (2015). A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 7543, 1–17.

Sawik, B., Faulin, J. and Pérez-Bernabeu, E. (2017a). A multicriteria analysis for the green VRP: a case discussion for the distribution problem of a Spanish retailer. *Transportation Research Procedia*, 22, 305–13.

Sawik, B., Faulin, J. and Pérez-Bernabeu, E. (2017b). Multi-criteria optimization for fleet size with environmental aspects. *Transportation Research Procedia*, 27, 61–8.

Sawik, B., Faulin, J. and Pérez-Bernabeu, E. (2017c). Selected multi criteria green vehicle routing problems. In: *Applications of Management Science*, 57–83.

Schneider, M., Stenger, A. and Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, 48, 500–20.

Schneider, M., Stenger, A. and Hof, J. (2015). An adaptive VNS algorithm for vehicle routing problems with intermediate stops. *OR Spectrum*, 37, 353–87.

Serrano-Hernández, A., Juan, A.A., Faulin, J. and Perez-Bernabeu, E. (2017). Horizontal collaboration in freight transport: Concepts, benefits, and environmental challenges. *Statistics and Operations Research Transactions*, 41, 1–22.

Shao, S., Guan, W. and Bi. J. (2018). Electric vehicle-routing problem with charging demands and energy consumption. *The Institute of Engineering and Technology*, 12, 202–12.

The World Bank. (2018). Connecting to compete: trade logistics in the global economy. Technical Report; World Bank; Washington, DC. Available at: https://elibrary.worldbank.org/doi/abs/10.1596/29971 (accessed: August 2018).

Toth, P. and Vigo, D. (2014). Vehicle routing: problems, methods, and applications. *SIAM*.

Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Vidal, T. and Subramanian, A. (2017). New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research*, 257, 845–58.