A Matheuristic Method for the Electric Vehicle Routing Problem with Time Windows and Fast Chargers

Merve Keskin[a,b] and Bülent Çatay[a,b,*]

[a] Sabanci University, Faculty of Engineering and Natural Sciences, Istanbul, Turkey
[b] Smart Mobility and Logistics Lab, Sabanci University, Istanbul, Turkey

---

* Corresponding author. catay@sabanciuniv.edu.

Abstract

The Electric Vehicle Routing Problem with Time Windows (EVRPTW) is an extension of the well-known VRPTW where electric vehicles (EVs) are used instead of internal combustion engine vehicles. An EV has a limited driving range due to its battery capacity and may need recharging to complete its route. Recharging can be made at any battery level and may be at any quantity up to the battery capacity. Furthermore, the stations may be equipped with chargers with different power supply, power voltage, maximum current options which affect the recharge duration. In this study, we model the EVRPTW by allowing partial recharges with three recharging configurations which can be referred to as normal, fast and super-fast recharges. In faster options, the battery is charged with the same energy in a shorter time but at a higher cost. Our objective is to minimize the total recharging cost while operating minimum number of vehicles. We formulated this problem as a mixed integer linear program and solved the small instances using CPLEX. To solve the larger problems, we develop a matheuristic approach which couples the Adaptive Large Neighborhood Search (ALNS) approach with an exact method. Our ALNS is equipped with various destroy-repair algorithms to efficiently explore the neighborhoods and uses CPLEX to strengthen the routes obtained. We carried out extensive experiments to investigate the benefits of fast recharges and test the performance of our algorithm using benchmark instances from the literature. The results show the effectiveness of the proposed matheuristic and demonstrate the benefits of fast chargers on the fleet size and energy costs.

1. Introduction

Transportation systems have a major impact on global energy consumption and CO_2 emissions with a share of around 20-25%. In the US, 26% of the total greenhouse gas (GHG) emissions in 2014 was generated by transportation systems that utilize fossil fuels (www.epa.gov). Furthermore, 74% of the domestic freight in 2012 was moved by trucks and the freight volume is expected to grow by 39% in 2040 (Bureau of Transportation Statistics, 2014). Similarly, the EU reported that transportation was a main contributor with 23.2% of total GHG emissions in 2014 and freight transport activity is predicted to grow by around 80% in 2050 compared to 1990 (ec.europa.eu).

Transportation will remain a major and growing source of GHGs in the future. Hence, governments initiated new environmental measures and targets for reducing emissions and cutting the dependency on fossil fuels. For instance, US government targets reducing GHG emissions 20%

2

below 2008 levels by 2020 (www.state.gov). The EU aims a reduction of 80-95% by 2050 with respect to 1990 (White Paper on Transport, 2011). In December 2016, a commitment was signed by 194 countries in New York to set a global action to stop the global temperature rise (unfccc.int). Since transportation plays a major part in GHG emissions and road transport contributes with a 75% share, the new regulations bring limitations to the use of internal combustion engine vehicles (ICEVs). In the EU, the use of ICEVs will be reduced by 50% in urban transport by 2030 and phased out by 2050. City logistics in major European urban centers will be CO_2-free by 2030 (White Paper on Transport, 2011). The parliaments of Netherlands and Norway recently passed new motions that will end sales of new cars powered by fossil fuels after 2025 (Edelstein, 2016). Similarly, German Federal Council accepted a resolution that bans the sales of fossil fuel cars by 2030 (Khan, 2017).

The targets set by governments and the new regulations encourage the usage of alternative fuel vehicles (AFVs) such as solar, electric, biodiesel, LNG, CNG vehicles. Many municipalities, government agencies, non-profit organizations, and private companies are converting their fleets to include AFVs, either to reduce their environmental impact voluntarily or to meet new environmental regulations (Erdoğan and Miller-Hooks, 2012). Consequently, the advancements in the electric vehicle (EV) technology have gained momentum in parallel with the growing environmental concerns in societies.

EVs can be classified as battery electric vehicles (BEVs), hybrid electric vehicles (HEVs), and fuel-cell electric vehicles (FCEVs) such as cars, vans, trucks, electric trains, airplanes, boats, and two-wheelers. In this paper, we refer to EV as a commercial road BEV. A fleet of EVs can be used in a variety of transport needs such as public transportation, home deliveries from grocery stores, postal deliveries and courier services, distribution operations in different sectors. The main advantages of EVs are zero tailpipe emission, high efficiency, and low operating noise (Pollet et al., 2012). The number of moving parts in EVs are much less than that of ICEVs and EVs do not require regular oil changes (Feng and Figliozzi, 2013). In addition, due to the regenerative breaking, break wear is used less which reduces the maintenance costs (Hiermann et al., 2015). On the other hand, operating an EV fleet has several drawbacks such as low achievable driving range, limited number of recharging stations, and long battery recharging times (Touati-Moungla and Jost, 2011). These limitations and additional complexities make the route planning of an EV fleet a

challenging combinatorial optimization problem. As a result, Electric Vehicle Routing Problem (EVRP) and its variants have attracted considerable attention in the recent literature.

EVRP is an extension to the Capacitated Vehicle Routing Problem (VRP) where a fleet of EVs is used instead of ICEVs. The energy stored in the battery is consumed along the journey proportional to the distance travelled and the EV may need recharging to complete its tour. Recharging may be performed at any battery state of charge (SoC). The stations are scarce and recharging may require a significant amount of time, compared to short refueling times at petrol stations. In this paper, we address the EVRP with Time Windows (EVRPTW) which was firstly introduced by Schneider et al. (2014). EVRPTW assumes that recharging time is a linear function of the energy transferred and the battery is fully charged. Bruglieri et al. (2015) relaxed the full charge restriction and allowed partial recharging with any quantity up to the battery capacity, which is the current practice in real-world applications.

In this study, we extend the EVRPTW-PR by introducing fast charging option and refer to this problem as EVRPTW and Fast Charging (EVRPTW-FC). Basically, we assume that the stations are equipped with multiple charger types. They vary in power supply, power voltage, and maximum current options, which affect the recharge duration. We formulate this problem as a 0-1 mixed integer linear program and propose a matheuristic approach to solve it efficiently. Our approach combines the Adaptive Large Neighborhood Search (ALNS) with an exact method. At each iteration of the ALNS, the feasible solution is destroyed by removing certain customers and stations from their routes and then repaired by inserting the removed customers back to the solution along with stations when recharging is necessary. When a station is inserted, the charger type and recharge quantity are also determined. The solution found by ALNS is then improved periodically by solving a mixed linear integer program which optimizes the decisions associated with recharge stations, charger types, and recharge quantities given the sequence of the customers visited.

The main contributions of the paper can be summarized as follows:

- We extend EVRPTW-PR to allow fast charging using multiple charging equipment types and present two mathematical programming formulations of EVRPTW-FC.
- As a solution methodology, we develop a matheuristic approach which combines ALNS with an exact method.

- The proposed ALNS involves new destroy and repair mechanisms specific to the nature of the problem.
- For a given sequence of customers, we propose a novel formulation of the charging sub-problem that can be solved to optimality fast.
- We devise an extensive experimental design to validate the performance of the proposed methodology and to show the benefits of fast charging.

The remainder of the paper is organized as follows: Section 2 reviews the relevant literature. Section 3 describes the problem and presents the mathematical models. The proposed solution approach is described in Section 4. Section 5 presents the computational study and provides the numerical results. Finally, the paper closes with concluding remarks in Section 6.

2. Related Literature

VRPs with AFVs context have been studied by several researchers in recent years. Conrad and Figliozzi (2011) present the recharging VRP where the EVs are recharged at selected customer locations during the service. The primary objective is to minimize the number of vehicles while the secondary objective minimizes total costs associated with travel distance, service time, and recharging. The latter is a penalty cost incurred at each recharge. The charging time is assumed constant and the battery SoC when departing from a customer can be full (fully charged) or 80% of the capacity (partially charged). The authors used an iterative route construction and improvement procedure to solve the problem.

Erdoğan and Miller-Hooks (2012) considered Green VRP (GVRP) in which AFVs are refueled at stations en-route. Refueling times are assumed constant and after refueling the tank becomes full. The fuel is consumed proportional to the distance travelled. The problem does not consider any vehicle capacity or customer time windows. The objective is to minimize the total travel distance. The authors developed two heuristics and designed 52 instances with varying sizes from 20 up to 500 customers to test their performances.

Schneider et al. (2014) introduced EVRP with Time Windows (EVRPTW) as an extension to GVRP. In contrast to short refueling times in GVRP, the recharging operation may take a significant amount of time in EVRPTW. Recharging starts at any SoC, continues until the battery is full, and its duration is proportional to the energy transferred. The objective is to minimize the total distance travelled by using minimum number of vehicles. The authors proposed a hybrid

Variable Neighborhood Search (VNS) and Tabu Search (TS) method and tested its performance on the instances for GVRP and Multi-Depot VRP with Inter-Depot Routes. They also generated new instances for EVRPTW by modifying Solomon (1987) data and reported their results. Desaulniers et al. (2016) tackled the same problem by considering different charging strategies and developed branch-price-and-cut algorithms to solve them optimally. Goeke and Schneider (2015) extended EVRPTW to include a mixed fleet of EVs and ICEVs. The objective is to minimize the total cost defined as a function of speed, gradient and cargo load. Hiermann et al. (2016) addressed the Fleet Size and Mix Vehicle Routing Problem with Time Windows where the fleet consists of EVs only. They minimize the total cost of vehicle acquisition and distance travelled. Both studies used ALNS as the solution methodology. Bruglieri et al. (2015) and Keskin and Çatay (2016) relaxed the full recharge restriction and allowed batteries to be recharged up to any level. The former minimized the number of vehicles, travel time, waiting time, and recharging time, developed a Variable Neighborhood Search Branching method, and used it to solve small size instances. The latter extended the model of Schneider et al. (2014) to formulate EVRPTW with Partial Recharges (EVRPTW-PR) and proposed an ALNS approach that improved some of the best-known results in the literature. Bruglieri et al. (2016) formulated a more effective mathematical model for GVRP by reducing the number of variables and eliminating dominated stations for each pair of customers.

Felipe et al. (2014) addressed EVRP by allowing partial recharges using multiple technologies. They restricted total route duration but did not consider time windows. They proposed both local search algorithms and a Simulated Annealing (SA) based metaheuristic approach. Recently, the impact of fast charging option in the presence of time windows has also been studied by Çatay and Keskin (2017).

Montoya et al. (2017) is the first study that extended EVRP to consider nonlinear charging functions. The objective function minimizes the total time which includes travel and charging time. The authors proposed a hybrid metaheuristic to solve the problem and introduced new benchmark instances. New formulations of this problem are proposed in Froger et al. (2017).

Yang and Sun (2015) modeled location and routing decisions simultaneously for a capacitated EV fleet. They consider battery swap stations (BSSs) instead of recharging stations where the EVs always depart from stations with full battery. Hof et al. (2017) also investigated EVRP with BSSs

and developed an Adaptive VNS (AVNS) approach to solve it. Recently, Schiffer and Walther (2017) proposed a location-routing model for EVRPTW allowing partial recharging at customer locations. They analyzed the effects of different objective functions. Paz et al. (2017) also modeled the same problem considering multi depots and battery swapping option in some locations.

Pelletier et al. (2017) integrated battery degradation into the model and optimized charging schedules at the depot. They also provided managerial insights considering degradation, grid restrictions, charging costs, and charging schedules of the fleet. A detailed survey of the goods distribution with EVs can be found in Pelletier et al. (2016) and Pelletier et al. (2017).

3. Problem Description and Model Formulation

3.1 Problem Description

Given a homogeneous fleet of EVs, EVRPTW-FC aims to determine a set of routes involving customers with known demands, delivery time windows, service durations, and recharging stations with different types of chargers. The charging levels can be classified into three categories: Level 1 (1.4 kW to 1.9 kW), Level 2 (4 kW to 19.2 kW), and Level 3 (50 kW to 100 kW) (Yilmaz and Krein, 2013). In line with the current technology, we assume that every station is equipped with three types of chargers, which may be referred to as normal, fast, and super-fast charger, respectively. While recharging takes less time in fast and super-fast charging options, the unit cost of energy is higher since the installation of the chargers requires substantial electrical infrastructure and the equipment is more expensive. The charge durations are linear with respect to time at the first phase of charging which corresponds to almost full battery while the second phase is non-linear and can take hours to obtain a fully charged battery (Montoya et al. 2017). On the other hand, it is a common industrial practice to operate within the first phase because recharging the battery up to full capacity can adversely affect its lifespan (Sweda et al., 2017). So, without loss of generality we assume linear recharging times in this study. In addition, we allow only one recharge between two consecutive customers which is the realistic situation within the context of urban logistics. Our objective function is hierarchical where minimizing the number of vehicles is the primary objective while minimizing the total cost of energy consumed is the secondary.
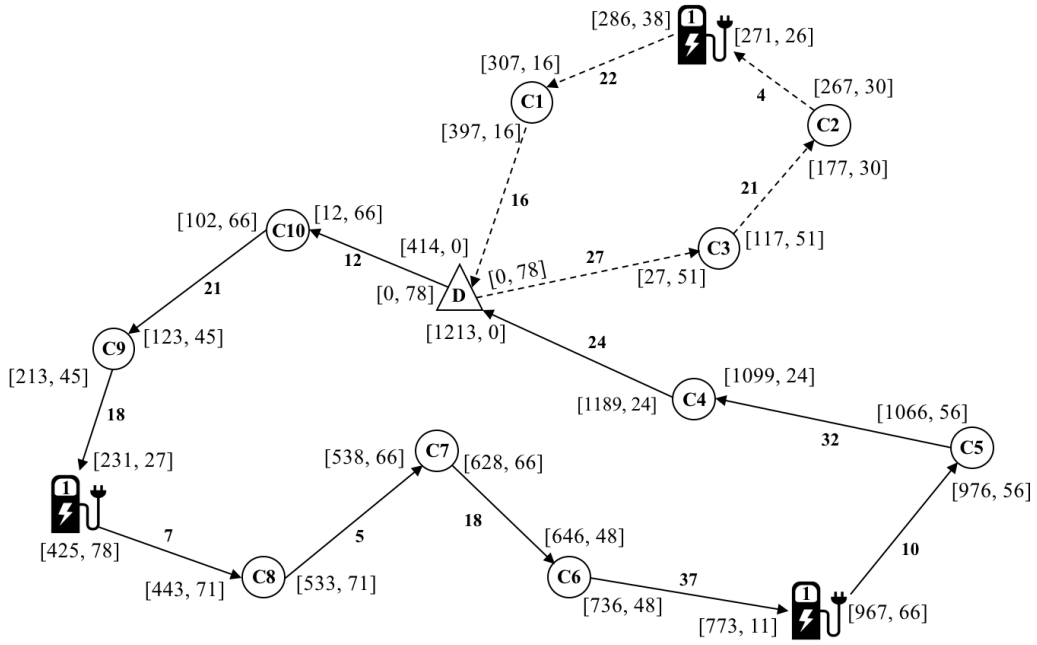
(a) Optimal solution using single charger (Total cost=273.93)
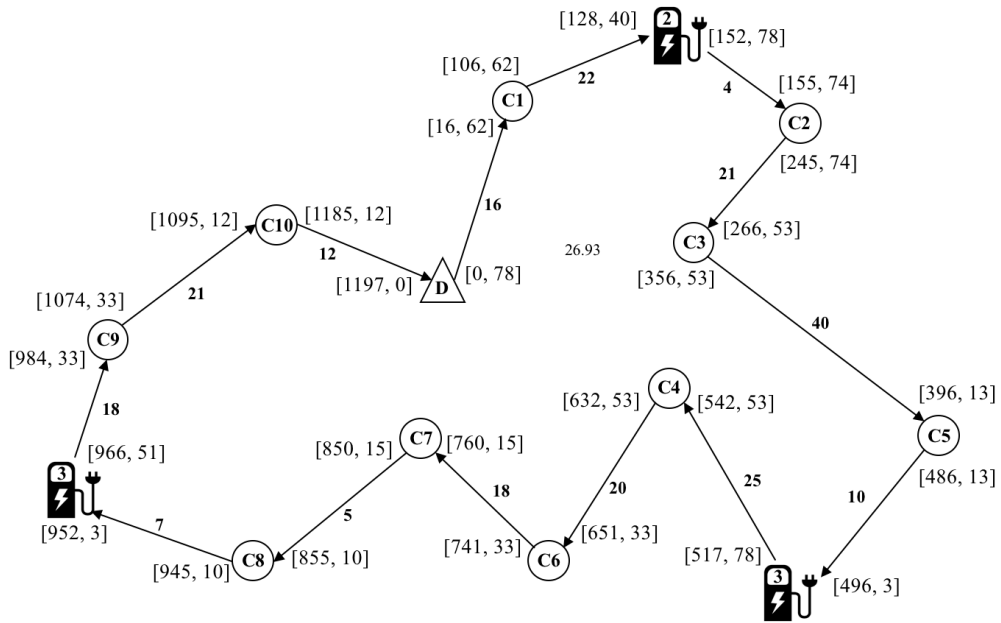
(b) Optimal solution using multiple chargers (Total cost=267.60)

Figure 1. Route plans when each recharging station is equipped with (a) only normal chargers, (b) normal, fast and super-fast chargers

To describe the general setting and highlight the advantage of using fast chargers we employ the instance c104c10-s3 of Schneider et al. (2014) which involves ten customers and three stations. The problem is solved using CPLEX and the optimal solution is illustrated in Figure 1. The recharging stations are represented with charger icons. The numbers on the icons refer to level 1 (normal), level 2 (fast), and level 3 (super-fast) chargers. The cargo and battery capacities of the vehicles are 200 and 77.75 units, respectively. The vehicles travel one unit distance in one unit of time consuming one unit of energy. Normal, fast, and super-fast chargers transfer one unit of energy in 3.47, 0.62, and 0.28 time units, respectively, at the cost of 1, 1.1, and 1.2 units, respectively. The demands and time windows are provided in Table 1. The values given in brackets placed at the beginning (end) of an arc represent the time and battery SoC, respectively, at departure (arrival) from (at) the corresponding customer. The arc distances are shown in bold. For ease of understanding, we rounded all the values in the figure to the nearest integer whereas the objective function values are exact.

Table 1. Demand and time-window data for the example illustrated in Figure 1

	Depot	Customers									
	D	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Demand	-	10	10	30	10	10	40	20	10	20	20
Early service time	0	0	177	0	0	0	0	0	0	0	0
Late service time	1236	1129	243	1119	1122	1094	1111	1126	1122	1113	1133

In Figure 1.a, we see the optimal solution of the problem when only normal chargers are available at the stations. Two vehicles travel a total distance of 273.93 units at a total energy cost of 273.93. The two values coincide because the problem involves only normal chargers and one unit of energy is consumed to travel one unit of distance. On the other hand, Figure 1.b shows that all customers can be served by only one vehicle in a single tour when fast and super-fast chargers are available. This is due to the reduced recharging times at stations which enable the EV to catch the time windows of all customers. In this case, the total distance travelled is 239.13 and total cost of energy is 267.60. Although fast and super-fast recharges are more expensive, the total cost of energy decreases because the EV makes a shorter trip consuming less energy.

Table 2. Mathematical notation

Sets:

V	Set of customers
F	Set of recharging stations
F'	Set of recharging stations with their copies
V'	Set of customers and stations with their copies ($V \cup F'$)
DD	Set of departure depots
AD	Set of arrival depots
V_{DD}	Set of customers and departure depots ($V \cup DD$)
V_{AD}	Set of customers and arrival depots ($V \cup AD$)
V'_{DD}	Set of customers, departure depots, and stations with their copies ($V' \cup DD$)
V'_{AD}	Set of customers, arrival depots, and stations with copies ($V' \cup AD$)
F'_{DD}	Set of departure depots and stations with their copies ($F' \cup DD$)
$V'_{DD,AD}$	Set of customers, arrival/departure depots, and stations with their copies ($V' \cup AD \cup DD$)

Parameters:

d_{ij}	Distance from node i to node j
t_{ij}	Travel time from node i to node j
q_i	Demand of customer i
s_i	Time required to serve customer i
$[e_i, l_i]$	Service time window of customer i
C	Cargo capacity of the vehicles
Q	Battery capacity of the vehicles

Decision variables:

τ_i	Service starting time at node i
u_i	Remaining cargo level at node i
y_i	Battery state of charge at the arrival at node i
Y_i	Battery state of charge at the departure from node i
x_{ij}	1 if the vehicle traverses arc (i, j); 0 otherwise
a_i	1 if is recharged with normal charger at station i; 0 otherwise
b_i	1 if the vehicle is recharged with fast charger at station i; 0 otherwise
θ_i^m	Amount of energy recharged at station i using charger type m

3.2 Problem Formulation

We follow the notation used in Keskin and Çatay (2016) for ease of understanding. Let V and F denote the set of customers and the set of recharging stations, respectively. Since recharging stations may be visited multiple times by the same vehicle or different vehicles we create sufficient number of copies and allow at most one visit to each in the mathematical model. So, we define F' as the set of all recharging stations along with their copies and $V' = V \cup F'$. We assume that each station is equipped with all types of chargers but only one is used at each visit to the station. This

assumption can be easily relaxed but may not be practical in the real business environment. We also assume that all EVs are recharged with normal (cheapest) charger type over night at the depot and depart in the morning with full battery. To keep track of each EV's energy consumption along its tour we create copies of the depot by defining sets DD and AD as the departure depot and arrival depot vertices, respectively. We define $V_{DD} = V \cup DD$ and $V_{AD} = V \cup AD$. Let $V'_{DD} = V' \cup DD$, $V'_{AD} = V' \cup AD$, and $F'_{DD} = F' \cup DD$. Then, the problem can be represented by a complete graph $G = (V'_{DD,AD}, A)$ where $A = \{(i,j)|i \in V'_{DD}, j \in V_{AD}\}$ and $V'_{DD,AD} = V' \cup AD \cup DD$.

Each arc is associated with distance d_{ij} and travel time t_{ij}. The energy is consumed at a rate of h and the battery is discharged by hd_{ij} when the vehicle traverses arc (i,j). Each customer $i \in V$ is associated with demand q_i, service time s_i, and time window $[e_i, l_i]$. The fleet is homogeneous and consists of vehicles with cargo capacity C and battery capacity Q. The continuous decision variables τ_i, u_i, and y_i keep track of the service start time, remaining cargo level, and battery SoC upon arrival to each vertex, respectively. Y_i keeps track of the battery SoC at the departure from either the depot or a station. Finally, the binary variable x_{ij} takes value of 1 if arc (i,j) is traversed and 0, otherwise. The mathematical notation is given in Table 2.

In what follows, we present two alternative 0-1 mixed integer linear programming formulations of the problem.

3.2.1   Model 1

In this model, we define binary variables a_i and b_i to determine which charging equipment is used to recharge the vehicle at station $i \in F'$: $a_i = 1$ if charger type 1 (normal) is used, $b_i = 1$ if type 2 (fast) is used, and $a_i = b_i = 0$ if the charger type is 3 (super-fast). The battery recharging rate and unit energy cost depend on the charger type $m \in M$ and are referred to as g^m and c^m, respectively. Since we consider three charger types, $M = \{1,2,3\}$ and $m = 1$ corresponds to the normal (slowest) charger whereas $m = 3$ represents the super-fast (fastest) charger. θ_i^m denotes the amount of energy recharged at station i by using charger type m. Then, the problem is formulated as follows:

$$\text{Min} \quad \sum_{i \in F'} \sum_{m \in M} c^m \theta_i^m + c^1 \left(Q \sum_{i \in DD} \sum_{j \in V'} x_{ij} - \sum_{i \in AD} y_i \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{j \in V_{AD}} x_{ij} \leq 1 \qquad\qquad \forall i \in F' \qquad\qquad (2)$$

$$\sum_{j \in V'_{AD}} x_{ij} = 1 \qquad\qquad \forall i \in V \qquad\qquad (3)$$

$$\sum_{i \in V'_{DD}} x_{ij} = \sum_{i \in V'_{AD}} x_{ji} \qquad\qquad \forall j \in V' \qquad\qquad (4)$$

$$\sum_{j \in V'} x_{ij} \leq 1 \qquad\qquad \forall i \in DD \qquad\qquad (5)$$

$$\sum_{i \in V'} x_{ij} \leq 1 \qquad\qquad \forall j \in AD \qquad\qquad (6)$$

$$\sum_{i \in DD} \sum_{j \in V'} x_{ij} = \sum_{i \in AD} \sum_{j \in V'} x_{ji} \qquad\qquad (7)$$

$$\tau_i + (t_{ij} + s_i)x_{ij} - l_0(1 - x_{ij}) \leq \tau_j \qquad\qquad \forall i \in V_{DD}, \forall j \in V'_{AD} \qquad (8)$$

$$\tau_i + t_{ij}x_{ij} + \sum_{m \in M} g^m \theta_i^m - (l_0 + g^1 Q)(1 - x_{ij}) \leq \tau_j \qquad \forall i \in F', \forall j \in V_{AD} \qquad (9)$$

$$e_j \leq \tau_j \leq l_j \qquad\qquad \forall j \in V'_{DD,AD} \qquad (10)$$

$$0 \leq u_j \leq u_i - q_i x_{ij} + C(1 - x_{ij}) \qquad\qquad \forall i \in V'_{DD}, \forall j \in V'_{AD} \qquad (11)$$

$$0 \leq u_i \leq C \qquad\qquad \forall i \in DD \qquad\qquad (12)$$

$$0 \leq y_j \leq y_i - (hd_{ij})x_{ij} + Q(1 - x_{ij}) \qquad\qquad \forall i \in V, \forall j \in V'_{AD} \qquad (13)$$

$$0 \leq y_j \leq Y_i - (hd_{ij})x_{ij} + Q(1 - x_{ij}) \qquad\qquad \forall i \in F'_{DD}, \forall j \in V_{AD} \qquad (14)$$

$$0 \leq y_i \leq Y_i \leq Q \qquad\qquad \forall i \in F'_{DD} \qquad\qquad (15)$$

$$Y_i = Q \qquad\qquad \forall i \in DD \qquad\qquad (16)$$

$$Y_i - y_i = \sum_{m \in M} \theta_i^m \qquad\qquad \forall i \in F' \qquad\qquad (17)$$

$$0 \leq \theta_i^1 \leq Q a_i \qquad\qquad \forall i \in F' \qquad\qquad (18)$$

$$0 \leq \theta_i^2 \leq Q b_i \qquad\qquad \forall i \in F' \qquad\qquad (19)$$

$$0 \leq \theta_i^3 \leq Q(1 - a_i - b_i) \qquad\qquad \forall i \in F' \qquad\qquad (20)$$

$$a_i, b_i \in \{0,1\} \qquad\qquad \forall i \in F' \qquad\qquad (21)$$

$$x_{ij} \in \{0,1\} \qquad\qquad \forall (i,j) \in A \qquad\qquad (22)$$

The objective function (1) minimizes the total energy cost which consists of three terms. The first term corresponds to the total cost of energy recharged along the route. The second is the total cost of initial charging at the depot. All vehicles are recharged fully using the cheapest (slowest) charger type at the depot overnight. The third is associated with the battery SoC at the end of the trip and deducts the value of the remaining energy from the total cost since that amount of energy has not been consumed en-route. Constraints (2)-(4) are the connectivity constraints which ensure that each customer is visited exactly once and each recharging station may be visited at most once. Constraints (5) and (6) keep track of departures from and arrivals at the depots. Constraints (7) guarantee that all EVs departed from the depot arrive at the depot at the end of their tour. Service times are controlled by constraints (8)-(10). Constraints (11) and (12) observe the load on the vehicle and make sure that total load does not exceed the cargo capacity. Constraints (13) and (14) keep track of battery SoC when departing from customers and stations, respectively. Constraints (15) define the bounds for variables y_i and Y_i while constraints (16) ensure that EVs depart from the depot with full battery. Constraints (17) determine the amount of energy transferred while constraints (18)-(20) control which charger type is utilized for recharging. Note that a_i and b_i cannot be 1 simultaneously because of nonnegativity of θ_i^3 variables. Finally, constraints (21) and (22) define the binary decision variables.

3.2.2 Model 2

The second model is a simple modification of EVRPTW-PR formulation and was first presented in Çatay and Keskin (2017). Here, instead of defining a_i and b_i variables to determine the charger type utilized at the station, we use three copies of each station where each copy represents a different charger type. In other words, each recharging station is equipped with only one charger type but we have three stations at the same location. Thus, the total number of stations increases three-fold. So, F includes all these stations and F' is this set of stations and their copies to allow multiple visits to each station. Let g_i and c_i be the recharging rate and unit energy cost for station $i \in F'$, respectively. g_0 is the recharging rate of the slowest charger and c_0 is the associated unit cost. Then, the mathematical model is formulated as follows:

$$\text{Min} \quad \sum_{i \in F'} c_i (Y_i - y_i) + c_0 \left[Q \sum_{i \in DD} \sum_{j \in V'} x_{ij} - \sum_{i \in AD} y_i \right] \tag{23}$$

s.t.   (2)-(16) and (22)

13

The objective function (23) represents the same energy cost as (1) but with different terms. All constraints in Model 1 remain in Model 2 except constraints (17)-(21) which are associated with charger types.

3.2.3   Evaluation of the Two Models

We compare the efficiency of models 1 and 2 by solving small instances of EVRPTW data set of Schneider et al. (2014). The data set includes three subsets of 12 problems, each involving 5, 10, and 15 customers. We used CPLEX 12.6.2 solver running on a single thread and the time limit is set to 7200 seconds. The experiments were carried on a workstation with Intel Xeon E5 3.30 GHz processor and 64 GB RAM.

Table 3. Comparison of the two models

	Model 2			Model 1			
#Cust	#Opt	#NFS	AvgTime	#Opt	#NFS	AvgTime	#Better
5	12	0	265	12	0	<1	0
10	7	0	3618	12	0	179	0
15	0	1	7200	3	0	5582	6

The results are summarized in Table 3. The column "*#Cust*" gives the number of customers in the problem set. "*#Opt*" refers to the number of optimal solutions found within the time limit whereas "*#NFS*" indicates the number of instances for which no feasible solution could be found. "*AvgTime*" is the average of the run times in seconds for each subset and "*#Better*" in the last column reports the number of instances in which Model 1 gives better solutions than Model 2. Since 5-customer instances are very small, CPLEX found the optimal solutions with both models; however, the solution times were significantly smaller with Model 1. For the 10-customer instances, CPLEX failed to prove the optimality of five problems within the time limit using Model 2 while all problems were solved to optimality with Model 1, again in substantially less time. Finally, for the 15-customer instances, three problems were solved optimally with Model 1 and none with Model 2. Furthermore, Model 2 could not yield a feasible solution in one problem. Overall, Model 1 provided the optimal solutions faster and in many instances, it provided better upper bounds when the time limit is reached. So, we decided to use the results obtained with Model 1 to benchmark our solution methodology that we will describe in the next section.

4. Description of the Matheuristic

For solving EVRPTW-FC, we propose a two-phase matheuristic approach where in the first phase, we attempt to find good heuristic solutions using ALNS and then improve them using an exact method in the second phase. For the exact method, we resort to CPLEX solver but any open-source or commercial solver can be utilized instead. Matheuristics use mathematical models in a heuristic framework and they have been applied to various routing problems. We refer the interested reader to Archetti and Speranza (2014) for the details of the approach and an overview of implementations.

ALNS is a neighborhood search approach which was proposed by Ropke and Pisinger (2006a, 2006b) and has been utilized to solve several types of VRPs (Pisinger and Ropke, 2007; Ribeiro and Laporte, 2012; Demir et al., 2012; Aksen et al., 2014; Grangier et al.,2016; Emeç et al.,2016; Koç et al., 2016) including EVRPs (Goeke and Schneider, 2015; Hiermann et al., 2015; Schiffer and Walther, 2016; Keskin and Çatay, 2016; Wen et al., 2016; Schiffer et al., 2017). It consists of an iterative destroy and repair framework where at each iteration some nodes from the current solution are removed and reinserted by means of several mechanisms in an attempt to improve the incumbent solution. At each iteration, a removal and an insertion mechanism are selected randomly based on the scores they are associated with. These scores are calculated depending on their performance to improve the solution quality. If the selected mechanisms yield good solutions, the probability of their selection in the subsequent iterations is increased by updating their scores. The adaptive nature of the algorithm comes from this update and selection mechanism.

In our matheuristic approach, while ALNS explores the neighborhoods to find promising routes, after every Ω iterations we further enhance the current best solution by optimizing the charging decisions along the tour of an EV by fixing the sequence of customers visited. A similar problem was also solved by Montoya et al. (2017). They name this problem as Fixed-Route Vehicle-Charging Problem (FRVCP) after the Fixed-Route Vehicle-Refueling Problem (FRVCP) introduced by Suzuki (2014). Montoya et al. (2017) solve the EVRP by using a sequence-first split-second approach where they first construct a TSP tour and then split it to extract vehicle routes by ignoring the EV range limit. If any of the resulting routes is energy-infeasible, then they try to repair it by solving FRCVP. This approach cannot be implemented for solving EVRPTW because building vehicle routes without considering the recharge needs/durations of EVs and then trying to

15

insert stations may cause many time-window violations. So, at each iteration of ALNS our repair procedure yields a feasible solution which is further improved by means of mathematical programming using CPLEX.

For most of the ALNS destroy and repair mechanisms we resort to the neighborhoods utilized in Keskin and Çatay (2016). In addition, we propose new removal and insertion methods specific to the fast charging nature of the problem.

4.1 Removal Heuristics

Since the problem has two types of vertices, namely customers and recharging stations, their removal will have different impact on the solution. So, we employ separate customer removal (CR) and recharging station removal (SR) operators for destroying the solution.

4.1.1 Customer Removal

In addition to the well-known CR heuristics widely used in the literature such as Random, Worst-Distance, Worst-Time, Shaw, Proximity-based, Demand-based, Time-based, Zone, Random Route, and Greedy Route removals, we utilize Remove Customer with Preceding Station and Remove Customer with Succeeding Station operators introduced by Keskin and Çatay (2016) where customers are removed along with the station visited immediately before or after serving that customer. At each iteration, one of these CR operators is selected randomly to remove γ customers from the solution and put them in a removal list. The value of γ depends on the total number of customers and is determined randomly between $\underline{n}_c$ and $\bar{n}_c$ using a uniform distribution.

4.1.2 Station Removal

We use Random and Worst-Distance Station removals proposed in Keskin and Çatay (2016). In addition, we propose the following two new SR operators:

Least Used Station Removal: The motivation behind this heuristic is to reduce the cost of visiting recharging stations often. So, we attempt to eliminate unnecessary recharges and satisfy the energy needs of EVs by visiting less number of stations. This can be achieved by utilizing the visited stations to recharge the battery as much as possible instead of recharging small quantities with frequent visits. The operator lists the stations (chargers) in the non-decreasing order of the quantity of energy they charge and removes a pre-determined number of stations from the top of the list.

Expensive Station Removal: Our aim in this removal heuristic is to save from energy cost by eliminating unnecessary recharges using more expensive charging options. The operator lists the stations (chargers) in the non-increasing order of the cost they incur and removes a pre-determined number of stations from the top of the list.

In all the SR operators, σ recharging stations are removed from the solution after every N_{SR} iterations. σ is determined in a similar way as γ based on the number of stations visited in the current solution.

4.2 Insertion Heuristics

As in removal heuristics, different insertion mechanisms are designed for customers and recharging stations. A customer insertion (CI) mechanism is used after every CR operation whereas the station insertion (SI) follows only an SR operation.

4.2.1 Customer Insertion

We use the Greedy, Regret-2, Time-based, and Zone insertions as proposed in Keskin and Çatay (2016). In addition, we employ these mechanisms only with the fastest recharging option when a station insertion is needed to feasibly add a removed customer into a tour. We refer to these new operators as Fast Recharge (FR) Greedy, FR Regret-2, FR Time-based, and FR Zone insertions. Our aim is to shorten the charge durations which may allow serving more customers along the tour and thus reduce the number of vehicles. Eventually, the charge-related decisions will later be optimized in the second phase using a solver as described in Section 4.4.

4.2.2 Station Insertion

We adapt the Greedy and Best Station insertions introduced in Keskin and Çatay (2016) using the cost criterion as follows: when a recharging station is inserted in a route, first, we try the normal charger since it is the cheapest option. If the normal charge is infeasible due to its longer duration, we try fast and super-fast chargers consecutively. This procedure is repeated for all feasible stations and candidate stations are determined along with the charger type. Then, the insertion is performed according to the criteria used in the corresponding SI operator.

4.3 Constructing the Initial Solution

We use three initialization approaches for comparison. The first uses the best-known solutions reported in Keskin and Çatay (2016). Basically, these solutions were obtained by only allowing normal recharge at the stations and can be considered as an upper bound for the fast recharge case.

17

In the second approach, we implement the ALNS of Keskin and Çatay (2016) by allowing the super-fast recharge only and feed its solution to initiate the matheuristic. The last approach randomly puts all customers into the removal list and applies the FR Greedy CI heuristic. Henceforth, we will refer to these initialization approaches as IA 1, IA 2, and IA 3, respectively.

4.4 Route Enhancement

To improve the solution quality, we employ a post-optimization procedure systematically throughout the ALNS process. This procedure uses a commercial solver to optimize the charge-related decisions along each EV route by fixing the sequence of the customers. These decisions include the locations of the stations, selection of the charger type, and the amount of energy transferred.

An easy way to solve this problem is to use Model 1 by reducing the customer set to include only those visited along the route of the vehicle considered and fixing their sequence. However, this formulation will be weak and solving it may require significant computation time particularly when the EV makes frequent stops. To overcome this drawback and speed-up the algorithm, we propose a tighter formulation which also eliminates the need for using copies of the stations. Our approach is similar to the ideas presented in Bruglieri et al. (2016). Let $\bar{V} = \{1, \dots, K\}$ be the set of customers served by the vehicle. 0 and $K + 1$ represent the depot. We define $\bar{V}_0 = \bar{V} \cup \{0\}$ and $\bar{V}_{K+1} = \bar{V} \cup \{K + 1\}$. Let $\theta_{i,i+1}^m$ denote the amount of energy recharged using charger type m if the EV visits a station on its way from customer i and to customer $i + 1$. As in Model 1, binary variables $a_{i,i+1}$ and $b_{i,i+1}$ are used to determine the charger type if the EV is recharged at a station between customers i and $i + 1$. Note that these variables are defined for only consecutive customers and the number of $a_{i,i+1}$ and $b_{i,i+1}$ variables is the same as the number of arcs on the route. The mathematical model is formulated as follows:

$$
\text{Min} \quad \sum_{i \in F} \sum_{m \in M} c^m \theta_{i,i+1}^m + c^1 (Q - y_{K+1}) \tag{24}
$$

$$
\text{s.t.} \quad \tau_i + s_i + \sum_{j \in F}(t_{ij} + t_{jk}) x_{ij} + \sum_{m \in M} g^m \theta_{i,i+1}^m \le \tau_{i+1} \qquad \forall i \in \bar{V}_0 \tag{25}
$$

$$
e_i \le \tau_i \le l_i \qquad \forall i \in \bar{V}_{K+1} \tag{26}
$$

$$y_i - h \left[d_{i,i+1} \left(1 - \sum_{j \in F} x_{ij} \right) + \sum_{j \in F} (d_{ij} + d_{j,i+1}) x_{ij} \right]$$
$$+ \sum_{m \in M} \theta_{i,i+1}^m = y_{i+1} \qquad \forall i \in \bar{V}_0 \qquad (27)$$

$$y_i - h \sum_{j \in F} d_{ij} x_{ij} \geq 0 \qquad \forall i \in \bar{V}_0 \qquad (28)$$

$$\sum_{m \in M} \theta_{i,i+1}^m \leq Q \sum_{j \in F} x_{ij} \qquad \forall i \in \bar{V}_0 \qquad (29)$$

$$\sum_{m \in M} \theta_{i,i+1}^m \leq Q - \left(y_i - h \sum_{j \in F} d_{ij} x_{ij} \right) \qquad \forall i \in \bar{V}_0 \qquad (30)$$

$$0 \leq \theta_{i,i+1}^1 \leq Q a_{i,i+1} \qquad \forall i \in \bar{V}_0 \qquad (31)$$

$$0 \leq \theta_{i,i+1}^2 \leq Q b_{i,i+1} \qquad \forall i \in \bar{V}_0 \qquad (32)$$

$$0 \leq \theta_{i,i+1}^3 \leq Q(1 - a_{i,i+1} - b_{i,i+1}) \qquad \forall i \in \bar{V}_0 \qquad (33)$$

$$y_0 = Q \qquad (34)$$

$$a_{i,i+1}, b_{i,i+1} \in \{0,1\} \qquad \forall i \in \bar{V}_0 \qquad (35)$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in \bar{V}_0, j \in F \qquad (36)$$

The objective function (24) minimizes total energy cost of the route. The first term represents the cost of recharges en-route while the second is the cost of energy, calculated based on the difference of SoCs between departure from and arrival at the depot. Constraints (25)-(26) satisfy time-window feasibility of the customers and the depot. Constraints (27) keep track of the battery SoC: if the EV does not visit a recharging station after leaving customer i, then SoC at arrival at customer $i + 1$ is calculated by subtracting the energy consumed along the arc $(i, i + 1)$ from SoC at customer i. If it visits a station, then the energy recharged at station j is added to the amount described in the previous case and the energy consumed along the arc $(j, i + 1)$ is subtracted. Constraints (28) make sure that if the EV visits a recharging station after customer i, SoC at the arrival at station j is nonnegative. Constraints (29) ensure that the recharging variables $\theta_{i,i+1}^m$ take a positive value only if the vehicle visits a station between customers i and $i + 1$. Constraints (30) guarantee that SoC after the recharge does not exceed the battery capacity Q. Constraints (31)-(33) determine the charger type if the EV is recharged between i and $i + 1$. Constraint (34) makes sure that the vehicle

departs from the depot with a full battery while constraints (35)-(36) define the binary decision variables.
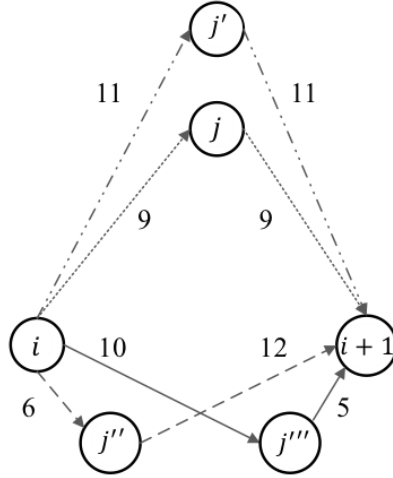
Figure 2. Set of recharging stations between customers i and $i + 1$

We also introduce a pre-processing procedure to reduce the set of the recharging stations as follows: Let i and $i + 1$ be two consecutive customers in the route and $F_{i,i+1}$ be the set of recharging stations that the EV may visit when traveling from customer i to $i + 1$. Initially, $F_{i,i+1} = F$. Then, we make a pairwise comparison of the stations with respect to their distance to customers i and $i + 1$. For instance, consider two stations $j, j' \in F_{i,i+1}$. If $d_{ij'} > d_{ij}$ and $d_{j',i+1} > d_{j,i+1}$ then j is said to dominate j' and j' cannot be visited in the optimal solution since j is closer to both i and $i + 1$. Hence, j' is removed from $F_{i,i+1}$. We repeat this procedure for all station pairs in $F_{i,i+1}$ to reduce its size. The same procedure is applied to all customer pairs $(i, i + 1)$ in the route.

Figure 2 illustrates the case of four recharging stations j, j', j'', and j''', which can be visited between customers i and $i + 1$. It can be easily identified that j' is dominated by j as explained above. However, neither one of j, j'', and j''' dominates the other since both conditions are not satisfied. So, $F_{i,i+1}$ includes j, j'', and j'''.
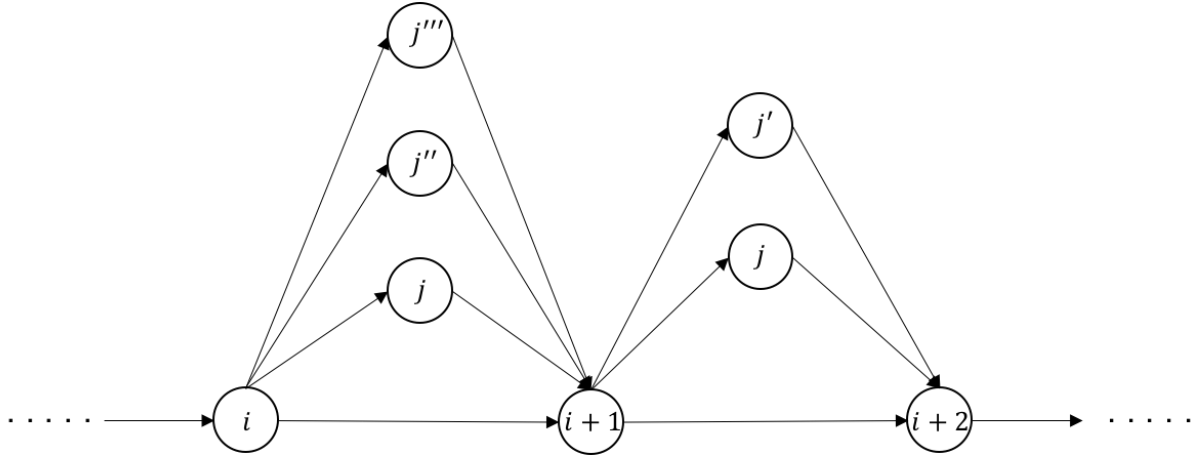
Figure 3. Station insertion between two nodes

In Figure 3, the structure of fixed-route problem is illustrated using a segment of the route consisting of three customer vertices. Since the pre-processing may eliminate some of stations that can be visited between a pair of customers, each arc is associated with a different recharging station set. So, we do not need to create any copies of the stations to allow multiple visits in the mathematical model. This decreases the number of decision variables significantly.

4.5 Reducing the Number of Vehicles

Since the primary objective is to minimize the number of vehicles, we devote some iterations throughout the ALNS to this purpose. After every N_{RR} iterations we dedicate n_{RR} consecutive iterations to remove customers using the Random Route Removal or Greedy Route Removal operators and insert them using FR CI heuristics described in Section 4.2.1. Using only super-fast chargers when recharging is needed decreases the duration of the recharge and allow serving more customers along the route which is not possible otherwise due to time-windows restrictions. This yields longer routes with more frequent visits to customers and thus offers an opportunity to reduce the total number of vehicles in the fleet.

The general structure of the proposed matheuristic is provided in Appendix A.

5. Experimental Design and Numerical Results

We test the performance of the proposed matheuristic on the EVRPTW data sets that Schneider et al. (2014) generated and on the GVRP-MTPR data set of Felipe et al. (2014). The larger instances of Schneider et al. (2014) include 100 customers and 21 recharging stations. The vertices in the data are randomly distributed, clustered, and both clustered and randomly distributed over a 100×100 grid and classified as r, c, and rc data set, respectively. Each of these data sets has also two subsets, type 1 and type 2, which differ by the length of the time windows, vehicle load, and battery capacities. The small instances were generated by randomly selecting a subset of customers and stations from the large instances.

The data set of Felipe et al. (2014) is referred to as FORT instances and consists of two different configurations involving five and nine stations. Each configuration includes three sets of ten instances with 100, 200 and 400 customers distributed randomly. In total, the data set includes 60 instances.

In ALNS, we used the same parameter values as reported in Keskin and Çatay (2016) (see Appendix B). For the recharging speed and cost of different chargers, we used the values given in Felipe et al. (2014). In the optimization phase, we employed CPLEX 12.6.2 with its default setting using single thread. The matheuristic was implemented in Java programming language and the experiments are conducted on the same workstation described in Section 3.2.3.

5.1 Results for Large Instances

We first investigate how different initialization approaches and optimization frequencies affect the solution quality in order to determine the best configuration. Next, we examine the benefits of utilizing fast chargers in terms of fleet size and energy costs.

5.1.1 Analysis of Different Configurations

We optimize the recharging decisions of the best solution of that round every Ω iterations, which we refer to as CPLEX call frequency. On the one hand, choosing this number too small may increase the run time. On the other hand, choosing it too large may deteriorate the solution quality. Our preliminary experiments revealed that calling CPLEX after 200 and 500 iterations shows a good compromise. So, we decided to consider these values for further investigation.

Table 4. Comparison of results obtained with different configurations

| | IA 1 | | | | IA 2 | | | | IA 3 | | | |
| | $\Omega = 200$ | | $\Omega = 500$ | | $\Omega = 200$ | | $\Omega = 500$ | | $\Omega = 200$ | | $\Omega = 500$ | |
Instance	#Veh	TC	#Veh	TC	#Veh	TC	#Veh	TC	#Veh	TC	#Veh	TC
c101	**12**	**1043.38**	**12.00**	**1043.38**	**12.00**	**1043.38**	**12.00**	**1043.38**	12.00	1044.51	12.00	1044.51
c102	10	1078.12	10.00	1096.73	**10.00**	**1009.86**	**10.00**	**1009.86**	10.00	1080.95	10.00	1055.41
c103	10	962.78	10.00	962.78	**10.00**	**961.78**	**10.00**	**961.78**	10.00	992.25	10.00	997.27
c104	9	1113.85	10.00	878.81	**9.00**	**1006.12**	**9.00**	**1006.12**	10.00	893.66	10.00	900.29
c105	10	1102.47	10.00	1124.58	**10.00**	**1031.58**	**10.00**	**1031.58**	10.00	1141.21	10.00	1166.01
c106	10	1141.19	10.00	1082.12	**10.00**	**1044.96**	**10.00**	**1044.96**	10.00	1083.49	10.00	1166.84
c107	10	1017.80	10.00	1022.24	**10.00**	**1015.81**	**10.00**	**1015.81**	10.00	1146.72	10.00	1048.87
c108	10	1025.15	10.00	1025.15	**10.00**	**1022.36**	**10.00**	**1022.36**	10.00	1191.61	10.00	1074.91
c109	**10**	**940.38**	**10.00**	**940.38**	10.00	959.66	10.00	959.66	10.00	1126.70	10.00	1057.56
c201	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**
c202	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	4	641.13
c203	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	4	638.17	4	638.17
c204	3	746.75	3	787.77	**3**	**719.89**	3	720	3	741.99	3	846.81
c205	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**
c206	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**
c207	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	4	638.17	**4**	**629.95**
c208	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	**4**	**629.95**	4	630.06
r101	17	1772.39	17	1829.99	**16**	**1815.77**	17	1642.9	17	1845.09	17	1817.59
r102	15	1583.3	15	1561.71	**15**	**1436.95**	**15**	**1436.95**	15	1572.01	15	1585.49
r103	13	1256.95	12	1395	**12**	**1233.72**	**12**	**1233.72**	12	1423.86	12	1472.09
r104	11	1076.25	11	1076.25	**10**	**1048.69**	10	1048.69	11	1210.75	11	1261.95
r105	13	1515.67	13	1547.43	**13**	**1364.98**	13	1366.89	14	1549.83	14	1572.14
r106	12	1446.52	13	1300.36	**12**	**1280.16**	**12**	**1280.16**	12	1386.04	12	1435.11
r107	11	1159.07	11	1184.88	**10**	**1130.86**	**10**	**1130.86**	11	1261.46	11	1234.65
r108	10	1213.32	11	1030.48	**10**	**1010.68**	**10**	**1010.68**	10	1223.7	10	1168.6
r109	12	1359.11	12	1310.09	**11**	**1198.55**	**11**	**1198.55**	12	1433.76	12	1389.78
r110	11	1089.9	11	1089.9	**10**	**1127.7**	**10**	**1127.7**	11	1206.9	11	1336.29
r111	11	1180.33	11	1192.5	**11**	**1086.1**	**11**	**1086.1**	11	1325.5	11	1276.08
r112	11	1016.3	11	1016.3	**10**	**1015.49**	**10**	**1015.49**	11	1181.79	11	1168.92
r201	3	1262.06	3	1262.06	3	1295.95	**3**	**1257.50**	3	1576.58	3	1378.97
r202	**3**	**1051.46**	**3**	**1051.46**	3	1060.18	3	1060.18	3	1068.4	3	1097.62
r203	**3**	**895.54**	**3**	**895.54**	3	898.96	3	898.96	3	932.9	3	923.22
r204	**2**	**779.71**	2	780.13	2	785.7	2	785.7	2	816.47	3	728.46
r205	**3**	**987.36**	**3**	**987.36**	3	1001.85	3	1006.86	3	1020.8	3	1047.23
r206	**3**	**922.7**	**3**	**922.7**	3	928.56	3	928.56	3	956.58	3	974.17
r207	**2**	**846.43**	**2**	**846.43**	2	857.07	2	859.3	3	825.55	3	829.16
r208	**2**	**736.13**	**2**	**736.13**	2	737.43	2	739.64	2	739.15	2	747.66
r209	**3**	**866.67**	**3**	**866.67**	3	900.77	3	900.77	3	931.26	3	892.07
r210	**3**	**843.21**	**3**	**843.21**	3	856.76	3	859.13	3	879.16	3	871.9
r211	2	862.56	2	862.56	**2**	**840.61**	2	857.74	3	783.63	3	797.63
rc101	15	1744.85	15	1744.85	**14**	**1800.73**	15	1640.57	15	1827.93	15	1810.49
rc102	14	1526.31	14	1526.27	**13**	**1557.39**	**13**	**1557.39**	14	1657.06	13	1645.16
rc103	12	1389.5	12	1444.88	**12**	**1355.71**	**12**	**1355.71**	12	1497.5	12	1552.26
rc104	11	1201.04	11	1200.24	**10**	**1192.6**	10	1193.86	11	1257.48	11	1389.82
rc105	14	1449.53	13	1587.57	**13**	**1425.92**	**13**	**1425.92**	13	1488.58	13	1590.97
rc106	13	1402.95	13	1398.85	**12**	**1388.85**	**12**	**1388.85**	13	1541.1	13	1490.22
rc107	11	1294.2	11	1300.44	11	1249.03	**11**	**1247.87**	11	1366.75	11	1440.09
rc108	11	1182.84	11	1182.84	**10**	**1199.24**	**10**	**1199.24**	11	1302.43	11	1262.57
rc201	**4**	**1446.84**	**4**	**1446.84**	4	1485.23	4	1485.23	4	1482.53	4	1501.32
rc202	**3**	**1416.96**	**3**	**1416.96**	3	1426.88	3	1424.86	4	1287.24	4	1313.06
rc203	**3**	**1064.33**	**3**	**1064.33**	3	1081.57	3	1081.57	3	1144.73	3	1114.98
rc204	3	886.23	**3**	**886.19**	3	895.18	3	895.18	3	898.97	3	894.21
rc205	3	1257.92	3	1257.92	**3**	**1256.30**	**3**	**1256.30**	3	1341.91	3	1451.59
rc206	**3**	**1206.06**	**3**	**1206.06**	3	1229.67	3	1229.67	3	1223.51	3	1296.31
rc207	3	992.14	3	992.14	**3**	**991.65**	**3**	**991.65**	3	1073.96	3	1090.87
rc208	**3**	**839.71**	**3**	**839.71**	3	884.76	3	885.76	3	886.36	3	944.14
#Best		23		23		38		33		5		4

23

Our stopping criterion is a limit on the number of iterations. For different initialization algorithms, we set different limits. We perform 25,000 iterations of ALNS when we utilize IA 1 and IA 2 to generate the initial solution. Then, we apply the matheuristic for 10,000 iterations. When we utilize IA 3 for initialization, the solution is constructed very fast by the greedy algorithm whereas the matheuristic is performed for 25,000 iterations. In other words, we allow a more intensive search during the initial solution generation in the former case whereas in the latter case matheuristic is the only actor and it benefits from the mathematical programming more rigorously.

For each configuration, we performed 30 runs for each instance and reported the best results in Table 4. "*#Veh*" and "*TC*" represent the number of vehicles needed and total cost of energy, respectively. The best solutions among six different configurations are indicated in bold. The row "*#Best*" shows the total number of instances for which the corresponding configuration yielded the best solution.

We observe that the initialization approaches have a significant effect on the performance of the matheuristic. While IA 2 yields better solutions in type-1 instances where customers have narrow time windows, IA 1 performs better in type-2 instances which involve customers with wide time windows. In other words, determining the initial solution through ALNS by considering only normal chargers works better in type-2 problems whereas using the same initialization approach with super-fast chargers has a better performance in type-1 problems. Moreover, we also see that the superiority of IA 2 in type-1 problems is usually in terms of the number of vehicles while in type-2 problems IA 1 performs slightly better than IA 2 in total cost. The former is an expected outcome as the utilization of super-fast chargers may significantly cut down the recharge time at stations and allow the EV serve more customers along its route which will translate into a reduction in fleet size. However, the latter can be considered as a surprising result and we will further elaborate on this issue in the next section.

Table 5. Average run times of different configurations (in minutes)

Data set	IA 1		IA 2		IA 3	
	$\Omega = 200$	$\Omega = 500$	$\Omega = 200$	$\Omega = 500$	$\Omega = 200$	$\Omega = 500$
c1	4.89	4.85	2.20	2.18	5.49	4.60
c2	38.31	38.70	22.27	22.63	46.58	45.13
r1	3.73	3.67	1.26	1.23	2.89	2.83
r2	74.33	72.39	31.19	33.15	64.10	68.60
rc1	3.27	3.21	1.07	1.08	2.51	2.75
rc2	33.56	33.69	12.62	12.65	37.03	37.22
Average	26.35	26.08	11.77	12.15	26.43	26.85

When we examine the role of CPLEX call frequency on the solution quality we do not observe any substantial difference, yet $\Omega = 200$ performs slightly better than $\Omega = 500$, which is not surprising and supports the role of optimization in achieving higher quality solutions. Independent of the value of Ω, the results obtained by using IA 3 are inferior than those given by the other two initialization approaches. In other words, searching for a good initial solution pays back the effort spent.

In Table 5, we report the average run time of each problem subset to evaluate the computational effort required by different configurations. The numbers are in minutes. The results indicate that type-2 problems require more time than type-1 problems. This is well expected and in parallel with many studies that utilized Solomon (1987) data because wide time windows expand the search space bringing more feasible insertions to be evaluated and long planning horizon allows an EV make longer trips visiting more customers (more than 50 in certain instances) requiring more recharges. On the other hand, we observe that the run time of the implementation with IA 2 is significantly smaller than those with IA 1 and IA 3. Although IA 1 and IA 2 perform the same number of iterations using the same ALNS mechanisms, the station insertion procedure requires less computational effort in IA 2. This is due to the fact that the search for a feasible station for insertion takes more time when only normal chargers are available because of their longer recharging durations.
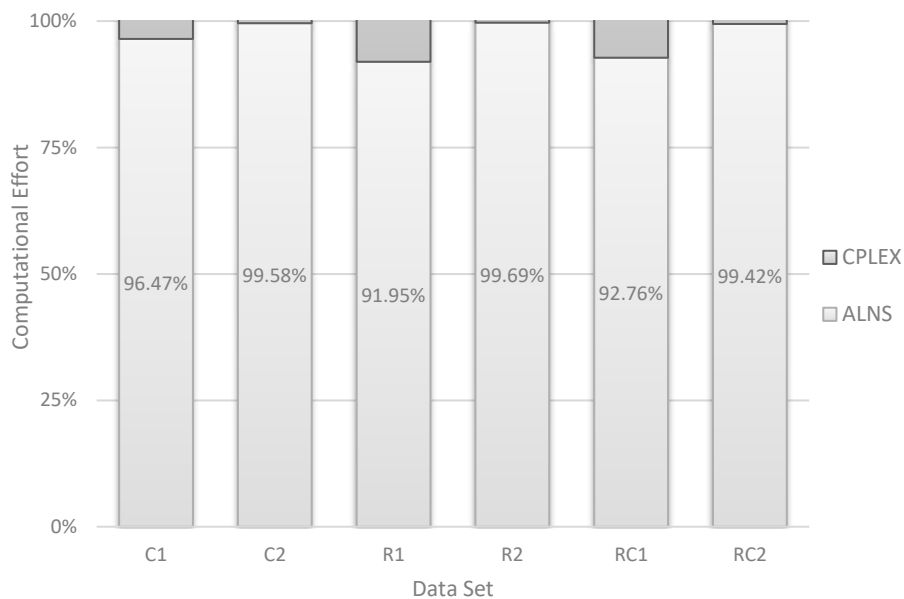

Figure 4. Percentage of computational effort required by ALNS vs. CPLEX

Table 6. Comparison of results obtained by multiple fast charging vs. single normal charging

Instance	1-Charger					3-Chargers								
	Q	#Veh	TC/TD	#L1	Qty	#Veh	TC	TD	#L1	Qty	#L2	Qty	#L3	Qty
c101	79.69	12	1043.38	8	165.96	12	1043.38	1043.38	8	165.96	0	0.00	0	0.00
c102	79.69	11	1032.49	10	202.56	**10**	998.48	1009.86	6	143.66	3	70.67	1	21.57
c103	79.69	10	973.39	9	229.52	10	946.79	**961.78**	5	98.76	2	87.40	2	31.25
c104	79.69	10	886.72	7	140.48	**9**	984.82	1006.12	7	93.07	5	136.11	1	38.43
c105	79.69	11	1037.78	9	206.52	**10**	1024.01	**1031.58**	7	172.50	3	75.72	0	0.00
c106	79.69	11	1024.18	9	212.81	**10**	1028.89	1044.96	6	125.67	3	104.42	1	28.11
c107	79.69	10	1058.11	11	273.32	10	1005.84	**1015.81**	7	138.76	3	99.71	0	0.00
c108	79.69	10	1033.50	10	252.33	10	1014.21	**1022.36**	8	174.26	3	81.54	0	0.00
c109	79.69	10	946.84	9	170.88	10	940.38	**940.38**	9	164.42	0	0.00	0	0.00
c201	118.31	4	629.95	3	168.26	4	629.95	629.95	3	168.26	0	0.00	0	0.00
c202	118.31	4	629.95	3	168.26	4	629.95	629.95	3	168.26	0	0.00	0	0.00
c203	118.23	4	629.95	3	168.50	4	629.95	629.95	3	168.50	0	0.00	0	0.00
c204	118.12	4	629.95	3	168.83	**3**	697.22	719.89	3	157.52	3	143.98	1	41.36
c205	117.78	4	629.95	3	169.85	4	629.95	629.95	3	169.85	0	0.00	0	0.00
c206	117.70	4	629.95	3	170.09	4	629.95	629.95	3	170.09	0	0.00	0	0.00
c207	117.66	4	629.95	3	170.21	4	629.95	629.95	3	170.21	0	0.00	0	0.00
c208	117.66	4	629.95	3	170.21	4	629.95	629.95	3	170.21	0	0.00	0	0.00
r101	62.14	18	1641.42	22	533.93	**16**	1788.25	1815.77	23	518.81	12	275.20	0	0.00
r102	62.14	16	1461.38	24	469.22	**15**	1422.69	**1436.95**	19	353.90	6	142.64	0	0.00
r103	62.14	13	1262.75	17	457.31	**12**	1195.76	**1233.72**	8	136.87	10	254.93	2	62.32
r104	67.15	11	1078.99	12	343.29	**10**	1015.81	**1048.69**	3	102.62	7	154.61	3	87.09
r105	62.14	15	1373.94	18	454.85	**13**	1338.52	**1364.98**	15	278.83	9	264.56	0	0.00
r106	62.60	13	1310.46	18	496.66	**12**	1246.47	**1280.16**	10	186.17	14	336.85	0	0.00
r107	66.28	12	1118.91	14	326.82	**10**	1096.09	1130.86	6	90.64	12	337.60	1	5.06
r108	64.06	11	1031.14	13	331.25	**10**	987.04	**1010.68**	7	135.59	5	192.14	2	22.16
r109	65.17	13	1197.57	15	360.49	**11**	1156.36	1198.55	5	82.61	9	306.66	2	57.64
r110	67.12	11	1090.92	17	352.61	**10**	1078.66	1127.70	4	77.84	7	168.81	4	160.81
r111	65.80	12	1084.13	13	304.56	**11**	1064.34	1086.10	8	153.74	5	155.99	1	30.81
r112	65.48	11	1017.31	14	297.03	**10**	986.79	**1015.49**	6	82.95	8	211.04	1	37.99
r201	187.86	3	1262.10	6	698.52	3	1257.50	**1257.50**	7	693.92	0	0.00	0	0.00
r202	238.34	3	1052.32	3	337.30	3	1051.46	**1051.46**	3	336.44	0	0.00	0	0.00
r203	187.90	3	895.54	4	331.84	3	895.54	895.54	4	331.84	0	0.00	0	0.00
r204	247.66	2	780.14	2	284.82	2	779.71	**779.71**	2	284.39	0	0.00	0	0.00
r205	198.88	3	987.36	3	390.72	3	987.36	987.36	3	390.72	0	0.00	0	0.00
r206	181.23	3	922.70	3	379.01	3	922.70	922.70	3	379.01	0	0.00	0	0.00
r207	267.18	2	846.59	2	312.23	2	846.43	**846.43**	2	312.07	0	0.00	0	0.00
r208	218.03	2	736.12	2	300.06	2	736.12	736.12	2	300.06	0	0.00	0	0.00
r209	181.83	3	868.95	4	323.46	3	866.67	**866.67**	4	321.18	0	0.00	0	0.00
r210	187.87	3	843.36	3	299.98	3	843.21	**843.21**	3	299.84	0	0.00	0	0.00
r211	265.71	2	862.56	2	331.14	2	840.61	**840.61**	4	309.19	0	0.00	0	0.00
rc101	79.69	15	1754.75	22	573.65	**14**	1769.82	1800.73	17	403.60	7	191.95	2	58.61
rc102	79.69	14	1526.31	17	455.00	**13**	1531.90	1557.39	12	269.33	8	213.47	1	20.68
rc103	79.69	13	1329.58	13	366.84	**12**	1332.38	1355.71	12	222.11	4	129.22	1	52.05
rc104	79.69	11	1202.93	14	326.34	**10**	1165.39	**1192.60**	7	121.53	8	221.88	1	25.08
rc105	79.69	14	1449.53	17	356.63	**13**	1403.53	**1425.92**	10	156.40	7	223.88	0	0.00
rc106	79.69	13	1402.95	16	372.68	**12**	1369.51	**1388.85**	10	236.05	8	193.48	0	0.00
rc107	79.69	12	1261.03	14	314.68	**11**	1221.72	**1247.87**	8	161.81	6	149.77	1	55.85
rc108	79.69	11	1164.32	13	289.47	**10**	1171.71	1199.24	7	156.53	5	161.34	1	56.95
rc201	211.04	4	1446.84	4	602.68	4	1446.84	1446.84	4	602.68	0	0.00	0	0.00
rc202	273.13	3	1416.96	4	597.57	3	1416.96	1416.96	4	597.57	0	0.00	0	0.00
rc203	209.92	3	1069.27	5	439.51	3	1064.33	**1064.33**	7	434.57	0	0.00	0	0.00
rc204	159.68	3	886.23	6	407.19	3	886.19	**886.19**	5	407.15	0	0.00	0	0.00
rc205	194.58	3	1262.22	8	678.48	3	1255.15	**1256.30**	8	659.89	1	11.51	0	0.00
rc206	229.26	3	1206.09	6	518.31	3	1206.06	**1206.06**	5	518.28	0	0.00	0	0.00
rc207	212.23	3	992.14	4	355.45	3	991.65	**991.65**	5	354.96	0	0.00	0	0.00
rc208	165.63	3	839.71	4	342.82	3	839.71	839.71	4	342.82	0	0.00	0	0.00

Regarding CPLEX call frequency, we do not see any major difference between the run times with $\Omega = 200$ and $\Omega = 500$ in any setting. This indicates that CPLEX does not require extensive computational effort to find the optimal solution for the fixed-route problem and validates the effectiveness of the proposed mathematical formulation. To further investigate the computational burden of route enhancement with CPLEX we illustrate the percentage of total computational time spent by ALNS and CPLEX in Figure 4. This figure reveals that CPLEX does not require more than 1% of the total run time in type-2 problems whereas in type-1 problems the optimization can take up to 8% of total time. Although route enhancement seems to require relatively more effort in type-1 problems, the total computation time for type-2 problems is substantially higher (see Table 4) and the time devoted to route enhancement corresponds to a small proportion within this large amount of time.

In summary, we can conclude that our matheuristic using IA 2 for initialization and performing route enhancement every 200 iterations (i.e. $\Omega = 200$) exhibits the best performance in terms of both solution quality and run time.

5.1.2 Effect of Multiple Chargers

In this section, we compare the best results obtained by the proposed matheuristic with the best results that ALNS of Keskin and Çatay (2016) yields for the single charge case where the stations are equipped with only Level 1 (normal) chargers. The results are presented in Table 6. In this table, "Q" stands for the battery capacity and "TC/TD" refers to the total cost which is equivalent to total distance travelled in the normal charge case. Columns "#$L1$", "#$L2$", and "#$L3$" report the total number of recharges performed by using Level 1, Level 2, and Level 3 chargers, respectively, and "Qty" shows the corresponding quantity of energy transferred. The improvements over the single charger results are highlighted in bold.

When we compare the 3-charger results with 1-charger results we see that fast charging is more beneficial when the customers have narrow time windows (i.e. type-1 problems). This is expected because the time spent at stations for recharging the EV can be reduced significantly with fast chargers and the vehicle may be able to serve additional customers along its route including customers that cannot be served otherwise due to strict time-window restrictions. If the EV can serve more customers along its route then more efficient solutions may be constructed which require less number of vehicles and/or consume less energy due to shortened travel distance. The cost of total energy may also go down depending on the charger types utilized and the quantity of energy transferred. In type-1 data set, out of 29 instances we have achieved better solutions in 28 whereas the solution for one instance (c101) has not changed. In addition, the number of vehicles is decreased by two in 8 instances and by one in 20. The

fleet size is reduced in all r- and rc-type problems. Furthermore, both the fleet size and energy cost are improved in 13 type-1 instances. We can conclude that the improvements in type-1 problems were accomplished by utilizing fast and super-fast chargers effectively based on the number of recharges and the energy quantities given in the last four columns.

When we analyze the results for type-2 instances we observe that fast charging is able to reduce the fleet size in only one instance (c204) out of 27 and by only one vehicle. Since the time windows can be easily satisfied in these problems the vehicles can serve more customers on their routes and the number of EVs is already few (between 2 and 4). Hence, a reduction in the fleet size is usually impossible. We also observe that fast charging does not help cutting down the costs either: total energy cost is reduced in 12 instances and the average improvement is only 0.17%. The main contributor to this average value is problem r211 where a 2.54% reduction in energy cost is achieved. These results are in parallel with Desaulniers et al. (2016) who highlighted the minor influence of wide time-window constraints on recharging decisions. Further investigation on these results reveals that #L2 and #L3 values are 0 in most of the instances. So, these slight improvements were achieved by an extended search of the solution space of single charger case rather than by using fast or super-fast chargers. Noting again that IA 1 constructs the initial solution by using only normal chargers, these results now explain why the matheuristic using IA 1 showed a better performance in type-2 problems in Section 5.1.1.

5.2 Results for Small Instances

In this section, we solve the small instances to compare the solutions of the matheuristic with the optimal solutions or best bounds given by CPLEX. With CPLEX, we solve Model 1 using a single thread and set the time limit to 7200 seconds. Since the problems are smaller we perform IA 2 for 10,000 iterations and run the matheuristic 10,000 iterations with $\Omega = 200$. The results are presented in Table 7. The computation times reported in columns "*Time*" are in seconds. The values given following "c" and "s" in the instance names represent the number of customers and stations, respectively. Note that CPLEX results reported with a run time of 7200 seconds show the best upper bounds found within the given time limit and are not necessarily the optimal solutions. CPLEX solves all 5- and 10-customer instances to optimality. Our matheuristic also solves them optimally; however, it requires more computational time in most of the instances. On the other hand, the matheuristic outperforms CPLEX in 15-customer instances both in terms of solution quality and run time. The improved results are highlighted in bold in the table. We see that our matheuristic achieved better cost figures in two instances (c103c15-s5 and rc204c15-s7) and provided a solution with one less vehicle in another instance

(rc202c15-s5). The latter case corresponds to a major improvement as the fleet size is reduced to a single vehicle from two. We believe that these results show the effectiveness of the proposed matheuristic approach.

Table 7. Comparison of results on small size instances

	CPLEX			Matheuristic		
Instance	#Veh	TC	Time	#Veh	TC	Time
c101c5-s3	2	250.69	< 1	2	250.69	2.50
c103c5-s2	1	175.37	< 1	1	175.37	2.76
c206c5-s4	1	242.56	< 1	1	242.56	3.02
c208c5-s3	1	164.34	< 1	1	164.34	2.62
r104c5-s3	2	136.69	< 1	2	136.69	0.98
r105c5-s3	2	156.08	< 1	2	156.08	2.09
r202c5-s3	1	128.88	< 1	1	128.88	3.08
r203c5-s4	1	179.06	< 1	1	179.06	3.43
rc105c5-s4	2	233.77	< 1	2	233.77	1.67
rc108c5-s4	2	253.93	< 1	2	253.93	2.44
rc204c5-s4	1	185.16	< 1	1	185.16	3.17
rc208c5-s3	1	167.98	< 1	1	167.98	3.23
c101c10-s5	3	382.93	1	3	382.93	4.34
c104c10-s4	1	267.60	27	1	267.60	8.48
c202c10-s5	1	304.06	1	1	304.06	7.99
c205c10-s3	1	283.29	3	1	283.29	5.91
r102c10-s4	3	249.19	1	3	249.19	3.78
r103c10-s3	2	206.30	57	2	206.30	5.22
r201c10-s4	1	241.25	1159	1	241.25	7.76
r203c10-s5	1	222.64	14	1	222.64	26.90
rc102c10-s4	4	415.99	12	4	415.99	3.13
rc108c10-s4	3	347.90	2	3	347.90	4.27
rc201c10-s4	1	412.86	865	1	412.86	6.50
rc205c10-s4	2	325.98	< 1	2	325.98	6.86
c103c15-s5	2	368.91	7200	2	**368.80**	12.47
c106c15-s3	2	310.79	1375	2	310.79	10.04
c202c15-s5	2	381.23	453	2	381.23	19.29
c208c15-s4	1	339.21	7200	1	339.21	21.09
r102c15-s8	5	411.03	7200	5	411.03	5.73
r105c15-s6	3	340.62	353	3	340.62	5.63
r202c15-s6	1	449.81	7200	1	449.81	30.08
r209c15-s5	1	313.24	7200	1	313.24	44.49
rc103c15-s5	4	397.67	7200	4	397.67	5.91
rc108c15-s5	3	370.25	7200	3	370.25	6.87
rc202c15-s5	2	394.39	7200	**1**	648.05	15.95
rc204c15-s7	1	392.76	7200	**1**	**340.25**	62.45

5.3 Results for FORT instances of Felipe et al. (2014)

To the best of our knowledge, Felipe et al. (2014) is the only study that addressed EVRP with multiple charger types and partial recharges but without considering time windows and using a different objective function which minimizes total cost of energy and battery degradation. To further evaluate the performance of our method, we adapt it to solve their case and perform five runs for each instance of FORT data. We implement IA2 with Ω=200 and run the algorithm with two different configurations: (A) 25,000 iterations of ALNS for initialization and 10,000 iterations of matheuristic; (B) 2500 iterations of ALNS and 1000 iterations of matheuristic. Our aim in performing test (B) is to investigate the trade-off between the solution quality and computation time, and to make a fair comparison with the heuristic of Felipe et al. (2014) in terms of run time. Note that they coded their algorithm in Fortran 95 and executed on an Intel Core i5 2.8 GHz processor and 8 GB RAM.

Table 8. Comparison of average results with Felipe et al. (2014) on FORT instances[*]

N	S	FORT		Matheuristic (A)			Matheuristic (B)		
		Avg TC	Avg Time	Avg TC	Avg Time	% Imp	Avg TC	Avg Time	% Imp
100	9	71.19	274	64.66	181	9.17	65.01	20	8.73
	5	71.59	268	65.12	180	9.02	65.24	31	8.85
200	9	110.38	533	98.75	798	11.19	101.81	80	8.62
	5	114.36	522	101.27	770	11.44	104.02	122	9.03
400	9	195.75	1181	176.61	2936	9.84	182.43	329	6.84
	5	203.18	1101	181.61	3247	10.58	188.18	533	7.33
Average			647		1352	10.21		186	8.23

The average results are presented in Table 8 and detailed results are given in Appendix C. In this table, "*N*" and "*S*" refer to the number of customers and number of stations in the data, respectively. "*Avg TC*" and "*Avg Time*" report the average total cost and the average computation time (in seconds) of the corresponding problem set. "*% Imp*" shows the percentage improvement achieved by our matheuristic for each configuration and calculated as (FORT − Matheuristic)/FORT.

The results reveal that our matheuristic provides significantly better solutions than FORT using both configurations (A) and (B). Reducing the number of iterations in configuration (B) deteriorates the solution quality by 2% on the average while the average run time is approximately ⅛ of that of configuration (A). Felipe et al. (2014) reported an average run time

---

[*] The cost figures are kindly provided by Gregorio Tirado. The average computation times correspond to the average running times of SA approach reported in Felipe et al. (2014).

of 647 seconds whereas our matheuristic (A) and (B) spent on the average 1352 and 186 seconds, respectively. Even though the problem addressed is slightly different and the two methods were executed on different processors, we believe that these results show the superiority of our matheuristic and confirm its effectiveness.

6. Conclusions

In this paper, we tackled the Electric Vehicle Routing Problem with Time Windows and Fast Chargers (EVRPTW-FC). In EVRPTW-FC, the stations are equipped with multiple chargers which vary in power supply, power voltage, and maximum current options. We considered three charger types, namely normal, fast, and super-fast. We formulated two different mathematical models of this problem and compared them in terms of solution quality and computational time. Since the medium and large size problems are intractable, we developed a matheuristic approach to solve the problem efficiently. Our approach combines ALNS with an exact method. In ALNS, while we employed destruction and repair algorithms from the literature, we also introduced new mechanisms specific to the nature of the problem. In the exact method, we fixed the sequence of the customers visited by each vehicle in the solution provided by ALNS and utilized CPLEX solver to optimize the charging related decisions. We also developed an efficient mathematical formulation for this fixed-route single-vehicle problem to be able to find the optimal solution in reasonable run time.

We tested the performance of our algorithm on both small and large benchmark instances from the literature. Our numerical results in small-size instances showed that our matheuristic outperformed CPLEX both in solution quality and run time. In large-size instances, the results revealed the advantage of using fast charging in terms of fleet size and energy consumption. Specifically, we were able to obtain route plans requiring less EVs or reducing energy cost or both in all instances where the time windows are narrow. On the other hand, the influence of the availability of fast chargers was minor when the time windows are wide.

In this study, we assumed that all stations were already located and equipped with all types of chargers. However, this may not be the case considering the high installation costs and lack of infrastructure. So, the problem can be extended to a location routing problem where the recharging stations are sited, their charger equipment and capacities are determined, and the EVs are routed simultaneously. Further research on this topic may also address the heterogeneous fleet case where the vehicles vary by their cargo capacities, battery condition and age which affect their cruising range and discharge/recharge durations. Furthermore, we assumed that recharging stations and chargers were always available. In real life, there may be

queues in the stations and the EVs may need to wait for service. Alternatively, it may drive to another station. So, variability in recharging times can be investigated within the stochastic context. The authors are currently working on this extension.

Acknowledgments

Appendix A. ALNS Parameters

In Table A.1, we provide the list of the ALNS parameters and the values we used throughout the experimental study.

Table A.1. ALNS parameters and their values

Par.	Description	Value
σ_1	Score of the best solution	25
σ_2	Score of the better solution	20
σ_3	Score of the worse solution	21
ρ	Roulette wheel parameter	0.25
ϕ_1	First Shaw parameter	0.5
ϕ_2	Second Shaw parameter	0.55
ϕ_3	Third Shaw parameter	0.53
ϕ_4	Fourth Shaw parameter	0.25
ε	Cooling rate of SA	0.9994
μ	Initial temperature control parameter of SA	0.4
κ	Worst removal determinism factor	4
η	Shaw removal determinism factor	12
n_Z	Number of zones in zone removal	25
$\underline{n}_c$	Lower limit for the number of customers to be removed	$min\{0.1|N|, 30\}$
$\overline{n}_c$	Upper limit for the number of customers to be removed	$min\{0,4|N|, 60\}$
N_C	Number of iterations between two consecutive updates of CR and CI operators	200
N_S	Number of iterations between two consecutive updates of SR and SI operators	5500
N_{SR}	Number of iterations between two consecutive SR and SI calls	60
N_{RR}	Number of iterations between two consecutive sections in which only route removal operators are performed	2000
m_r	Route removal upper bound	0.3
n_{RR}	Number of consecutive iterations during the sections in which only route removal operators are performed	1250
Ω	Number of iterations between two consecutive route enhancement procedures	200

Appendix B. Outline of the Matheuristic

In Algorithm 1, we provide the pseudocode of the proposed methodology.

Algorithm 1: Pseudocode of the proposed matheuristic

1:	Generate an initial solution, current solution ← initial solution
2:	Initialize the scores and probabilities of the operators, $iter \leftarrow 1$
3:	**while** *termination criterion is not met* **do**
4:	**if** *iter is a multiple of N_{SR}* **then**
5:	Select an SR operator and remove stations
6:	Select an SI operator and repair solution
7:	**else if** *iter is a multiple of N_{RR}* **then**
8:	**for** n_{RR} iterations **do**
9:	Select RRR or GRR operator and remove customers
10:	Select a CI operator and repair solution
11:	**if** *iter is a multiple of N_{SR}* **then**
12:	Select an SR operator and remove stations
13:	Select an SI operator and repair solution
14:	**end if**
15:	**if** *iter is a multiple of Ω* **then** Apply Route Enhancement to current best solution
16:	**end for**
17:	**else**
18:	Select a CR operator and remove customers
19:	**if** *the destroyed solution is infeasible* **then** Perform Greedy Station Insertion
20:	Select a CI operator and repair solution
21:	**end if**
22:	Accept/reject the solution using Simulated Annealing criterion
23:	$iter \leftarrow iter + 1$
24:	**if** *iter is a multiple of Ω* **then** Apply Route Enhancement to current best solution
25:	**if** *iter is a multiple of N_C* **then** Update adaptive weights of CR and CI operators
26:	**if** *iter is a multiple of N_S* **then** Update adaptive weights of SR and SI operators
27:	**end while**

SR: Recharging Station Removal	CR: Customer Removal	RRR: Random Route Removal
SI: Recharging Station Insertion	CI: Customer Insertion	GRR: Greedy Route Removal

Appendix C. Comparison with Felipe et al. (2014)

The detailed results of the experiments for the problem of Felipe et al. (2014) using FORT instances are presented in Tables C.1-C.3. The results were obtained by using IA2 with $\Omega=200$ and by implementing two configurations: (A) 25,000 iterations of ALNS for initialization and 10,000 iterations of matheuristic; (B) 2500 iterations of ALNS and 1000 iterations of matheuristic. "*S*" shows the number of stations in the problem, "*TC*" and "*Time*" refer to the total cost and computation time (in seconds), respectively, and "*% Imp*" represents the percentage improvement achieved by our matheuristic for each configuration and calculated as (FORT − Matheuristic)/FORT. Note that for the instances where TC of FORT is shown with "–" Felipe et al. (2014) did not report any results.

Table C.1. Comparison of results on 100-customer instances of Felipe et al. (2014)

		FORT	Matheuristic (A)			Matheuristic (B)		
S	*Instance*	*TC*	*TC*	*Time*	*% Imp*	*TC*	*Time*	*% Imp*
9	p1	73.42	65.12	197	11.31	65.43	22	10.88
	p2	65.97	60.49	173	8.31	60.81	19	7.82
	p3	66.98	62.95	175	6.01	61.46	21	8.24
	p4	70.96	65.41	191	7.83	65.44	19	7.78
	p5	78.22	73.82	190	5.63	75.27	19	3.77
	p6	72.16	65.72	174	8.93	65.94	21	8.63
	p7	73.87	65.40	171	11.46	67.33	20	8.86
	p8	62.70	57.21	186	8.76	57.44	21	8.40
	p9	72.45	62.80	186	13.32	63.15	21	12.83
	p10	66.70	60.21	168	9.73	60.21	18	9.73
5	p1	73.79	65.29	197	11.52	65.47	32	11.28
	p2	66.58	60.92	168	8.51	60.95	30	8.46
	p3	66.92	63.12	189	5.68	63.28	33	5.44
	p4	71.74	68.24	183	4.89	65.88	30	8.17
	p5	82.02	76.28	165	7.00	76.37	29	6.88
	p6	73.59	66.25	167	9.98	66.36	31	9.83
	p7	74.00	67.65	180	8.59	67.91	30	8.23
	p8	62.74	57.21	187	8.81	57.23	30	8.78
	p9	73.28	65.37	196	10.79	65.54	31	10.56
	p10	71.19	60.91	164	14.44	63.46	29	10.86

Table C.2. Comparison of results on 200-customer instances of Felipe et al. (2014)

S	Instance	FORT TC	Matheuristic (A) TC	Matheuristic (A) Time	Matheuristic (A) % Imp	Matheuristic (B) TC	Matheuristic (B) Time	Matheuristic (B) % Imp
9	p1	-	105.54	768	-	110.49	80	-
	p2	108.24	96.40	757	10.94	97.47	83	9.95
	p3	110.21	97.34	827	11.68	99.06	77	10.12
	p4	108.69	95.78	794	11.87	99.54	77	8.42
	p5	117.44	103.65	781	11.74	106.33	75	9.46
	p6	111.26	98.17	869	11.76	101.66	83	8.62
	p7	110.42	98.48	773	10.81	102.46	78	7.21
	p8	102.37	93.66	813	8.51	94.65	84	7.54
	p9	110.08	97.41	821	11.51	101.62	86	7.68
	p10	114.71	101.06	780	11.90	104.86	73	8.59
5	p1	124.11	110.48	706	10.98	114.43	113	7.80
	p2	110.15	98.49	788	10.59	101.97	126	7.43
	p3	109.64	99.18	718	9.54	103.34	117	5.74
	p4	112.65	99.03	769	12.09	100.29	122	10.97
	p5	121.81	104.88	703	13.90	109.69	119	9.95
	p6	115.05	102.23	831	11.14	103.10	127	10.38
	p7	113.76	102.52	805	9.88	103.32	124	9.17
	p8	106.70	92.42	835	13.38	96.71	129	9.36
	p9	113.46	100.32	792	11.58	100.75	129	11.21
	p10	116.23	103.12	749	11.28	106.60	117	8.28

Table C.3. Comparison of results on 400-customer instances of Felipe et al. (2014)

S	Instance	FORT TC	Matheuristic (A) TC	Matheuristic (A) Time	Matheuristic (A) % Imp	Matheuristic (B) TC	Matheuristic (B) Time	Matheuristic (B) % Imp
9	p1	198.48	179.56	2715	9.53	182.81	333	7.90
	p2	196.50	176.34	2928	10.26	180.94	341	7.92
	p3	195.72	177.34	2631	9.39	183.52	335	6.23
	p4	190.28	172.00	3292	9.61	179.11	329	5.87
	p5	192.67	176.13	2880	8.59	181.20	348	5.95
	p6	200.66	179.52	2810	10.53	184.82	311	7.89
	p7	194.40	176.17	2903	9.38	182.97	318	5.88
	p8	194.92	176.03	3103	9.69	183.16	332	6.03
	p9	198.16	175.12	2976	11.63	182.46	334	7.92
	p10	-	177.88	3123	-	183.30	311	-
5	p1	206.60	182.70	3265	11.57	189.73	545	8.17
	p2	201.98	180.38	3393	10.69	181.00	542	10.39
	p3	207.55	183.56	2780	11.56	188.32	539	9.27
	p4	192.96	177.07	3765	8.24	187.02	555	3.08
	p5	199.21	177.34	3386	10.98	184.44	556	7.42
	p6	206.75	184.41	3200	10.80	188.15	505	9.00
	p7	205.96	182.73	3261	11.28	190.41	541	7.55
	p8	195.06	181.10	3203	7.16	191.48	509	1.84
	p9	207.38	184.38	3173	11.09	191.94	531	7.44
	p10	208.33	182.41	3046	12.44	189.35	503	9.11

References

Aksen, D., Kaya, O., Salman, S., Tüncel, Ö. (2014). An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem. European Journal of Operational Research 239 (2):413–426.

Archetti, C. and Speranza, M.G. (2014). A survey on matheuristics for routing problems. EURO Journal on Computational Optimization, 2(4): 223–246.

Barco, J., Guerra, A., Muñoz, L., Quijano, N. (2013). Optimal routing and scheduling of charge for electric vehicles: Case study. Technical Report. Available online: <http://arxiv.org/pdf/1310.0145.pdf> (last accessed 10.04.2018)

Bruglieri, M., Pezzella, F., Pisacane, O. and Suraci, S. (2015). A variable neighborhood search branching for the electric vehicle routing problem with time windows. Electronic Notes in Discrete Mathematics, 47: 221-228.

Bruglieri, M., Mancini, S., Pezzella, F., Pisacane, O., (2016). A new mathematical programming model for the green vehicle routing problem. Electronic Notes in Discrete Mathematics, 55: 89–92.

Conrad, R. G., Figliozzi, M. A. (2011). The recharging vehicle routing problem. In: Doolen, T. and Van Aken, E. (eds.) Proceedings of the 2011 Industrial Engineering Research Conference.

Çatay, B., Keskin, M. (2017). The impact of quick charging stations on the route planning of electric vehicles. Proceedings of the 2017 IEEE Symposium on Computers and Communications: 152–157.

Demir, E., Bektaş, T., Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. European Journal of Operational Research 223(2): 346–359.

Desaulniers, G., Errico, F., Irnich, S. and Schneider, M., (2016). Exact algorithms for electric vehicle-routing problems with time windows. Operations Research, 64(6): 1388–1405.

Edelstein, S. (2016). Netherlands joins Norway in plans to end new gas, diesel car sales by 2025, <http://www.greencarreports.com/news/1103507_netherlands-joins-norway-in-plans-to-end-new-gas-diesel-car-sales-by-2025> (last accessed 10.04.2018)

Emeç, U., Çatay, B. and Bozkaya, B. (2016). An adaptive large neighborhood search for an e-grocery delivery routing problem. Computers & Operations Research, 69: 109–125.

Erdogan, 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.

Feng, W. and Figliozzi, M. (2013). An economic and technological analysis of the key factors affecting the competitiveness of electric commercial vehicles: A case study from the USA market. Transportation Research Part C: Emerging Technologies 26: 135–145.

Froger, A., Mendoza, J.E., Ola, J. and Laporte, G. (2017). New formulations for the electric vehicle routing problem with nonlinear charging functions. CIRRELT-Technical Report 2017-31. Available online: < http://www.cirrelt.ca/DocumentsTravail/CIRRELT-2017-31.pdf> (last accessed 10.04.2018)

Goeke, D. and Schneider, M. (2015). Routing a mixed fleet of electric and conventional vehicles. European Journal of Operational Research, 245(1): 81–99.

Grangier, P., Gendreau, M., Lehuédé, F., Rousseau, L.M. (2016). An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. European Journal of Operational Research, 254(1): 80–91.

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 252.3: 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: Methodological, 97: 102–112.

Keskin, M., and Çatay, B. (2016). Partial recharge strategies for the electric vehicle routing problem with time windows. Transportation Research Part C: Emerging Technologies, 65: 111–127.

Khan, S. (2016). Germany pushes to ban petrol-fuelled cars within next 20 years. The Independent, <http://www.independent.co.uk/news/world/europe/germany-petrol-car-ban-no-combustion-diesel-vehicles-2030-a7354281.html> (last accessed 10.04.2018)

Koç, Ç., (2016). A unified-adaptive large neighborhood search metaheuristic for periodic location-routing problems. Transportation Research Part C: Emerging Technologies, 68: 265–284.

Koç, Ç., Bektaş, T., Jabali, O. and 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(1): 33–51.

Lin, J., Zhou, W., Wolfson, O. (2016). Electric vehicle routing problem. Transportation Research Procedia 12, 508–521.

Montoya, A., Gueret, C., Mendoza, J. E., Villegas, J. G. (2017). The electric vehicle routing problem with nonlinear charging function. Transportation Research Part B: Methodological, 103: 87–110.

Paz, J., Granada-Echeverri, M., Escobar, J. (2017). The multi-depot electric vehicle location routing problem with time windows. International Journal of Industrial Engineering Computations, (in press).

Pelletier, S., Jabali, O. and Laporte, G., (2016). 50th anniversary invited article—goods distribution with electric vehicles: review and research perspectives. Transportation Science, 50(1): 3–22.

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: Methodological, 103, 158–187.

Pelletier, S., Jabali, O. and Laporte, G., (2017). Charge scheduling for electric freight vehicles. CIRRELT Technical Report 2017-37, Available online <https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2017-37.pdf>> (last accessed 10.04.2018)

Pisinger, D., Ropke, S. (2007). A general heuristic for vehicle routing problems. Computers & Operations Research 34(8): 2403–2435.

Pollet, B.G., Staffell, I. and Shang, L.J. (2012). Current status of hybrid, battery and fuel cell electric vehicles: from electrochemistry to market prospects. Electrochimica Acta 84: 235–249.

Ribeiro, G.M., Laporte, G. (2012). An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. Computers & Operations Research 39(3): 728–735.

Ropke, S., Pisinger, D. (2006a). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science 40(4): 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(3): 750–775.

Sassi, O., Cherif, W. R., Oulamara, A. (2014). Vehicle routing problem with mixed fleet of conventional and heterogeneous electric vehicles and time dependent charging costs. HAL Technical Report hal-01083966, Available online <http://hal.archives-ouvertes.fr/hal-01083966/document> (last accessed 10.04.2018)

Schiffer, M., Walther, G. (2017). The electric location routing problem with time windows and partial recharging. European Journal of Operational Research, 260(3): 995–1013.

Schiffer, M., Schneider, M., Laporte, G. (2017). Designing sustainable mid-haul logistics networks with intra-route multi-resource facilities. European Journal of Operational Research (in press).

Schneider, M., Stenger, A., Goeke, G. (2014). The electric vehicle-routing problem with time windows and recharging stations. Transportation Science 48.4: 500–520.

Solomon, M.M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research 35 (2): 254–265.

Suzuki, Y. (2014). A variable-reduction technique for the fixed-route vehicle-refueling problem. Computers & Industrial Engineering 67: 204–215.

Sweda, T.M., Dolinskaya, I.S., Klabjan, D. (2017). Optimal recharging policies for electric vehicles. Transportation Science 51:2: 457–479.

Touati-Moungla, N., Jost, V. (2012). Combinatorial optimization for electric vehicles management. Journal of Energy and Power Engineering 6(5): 738–743.

Wen, M., Linde, E., Ropke, S., Mirchandani, P., Larsen, A. (2016). An adaptive large neighborhood search heuristic for the electric vehicle scheduling problem. Computers & Operations Research, 76: 73–83.

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

Yilmaz, M. and Krein, P.T. (2013). Review of battery charger topologies, charging power levels, and infrastructure for plug-in electric and hybrid vehicles. IEEE Transactions on Power Electronics, 28(5): 2151–2169.

European Commission White Paper on Transport 2011, Roadmap to a single European transport area: Towards a competitive and resource-efficient transport system. <http://ec.europa.eu/transport/themes/strategies/2011_white_paper_en.htm> (last accessed 09.10.2017)

Greenhouse gas emission statistics. (2016). <http://ec.europa.eu/eurostat/statistics-explained/index.php/Greenhouse_gas_emission_statistics> (last accessed 10.04.2018)

U.S. Department of State Policy Statement & Executive Summary <http://www.state.gov/documents/organization/203933.pdf> (last accessed 10.04.2018)

U.S. Department of Transportation, Bureau of Transportation Statistics. <http://www.bts.gov/> (last accessed 10.04.2018)

U.S. Environmental Protection Agency, Sources of Greenhouse Gas Emissions. <http://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions> (last accessed 10.04.2018)

United Nations, Framework Convention on Climate Change, Paris Agreement, <http://unfccc.int/paris_agreement/items/9485.php> (last accessed 10.04.2018)