INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 26 (2019) 1211–1243
DOI: 10.1111/itor.12620

The electric vehicle routing problem with shared charging stations

Çağrı Koç[a], Ola Jabali[b], Jorge E. Mendoza[c] and Gilbert Laporte[d]

[a]*Department of Business Administration, Social Sciences University of Ankara, Hükümet Meydanı, 06050 Ulus, Ankara, Turkey*
[b]*Dipartimento di Elettronica, Informazione e Bioingegneria Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133 Milano, Italy*
[c]*HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal H3T 2A7, Canada*
[d]*CIRRELT, Canada Research Chair in Distribution Management and HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal H3T 2A7, Canada*
E-mail: cagri.koc@asbu.edu.tr [Koç]; ola.jabali@polimi.it [Jabali]; jorge.mendoza@hec.ca [Mendoza]; gilbert.laporte@cirrelt.ca [Laporte]

Received 7 August 2017; received in revised form 13 November 2018; accepted 15 November 2018

Abstract

We introduce the electric vehicle routing problem with shared charging stations (E-VRP-SCS). The E-VRP-SCS extends the electric vehicle routing problem with nonlinear charging function (E-VRP-NL) by considering several companies that jointly invest in charging stations (CSs). The objective is to minimize the sum of the fixed opening cost of CSs and the drivers cost. The problem consists of deciding the location and technology of the CSs and building the routes for each company. It is solved by means of a multistart heuristic that performs an adaptive large neighborhood search coupled with the solution of mixed integer linear programs. It also contains a number of advanced efficient procedures tailored to handle specific components of the E-VRP-SCS. We perform extensive computational experiments on benchmark instances. We assess the competitiveness of the heuristic on the E-VRP-NL and derive 38 new best known solutions. New benchmark results on the E-VRP-SCS are presented, solved, and analyzed.

Keywords: ehicle routing; electric vehicles; multidepot; ALNS; nonlinear charging function

1. Introduction

In recent years, we have witnessed a growing interest in the integration of environmental aspects within the vehicle routing problem (VRP) and its variants (Demir et al., 2014; Eglese and Bektaş, 2014; Lin et al., 2014). The electric VRP (E-VRP), introduced as a consequence of this research effort, considers the technical limitations of electric vehicles (EVs), such as their limited payload and driving range. In particular, E-VRP solutions frequently include routes with planned detours to

out-of-the-depot charging stations (CSs). This is especially true in mid-haul applications arising in rural and semiurban areas because in these cases the distance covered by the routes on a single day is often higher than the driving range. While en route recharge is performed mainly for feasibility reasons, it also offers economical benefits. For instance, in a recent study carried for the French electricity giant ENEDIS, Villegas et al. (2018) showed that by allowing en route charging, in certain settings, the company could save up to 14% on operational routing costs.

En route charging can be performed in public or private CSs. The latter are usually located in facilities owned by the company operating the routes (e.g., administrative buildings and secondary depots). With a few exceptions such as Sweda et al. (2017) and Kullman et al. (2017), most papers in the E-VRP literature assume private CSs (Montoya et al., 2017). In practice, installing en route charging infrastructure requires significant investments. For example, according to the U.S. Department of Energy, the cost of acquiring a fast charger in the United States may vary between USD 14K and USD 91K depending on the charger model (Smith and Castellano, 2015). For companies operating small EV fleets, the economical benefits of en route charging may be quickly offset by the long payback times of these large investments. In these cases, one possible strategy to leverage the investment is to explore mutualization opportunities. In this paper, we model the situation where the investment in out-of-depot charging infrastructure is jointly made by a number of companies, which ultimately share it on a daily basis.

Our aim is to develop a heuristic to efficiently solve the electric vehicle routing problem with shared CSs (E-VRP-SCS), which combines location and routing decisions. The location component includes decisions on where to open CS and which type of technology to install in them, while the routing component builds the routes for each company using the CSs established by the first component.

1.1. A brief literature review

Erdoğan and Miller-Hooks (2012) introduced the green VRP (GVRP) in which alternative fuel vehicles are considered. The problem aims to minimize the total traveled distance where fuel is consumed at a given rate per traveled distance and the tank of the vehicle can be refueled at alternative fuel stations. The authors proposed two construction heuristics whose performance was tested on instances with up to 500 customers. Koç and Karaoglan (2016) developed a branch-and-cut algorithm for the GVRP, which combines several valid inequalities and includes a simulated annealing based heuristic algorithm. The method can optimally solve instances with up to 20 customers. Montoya et al. (2016) developed a multispace sampling two-phase heuristic for the GVRP. The first phase applies randomized route-first cluster-second heuristics, and the second phase is based on the solution of a set partitioning formulation (SPF). Bartolini and Andelmin (2017) modeled the GVRP as a set partitioning problem and developed an exact algorithm for it. The authors strengthened their formulation by adding a number of valid inequalities. They optimally solved instances with up to 110 customers. Andelmin and Bartolini (2016) developed a multistart local search heuristic for the GVRP and solved instances with up to 470 customers.

Felipe et al. (2014) extended the GVRP to EVs by allowing partial recharges using multiple technologies. In their problem, EVs have capacity and route duration limits. The authors developed construction and local search algorithms, as well as a simulated annealing metaheuristic. Schneider

et al. (2014) introduced the E-VRP with time windows (E-VRPTW) and recharging stations and proposed a hybrid metaheuristic that integrates variable neighborhood search and tabu search. The performance of the method was tested on the GVRP instances of Erdoğan and Miller-Hooks (2012), on the multidepot VRP with interdepot routes instances of Crevier et al. (2007) and of Tarantilis et al. (2008), and on newly generated instances for the E-VRPTW. Goeke and Schneider (2015) solved heterogeneous fleet VRP in the context of the E-VRPTW, where both EVs and internal combustion engine vehicles are considered. The objective was to minimize the energy consumption that depends on speed, gradient, and cargo load distribution. The authors applied an adaptive large neighborhood search (ALNS) algorithm for this problem. Desaulniers et al. (2016) developed branch-price-and-cut algorithms for the E-VRPTW and also extended the problem by considering four recharging strategies: a single recharge per route when the batteries are fully recharged, multiple recharges per route and full recharges, at most a single recharge per route and partial recharges per route, and multiple and partial recharges. Hiermann et al. (2016) introduced the fleet size and mix VRP with time windows where EVs are considered. The authors developed a branch-and-price algorithm and a hybrid metaheuristic that combines ALNS and labeling procedures. Keskin and Çatay (2016) studied the E-VRPTW with partial recharge strategies and developed an ALNS algorithm with several new mechanisms adapted to the problem. Same authors later developed a matheuristic algorithm to solve the E-VRPTW with fast chargers (Keskin and Çatay, 2018).

Depot location and vehicle routing are two interdependent decisions. Their joint study has evolved into what is now commonly known as the location-routing problem (LRP) (for recent reviews, see Prodhon and Prins, 2014; Albareda-Sambola, 2015; Drexl and Schneider, 2015). Yang and Sun (2015) studied the EV battery swap stations (BSSs) LRP which aims to determine the locations of BSSs, as well as a routing plan. The authors proposed a hybrid algorithm including a modified sweep heuristic, iterated greedy search, tabu search, and several improvement procedures. Hof et al. (2017) later studied the same problem and developed an adaptive variable neighborhood search algorithm which obtained several new best known solutions (BKSs). More recently, Schiffer and Walther (2017a) studied the electric LRP with time windows and partial recharging. The problem considers one type of technology at the CS, and vehicles can charge at any of the nodes. The problem considers three separate objectives: minimization of distance, number of vehicles, or number of CSs. The authors proposed a mathematical formulation for the problem and presented exact results on small-sized instances with up to 15 customers. In the context of city logistics, Schiffer and Walther (2017b) introduced the LRP with intraroute facilities where the location of the depots is known and the location of facilities for intermediate stops has to be determined. These facilities are not depots and do not necessarily coincide with customer locations. They can be visited on a route and enable the vehicle to stay operational by replenishing, loading, or unloading certain resources like energy, fuel, freight, or waste. The authors proposed an ALNS heuristic including dynamic programming components. They first solved the problem of Schiffer and Walther (2017a), and then applied their method to real-world instances obtained from a German retail company. Schiffer et al. (2018a) studied the LRP with intraroute facilities with combined facilities offering different replenishment services. The authors developed an ALNS that uses a lower bounding procedure. Schiffer and Walther (2018) introduced a robust LRP for strategic network design of electric logistics fleets. It considers simultaneous decisions on locating CSs and vehicle routes, and the uncertain customer patterns with respect to the spatial customer distribution, demand, and service time windows. CS prices heavily influence the viability of electric commercial vehicles in mid-haul logistics (Schiffer

et al., 2017). The reader is referred to the recent survey paper of Schiffer et al. (2018b) for a detailed coverage on LRP with intraroute facilities and its variants.

All of the papers described use linear charging functions to approximate the charging behavior of the battery. Recently, Montoya et al. (2017) introduced the E-VRP with nonlinear charging function (E-VRP-NL), which better reflects the behavior of the charging process. They used a piecewise linear approximation of the nonlinear charging function. The authors developed a hybrid metaheuristic that combines iterated local search and an SPF. Their results indicated that neglecting nonlinear charging may lead to infeasible or overly expensive solutions.

Table 1 summarizes the main papers on the GVRP, E-VRP, and their variants. For further details on the E-VRP and electric vehicular technology, the reader is referred to the papers of Afroditi et al. (2014), Juan et al. (2016), Margaritis et al. (2016), Pelletier et al. (2016), Pelletier et al. (2017), and Schiffer et al. (2017).

1.2. Scientific contributions and structure of the paper

We make two main contributions. We first introduce the E-VRP-SCS as an extension of the E-VRP-NL, using a comprehensive objective function that minimizes the fixed cost of opening CSs as well as driver costs. Our second contribution is the development of a heuristic that successfully integrates within a multistart scheme an ALNS with an exact vehicle-charging optimization component applied to a fixed route. We also introduce several new ALNS operators, as well as initialization, intensification, and diversification procedures.

The remainder of this paper is structured as follows. In Section 2, we describe the problem and introduce a mixed integer programming formulation to solve it. In Section 3, we present our solution method and discuss in detail its algorithmic components. In Section 4, we report extensive computational experiments carried on instances adapted from the literature. In this section we also present state-of-the-art results for the closely related E-VRP-NL. Finally, in Section 5 we conclude the paper and outline open research perspectives.

2. Problem definition and formulation

In our problem, several companies jointly invest in purchasing and installing CSs. To this end, we consider a set of potential locations for the shared CSs and multiple charging technologies (e.g., slow, moderate, and fast) with a nonlinear charging function (Pelletier et al., 2017). Each company runs a distribution system involving (a) a homogeneous and unlimited fleet of EVs and (b) a depot with a preassigned set of customers with service times.

The problem consists of establishing the location and technology of a subset of CSs, and determining a set of vehicle routes such that all EVs start and end their routes at the same depot, each customer is visited exactly once by a vehicle, and each route satisfies a maximum duration limit. Furthermore, we also decide for each vehicle when, where, and how much to charge. For the sake of simplicity, in the reminder of the paper we refer to establishing the location and technology of a CS as "opening" a CS. The objective is to minimize the total cost, which is made up of two components: the opening costs of CSs and the driver costs. The latter is based on the total travel time, customer

Table 1
Summary of the literature on the GVRP, E-VRP and its variants

Reference	Algorithm	Partial recharges	Time windows	Location of CSs	Multiple technologies	Battery swap	Fixed charging	Linear charging	Nonlinear charging	Multiple depots
Erdoğan and Miller-Hooks (2012)	Heuristic						•			
Felipe et al. (2014)	Heuristic	•			•			•		
Schneider et al. (2014)	Heuristic		•					•		
Goeke and Schneider (2015)	Heuristic		•					•		
Yang and Sun (2015)	Heuristic			•		•				
Bartolini and Andelmin (2017)	Exact						•			
Desaulniers et al. (2016)	Exact	•	•					•		
Hiermann et al. (2016)	Exact, heuristic		•					•		
Keskin and Çatay (2016)	Heuristic	•	•					•		
Koç and Karaoglan (2016)	Exact						•			
Montoya et al. (2016)	Heuristic						•			
Montoya et al. (2017)	Heuristic	•			•				•	
Hof et al. (2017)	Heuristic			•		•				
Schiffer and Walther (2017a)	Exact	•						•		
Schiffer and Walther (2017b)	Heuristic	•	•	•				•		
This paper	Heuristic	•		•	•				•	•

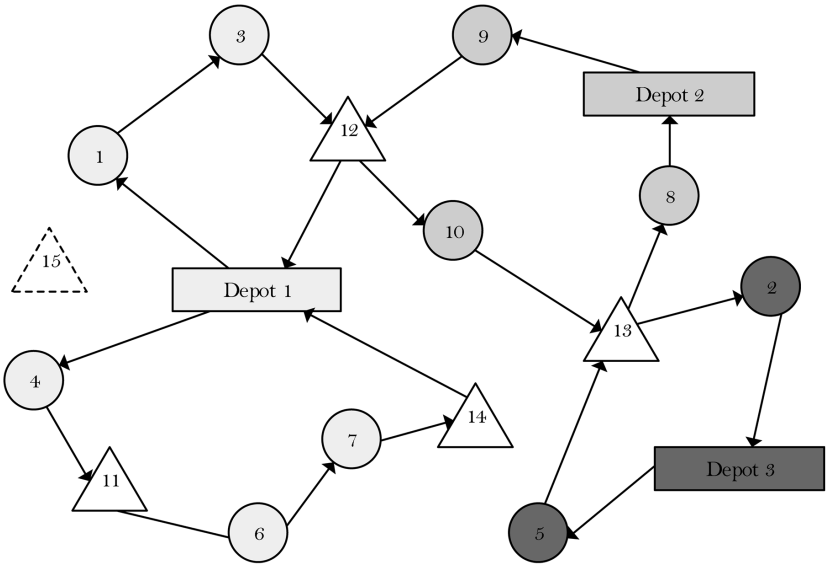

Fig. 1. An illustration of the problem.

service time, and charging time. Each depot has its own customers. It is assumed that all EVs are fully charged upon leaving their depot. An illustration of the problem is presented in Fig. 1 where circles denote customers, triangles with full lines denote opened CSs, and triangles with dashed lines denote potential locations were no CSs were opened.

It is worth noting that the E-VRP-SCS is a strategic problem aiming to establish the locations and technologies of CSs. Accounting for these strategic decisions, operational E-VRPs deal with issues such as synchronizing access to CSs (Froger et al., 2017), power grid constraints (Pelletier et al., 2018), and battery degradation (Pelletier et al., 2017).

2.1. Notation

The set of all depots and customers are denoted by $\mathcal{I}$. The sets of all potential CS locations and companies are denoted by $\mathcal{S}$ and $\mathcal{C}$, respectively. The sets $\mathcal{I}_c^0$ and $\mathcal{I}_c^m$ represent the depot and customers of company $c \in \mathcal{C}$, respectively. Let $\mathcal{I} = \bigcup_{c\in\mathcal{C}} \mathcal{I}_c$ and $\mathcal{I}_c = \mathcal{I}_c^0 \cup \mathcal{I}_c^m$. We introduce the set $\mathcal{S}'$ that contains the set of $\mathcal{S}$ and β additional copies of each CS (i.e., $|\mathcal{S}'| = |\mathcal{S}|(1+\beta)$), where the value of $1+\beta$ corresponds to the number of times each CS can be visited. The problem is defined on a complete directed graph $G = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N} = \mathcal{I} \cup \mathcal{S}'$ is the set of nodes, and $\mathcal{A} = \{(i,j) : i \in \mathcal{I}_c, j \in \mathcal{I}_c, c \in \mathcal{C}\} \cup \{(i,j) : i \in \mathcal{I}, j \in \mathcal{S}'\}$ is the set of arcs.

Each customer $i \in \mathcal{I}_c^m$ of company $c \in \mathcal{C}$ has a service time p_i^c. The travel time on arc $(i,j) \in \mathcal{A}$ is denoted by t_{ij}. The maximum route duration for each vehicle is denoted by T_{max}, which is equal for all companies. The battery capacity of an EV is denoted by Q (kWh). Each CS $s \in \mathcal{S}'$ has a nonlinear charging function F_s, which is modeled using a piecewise linear approximation (see Montoya et al., 2017). Function F_s uses the state of charge (SoC) when the EV arrives at a CS and the charging time as input parameters. It is characterized by a set of break points $\mathcal{B}_s = \{0, \ldots, \bar{b}_s\}$, assuming

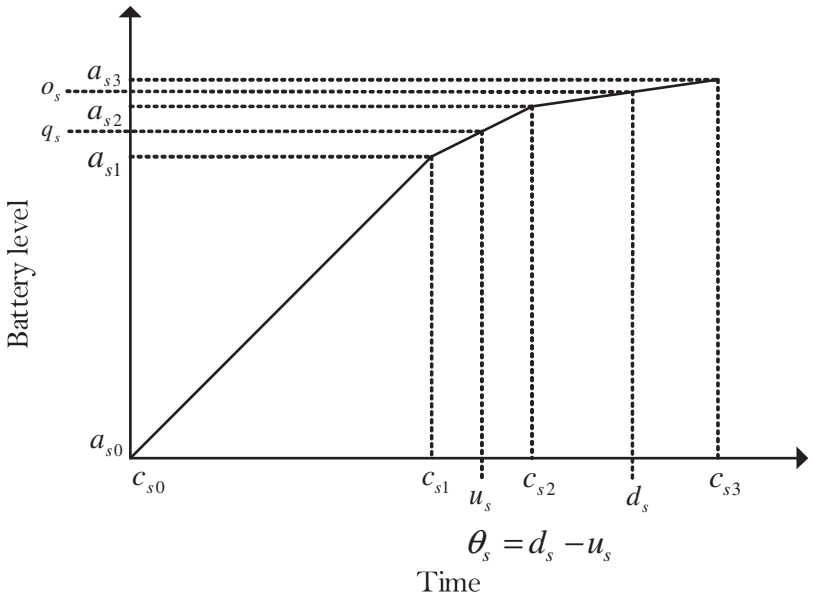

Fig. 2. Battery charge levels and charging times.

an empty battery, reaching a charging level a_{sb} at break point $b \in \mathcal{B}_s$ requires c_{sb} time units at CS $s \in \mathcal{S}'$. The energy consumption of an EV on arc $(i, j) \in \mathcal{A}$ is denoted by e_{ij}. The cost of opening CS $s \in \mathcal{S}$ is denoted by f^s. This cost includes the purchasing and installation of the CS and it depends on the charging technology (e.g., slow, moderate, fast). The driver cost is denoted by f^d, which is multiplied by the sum of total travel time, charging time, and service time.

To formulate the problem, we first define the following binary variables. Let x_{ij} be equal to 1 if an EV travels on arc $(i, j) \in \mathcal{A}$, and to 0 otherwise. Let y^s be equal to 1 if station s is opened, and to 0 otherwise. Let z_{sb} be equal to 1 if the SoC of an EV arriving at CS $s \in \mathcal{S}'$ is between $a_{s,b-1}$ and a_{sb}, with $b \in \mathcal{B}_s$, $b > 0$, and to 0 otherwise. Let w_{sb} be equal to 1 if the SoC of an EV leaving $s \in \mathcal{S}'$ is between $a_{s,b-1}$ and a_{sb}, with $b \in \mathcal{B}_s$, $b > 0$, and to 0 otherwise.

We now define the following continuous variables. Let v_i be the SoC when an EV departs from node $i \in \mathcal{N}$. Let q_s be the SoC when an EV arrives at s. Let o_s be the SoC when an EV departs from s. Let u_s be the time needed to charge from 0 to q_s according to the technology available at s. Let d_s be the time needed to charge from 0 to o_s according to the technology available at s. Let $\theta_s = d_s - u_s$ be the charging time spent at s (see Fig. 2). Let τ_i be the time when an EV departs from node $i \in \mathcal{N}$. Finally, let α_{sb} and λ_{sb} be auxiliary variables that allow to express (u_s, q_s) and (d_s, o_s) as linear combinations of the break points $\{(c_{sb}, a_{sb})\}_{b \in \mathcal{B}_s}$, respectively. We summarize all the notation of the problem in Table 2.

Figure 2 provides an example of a charging process at a CS. In this example, the battery SoC parameters of the charging function are $a_{s1} = 13,600$ Wh, $a_{s2} = 15,200$ Wh, and $a_{s3} = 16,000$ Wh. Time parameters of the charging function are $c_{s1} = 0.31$, $c_{s2} = 0.39$, and $c_{s3} = 0.51$. Suppose that a vehicle arrives to the station with $q_s = 14,400$ Wh, and departs with $o_s = 15,600$ Wh, then $u_s = 0.35$, and $d_s = 0.45$. Therefore, $\theta_s = 0.1$, $z_{s2} = 1$, and $w_{s3} = 1$.

Table 2
Summary of notation

Sets	
$\mathcal{C}$	Set of companies
$\mathcal{I}$	Set of all depots and customers
$\mathcal{I}_c$	Set of depot and customers of company $c \in \mathcal{C}$
$\mathcal{I}_c^0$	Set containing only the depot of company $c \in \mathcal{C}$
$\mathcal{I}_c^m$	Set of customers of company $c \in \mathcal{C}$
$\mathcal{S}$	Set of all potential CSs
$\mathcal{S}'$	Set of all potential CSs and their copies
$\mathcal{B}_s$	Set of break points of the charging function ($\mathcal{B}_s = \{0, \ldots, \bar{b}_s\}$) at CS $s \in \mathcal{S}'$
Parameters	
p_i^c	Service time of customer $i \in \mathcal{I}_c^m$
t_{ij}	Travel time on arc $(i, j) \in \mathcal{A}$
T_{max}	Maximum tour duration
Q	Battery capacity of an EV
F_s	Charging function of an EV at CS $s \in \mathcal{S}'$
c_{sb}	Charging time at break point $b \in \mathcal{B}_s$ in F_s
a_{sb}	Charging level at break point $b \in \mathcal{B}_s$ in F_s
e_{ij}	Energy consumption of an EV on arc $(i, j) \in \mathcal{A}$
f^s	Cost of opening CS $s \in \mathcal{S}$ (monetary unit)
f^d	Driver cost (monetary unit per time unit)
Binary variables	
x_{ij}	1 if an EV travels on arc $(i, j) \in \mathcal{A}$, 0 otherwise
y^s	1 if station $s \in \mathcal{S}$ is opened, 0 otherwise
z_{sb}	1 if the SoC is between $a_{s,b-1}$ and a_{sb}, with $b \in \mathcal{B}_s$, $b > 0$, when an EV arrives at CS $s \in \mathcal{S}'$, 0 otherwise
w_{sb}	1 if the SoC is between $a_{s,b-1}$ and a_{sb}, with $b \in \mathcal{B}_s$, $b > 0$, when an EV departs from CS $s \in \mathcal{S}'$, 0 otherwise
Continuous variables	
v_i	SoC when an EV departs from node $i \in \mathcal{N}$
q_s	SoC when an EV arrives at CS $s \in \mathcal{S}'$
o_s	SoC when an EV departs from CS $s \in \mathcal{S}'$
u_s	Time needed to charge from 0 to q_s according to the technology available at CS $s \in \mathcal{S}'$
d_s	Time needed to charge from 0 to o_s according to the technology available at CS $s \in \mathcal{S}'$
θ_s	Charging time spent at CS $s \in \mathcal{S}'$ ($\theta_s = d_s - u_s$)
τ_i	Time at which an EV departs from node $i \in \mathcal{N}$
α_{sb}	Auxiliary variable allowing to express (u_s, q_s) as a linear combination of the break points $\{(c_{sb}, a_{sb})\}_{b \in \mathcal{B}_s}$
λ_{sb}	Auxiliary variable allowing to express (d_s, o_s) as a linear combination of the break points $\{(c_{sb}, a_{sb})\}_{b \in \mathcal{B}_s}$

2.2. Mixed integer linear programming formulation

The mathematical formulation of the problem, which is an extension of the formulation for the E-VRP-NL by Montoya et al. (2017), is then:

$$\text{Minimize} \sum_{s\in\mathcal{S}} f^s y^s + f^d\left(\sum_{i,j\in\mathcal{A}} t_{ij}x_{ij} + \sum_{s\in\mathcal{S}'}\theta_s + \sum_{c\in\mathcal{C}}\sum_{i\in\mathcal{I}_c} p_i^c\right) \tag{1}$$

subject to

$$\sum_{j\in\mathcal{I}_c\cup\mathcal{S}', i\neq j} x_{ij} = 1 \qquad c\in\mathcal{C}, i\in\mathcal{I}_c^m \tag{2}$$

$$\sum_{c\in\mathcal{C}}\sum_{j\in\mathcal{I}_c\cup\mathcal{S}', s\neq j} x_{sj} \leq 1 \qquad s\in\mathcal{S}' \tag{3}$$

$$\sum_{j\in\mathcal{I}_c\cup\mathcal{S}', i\neq j} x_{ji} - \sum_{j\in\mathcal{I}_c\cup\mathcal{S}'} x_{ij} = 0 \qquad c\in\mathcal{C}, i\in\mathcal{I}_c\cup\mathcal{S}' \tag{4}$$

$$e_{ij}x_{ij} - \left(1-x_{ij}\right)Q \leq v_i - v_j \leq e_{ij}x_{ij} + \left(1-x_{ij}\right)Q \qquad c\in\mathcal{C}, i\in\mathcal{I}_c\cup\mathcal{S}', j\in\mathcal{I}_c^m \tag{5}$$

$$e_{is}x_{is} - \left(1-x_{is}\right)Q \leq v_i - q_s \leq e_{is}x_{is} + \left(1-x_{is}\right)Q \qquad c\in\mathcal{C}, i\in\mathcal{I}_c\cup\mathcal{S}', s\in\mathcal{S}' \tag{6}$$

$$v_i \geq e_{ij}x_{ij} \qquad c\in\mathcal{C}, i\in\mathcal{I}_c\cup\mathcal{S}', j\in\mathcal{I}_c^0 \tag{7}$$

$$v_s = o_s \qquad s\in\mathcal{S}' \tag{8}$$

$$v_i = Q \qquad c\in\mathcal{C}, i\in\mathcal{I}_c^0 \tag{9}$$

$$q_s \leq o_s \qquad s\in\mathcal{S}' \tag{10}$$

$$q_s = \sum_{b\in\mathcal{B}_s} \alpha_{sb}a_{sb} \qquad s\in\mathcal{S}' \tag{11}$$

$$u_s = \sum_{b\in\mathcal{B}_s} \alpha_{sb}c_{sb} \qquad s\in\mathcal{S}' \tag{12}$$

$$\sum_{b\in\mathcal{B}_s} \alpha_{sb} = \sum_{b\in\mathcal{B}_s} z_{sb} \qquad s\in\mathcal{S}' \tag{13}$$

$$\sum_{b\in\mathcal{B}_s} z_{sb} = \sum_{c\in\mathcal{C}}\sum_{j\in\mathcal{I}_c\cup\mathcal{S}'} x_{sj} \qquad s\in\mathcal{S}' \tag{14}$$

$$\alpha_{s0} \leq z_{s1} \qquad s\in\mathcal{S}' \tag{15}$$

$$\alpha_{sb} \leq z_{sb} + z_{s,b+1} \qquad s\in\mathcal{S}', b\in\mathcal{B}_s\backslash\{0,\bar{b}_s\} \tag{16}$$

$$\alpha_{s\bar{b}_s} \leq z_{s\bar{b}_s} \qquad s\in\mathcal{S}' \tag{17}$$

$$o_s = \sum_{b\in\mathcal{B}_s} \lambda_{sb}a_{sb} \qquad s\in\mathcal{S}' \tag{18}$$

$$d_s = \sum_{b \in \mathcal{B}_s} \lambda_{sb} c_{sb} \qquad s \in \mathcal{S}' \tag{19}$$

$$\sum_{b \in \mathcal{B}_s} \lambda_{sb} = \sum_{b \in \mathcal{B}_s} w_{sb} \qquad s \in \mathcal{S}' \tag{20}$$

$$\sum_{b \in \mathcal{B}_s} w_{sb} = \sum_{c \in \mathcal{C}} \sum_{j \in \mathcal{I}_c \cup \mathcal{S}'} x_{sj} \qquad s \in \mathcal{S}' \tag{21}$$

$$\lambda_{s0} \leq w_{s1} \qquad s \in \mathcal{S}' \tag{22}$$

$$\lambda_{sb} \leq w_{sb} + w_{s,b+1} \qquad s \in \mathcal{S}', b \in \mathcal{B}_s \backslash \{0, \bar{b}_s\} \tag{23}$$

$$\lambda_{s\bar{b}_s} \leq w_{s\bar{b}_s} \qquad s \in \mathcal{S}' \tag{24}$$

$$\theta_s = d_s - u_s \qquad s \in \mathcal{S}' \tag{25}$$

$$\tau_i + (t_{ij} + p_j) x_{ij} - T_{max}(1 - x_{ij}) \leq \tau_j \qquad c \in \mathcal{C}, i \in \mathcal{I}_c \cup \mathcal{S}', j \in \mathcal{I}_c^m \tag{26}$$

$$\tau_i + \theta_s + t_{is} x_{is} - (S_{max} + T_{max})(1 - x_{is}) \leq \tau_s \qquad c \in \mathcal{C}, i \in \mathcal{I}_c \cup \mathcal{S}', s \in \mathcal{S}' \tag{27}$$

$$\tau_j + t_{ji} \leq T_{max} \qquad c \in \mathcal{C}, j \in \mathcal{I}_c \cup \mathcal{S}', i \in \mathcal{I}_c^0 \tag{28}$$

$$\tau_i \leq T_{max} \qquad c \in \mathcal{C}, i \in \mathcal{I}_c^0 \tag{29}$$

$$x_{ij} = 0 \qquad i, j \in \mathcal{S}' : \rho_{ij} = 1 \tag{30}$$

$$\tau_i \geq \tau_j \qquad i, j \in \mathcal{S}' : \rho_{ij} = 1, j \leq i \tag{31}$$

$$\tau_j \leq T_{max} \sum_{i \in \mathcal{I}_c \cup \mathcal{S}'} x_{ij} \qquad j \in \mathcal{S}', c \in \mathcal{C} \tag{32}$$

$$\sum_{c \in \mathcal{C}} \sum_{i \in \mathcal{I}_c \cup \mathcal{S}'} x_{is_1} \geq \sum_{c \in \mathcal{C}} \sum_{j \in \mathcal{I}_c \cup \mathcal{S}'} x_{js_2} \qquad s_1, s_2 \in \mathcal{S}' : \rho_{s_1 s_2} = 1, s_1 \leq s_2 \tag{33}$$

$$\sum_{c \in \mathcal{C}} \sum_{i \in \mathcal{I}_c \cup \mathcal{S}'} \sum_{s' \in \mathcal{S}'} x_{is'} \leq y_s \qquad s \in \mathcal{S} \tag{34}$$

$$x_{ij} \in \{0, 1\} \qquad i, j \in \mathcal{A} \tag{35}$$

$$y^s \in \{0, 1\} \qquad s \in \mathcal{S}' \tag{36}$$

$$z_{sb} \in \{0, 1\}, w_{sb} \in \{0, 1\} \qquad s \in \mathcal{S}', b \in \mathcal{B}_s \tag{37}$$

$$\tau_i \geq 0, v_i \geq 0 \qquad c \in \mathcal{C}, i \in \mathcal{I}_c \cup \mathcal{S}' \tag{38}$$

$$u_s \geq 0, d_s \geq 0, \theta_s \geq 0, q_s \geq 0, o_s \geq 0 \qquad s \in \mathcal{S}' \tag{39}$$

$$\alpha_{sb} \geq 0, \lambda_{sb} \geq 0 \qquad s \in \mathcal{S}', b \in \mathcal{B}_s. \tag{40}$$

The objective function (1) minimizes the total costs of opening CSs, and the total driver cost based on the total travel time, charging time and service time.

Constraints (2) guarantee that each customer is visited exactly once. Constraints (3) impose that each copy of CS be visited at most once. Constraints (4) ensure the flow conservation. Constraints (5) and (6) compute the battery SoC at each node. Constraints (7) ensure that EV has a sufficient energy to reach the depot. Constraints (8) update the charging level upon leaving a CS. Constraints (9) impose that the battery level be equal to the EV battery capacity at the depot. Constraints (10) ensure that the arrival SoC of an EV at a CS does not exceed the departure SoC. Constraints (11)–(17) define the SoC of an EV and its corresponding charging time when an EV arrives at CS. Constraints (18)–(24) define the SoC of an EV and its corresponding charging time when an EV departs from a CS. Constraints (25) compute the time spent at a CS. Constraints (26) and (27) track the departure time at each node where $S_{max} = \max_{s \in \mathcal{S}'}\{c_{s,\bar{b}_s}\}$. Constraints (28) and (29) impose the maximum tour duration limit. Constraints (30)–(33) avoid the symmetry generated by the copies of the CSs. The binary parameter ρ_{ij} is equal to 1 if $i, j \in \mathcal{S}'$ represent the same CS, and to 0 otherwise. Constraints (34) determine whether a station is opened or not. Constraints (35)–(40) define the domains of the decision variables.

3. A heuristic algorithm

We have developed a heuristic algorithm to solve the E-VRP-SCS. We use the ALNS framework proposed by Ropke and Pisinger (2006a, 2006b) and used by many other papers (Cherkesly et al., 2015; Koç et al., 2016; Hemmati and Hvattum, 2017), which is based on a iterative procedure of removal and insertion of customers. Candidate solutions are accepted or rejected according to a probabilistic simulated annealing criterion. The need to insert CSs is established by ensuring the feasibility of the route, which in turn necessitates solving the fixed route vehicle-charging problem (FRVCP). Given a route $\Pi = \{\pi(0), \pi(1), \ldots, \pi(i), \ldots, \pi(n_r - 1), \pi(n_r)\}$, where $\pi(0)$ and $\pi(n_r)$ are depots and $\pi(1), \ldots, \pi(i), \ldots, \pi(n_r - 1)$ are customers, the FRVCP determines the charging decisions that minimize the sum of the charging times and detour times. For more details and for the MILP formulation of the FRVCP, we refer the reader to Montoya et al. (2017). In the ALNS, we only consider feasible routes, which is done by solving the FRCVP either (exactly) using CPLEX or (heuristically) using the greedy fixed route vehicle-charging heuristic (GFRVCH) of Montoya et al. (2017). The ALNS is further enhanced using a multistart scheme. Finally, a set of promising routes is maintained through the search and we periodically solve a set partitioning problem aimed at improving the solution to the E-VRP-SCS by selecting a subset of these promising routes.

Algorithm 1 presents the general framework of our matheuristic. The number of initial multistart solutions is denoted by $MH_{InitSol}$. The number iterations of Steps 1 and 2 is denoted by MH_{Step1} and MH_{Step2}, respectively. The multistart solution counter and the iteration counter are denoted by κ_1 and κ_2, respectively. The heuristic considers a solution denoted by ω and its cost by $c(\omega)$.

In Step 1, the heuristic applies the following operations for $MH_{InitSol}$ iterations. It first generates an initial solution $\omega_{initial}$, and then performs ALNS for MH_{Step1} iterations. Step 1 returns ω_{bestMH} as the best found initial feasible solution. The temperature T, used in simulated annealing, is initially

set at $c(\omega_{initial})T_0$, where T_0 is the start-up temperature defined as an input parameter. Step 2 initializes ω_{bestMH} as $\omega_{initial}$, and applies ALNS to it for MH_{Step2} iterations. In total, $MH_{Total} = MH_{InitSol}MH_{Step1} + MH_{Step2}$ iterations are executed. A single-start algorithm is represented by the case $MH_{InitSol} = 1$.

At the end of the algorithm, the total cost of the best found solution is computed by exactly solving the FRVCP for each route while satisfying all constraints of the problem. Finally, the best solution ω_{best} is returned, that is, a set of routes together with an optimal charging schedule for each of them.

Algorithm 1. General framework of the matheuristic

Input: The graph $G = (\mathcal{N}, \mathcal{A})$ and related parameters
Step 1
$\kappa_1 \leftarrow 1$, $c(\omega_{bestMH}) \leftarrow \infty$
while $\kappa_1 \leq MH_{InitSol}$ **do**
  Generate an initial solution $\omega_{initial}$ (Algorithm 2)
  $T \leftarrow c(\omega_{initial})T_0$, $\omega_{current} \leftarrow \omega_{best} \leftarrow \omega_{initial}$, $\Omega \leftarrow \emptyset$, $\kappa_2 \leftarrow 1$
  while $\kappa_2 \leq MH_{Step1}$ **do**
    Perform ALNS on $\omega_{current}$ (Algorithm 3)
    if $c(\omega_{current}) < c(\omega_{best})$ **then**
      $\omega_{best} \leftarrow \omega_{current}$
    $\kappa_2 \leftarrow \kappa_2 + 1$
  end while
  if $c(\omega_{best}) < c(\omega_{bestMH})$ **then**
    $\omega_{bestMH} \leftarrow \omega_{best}$
  $\kappa_1 \leftarrow \kappa_1 + 1$
end while
Return best initial feasible solution ω_{bestMH}
Step 2
$T \leftarrow c(\omega_{bestMH})T_0$, $\omega_{current} \leftarrow \omega_{best} \leftarrow \omega_{initial} \leftarrow \omega_{bestMH}$, $\Omega \leftarrow \emptyset$, $\kappa_2 \leftarrow 1$
while $\kappa_2 \leq MH_{Step2}$ **do**
  Perform ALNS on $\omega_{current}$ (Algorithm 3)
  if $c(\omega_{current}) < c(\omega_{best})$ **then**
    $\omega_{best} \leftarrow \omega_{current}$
  $\kappa_2 \leftarrow \kappa_2 + 1$
end while
Output: ω_{best} with an optimal charging schedule

In what follows, we will successively describe (a) in Section 3.1: the initialization procedure; (b) in Section 3.2: the ALNS; (c) in Section 3.3: the removal and insertion operators (used in Sections 3.4, 3.1, and 3.2); and finally (d) in Section 3.4: the intensification and diversification procedures (used in Section 3.2).

3.1. Initialization procedure

Algorithm 2 presents the procedure we have developed to generate an initial feasible solution $\omega_{initial}$ for the E-VRP-SCS. The number of removal nodes is first fixed equal to the total number of customers $|\cup_{c\in\mathcal{C}} \mathcal{I}_c^m|$. We then add all the customers to the removal list $L^{removal}$, and we randomly sort the list. The IO2 operator is then applied to create solution $\omega^*_{initial}$ using GFRVCH to solve the FRVCP. We return the obtained solution as an initial solution $\omega_{initial}$ if it is feasible. Otherwise, this procedure is reiterated until a feasible solution has been identified.

Algorithm 2. Initialization procedure

1: **Input:** The graph $G = (\mathcal{N}, \mathcal{A})$ and related parameters
2: **while** a feasible E-VRP-SCS solution is not generated **do**
3: Randomly assign all customers to the removal list $L^{removal}$
4: Apply the IO2 operator using GFRVCH and create $\omega^*_{initial}$
5: $\omega_{initial} \leftarrow \omega^*_{initial}$
6: **end while**
7: **Output:** $\omega_{initial}$

3.2. Adaptive large neighborhood search

Algorithm 3 presents the detailed algorithmic steps of the ALNS. The algorithm uses as input an initial feasible solution $\omega_{initial}$ (Section 2). Each ALNS operator has a certain probability of being chosen at every iteration. These probabilities are updated based on the historical performance of the operators and is governed by a roulette-wheel mechanism. Initially, the probabilities of each operator are equal, and the current ($\omega_{current}$) and best (ω_{best}) solutions coincide with the initial solution $\omega_{initial}$. A temporary solution, which will either be discarded or become the current solution, is denoted by $\omega_{temporary}$. The probability of accepting a nonimproving solution is denoted by ϑ and computed as a function of the current temperature. The fixed cooling parameter is denoted by $\delta_{Cooling}$ ($0 < \delta_{Cooling} < 1$).

Every insertion necessitates determining the feasibility of a route and its length, and thus the charging times at the CSs. Therefore, for each insertion we solve the FRCVP either exactly (with CPLEX) or heuristically (with GFRVCH). Every $\Upsilon^{Insertion}_{CPLEX}$ iterations, CPLEX is used to solve the FRVCP while the GFRVCH is called otherwise. To improve the quality of our matheuristic, we introduce intensification and diversification procedures. The frequency of calls to the intensification and diversification procedures are denoted by Υ_{Int} and Υ_{Div}, respectively. Furthermore, at the end of each iteration, we add the routes of the best solution to a pool of routes $\Omega \subset R$, where R is a set of feasible routes. In an attempt to improve the solution of E-VRP-NL, every Υ_{SPF} iterations we solve a SPF over the set of routes Ω, where the objective is to select the best subset of routes such that each customer is served once.

To update the probabilities of the removal and insertion operators, an adaptive weight adjustment procedure is applied. After every segment of δ_{RW} iterations, we recalculate the probability of each

operator based on its total score. At the start of each segment, the scores of all operators are set to zero. The probability of selecting operator i in segment t is denoted by ϱ_i^t. Let $\varrho_i^{t+1} = 0.9\varrho_i^t + 0.1\psi_i/\zeta_i$, for operator i where ψ_i is the score and ζ_i is the number of times it was used during the last segment. If a new best solution is found, the scores are increased by σ_1. If the new solution is better than the current solution, the scores are increased by σ_2. If the new solution is worse than the current solution, the scores are increased by σ_3.

Algorithm 3. The algorithmic steps of ALNS

Input: The graph $G = (\mathcal{N}, \mathcal{A})$, related parameters and $\omega_{current}$
Select a removal operator, apply it to the $\omega_{current}$ and obtain $\omega_{temporary}*$
Apply an insertion operator using GFRVCH to solve the FRVCP for each route in $\omega_{temporary}*$ and obtain $\omega_{temporary}$
if $c(\omega_{temporary}) < c(\omega_{current})$ **then**
  $\omega_{current} \leftarrow \omega_{temporary}$
else
  $\vartheta \leftarrow exp^{-(c(\omega_{temporary})-c(\omega_{current}))/T}$
  Generate a random number ϵ in the interval $[0, 1]$
  if $\epsilon < \vartheta$ **then**
    $\omega_{current} \leftarrow \omega_{temporary}$
if κ_2 is a multiple of $\Upsilon_{CPLEX}^{Insertion}$ **then**
  Apply a removal operator to the $\omega_{current}$ and obtain $\omega_{temporary}*$
  Apply an insertion operator using CPLEX to solve the FRVCP to the $\omega_{temporary}*$ and obtain $\omega_{temporary}$
  if $c(\omega_{temporary}) < c(\omega_{current})$ **then**
    $\omega_{current} \leftarrow \omega_{temporary}$
if κ_2 is a multiple of Υ_{Int} **then**
  Intensification procedure: Apply Algorithm 4
if κ_2 is a multiple of Υ_{Div} **then**
  Diversification procedure: Apply Algorithm 5
if κ_2 is a multiple of Υ_{SPF} **then**
  Solve the SPF and obtain ω_{SPF}
  if $c(\omega_{SPF}) < c(\omega_{current})$ **then**
    $\omega_{current} \leftarrow \omega_{SPF}$
if κ_2 is a multiple of δ_{RW} **then**
  Update the probabilities of the operators
Add each route R of ω_{best} to the pool of routes Ω
$T \leftarrow \delta_{Cooling} T$.
Output: $\omega_{current}$

3.3. Operators

We now describe our removal and insertion operators. Removal operators first remove n' customers where the value of n' is selected from the interval $[\underline{\iota}, \bar{\iota}]$, where $\underline{\iota}$ and $\bar{\iota}$ are input parameters. Removal operators then place these removed customers in a removal list $L^{removal}$. In the insertion phase, the

n' removed customers are inserted into a least-cost position of the destroyed solution. Removal operators remove customers irrespective of the company. However, we consider only insertions of customers in their respective company's routes.

Here, we introduce a mechanism that considers the fixed costs of CSs during the removal and insertion phases. Let $\eta_{RC}(i)$ be the travel time saving obtained as a result of using a removal operator on customer $i \in \cup_{c\in\mathcal{C}}\mathcal{I}_c^m \backslash L^{removal}$. Let f_{RC}^s be equal to the fixed opening cost of CS for $s \in \mathcal{S}$ if and only if customer i is the only customer who requires the insertion of CS s when we solve the FRVCP using GFRVCH, and to 0 otherwise. Thus, for each removal operator, the total saving resulting from removing customer $i \in \cup_{c\in\mathcal{C}}\mathcal{I}_c^m \backslash L^{removal}$, denoted $RC(i)$, is calculated as

$$RC(i) = \eta_{RC}(i) + \sum_{s\in\mathcal{S}} f_{RC}^s. \tag{41}$$

In the destroyed solution, the total travel time insertion cost of customer $i \in L^{removal}$ is defined as $\eta_{IC}(i)$. Similar to the f_{RC}^s, let f_{IC}^s be equal to the fixed opening cost of CS $s \in \mathcal{S}$ if and only if customer i is the only customer who requires the insertion of s when we solve the FRVCP using GFRVCH, and to 0 otherwise. Thus, the total insertion cost of customer i is $IC(i)$, for each insertion operator is

$$IC(i) = \eta_{IC}(i) + \sum_{s\in\mathcal{S}} f_{IC}^s. \tag{42}$$

3.3.1. Removal operators

We use the same seven removal operators as those used by Koç et al. (2018) and several other authors (e.g., Cherkesly et al., 2015; Koç et al., 2016). The original version of these operators considered distance (see Ropke and Pisinger, 2006a, 2006b). However, here we adapt them to consider travel time instead of distance and also by considering the aforementioned mechanism.

Random removal (RO1): Randomly remove n' customers from a solution.

Route removal (RO2): Remove a randomly selected route from the solution.

Worst travel time removal (RO3): Removes customers based on travel time. Let $i \in \cup_{c\in\mathcal{C}}\mathcal{I}_c$ be the predecessor, and $k \in \cup_{c\in\mathcal{C}}\mathcal{I}_c$ be the successor of customer $j \in \cup_{c\in\mathcal{C}}\mathcal{I}_c^m$ in the current solution. The operator iteratively removes the customer j^*, where $j^* = \arg\max_{j\in\cup_{c\in\mathcal{C}}\mathcal{I}_c^m}\{t_{ij} + t_{jk}\}$.

Shaw removal (RO4): Remove n' similar customers by selecting a customer that is most similar to the one last added to $L^{removal}$, using a similarity between customers i and j defined by a measure $\Lambda(i, j)$, which includes two terms: the travel time t_{ij}, and an index l_{ij} equal to -1 if i and j are in the same route, and to 1 otherwise. The relatedness measure is given by $\Lambda(i, j) = \varphi_1 t_{ij} + \varphi_2 l_{ij}$, where φ_1 and φ_2 are weights that are normalized to find the best candidate solution. The operator starts by randomly selecting a node $i \in \cup_{c\in\mathcal{C}}\mathcal{I}_c^m \backslash L^{removal}$, and selects the node j^* to remove, where $j^* = \arg\min_{j\in\cup_{c\in\mathcal{C}}\mathcal{I}_c^m \backslash L^{removal}}\{\Omega(i, j)\}$.

Proximity-based removal (RO5): This is a special case of RO4 where the selection criterion only considers travel time.

Neighborhood removal (RO6): Remove customers who are extreme with respect to the average total travel time of a route $\bar{c}(\Pi)$. For each route Π, the operator calculates $\bar{c}(\Pi) = \sum_{(i,j)\in\Pi} t_{ij}/\Theta$, where Θ denotes the number of customers in Π, and selects a node $j^* = \arg\max_{(\Pi\in\mathcal{R};\, j\in\Pi)}\{\bar{c}(\Pi) - c_{\Pi\setminus\{j\}}\}$, where $c_{\Pi\setminus\{j\}}$ denotes the average total travel time of route Π excluding node j.

Pair removal (RO7): Remove customer pairs that are similar to one another in terms of travel time. First randomly remove $n'/2$ customers, and then the other $n'/2$ customers by selecting the closest ones based on the travel time to the previously removed customers.

3.3.2. Insertion operators

We use four insertion operators where the first and second ones are from Ropke and Pisinger (2006a, 2006b), and the third and fourth ones are new. We have developed a mechanism that allows us to generate feasible solutions for the E-VRP-SCS. During the insertion process of each customer, we forbid the insertion of a customer in the route if this would violate the maximum tour duration and battery capacity constraints. This is achieved by applying the GFRVCH to insert CSs when an insertion is infeasible without visiting any CS. All of the insertion operators use this new mechanism.

Greedy insertion (IO1): Starting with the first customer of $L^{removal}$, insert all nodes of $L^{removal}$ in the solution. It considers the best possible travel time based insertion position. For node $i \in \mathcal{I}_c\setminus L^{removal}$ of company $c \in \mathcal{C}$ followed in the destroyed solution by $k \in \mathcal{I}_c\setminus L^{removal}$ of company $c \in \mathcal{C}$, and node $j \in L^{removal}$ we define $\gamma(i, j) = c_{ij} + c_{jk} - c_{ik}$. It finds the least-time insertion position for $j \in L^{removal}$ by $i^* = \arg\min_{i\in\mathcal{I}_c\setminus L^{removal}}\{\gamma(i, j)\}$.

Greedy insertion with noise function (IO2): This operator is a variant of the IO1 operator. It allows some diversification by selecting the best position for a node. It calculates a noise cost $\upsilon(i, j) = \gamma(i, j) + d_{max}\delta_{Noise}\xi$, where d_{max} is the maximum travel time between all nodes, δ_{Noise} is a noise parameter used for diversification, and ξ is a random number in $[-1, 1]$. It finds a least-cost insertion position for $j \in L^{removal}$ by computing $i^* = \arg\min_{i\in\cup_{c\in\mathcal{C}}\mathcal{I}_c\setminus L^{removal}}\{\upsilon(i, j)\}$.

Removal list insertion (IO3): This operator aims to change the order of the nodes in $L^{removal}$ with the aim of decreasing the insertion costs. For each node in $L^{removal}$ of companies, the operator iteratively calculates the least insertion cost, that is, driver and CS opening costs, in the current solution. It then lists them in a nonincreasing order, and finally applies the operator IO1.

Single-customer insertion (IO4): This operator aims to diversify solutions by generating single-customer routes. It inserts each customer of $L^{removal}$ in a back and forth route, that is, depot-customer-depot.

3.4. Intensification and diversification procedures

We first introduce the intensification procedure we have developed to improve the quality of solutions. This procedure intensifies the search within promising regions of the solution space. It is applied every Υ_{Int} iterations. Algorithm 4 presents the intensification procedure. Operators RO2 and IO1 are applied. The latter uses GFRVCH to solve the FRVCP. We use operators RO2 and IO1 to change the positions of n' nodes in current the solution.

Algorithm 4. Intensification procedure

1: **Input:** The graph $G = (\mathcal{N}, \mathcal{A})$, related parameters and $\omega_{current}$
2: Apply RO2 operator to the $\omega_{current}$ and obtain $\omega_{temporary}*$
3: Apply IO1 operator to the $\omega_{temporary}*$ using GFRVCH to solve the FRVCP and obtain $\omega_{temporary}$
4: **if** $c(\omega_{temporary}) < c(\omega_{current})$ **then**
5: $\quad \omega_{current} \leftarrow \omega_{temporary}$
6: **Output:** $\omega_{current}$

The performance of the heuristic is improved by applying a diversification procedure. Over the iterations, solutions tend to become more similar, making it difficult to avoid premature convergence. To overcome this difficulty, we introduce a diversification procedure in order to increase the population diversity. This procedure, described in Algorithm 5, diversifies the search within promising regions of the solution space. It is applied every Υ_{Div} iterations. We use two diversification structure based operators, the RO1 and the IO2 using GFRVCH to solve the FRVCP, to change the positions of a n' number of nodes in current the solution.

Algorithm 5. Diversification procedure

1: **Input:** $\omega_{current}$
2: Apply RO1 operator to the $\omega_{current}$ and obtain $\omega_{temporary}*$
3: Apply IO2 operator to the $\omega_{temporary}*$ using GFRVCH to solve the FRVCP and obtain $\omega_{temporary}$
4: **if** $c(\omega_{temporary}) < c(\omega_{current})$ **then**
5: $\quad \omega_{current} \leftarrow \omega_{temporary}$
6: **Output:** $\omega_{current}$

4. Computational experiments and analyses

We now describe our computational experiments carried out on a machine with an Intel 3.6 GHz processor and 32 GB of RAM. The algorithm was coded in C++. We used CPLEX 12.6 with its default settings to solve the FRVCP. We have performed 10 runs on each instance.

The remainder of this section is organized as follows. We first describe our benchmark instances for the E-VRP-SCS in Section 4.1. We then explain the parameter settings in Section 4.2. We present the results of our heuristic on the E-VRP-NL in Section 4.3 and the results it produced on the E-VRP-SCS instances in Section 4.4. We finally present in Section 4.5 the results of the effect of the cost of fast technology CS on the E-VRP-SCS.

4.1. Benchmark instances and data generation

Since the E-VRP-SCS is a new problem, there are no publicly available instances. Therefore, to assess the performance of our heuristic, we adapted the benchmarks of Montoya et al. (2017) for

Table 3
Cost of purchasing and installing CSs in our E-VRP-SCS instances

	Costs				
CS technology	Acquisition	Installation	Total	Daily (in $)	Daily (in €)
Slow	$6,500	$3,000	$9,500	$6.33	€5.28
Moderate	$10,000	$21,000	$31,000	$20.66	€17.23
Fast	$15,000	$21,000	$36,000	$24.00	€20.00

the E-VRP-NL. Their test bed includes 120 instances based on real EV data and battery charging functions. The instances contain between 10 and 320 customers, and between two and 38 CSs. Customers are located inside a 120 km × 120 km area according to continuous uniform distribution, a random clustered distribution, or a mixture of both. The CSs are located either randomly or using a k-means algorithm. Each CS has one of three possible charging technologies: slow (11 kW), moderate (22 kW), and fast (44 kW). The EVs are Peugeot iOns with an energy consumption rate of 0.125 kWh/km, and a battery capacity of 16 kWh (Peugeot iOn, 2017). The instances assume that the EVs' energy consumption is the traveled distance multiplied by the consumption rate. The vehicles have piecewise linear charging functions for each CS technology derived from the real charging data of Uhrig et al. (2015). Finally, the maximum route duration T_{max} is set to 10 hours for each EV.

To adapt the instances to our problem, we first inserted to each instance two additional depots (i.e., we assume that there are three companies joining the venture). The locations of the new depots were randomly generated. Next, we randomly assigned each of the original customers to one of the three depots, ensuring a balanced assignment. For example, for the 10-customer instances, the first, second, and third depots have four, three, and three customers, respectively. We used the location of each CSs in the original instances as a potential CS location and we considered that the three original technologies (slow, moderate, and fast) can be installed in any potential location. We set the cost of the driver's time to $f^d = €9.76/\text{h}$ based on the figures reported in (Wage Indicator, 2017). To set the cost of the CSs, we used data provided by a 2015 report by the Department of Energy of the United States (Smith and Castellano, 2015). The report presents approximate purchasing and installation costs for three types of CSs: Level 1 chargers (1.4–1.9 kW), Level 2 chargers (3.4–19.2 kW), and CDFC chargers (24–90 kW). To map the technologies of our CSs to the technologies of the chargers considered in the report, we used the following convention: slow CSs are considered top-of-the-line Level 2 chargers, moderate CSs are considered bottom-of-the-line CDFC chargers, and fast CSs are considered mid-range CDFC chargers. Based on this mapping, we used in our instances the acquisition cost reported in Table 1 and the average installation costs reported in Table 2 of the report by the Department of Energy of the United States (Smith and Castellano, 2015). Since in our instances $T_{max} = 10\text{h}$, we assumed that each route can be completed on a single working day. Therefore, we translated the acquisition and installation cost of CSs into daily costs by assuming for each CS a lifetime of five years, each with 300 operational days. Table 3 summarizes the costs of CSs in our instances.

Table 4
Results of the parameter calibration

Parameter	Value	Dev (%)	Value	Dev (%)	Value	Dev (%)	Value	Dev (%)
$MH_{InitSol}$, MH_{Step1}, MH_{Step2}	(1,2500,2500)	2.14	(25,100,2500)	1.11	(1,1500,3500)	1.24	(15,100,3500)	0.00
MH_{Total}	3000	3.56	4000	2.50	5000	0.00	6000	0.00
$\Upsilon_{CPLEX}^{Insertion}$	150	0.31	200	0.19	250	0.00	300	0.28
Υ_{Int}	20	0.21	25	0.00	30	0.18	35	0.26
Υ_{Div}	20	0.18	25	0.22	30	0.00	35	0.31
Υ_{SPF}	300	0.39	400	0.34	500	0.00	600	0.19
δ_{RW}	400	0.09	450	0.00	500	0.03	550	0.08
σ_1, σ_2, σ_3	(5,1,0)	0.38	(3,1,0)	0.29	(1,0,3)	0.22	(1,0,5)	0.00

Table 5
Parameters used in the matheuristic

Description	Notation	Value
The number of initial multistart solutions	$MH_{InitSol}$	15
The number iterations of Step 1	MH_{Step1}	100
The number iterations of Step 2	MH_{Step2}	3500
Total number of iterations	MH_{Total}	5000
Frequency of calls to insertion operator using CPLEX	$\Upsilon_{CPLEX}^{Insertion}$	250
Frequency of calls to intensification procedure	Υ_{Int}	25
Frequency of calls to diversification procedure	Υ_{Div}	30
Frequency of calls to SPF	Υ_{SPF}	300
Roulette-wheel parameter	δ_{RW}	450
New global solution score	σ_1	1
Better solution score	σ_2	0
Worse solution score	σ_3	5
Start-up temperature parameter	T_0	100
Cooling parameter	$\delta_{Cooling}$	0.99
Lower limit of removable nodes	$\underline{\iota}$	10% of $\lvert\mathcal{I}_c^m\rvert$
Upper limit of removable nodes	$\overline{\iota}$	30% of $\lvert\mathcal{I}_c^m\rvert$
First Shaw parameter	φ_1	0.5
Second Shaw parameter	φ_2	0.15
Third Shaw parameter	φ_3	0.25
Noise parameter	δ_Noise	0.1

4.2. Parameter settings

Matheuristic algorithms are based on a set of correlated parameters and configuration decisions. In our implementation, we have conducted several experiments to further fine-tune these parameters to solve the E-VRP-SCS. We calibrated the parameters based on six representative 80-customer E-VRP-SCS training instances instead of tc1c160s16cf0, tc2c160s16ct1, tc0c160s16ct4, tc0c160s24ct2, tc1c160s24ct0, and tc1c160s24cf3. Table 4 presents the results of the parameter calibration experiments. For each setting, the column Dev (%) shows the percentage deviation from the best solution value. Table 5 provides the parameter values obtained after the calibration phase.

Table 6
Average results for the E-VRP-NL instances

Instance	MGMV17		Heuristic			BKS	
family	Avg best	Avg time (seconds)	Avg best	Avg time (seconds)	Avg Dev (%)	M	I
10-Customer	14.25	5.62	14.25	7.67	0.00	20	0
20-Customer	19.44	10.56	19.43	13.10	−0.06	13	4
40-Customer	31.48	35.35	31.49	43.91	−0.08	4	11
80-Customer	38.01	80.11	38.08	96.68	0.13	1	11
160-Customer	70.24	568.02	70.51	620.01	0.38	0	8
320-Customer	132.47	4397.64	133.11	4263.05	0.49	0	4
Average					0.14		
Runs	10		10				
Processor	Intel Xeon 2.33 GHz		Intel 3.6 GHz				
Total						38	38

4.3. Results for the E-VRP-NL

The E-VRP-SCS reduces to the E-VRP-NL when $|\mathcal{C}| = 1$, $f^d = 0$, and $f^s = 0\ \forall s \in \mathcal{S}$. In order to assess the quality of our matheuristic, we present a comparative analysis of the results with those obtained with the algorithm MGMV17 of Montoya et al. (2017) on the E-VRP-NL instances of these authors.

Average comparisons for 10-, 20-, 40-, 80-, 160-, and 320-customer E-VRP-NL instances are presented in Table 6. The first column displays the instance family. For each family, the other columns show the average best value, the average time of 10 runs in seconds, and average percentage of deviations (Dev (%)) of the values of solutions found by MGMV17 with respect to our matheuristic. Let s(MGMV17) and s(matheuristic) be the solution values produced by MGMV17 and the heuristic, respectively. Then, Dev (%) is calculated as $100 \times (s(\text{matheuristic}) - s(\text{MGMV17}))/s(\text{MGMV17})$. Thus, a negative deviation shows that the solution found by the heuristic is of better quality. In the column BKS, "M" shows the total number of matched BKSs and "I" shows the total number of new BKS found for each instance family. For detailed results, the reader is referred to Appendix A.

Table 6 reveals that our heuristic is highly competitive, with average deviations ranging from −0.08% to 0.49%. The average deviations for 10-, 20-, 40-, 80-, 160-, and 320-customer instances are 0.00%, −0.06%, −0.08%, 0.13%, 0.38%, and 0.49%, respectively. The average performance of our heuristic on all 120 instances is only 0.14% worse than that of MGMV17. However, the heuristic outperforms MGMV17 on the 20- and 40-customer instances. Overall, the heuristic improves 38 BKS and matches 38 BKS of 120 benchmark instances.

4.4. Results for the E-VRP-SCS

We now present the results of the E-VRP-SCS. The average comparisons for 10-, 20-, 40-, 80-, 160-, and 320-customer E-VRP-SCS instances are presented in Table 7. The first column displays the instance family. For average best solutions, the next six columns show the driver cost

Table 7
Average results for the E-VRP-SCS instances

	Matheuristic							
Instance family	Avg best driver cost	Avg best CS cost	Avg best total cost	Avg best no. of fast CS	Avg best no. of moderate CS	Avg best no. of slow CS	Avg total cost	Avg time (seconds)
10-Customer	249.15	25.09	274.23	1.00	0.05	0.80	274.52	10.48
20-Customer	397.38	41.03	438.42	1.30	0.75	0.40	439.45	24.00
40-Customer	657.87	49.86	707.73	1.05	1.20	1.55	709.71	55.88
80-Customer	964.20	65.59	1029.79	1.25	1.85	1.65	1033.62	104.16
160-Customer	1822.92	146.31	1969.23	2.50	4.90	2.25	1985.80	649.55
320-Customer	3705.48	239.93	3945.40	6.40	5.50	3.25	3976.69	4823.77

Table 8
Average percentages of the opened CSs

	Fast CSs			Moderate CSs			Slow CSs		
Instance family	No. of potential	No. of opened	%	No. of potential	No. of opened	%	No. of potential	No. of opened	%
10-Customer	2.50	1.00	40.00	2.50	0.05	2.00	2.50	0.80	32.00
20-Customer	3.50	1.30	37.14	3.50	0.75	21.43	3.50	0.40	11.43
40-Customer	6.50	1.05	16.15	6.50	1.20	18.46	6.50	1.55	23.85
80-Customer	10.00	1.25	12.50	10.00	1.85	18.50	10.00	1.65	16.50
160-Customer	20.00	2.50	12.50	20.00	4.90	24.50	20.00	2.25	11.25
320-Customer	31.00	6.40	20.65	31.00	5.50	17.74	31.00	3.25	10.48
Average			23.16			17.11			17.58

(€), CS cost (€), total cost (€), the number of opened fast CSs, the number of opened moderate CSs, and the number of opened slow CSs of best solution of 10 runs. The last two columns display the average total cost and total time (seconds) of 10 runs. For detailed results of all instances, the reader is referred to Appendix B that reports the results corresponding to the best solution for each instance. We observe that the slow CSs are used less often than the fast and moderate CSs.

Table 8 presents the average percentages of the opened CSs for each technology and for each instance family. The first, second, and third columns of each CS technology display the average number of potential CSs, the average number of opened CSs, and the average percentages of opened CSs for each technology. Overall, 23.16%, 17.11%, and 17.58% of fast, moderate, and slow technology CSs are opened, respectively. These results show that fast technology CSs are preferable over other technologies even though they have higher opening costs. This makes sense since high charging speeds will reduce the driver cost. In addition, using fast CSs is instrumental in producing feasible solutions on instances tightly constrained by the maximum tour duration constraints.

Table 9
Results of the effect of the cost of fast technology CS

	Version 1	Version 2	Base case	Version 3
Instance	€30	€25	€20	€18
E-VRP-SCS-tc0c80s8cf0	(0,2,0)	(1,2,2)	(1,2,2)	(2,3,1)
E-VRP-SCS-tc0c80s8cf1	(1,3,1)	(1,3,2)	(1,3,1)	(1,3,2)
E-VRP-SCS-tc1c80s8cf2	(0,1,1)	(1,1,1)	(1,1,1)	(1,1,1)
E-VRP-SCS-tc2c80s8cf3	(1,2,3)	(1,1,3)	(0,0,3)	(1,0,3)
E-VRP-SCS-tc2c80s8cf4	(1,2,4)	(1,2,4)	(1,2,4)	(1,2,4)
E-VRP-SCS-tc0c80s8ct0	(2,2,1)	(2,3,1)	(2,3,0)	(2,2,0)
E-VRP-SCS-tc0c80s8ct1	(1,3,2)	(1,2,2)	(1,3,2)	(1,3,1)
E-VRP-SCS-tc1c80s8ct2	(1,1,1)	(1,1,1)	(1,1,0)	(1,1,0)
E-VRP-SCS-tc2c80s8ct3	(1,1,3)	(1,1,2)	(2,1,3)	(1,1,3)
E-VRP-SCS-tc2c80s8ct4	(1,2,0)	(2,2,2)	(1,2,2)	(1,2,2)
E-VRP-SCS-tc0c80s12cf0	(2,2,2)	(1,2,1)	(2,2,1)	(2,3,2)
E-VRP-SCS-tc0c80s12cf1	(1,2,2)	(1,2,1)	(3,2,1)	(3,2,1)
E-VRP-SCS-tc1c80s12cf2	(2,2,1)	(1,1,1)	(0,1,2)	(0,1,4)
E-VRP-SCS-tc2c80s12cf3	(0,1,3)	(1,1,4)	(0,0,3)	(0,0,4)
E-VRP-SCS-tc2c80s12cf4	(2,3,2)	(2,2,2)	(2,3,2)	(1,3,2)
E-VRP-SCS-tc0c80s12ct0	(2,5,1)	(1,3,1)	(2,3,0)	(2,3,0)
E-VRP-SCS-tc0c80s12ct1	(1,2,1)	(1,2,2)	(3,2,1)	(2,2,0)
E-VRP-SCS-tc1c80s12ct2	(1,2,0)	(1,2,1)	(1,2,0)	(2,1,2)
E-VRP-SCS-tc2c80s12ct3	(1,2,3)	(1,2,4)	(0,1,3)	(2,3,1)
E-VRP-SCS-tc2c80s12ct4	(2,2,1)	(2,3,1)	(1,3,2)	(1,2,1)
Average	(1.15,2.10,1.60)	(1.20,1.90,1.90)	(1.25,1.85,1.65)	(1.25,1.90,1.70)

4.5. The effect of the cost of fast technology CS on the E-VRP-SCS

Given the difficulty of accurately estimating the CS opening costs, especially those that use a fast charging technology, we have performed a sensitivity analysis. In our benchmark instances, the cost of fast CS is high with respect to slow and moderate CSs. We now investigate the effect of decreasing and increasing the fast CS costs. To this end, we have selected all 80-customer E-VRP-SCS instances and we have considered three fast CS cost versions: €30, €25, and €18. Table 9 presents the results of these experiments. We report the number of opened CSs for each instance. In this table, the first, second, and third elements within the parentheses represent the number of opened CSs of fast, moderate, and slow technology, respectively.

Table 9 shows that the average number of opened fast technology CSs is 1.15, 1.20, 1.25, and 1.25 for version 1, version 2, base case, and version 3, respectively. The average number of opened fast technology CSs is relatively stable. Thus, implying that high charging speeds justify the investment in opening fast CSs, even when the latter costs are increased.

4.6. The effect of the slow and moderate CS availability on the E-VRP-SCS

Our results show that fast technology CSs are preferred. However, in practice, the grid is not always capable of handling fast charging. Therefore, we now analyze the situation where only low and

Table 10
Results of the effect of the slow and moderate CS availability

	Base case		Slow and moderate CS		
Instance	CS cost	Total cost	CS cost	Total cost	Dev (%)
E-VRP-SCS-tc0c80s8cf0	65.02	1185.18	84.76	1210.92	2.17
E-VRP-SCS-tc0c80s8cf1	76.97	1245.45	101.99	1274.17	2.31
E-VRP-SCS-tc1c80s8cf2	42.51	829.72	45.02	840.25	1.27
E-VRP-SCS-tc2c80s8cf3	15.84	869.54	72.81	933.60	7.37
E-VRP-SCS-tc2c80s8cf4	75.58	1079.95	78.09	1093.26	1.23
E-VRP-SCS-tc0c80s8ct0	91.69	1209.90	94.20	1224.71	1.22
E-VRP-SCS-tc0c80s8ct1	82.25	1241.64	107.27	1275.06	2.69
E-VRP-SCS-tc1c80s8ct2	37.23	861.59	45.02	875.98	1.67
E-VRP-SCS-tc2c80s8ct3	73.07	952.50	67.53	958.26	0.60
E-VRP-SCS-tc2c80s8ct4	65.02	1056.20	72.81	1078.79	2.14
E-VRP-SCS-tc0c80s12cf0	79.74	1065.86	84.76	1087.68	2.05
E-VRP-SCS-tc0c80s12cf1	99.74	1258.50	101.99	1281.95	1.86
E-VRP-SCS-tc1c80s12cf2	27.79	831.20	45.02	855.83	2.96
E-VRP-SCS-tc2c80s12cf3	15.84	842.74	27.79	885.69	5.10
E-VRP-SCS-tc2c80s12cf4	102.25	1017.48	107.27	1022.50	0.49
E-VRP-SCS-tc0c80s12ct0	91.69	1089.22	101.99	1099.52	0.95
E-VRP-SCS-tc0c80s12ct1	99.74	1118.46	107.27	1125.99	0.67
E-VRP-SCS-tc1c80s12ct2	54.46	887.98	62.25	895.77	0.88
E-VRP-SCS-tc2c80s12ct3	33.07	858.13	50.30	875.36	2.01
E-VRP-SCS-tc2c80s12ct4	82.25	1094.51	96.71	1108.97	1.32
Average					2.05

moderate CSs are available. To this end, we have selected all 80-customer E-VRP-SCS instances and we report the results in Table 10. The column Dev (%) shows the percentage deviation found by only considering slow and moderate CS availability with respect to the base case. The results indicate that if the CSs are limited to slow or moderate technology the total cost increases by 2.05% on average. This shows the importance of the fast technology in CSs.

4.7. The effect of the mixed CS availability on the E-VRP-SCS

Due to a number of technical constraints, some CS locations may not allow for all three types of charging technologies. Therefore, in this section we analyze the situation in which certain CSs may only have a subset of technologies. In practice, three categories of CS locations may exist: CSs that only allow for the slow technology, CSs that allow for slow and moderate charging technologies, and CSs that allow for all three technologies. We have performed experiments on all 80-customer E-VRP-SCS instances on a mixed CS availability setting, where in each instance roughly a third of the CSs were assigned to each category. We report the results in Table 11. The column Dev (%) shows the percentage deviation found by mixed CS availability with respect to the base case. The results indicate that if we consider mixed CS availability in the network the total cost increases by 1.18% on average.

Table 11
Results of the effect of the mixed CS availability

	Base case		Mix CS		
Instance	CS cost	Total cost	CS cost	Total cost	Dev (%)
E-VRP-SCS-tc0c80s8cf0	65.02	1185.18	82.25	1203.36	1.53
E-VRP-SCS-tc0c80s8cf1	76.97	1245.45	99.48	1269.61	1.94
E-VRP-SCS-tc1c80s8cf2	42.51	829.72	47.79	840.04	1.24
E-VRP-SCS-tc2c80s8cf3	15.84	869.54	70.30	911.09	4.78
E-VRP-SCS-tc2c80s8cf4	75.58	1079.95	70.30	1084.41	0.41
E-VRP-SCS-tc0c80s8ct0	91.69	1209.90	87.53	1221.05	0.92
E-VRP-SCS-tc0c80s8ct1	82.25	1241.64	92.81	1254.58	1.04
E-VRP-SCS-tc1c80s8ct2	37.23	861.59	42.51	869.42	0.91
E-VRP-SCS-tc2c80s8ct3	73.07	952.50	65.02	954.73	0.23
E-VRP-SCS-tc2c80s8ct4	65.02	1056.20	70.30	1072.23	1.52
E-VRP-SCS-tc0c80s12cf0	79.74	1065.86	76.97	1071.89	0.57
E-VRP-SCS-tc0c80s12cf1	99.74	1258.50	98.09	1275.95	1.39
E-VRP-SCS-tc1c80s12cf2	27.79	831.20	42.51	850.30	2.30
E-VRP-SCS-tc2c80s12cf3	15.84	842.74	37.23	858.23	1.84
E-VRP-SCS-tc2c80s12cf4	102.25	1017.48	104.76	1021.99	0.44
E-VRP-SCS-tc0c80s12ct0	91.69	1089.22	99.48	1092.99	0.35
E-VRP-SCS-tc0c80s12ct1	99.74	1118.46	104.76	1122.37	0.35
E-VRP-SCS-tc1c80s12ct2	54.46	887.98	59.74	891.25	0.37
E-VRP-SCS-tc2c80s12ct3	33.07	858.13	42.51	865.62	0.87
E-VRP-SCS-tc2c80s12ct4	82.25	1094.51	94.20	1100.41	0.54
Average					1.18

5. Conclusions and perspectives

We have introduced the E-VRP-SCS as an extension of the E-VRP-NL. The E-VRP-SCS aims to minimize the fixed opening cost of CSs as well as driver costs based on total travel time, service time at customers, and charging time at CSs. We have developed a heuristic capable of efficiently solving the E-VRP-SCS. Our matheuristic successfully combines within a multistart scheme an ALNS with an exact vehicle-charging optimization component applied to a fixed route. We have also introduced several new operators, as well as initialization, intensification, and diversification procedures. Extensive computational experiments were carried out on benchmark instances. We have compared our method to that of with Montoya et al. (2017). Our heuristic has identified 76 BKSs of 120 on the standard benchmark instances, 38 of which are strictly better than previously known solutions. For the E-VRP-SCS, 120 instances were generated and solved here for the first time. The running times of the algorithm enable it to be used in practical applications.

The E-VRP-SCS aims at optimizing the strategic decisions related to locating CSs and determining their technologies. To optimize such decisions the E-VRP-SCS routing component can be viewed as a proxy for the daily operational cost. Another strategic dimension worth exploring relates to determining the capacity of each CS, that is, the number of chargers to operate simultaneously

at each station. We note that including capacity decisions in the problem setting of the E-VRP-SCS implies handling synchronization issues between vehicle routes.

Acknowledgments

The authors thank the three anonymous referees for their insightful comments and suggestions that helped improve the content and the presentation of the paper. They also gratefully acknowledge the funding provided by the Canadian Natural Sciences and Engineering Research Council under grants 2015-06189 and 436014-2013, and by the French National Research Agency (Agence Nationale de la Recherce) through project e-VRO under grant ANR-15-CE22-0005-01. The authors are indebted to Aurélien Froger for his technical support.

References

Afroditi, A., Boile, M., Theofanis, S., Sdoukopoulos, E., Margaritis, D., 2014. Electric vehicle routing problem with industry constraints: trends and insights for future research. *Transportation Research Procedia* 3, 452–459.

Albareda-Sambola, M., 2015. Location-routing and location-arc routing problems. In Laporte, G., Nickel, S., Saldanha da Gama, F. (eds) *Location Science*. Springer, Berlin-Heidelberg, pp. 399–418.

Andelmin, J., Bartolini, E., 2016. A multi-start local search heuristic for the green vehicle routing problem based on a multigraph reformulation. Technical Report, Aalto University, Finland.

Bartolini, E., Andelmin, J., 2017. An exact algorithm for the green vehicle routing problem. *Transportation Science* 51, 1288–1303.

Cherkesly, M., Desaulniers, G., Laporte, G., 2015. A population-based metaheuristic for the pickup and delivery problem with time windows and LIFO loading. *Computers & Operations Research* 62, 23–35.

Crevier, B., Cordeau, J.-F., Laporte, G., 2007. The multi-depot vehicle routing problem with inter-depot routes. *European Journal of Operational Research* 176, 756–773.

Demir, E., Bektaş, T., Laporte, G., 2014. A review of recent research on green road freight transportation. *European Journal of Operational Research* 237, 775–793.

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

Drexl, M., Schneider, M., 2015. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research* 241, 283–308.

Eglese, R., Bektaş, T., 2014. Green vehicle routing. In Toth, P., Vigo, D. (eds) *Vehicle Routing: Problems, Methods, and Applications*, MOS-SIAM Series on Optimization, Philadelphia, PA, pp. 437–458.

Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. *Transportation Research Part E* 48, 100–114.

Felipe, Á., Ortuño, M.T., Righini, G., Tirado, G., 2014. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E* 71, 111–128.

Froger, A., Mendoza, J.E., Jabali, O., Laporte, G., 2017. A matheuristic for the electric vehicle routing problem with capacitated charging stations. CIRRELT 2017-31, CIRRELT, Montreal.

Goeke, D., Schneider, M., 2015. Routing a mixed fleet of electric and conventional vehicles. *European Journal of Operational Research* 245, 81–99.

Hemmati, A., Hvattum, L.M., 2017. Evaluating the importance of randomization in adaptive large neighborhood search. *International Transactions in Operational Research* 24, 929–942.

Hiermann, G., Puchinger, J., Ropke, S., 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., 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, 102–112.

Juan, A.A., Mendez, C.A., Faulin, J., de Armas, J., Grasman, S.E., 2016. Electric vehicles in logistics and transportation: a survey on emerging environmental, strategic, and operational challenges. *Energies* 9, 86.

Keskin, M., Çatay, B., 2016. Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C* 65, 111–127.

Keskin, M., Çatay, B., 2018. A matheuristic method for the electric vehicle routing problem with time windows and fast chargers. *Computers & Operations Research* 100, 172–188.

Koç, Ç., Bektaş, T., Jabali, O., Laporte, G., 2016. The fleet size and mix location-routing problem with time windows: formulations and a heuristic algorithm. *European Journal of Operational Research* 248, 33–51.

Koç, Ç., Jabali, O., Laporte, G., 2018. Long-haul vehicle routing and scheduling with idling options. *Journal of the Operational Research Society* 69, 235–246.

Koç, Ç., Karaoglan, I., 2016. The green vehicle routing problem: a heuristic based exact solution approach. *Applied Soft Computing* 39, 154–164.

Kullman, N., Goodson, J.C., Mendoza, J.E., 2017. Dynamic electric vehicle routing: heuristics and dual bounds. Working paper. Available at: https://hal.archives-ouvertes.fr/hal-01928730 (accessed November 30, 2018).

Lin, C., Choy, K.L., Ho, G.T., Chung, S.H., Lam, H.Y., 2014. Survey of green vehicle routing problem: past and future trends. *Expert Systems with Applications* 41, 1118–1138.

Margaritis, D., Anagnostopoulou, A., Tromaras, A., Boile, M., 2016. Electric commercial vehicles: practical perspectives and future research directions. *Research in Transportation Business & Management* 18, 4–10.

Montoya, A., Guéret, C., Mendoza, J.E., Villegas, J.G., 2016. A multi-space sampling heuristic for the green vehicle routing problem. *Transportation Research Part C* 70, 113–128.

Montoya, A., Guéret, C., Mendoza, J.E., Villegas, J.G., 2017. A hybrid metaheuristic for the electric vehicle routing problem with partial charging and nonlinear charging function. *Transportation Research Part B* 103, 87–110.

Pelletier, S., Jabali, O., Laporte, G., 2016. Goods distribution with electric vehicles: review and research perspectives. *Transportation Science* 50, 3–22.

Pelletier, S., Jabali, O., Laporte, G., 2018. Charge scheduling for electric freight vehicles. *Transportation Research Part B: Methodological* 115, 246–269.

Pelletier, S., Jabali, O., Laporte, G., Veneroni, M., 2017. Battery degradation and behaviour for electric vehicles: review and numerical analyses of several models. *Transportation Research Part B* 103, 158–187.

Peugeot iOn, 2017. Available at http://www.peugeot.fr/gamme/nos-vehicules/ion.html (accessed March 27, 2017).

Prodhon, C., Prins, C., 2014. A survey of recent research on location-routing problems. *European Journal of Operational Research* 238, 1–17.

Ropke, S., Pisinger, D., 2006a. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science* 40, 455–472.

Ropke, S., Pisinger, D., 2006b. A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research* 171, 750–775.

Schiffer, M., Schneider, M., Laporte, G., 2018a. Designing sustainable mid-haul logistics networks with intra-route multi-resource facilities. *European Journal of Operational Research* 265, 517–532.

Schiffer, M., Schneider, M., Walther, G., Laporte, G., 2018b. Vehicle routing and location-routing problems with intermediate stops: a review. *Transportation Science*, in press.

Schiffer, M., Stütz, S., Walther, G., 2017. Are ECVs breaking even? Competitiveness of electric commercial vehicles in retail logistics. GERAD G–2017–47, Technical Report, Montréal.

Schiffer, M., Walther, G., 2017a. The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research* 260, 995–1013.

Schiffer, M., Walther, G., 2017b. An adaptive large neighbourhood search for the location routing problem with intraroute facilities. *Transportation Science* 52, 229–496.

Schiffer, M., Walther, G., 2018. Strategic planning of electric logistics fleet networks: a robust location-routing approach. *Omega* 80, 31–42.

Schneider, M., Stenger, A., Goeke, D., 2014. The electric vehicle routing problem with time windows and recharging stations. *Transportation Science* 48, 500–520.

Smith, M., Castellano, J., 2015. Costs associated with non-residential electric vehicle supply equipment: factors to consider in the implementation of electric vehicle charging stations. Technical Report DOE/EE-1289, US Department of Energy.

Sweda, T.M., Dolinskaya, I.S., Klabjan, D., 2017. Optimal recharging policies for electric vehicles. *Transportation Science* 51, 457–479.

Tarantilis, C.D., Zachariadis, E.E., Kiranoudis, C.T., 2008. A hybrid guided local search for the vehicle-routing problem with intermediate replenishment facilities. *INFORMS Journal on Computing* 20, 154–168.

Uhrig, M., Weiß, L., Suriyah, M. Leibfried, T., 2015. E-mobility in car parks-guidelines for charging infrastructure expansion planning and operation based on stochastic simulations. The 28th International Electric Vehicle Symposium and Exhibition, KINTEX, Goyang, Korea.

Villegas, J.G., Guéret, C., Mendoza, J.E., Montoya, A., 2018. The technician routing and scheduling problem with conventional and electric vehicles. Working paper. Available at https://hal.archives-ouvertes.fr/hal-01813887/document (accessed February 6, 2018).

Wage Indicator, 2017. Minimum wages in France. Available at http://www.wageindicator.org/main/salary/minimum-wage/france (accessed March 29, 2017).

Yang, J., Sun, H., 2015. Battery swap station location-routing problem with capacitated electric vehicles. *Computers & Operations Research* 55, 217–232.

Appendix A

Table A1 presents the detailed results on all electric vehicle routing problem with nonlinear charging function (E-VRP-NL) benchmark instances. For each instance, the best value of 10 runs is reported. A boldface entry refers to the best-known solution. The time column shows the average computation time of 10 runs.

Table A1
Results for the E-VRP-NL benchmark instances

	MGMV17		Matheuristic			
Instance	Best	Time	Best	Avg.	Time	Dev (%)
tc2c10s2cf0	**21.77**	8.53	**21.77**	21.78	7.73	0.00
tc0c10s2cf1	**19.75**	3.86	**19.75**	19.77	7.15	0.00
tc1c10s2cf2	**9.03**	2.43	**9.03**	9.04	8.10	0.00
tc1c10s2cf3	**16.37**	5.63	**16.37**	16.38	7.14	0.00
tc1c10s2cf4	**16.10**	4.79	**16.10**	16.11	7.95	0.00
tc2c10s2ct0	**12.45**	5.38	**12.45**	12.46	7.98	0.00
tc0c10s2ct1	**12.30**	3.99	**12.30**	12.31	7.40	0.00
tc1c10s2ct2	**10.75**	4.21	**10.75**	10.76	7.04	0.00
tc1c10s2ct3	**13.17**	7.56	**13.17**	13.18	8.05	0.00
tc1c10s3ct4	**13.21**	6.01	**13.21**	13.22	7.30	0.00
tc2c10s3cf0	**21.77**	8.90	**21.77**	21.79	8.27	0.00
tc0c10s3cf1	**19.75**	4.41	**19.75**	19.76	8.12	0.00
tc1c10s3cf2	**9.03**	2.36	**9.03**	9.04	8.01	0.00
tc1c10s3cf3	**16.37**	6.06	**16.37**	16.39	8.28	0.00
tc1c10s3cf4	**14.90**	6.72	**14.90**	14.91	7.26	0.00

Continued

Table A1
Continued

	MGMV17		Matheuristic			
Instance	Best	Time	Best	Avg.	Time	Dev (%)
tc2c10s3ct0	**11.51**	6.81	**11.51**	11.52	8.08	0.00
tc0c10s3ct1	**10.80**	4.83	**10.80**	10.81	7.03	0.00
tc1c10s3ct2	**9.20**	5.33	**9.20**	9.21	7.91	0.00
tc1c10s3ct3	**13.02**	9.77	**13.02**	13.03	7.27	0.00
tc1c10s2ct4	**13.83**	4.84	**13.83**	13.84	7.42	0.00
tc2c20s3cf0	**24.68**	13.86	**24.68**	24.70	9.51	0.00
tc1c20s3cf1	**17.50**	12.32	**17.50**	17.53	11.78	0.00
tc0c20s3cf2	27.60	11.77	**27.47**	27.52	10.82	−0.47
tc1c20s3cf3	16.63	8.41	**16.48**	16.50	16.27	−0.90
tc1c20s3cf4	**17.00**	3.77	**17.00**	17.03	13.75	0.00
tc2c20s3ct0	**25.79**	14.66	**25.79**	25.83	13.66	0.00
tc1c20s3ct1	**18.95**	15.25	**18.95**	18.97	12.26	0.00
tc0c20s3ct2	**17.08**	8.49	**17.08**	17.11	16.52	0.00
tc1c20s3ct3	12.65	8.86	**12.60**	12.62	15.59	−0.40
tc1c20s3ct4	**16.21**	5.16	**16.21**	16.24	9.85	0.00
tc2c20s4cf0	**24.67**	14.63	**24.67**	24.71	11.76	0.00
tc1c20s4cf1	**16.39**	13.47	16.47	16.49	16.36	0.49
tc0c20s4cf2	**27.48**	12.81	27.60	27.65	13.01	0.44
tc1c20s4cf3	16.56	8.69	**16.48**	16.51	10.14	−0.48
tc1c20s4cf4	**17.00**	4.17	**17.00**	17.03	14.01	0.00
tc2c20s4ct0	**26.02**	15.25	26.03	26.07	15.03	0.04
tc1c20s4ct1	**18.25**	16.14	**18.25**	18.28	16.55	0.00
tc0c20s4ct2	**16.99**	9.33	**16.99**	17.02	15.00	0.00
tc1c20s4ct3	**14.43**	7.99	**14.43**	14.46	10.74	0.00
tc1c20s4ct4	**17.00**	6.08	**17.00**	17.03	9.29	0.00
tc0c40s5cf0	**32.67**	23.85	**32.67**	32.75	49.54	0.00
tc1c40s5cf1	**65.16**	44.01	65.52	65.67	31.91	0.55
tc2c40s5cf2	**27.54**	31.64	**27.54**	27.62	40.27	0.00
tc2c40s5cf3	19.74	16.85	**19.65**	19.70	48.65	−0.46
tc0c40s5cf4	30.77	33.33	**30.60**	30.69	46.87	−0.55
tc0c40s5ct0	28.72	24.50	**28.70**	28.78	44.47	−0.07
tc1c40s5ct1	52.68	58.52	**52.60**	52.72	38.37	−0.15
tc2c40s5ct2	**26.91**	22.85	**26.91**	26.99	40.43	0.00
tc2c40s5ct3	**23.54**	26.48	23.71	23.75	49.87	0.72
tc0c40s5ct4	**28.63**	32.55	29.17	29.25	57.74	1.89
tc0c40s8cf0	31.28	33.59	**31.23**	31.31	61.18	−0.16
tc1c40s8cf1	**40.75**	69.99	41.63	41.71	32.67	2.16
tc2c40s8cf2	27.15	28.92	**27.14**	27.20	33.46	−0.04
tc2c40s8cf3	19.66	19.46	**19.65**	19.69	35.19	−0.05
tc0c40s8cf4	29.32	43.05	**28.25**	28.30	49.90	−3.65
tc0c40s8ct0	26.35	28.54	**26.22**	26.27	56.57	−0.49
tc1c40s8ct1	**40.56**	70.50	**40.56**	40.67	47.63	0.00
tc2c40s8ct2	26.33	25.64	**26.29**	26.34	38.74	−0.15
tc2c40s8ct3	22.71	25.25	**22.45**	22.52	28.77	−1.14

Continued

Table A1
Continued

	MGMV17		Matheuristic			
Instance	Best	Time	Best	Avg.	Time	Dev (%)
tc0c40s8ct4	**29.20**	47.46	29.22	29.28	45.94	0.07
tc0c80s8cf0	**39.43**	56.41	40.64	40.77	84.74	3.07
tc0c80s8cf1	**45.23**	121.27	46.65	46.80	95.12	3.14
tc1c80s8cf2	**30.81**	50.99	31.38	31.47	91.63	1.85
tc2c80s8cf3	32.44	64.05	**31.94**	32.06	84.36	−1.54
tc2c80s8cf4	**49.29**	99.84	49.67	49.84	125.86	0.77
tc0c80s8ct0	41.90	54.35	**41.44**	41.59	84.39	−1.10
tc0c80s8ct1	45.27	129.66	**45.25**	45.37	97.11	−0.04
tc1c80s8ct2	31.74	59.73	**31.72**	31.82	94.85	−0.06
tc2c80s8ct3	**32.31**	65.15	32.71	32.82	86.25	1.24
tc2c80s8ct4	44.83	111.24	**44.16**	44.31	100.80	−1.49
tc0c80s12cf0	**34.64**	57.24	35.24	35.40	102.60	1.73
tc0c80s12cf1	42.90	74.58	**42.30**	42.47	82.08	−1.40
tc1c80s12cf2	**29.54**	61.34	**29.54**	29.66	110.20	0.00
tc2c80s12cf3	31.97	75.64	**31.28**	31.37	102.84	−2.16
tc2c80s12cf4	43.89	131.13	**43.69**	43.81	83.46	−0.46
tc0c80s12ct0	39.31	65.54	**39.27**	39.41	83.00	−0.10
tc0c80s12ct1	41.94	73.32	**41.64**	41.83	101.10	−0.72
tc1c80s12ct2	29.52	58.85	**29.38**	29.47	111.08	−0.47
tc2c80s12ct3	30.83	57.57	**30.31**	30.39	111.55	−1.69
tc2c80s12ct4	**42.40**	134.33	42.56	44.68	100.48	0.38
tc1c160s16cf0	79.80	765.69	**79.76**	80.52	409.13	−0.05
tc2c160s16cf1	60.34	273.86	**60.25**	60.70	704.96	−0.15
tc0c160s16cf2	**61.20**	365.10	62.09	62.55	430.06	1.45
tc1c160s16cf3	**71.76**	461.58	71.98	72.77	717.90	0.31
tc0c160s16cf4	82.92	1213.20	**82.77**	83.41	697.40	−0.18
tc1c160s16ct0	**79.04**	643.27	80.21	80.99	460.66	1.48
tc2c160s16ct1	60.27	287.64	**59.86**	60.40	394.86	−0.68
tc0c160s16ct2	59.90	341.86	**59.75**	60.29	798.52	−0.25
tc1c160s16ct3	73.29	278.67	**73.24**	73.82	738.37	−0.07
tc0c160s16ct4	**82.37**	944.60	82.90	83.85	970.13	0.64
tc1c160s24cf0	**78.60**	741.12	79.48	80.32	448.21	1.12
tc2c160s24cf1	**59.82**	304.66	60.01	60.63	551.78	0.32
tc0c160s24ct2	**59.25**	409.80	59.67	60.21	904.41	0.71
tc1c160s24ct3	**68.72**	358.35	69.17	69.76	876.47	0.65
tc0c160s24cf4	81.44	1209.32	**81.43**	82.33	583.63	−0.01
tc1c160s24ct0	**78.21**	577.83	78.32	79.05	540.34	0.14
tc2c160s24ct1	**59.13**	340.40	59.97	60.53	519.34	1.42
tc0c160s24cf2	59.27	403.33	**59.26**	59.79	720.26	−0.02
tc1c160s24cf3	**68.56**	483.10	68.73	69.28	508.89	0.25
tc0c160s24ct4	**80.96**	956.94	81.38	82.21	424.93	0.52
tc2c320s24cf0	**182.45**	6566.41	182.90	185.27	6381.14	0.25
tc2c320s24cf1	**95.51**	1456.16	95.71	96.81	1359.35	0.21
tc1c320s24cf2	**152.13**	7105.63	153.12	154.65	6950.98	0.65

Continued

Table A1
Continued

Instance	MGMV17		Matheuristic			
	Best	Time	Best	Avg.	Time	Dev (%)
tc1c320s24cf3	117.48	3065.82	**117.39**	118.43	2947.39	−0.08
tc2c320s24cf4	**122.74**	3681.14	122.83	124.51	3556.63	0.07
tc2c320s24ct0	**181.45**	7204.02	182.29	183.80	7020.22	0.46
tc2c320s24ct1	**94.73**	1259.26	94.97	95.96	1163.30	0.25
tc1c320s24ct2	148.77	6853.35	**148.57**	149.89	6703.46	−0.13
tc1c320s24ct3	**116.64**	3273.79	117.50	118.53	3155.26	0.74
tc2c320s24ct4	**121.94**	4273.94	122.09	123.45	4150.49	0.12
tc2c320s38cf0	**176.92**	6733.82	178.17	179.81	6554.01	0.71
tc2c320s38cf1	**94.29**	1601.78	95.73	96.79	1504.99	1.53
tc1c320s38cf2	**141.63**	7235.62	142.25	144.17	7091.45	0.44
tc1c320s38cf3	**116.22**	3113.71	117.31	118.78	2994.93	0.94
tc2c320s38cf4	122.32	2660.68	**122.26**	123.46	2537.22	−0.05
tc2c320s38ct0	**190.97**	7636.50	192.23	194.66	7441.84	0.66
tc2c320s38ct1	**94.53**	1408.88	94.66	95.87	1313.01	0.14
tc1c320s38ct2	**140.96**	6974.34	142.75	144.50	6829.84	1.27
tc1c320s38ct3	**116.07**	3062.95	117.91	119.40	2943.55	1.59
tc2c320s38ct4	121.66	2784.91	**121.64**	123.06	2661.85	−0.02

Appendix B

Table B1 presents the detailed results on all eletric vehicle routing problem with shared charging stations (E-VRP-SCS) benchmark instances. For each instance, the best value of 10 runs is reported. The time column shows the average computation time of 10 runs.

Table B1
Results for the E-VRP-SCS benchmark instances

Instance	Matheuristic							
	Driver cost	CS cost	Total cost	No. of fast CS	No. of moderate CS	No. of slow CS	Avg total cost	Time
E-VRP-SCS-tc2c10s2cf0	210.17	5.28	215.45	0	0	1	215.88	10.45
E-VRP-SCS-tc0c10s2cf1	367.90	40.00	407.90	2	0	0	408.55	11.47
E-VRP-SCS-tc1c10s2cf2	191.56	20.00	211.56	1	0	0	211.73	10.40
E-VRP-SCS-tc1c10s2cf3	265.64	25.28	290.92	1	0	1	291.21	9.16
E-VRP-SCS-tc1c10s2cf4	272.18	25.28	297.46	1	0	1	297.94	11.60
E-VRP-SCS-tc2c10s2ct0	217.64	5.28	222.92	0	0	1	223.30	9.88
E-VRP-SCS-tc0c10s2ct1	233.25	40.00	273.25	2	0	0	273.44	11.62
E-VRP-SCS-tc1c10s2ct2	185.37	25.28	210.65	1	0	1	210.82	11.67
E-VRP-SCS-tc1c10s2ct3	256.49	25.28	281.77	1	0	1	282.19	11.50
E-VRP-SCS-tc1c10s3ct4	268.27	25.28	293.55	1	0	1	293.99	9.28

Continued

Table B1
Continued

Instance	Matheuristic							
	Driver cost	CS cost	Total cost	No. of fast CS	No. of moderate CS	No. of slow CS	Avg total cost	Time
E-VRP-SCS-tc2c10s3cf0	210.17	5.28	215.45	0	0	1	215.56	11.79
E-VRP-SCS-tc0c10s3cf1	367.90	40.00	407.90	2	0	0	408.14	10.18
E-VRP-SCS-tc1c10s3cf2	191.56	20.00	211.56	1	0	0	211.79	10.83
E-VRP-SCS-tc1c10s3cf3	265.64	25.28	290.92	1	0	1	291.42	11.15
E-VRP-SCS-tc1c10s3cf4	272.03	30.56	302.59	1	0	2	302.86	10.50
E-VRP-SCS-tc2c10s3ct0	240.79	27.79	268.58	0	1	2	269.01	11.15
E-VRP-SCS-tc0c10s3ct1	248.08	40.00	288.08	2	0	0	288.45	9.56
E-VRP-SCS-tc1c10s3ct2	188.46	25.28	213.74	1	0	1	213.93	11.12
E-VRP-SCS-tc1c10s3ct3	256.16	25.28	281.44	1	0	1	281.92	11.31
E-VRP-SCS-tc1c10s2ct4	273.69	25.28	298.97	1	0	1	299.51	9.29
E-VRP-SCS-tc2c20s3cf0	381.83	34.46	416.29	0	2	0	417.17	19.68
E-VRP-SCS-tc1c20s3cf1	326.69	25.28	351.97	1	0	1	352.71	19.65
E-VRP-SCS-tc0c20s3cf2	585.70	40.00	625.70	2	0	0	627.14	24.74
E-VRP-SCS-tc1c20s3cf3	415.87	60.00	475.87	3	0	0	477.25	20.00
E-VRP-SCS-tc1c20s3cf4	353.31	34.46	387.77	0	2	0	388.35	20.09
E-VRP-SCS-tc2c20s3ct0	452.92	54.46	507.38	1	2	0	508.29	24.06
E-VRP-SCS-tc1c20s3ct1	341.71	25.28	366.99	1	0	1	367.91	23.07
E-VRP-SCS-tc0c20s3ct2	390.21	40.00	430.21	2	0	0	431.20	25.55
E-VRP-SCS-tc1c20s3ct3	347.54	40.00	387.54	2	0	0	388.43	20.95
E-VRP-SCS-tc1c20s3ct4	342.89	22.51	365.40	0	1	1	366.10	22.42
E-VRP-SCS-tc2c20s4cf0	381.83	34.46	416.29	0	2	0	417.50	24.07
E-VRP-SCS-tc1c20s4cf1	329.88	45.28	375.16	2	0	1	375.99	26.51
E-VRP-SCS-tc0c20s4cf2	570.79	60.00	630.79	3	0	0	632.62	21.91
E-VRP-SCS-tc1c20s4cf3	428.53	45.28	473.81	2	0	1	474.66	26.70
E-VRP-SCS-tc1c20s4cf4	339.66	34.46	374.12	0	2	0	375.21	20.70
E-VRP-SCS-tc2c20s4ct0	463.03	34.46	497.49	0	2	0	498.99	23.16
E-VRP-SCS-tc1c20s4ct1	346.95	45.28	392.23	2	0	1	393.25	20.53
E-VRP-SCS-tc0c20s4ct2	415.26	60.00	475.26	3	0	0	475.97	26.25
E-VRP-SCS-tc1c20s4ct3	403.90	45.28	449.18	2	0	1	449.94	19.44
E-VRP-SCS-tc1c20s4ct4	329.12	39.74	368.86	0	2	1	369.64	26.35
E-VRP-SCS-tc0c40s5cf0	714.93	27.79	742.72	0	1	2	745.70	50.66
E-VRP-SCS-tc1c40s5cf1	909.19	33.07	942.26	0	1	3	944.62	72.02
E-VRP-SCS-tc2c40s5cf2	570.96	39.74	610.70	0	2	1	612.11	70.64
E-VRP-SCS-tc2c40s5cf3	479.20	37.23	516.43	1	1	0	518.30	51.48
E-VRP-SCS-tc0c40s5cf4	851.12	37.23	888.35	1	1	0	890.40	50.23
E-VRP-SCS-tc0c40s5ct0	706.16	53.07	759.23	1	1	3	760.98	44.47
E-VRP-SCS-tc1c40s5ct1	840.01	38.35	878.36	0	1	4	881.71	60.10
E-VRP-SCS-tc2c40s5ct2	555.96	39.74	595.70	0	2	1	597.01	69.79
E-VRP-SCS-tc2c40s5ct3	540.79	22.51	563.30	0	1	1	564.66	47.62
E-VRP-SCS-tc0c40s5ct4	686.27	80.00	766.27	4	0	0	767.73	57.94
E-VRP-SCS-tc0c40s8cf0	693.38	47.79	741.17	1	1	2	742.66	72.92
E-VRP-SCS-tc1c40s8cf1	764.99	47.79	812.78	1	1	2	814.49	40.57
E-VRP-SCS-tc2c40s8cf2	590.46	56.97	647.43	0	3	1	649.18	50.61

Continued

Table B1
Continued

Instance	Matheuristic							
	Driver cost	CS cost	Total cost	No. of fast CS	No. of moderate CS	No. of slow CS	Avg total cost	Time
E-VRP-SCS-tc2c40s8cf3	447.93	37.23	485.16	1	1	0	486.72	57.78
E-VRP-SCS-tc0c40s8cf4	653.45	65.28	718.73	3	0	1	720.53	52.54
E-VRP-SCS-tc0c40s8ct0	622.73	70.30	693.03	1	2	3	695.74	40.94
E-VRP-SCS-tc1c40s8ct1	763.53	73.07	836.60	2	1	3	839.62	50.64
E-VRP-SCS-tc2c40s8ct2	551.48	67.53	619.01	0	3	3	620.50	73.04
E-VRP-SCS-tc2c40s8ct3	537.18	42.51	579.69	1	1	1	580.97	35.59
E-VRP-SCS-tc0c40s8ct4	677.71	80.00	757.71	4	0	0	760.52	68.03
E-VRP-SCS-tc0c80s8cf0	1120.16	65.02	1185.18	1	2	2	1188.98	106.59
E-VRP-SCS-tc0c80s8cf1	1168.48	76.97	1245.45	1	3	1	1248.70	122.34
E-VRP-SCS-tc1c80s8cf2	787.21	42.51	829.72	1	1	1	834.56	80.65
E-VRP-SCS-tc2c80s8cf3	853.70	15.84	869.54	0	0	3	873.82	124.15
E-VRP-SCS-tc2c80s8cf4	1004.37	75.58	1079.95	1	2	4	1084.29	100.43
E-VRP-SCS-tc0c80s8ct0	1118.21	91.69	1209.90	2	3	0	1213.42	119.81
E-VRP-SCS-tc0c80s8ct1	1159.39	82.25	1241.64	1	3	2	1245.38	95.49
E-VRP-SCS-tc1c80s8ct2	824.36	37.23	861.59	1	1	0	865.22	90.38
E-VRP-SCS-tc2c80s8ct3	879.43	73.07	952.50	2	1	3	957.58	95.26
E-VRP-SCS-tc2c80s8ct4	991.18	65.02	1056.20	1	2	2	1059.27	112.35
E-VRP-SCS-tc0c80s12cf0	986.12	79.74	1065.86	2	2	1	1068.96	87.24
E-VRP-SCS-tc0c80s12cf1	1158.76	99.74	1258.50	3	2	1	1262.67	110.35
E-VRP-SCS-tc1c80s12cf2	803.41	27.79	831.20	0	1	2	833.62	98.34
E-VRP-SCS-tc2c80s12cf3	826.90	15.84	842.74	0	0	3	845.02	106.18
E-VRP-SCS-tc2c80s12cf4	915.23	102.25	1017.48	2	3	2	1021.77	82.38
E-VRP-SCS-tc0c80s12ct0	997.53	91.69	1089.22	2	3	0	1092.94	105.85
E-VRP-SCS-tc0c80s12ct1	1018.72	99.74	1118.46	3	2	1	1124.65	121.39
E-VRP-SCS-tc1c80s12ct2	833.52	54.46	887.98	1	2	0	891.99	118.65
E-VRP-SCS-tc2c80s12ct3	825.06	33.07	858.13	0	1	3	861.23	92.06
E-VRP-SCS-tc2c80s12ct4	1012.26	82.25	1094.51	1	3	2	1098.35	113.33
E-VRP-SCS-tc1c160s16cf0	1794.03	121.99	1916.02	1	5	3	1930.69	629.71
E-VRP-SCS-tc2c160s16cf1	1606.30	111.43	1717.73	1	5	1	1725.49	431.07
E-VRP-SCS-tc0c160s16cf2	1693.99	160.61	1854.60	2	7	0	1867.11	726.26
E-VRP-SCS-tc1c160s16cf3	1842.12	68.91	1911.03	1	1	6	1920.63	578.01
E-VRP-SCS-tc0c160s16cf4	2191.44	161.99	2353.43	3	5	3	2378.88	756.05
E-VRP-SCS-tc1c160s16ct0	1760.63	128.92	1889.55	3	4	0	1903.83	778.97
E-VRP-SCS-tc2c160s16ct1	1562.00	91.69	1653.69	2	3	0	1660.83	831.67
E-VRP-SCS-tc0c160s16ct2	2112.61	183.12	2295.73	2	8	1	2316.81	443.76
E-VRP-SCS-tc1c160s16ct3	1974.85	120.86	2095.71	3	2	5	2116.88	658.28
E-VRP-SCS-tc0c160s16ct4	2377.38	141.99	2519.37	2	5	3	2546.62	667.00
E-VRP-SCS-tc1c160s24cf0	1776.92	179.22	1956.14	3	6	3	1978.30	544.32
E-VRP-SCS-tc2c160s24cf1	1562.02	156.45	1718.47	1	7	3	1738.11	741.84
E-VRP-SCS-tc0c160s24ct2	1768.69	179.48	1948.17	5	4	2	1959.54	733.01
E-VRP-SCS-tc1c160s24ct3	1797.66	119.48	1917.14	2	4	2	1925.03	765.51
E-VRP-SCS-tc0c160s24cf4	1936.05	201.99	2138.04	5	5	3	2154.41	791.80
E-VRP-SCS-tc1c160s24ct0	1758.47	183.38	1941.85	4	6	0	1964.24	586.65

Continued

Table B1
Continued

Instance	Matheuristic							
	Driver cost	CS cost	Total cost	No. of fast CS	No. of moderate CS	No. of slow CS	Avg total cost	Time
E-VRP-SCS-tc2c160s24ct1	1547.61	128.66	1676.27	1	6	1	1691.83	564.58
E-VRP-SCS-tc0c160s24cf2	1735.19	193.94	1929.13	4	6	2	1952.56	606.17
E-VRP-SCS-tc1c160s24cf3	1804.89	110.04	1914.93	1	4	4	1933.49	485.76
E-VRP-SCS-tc0c160s24ct4	1855.60	181.99	2037.59	4	5	3	2050.71	670.52
E-VRP-SCS-tc2c320s24cf0	4995.10	247.27	5242.37	7	5	4	5293.18	4519.46
E-VRP-SCS-tc2c320s24cf1	3041.18	132.55	3173.73	1	5	5	3192.25	3392.40
E-VRP-SCS-tc1c320s24cf2	4223.98	232.55	4456.53	6	5	5	4500.18	5492.29
E-VRP-SCS-tc1c320s24cf3	3582.23	240.34	3822.57	5	6	7	3841.78	4578.52
E-VRP-SCS-tc2c320s24cf4	3128.90	155.58	3284.48	5	2	4	3314.31	4754.55
E-VRP-SCS-tc2c320s24ct0	4780.85	247.27	5028.12	7	5	4	5049.33	5578.28
E-VRP-SCS-tc2c320s24ct1	3258.26	236.45	3494.71	5	7	3	3520.76	5037.43
E-VRP-SCS-tc1c320s24ct2	3843.49	227.27	4070.76	6	5	4	4087.52	4994.87
E-VRP-SCS-tc1c320s24ct3	3934.33	244.24	4178.57	4	8	5	4206.33	5981.89
E-VRP-SCS-tc2c320s24ct4	3241.49	240.60	3482.09	7	4	6	3518.68	3968.36
E-VRP-SCS-tc2c320s38cf0	4526.81	345.63	4872.44	9	9	2	4913.22	5307.55
E-VRP-SCS-tc2c320s38cf1	2708.81	219.48	2928.29	7	4	2	2941.82	5968.22
E-VRP-SCS-tc1c320s38cf2	4198.68	297.84	4496.52	8	8	0	4544.23	3747.16
E-VRP-SCS-tc1c320s38cf3	3315.11	224.50	3539.61	5	6	4	3580.06	5146.70
E-VRP-SCS-tc2c320s38cf4	2994.90	283.12	3278.02	7	8	1	3300.80	5755.70
E-VRP-SCS-tc2c320s38ct0	4680.11	337.84	5017.95	10	8	0	5067.10	3793.19
E-VRP-SCS-tc2c320s38ct1	2798.93	208.92	3007.85	7	4	0	3027.53	5199.81
E-VRP-SCS-tc1c320s38ct2	4307.03	232.81	4539.84	8	3	4	4593.12	5132.67
E-VRP-SCS-tc1c320s38ct3	3552.16	236.71	3788.87	7	5	2	3813.66	4995.48
E-VRP-SCS-tc2c320s38ct4	2997.24	207.53	3204.77	7	3	3	3228.01	3130.77