Article

Goods Delivery with Electric Vehicles: Electric Vehicle Routing Optimization with Time Windows and Partial or Full Recharge

Tomislav Erdelić * and **Tonči Carić**

Faculty of Transport and Traffic Sciences, University of Zagreb, Vukelićeva 4, 10000 Zagreb, Croatia; fpz@fpz.unizg.hr

* Correspondence: terdelic@fpz.unizg.hr

Abstract: With the rise of the electric vehicle market share, many logistic companies have started to use electric vehicles for goods delivery. Compared to the vehicles with an internal combustion engine, electric vehicles are considered as a cleaner mode of transport that can reduce greenhouse gas emissions. As electric vehicles have a shorter driving range and have to visit charging stations to replenish their energy, the efficient routing plan is harder to achieve. In this paper, the Electric Vehicle Routing Problem with Time Windows (EVRPTW), which deals with the routing of electric vehicles for the purpose of goods delivery, is observed. Two recharge policies are considered: full recharge and partial recharge. To solve the problem, an Adaptive Large Neighborhood Search (ALNS) metaheuristic based on the ruin-recreate strategy is coupled with a new initial solution heuristic, local search, route removal, and exact procedure for optimal charging station placement. The procedure for the $\mathcal{O}(1)$ evaluation in EVRPTW with partial and full recharge strategies is presented. The ALNS was able to find 38 new best solutions on benchmark EVRPTW instances. The results also indicate the benefits and drawbacks of using a partial recharge strategy compared to the full recharge strategy.

Keywords: electric vehicle; goods delivery; vehicle routing problem; metaheuristics; charging station; electromobility

1. Introduction

The Electric Vehicle Routing Problem (EVRP) aims to determine a set of least-cost electric delivery routes from a depot to a set of geographically scattered customers, subject to side constraints [1]. The problem is a special case of the well-known Vehicle Routing Problem (VRP), in which the delivery is performed by the conventional Internal Combustion Engine Vehicles (ICEVs). The application of VRP in several real-life optimization problems has led to the definition of many problem variants over the years [2,3]. As the EU tends to decrease Greenhouse Gas (GHG) emissions in the transport sector by 40% by 2030, [4], new actions and regulations related to the use of cleaner transport modes have been proclaimed [5]. Here, Electric Vehicles (EVs) come to the fore, as compared to ICEVS, they have several advantages: (i) they do not have local GHG emissions; (ii) produce minimal noise; (iii) can be powered from renewable energy sources; and (iv) are independent of fluctuating oil prices [6–8]. In this paper, the focus will be on the Battery Electric Vehicles (BEVs), which have batteries mounted inside the vehicle. Compared to ICEVSs, delivery BEVs also have a significant drawback: limited driving range, usually in 160–240 km range [9]. To achieve a similar driving range as ICEVs (480–650 km [10]), BEVs have to visit a Charging Station (CS) to replenish their energy. Another drawback of BEVs is a relatively high purchase price, which can lead to a significant loss of value of the car in the first years of its use [11]. As consumer heterogeneity and differences in ambient temperature influence the cost-effective BEV range, Plug-in Hybrid EVs (PHEVs) could come to the fore as they are suitable for drivers with a high average daily traveled distance or large variance in the traveled distance [12].

In the delivery process, visits to CSs have to be accounted for within two aspects: (i) distance, time, and energy needed to travel to and from the CS, and (ii) recharging time at CS. The basic variant of the EVRP problem is the Electric Vehicle Routing Problem with Time Windows and Full Recharge at CSs (EVRPTW-FR) [1], which considers the following constraints and assumptions: (i) vehicles have equal load capacities; (ii) vehicles have equal battery capacities; (iii) energy level never drops below zero; (iv) each customer has to be visited within its time window (only once); (v) each customer has a demand (the amount of goods) and service time; (vi) linear recharging time at CS up to the full battery capacity; and (vii) linear relation between energy consumed, distance traveled, and travel time. The bi-objective function consists of the minimization of the total number of vehicles (primary objective) and the minimization of the total routing costs (secondary objective), which are usually expressed as total traveled distance, total energy consumed, total travel time, total time, GHG emissions, recharging costs, and so forth [1,13–16]. Such an objective function is contradictory, as with fewer vehicles, the secondary routing costs increase, and vice versa.

Depending on the additional constraints added to the problem, several EVRP variants emerged. As most of today's vehicle fleets include only ICEVs, the BEVs are gradually integrated into the existing fleet because the transition to a solely BEV fleet is a challenging economic task. Therefore, the most natural extension of the problem is the mixed-fleet EVRPTW [17–19], where a fleet of different vehicle types (BEVs, ICEVs, and hybrid EVs) with varying load and battery capacities is used for the delivery. Another important variant of the problem is the EVRPTW with Partial Recharge (EVRPTW-PR) [20–22], which instead of the Full Recharge (FR), considers Partial Recharge (PR) at CSs. Using the PR strategy could improve the solution quality, as less time could be spent on recharging, and most of the energy could be replenished during the lower energy network load. As today, multiple charging technologies are present: (i) battery swap stations, (ii) slow, (iii) fast, and (iv) rapid charging, variants of the EVRPTW that consider different charger types and CSs have also been introduced [16,23,24]. The other important variants of the problem consider nonlinear charging function [25,26], simultaneous CS placement and routing [21,27–29], waiting times at CSs [30,31], and so forth.

The EVRP is an Non-deterministic Polynomial (NP) hard problem, meaning that it currently cannot be solved in polynomial time, but it can be verified. Therefore, the exact procedures are capable of solving only small-sized problem instances (up to 50 customers), with a relatively long execution time [22]. To efficiently solve the problem, a great number of heuristic, metaheuristic, and hybrid procedures were proposed [7,32]. Usually, for each variant of the problem, a metaheuristic with specially defined heuristic procedures is proposed. Most researchers apply some form of the ALNS metaheuristic to solve the problem [7,16–18,20,21,33]. The ALNS method is based on the ruin-recreate principle, where throughout the search, the destroy-and-repair operators are applied based on their previous performance values [34]. The use of versatile destroy-and-repair operators helps to diversify the search and escape the local optima. The other applied metaheuristic methods include a genetic algorithm [19], tabu search [1,35], variable neighborhood search [1,24,28], simulated annealing [1,20,23], and so forth. To intensify the search, commonly, the Local Search (LS) operators are used within the metaheuristic [1,21,36], as well as exact procedures to determine the optimal CS placement [16,21].

One important aspect of the search method is the search strategy in terms of feasibility. This refers to the strategy of searching in the infeasible or feasible solution space. The feasible solution space represents solutions that satisfy all problem constraints. In recent literature reviews [7,32] it was noted that the best known solutions (BKSs) to EVRPTW problems on benchmark instances are mostly achieved by methods including an infeasible search strategy [1,17–19,21]. The reason lies in the fact that in problems with narrow feasible solution space, escaping local optima requires a large jump in the solution space, while in the infeasible solution space, this jump can be expressed as a couple of small moves that

lead to the same feasible solution. Appropriately, the secondary objective function with the infeasible search strategy should include penalties for constraint violations [21].

The other important aspect of any method used to solve a VRP is the move evaluation. Most of the methods during the search change the current solution, which can be interpreted as moves in the solution space. These moves often occur in LS, and destroy-and-repair phases. As such moves are typically applied very often during the search, the evaluation of moves should be as fast as possible, in the best case, in constant time $\mathcal{O}(1)$. The evaluation of the move represents the change of the value in the objective function, by which the good moves can be distinguished from bad ones. Regarding the efficient evaluation of the move, the problem is not the load capacity change, but rather, the change of the travel time; more precisely, the change in the start times at each customer or CS (henceforth, both are referred to as the user), which are nonlinear due to the fixed time windows. In related Vehicle Routing Problems with Time Windows (VRPTW), for most of the basic operators, the move evaluation in $\mathcal{O}(1)$ can be done by using the technique of traveling back in time to the latest feasible point when a violation occurs, and specially defined concatenation operators based on both the forward and backward user variables [37,38]. In EVRPTW, this efficient evaluation is even harder to do, as the charging amount at preceding CSs, also affects start times for users further down the route. Schneider et al. [1], for the move evaluation in EVRPTW-FR, proposed a similar technique as the one for the VRPTW problem with the $\mathcal{O}(B)$ complexity, where B is the number of users between the route change position and the last CS in the route. Goeke et al. [17], for EVRPTW-FR, proposed the use of surrogate functions to evaluate changes in the solution with $\mathcal{O}(1)$ complexity (for basic operators) and latter exact evaluation of best ϵ moves with $\mathcal{O}(n)$ complexity. Schiffer and Walhter [21] proposed forward and backward variables and concatenation operators for the Electric Location Routing Problem with Time Windows and Partial Recharging (ELRPTW-PR) with an evaluation done in $\mathcal{O}(1)$.

In this paper, the EVRPTW-FR and EVRPTW-PR variants are addressed. The ALNS proposed by the Ref. [21] for ELRPTW-PR is modified and applied to solve both observed problems. The contributions of the paper are the following:

(i)   New BKSs on several benchmark instances for both EVRPTW-FR and EVRPTW-PR;
(ii)   new heuristic for the creation of the feasible initial EVRPTW-FR solution;
(iii)   slightly modified forward and backward variables of Schiffer and Walther [21] for EVRPTW-PR that also include determination of charging amounts at CSs;
(iv)   new forward variables for EVRPTW-FR; and
(v)   comparison between FR and PR strategy in goods delivery performed by BEVs.

The rest of the paper is organized into 5 sections. In Section 2, the description of mathematical models for both EVRPTW-FR and EVRPTW-PR is given, together with the description of benchmark instances. In Section 3, the ALNS metaheuristic used to solve both problems is presented, while in Section 4 the results of the ALNS on benchmark instances are presented. In Section 5, a short discussion about differences between PR and FR strategy is given, while in Section 6, concluding remarks are given.

2. Model and Materials

In this section, EVRPTW-FR and EVRPTW-PR problems are mathematically modeled as Mixed Integer Linear Programs (MILPs) on the complete directed graph. At the end of the section, the benchmark test instances used for the comparison to the other methods from the literature are described.

2.1. EVRPTW-FR

The EVRPTW-FR problem can be modeled as a MILP program on the complete directed graph G, where customers are modeled as graph vertices, and paths between customers are modeled as graph arcs [1,20,39]. Let $V = \{1, \ldots, N\}$ be a set of geographically scattered customers that need to be served, and let F be a set of CSs. In order to allow multiple visits to the same CS, a virtual set of CSs F' is defined, where β rep-

resents the maximal number of virtual CSs (visits) to a single CS. Vertices 0 and $N+1$ denote the depot instances, and every route begins at vertex 0, and ends at vertex $N+1$ ($V_{0,N+1} = V \cup \{0\} \cup \{N+1\}$). Graph G is defined as $G = (V_{0,N+1} \cup F', A)$, where A is the set of arcs $A = \{(i,j) | i,j \in V_{0,N+1} \cup F', i \neq j\}$. The binary variable $x_{ij} \in \{0,1\}$ (Equation (1)) is equal to 1 if arc (i,j) is traversed in the solution, and 0 otherwise. The arc value d_{ij} represents the arc distance, t_{ij} represents the time needed to traverse the arc, and e_{ij} represents the energy consumption on the arc. The distance matrix representing the shortest path between each vertex (user) is computed in advance in the preprocessing step. The hierarchical objective function consists, first of vehicle number minimization (Equation (2)), and then the total traveled distance minimization (Equation (3)). Each BEV has a load capacity of C and battery capacity of Q. Recharge time is computed as a linear function value of recharged capacity with an inverse recharge rate of g. The energy consumption on the arc is computed as a linear function value of arc distance, $e_{ij} = rd_{ij}$, where r is the energy consumption rate. Each vertex i has a service time s_i, load demand q_i and time window $[e_i, l_i]$. CSs and depots have the time window $[e_0, l_0]$, that is, working hours. Beside the x_{ij} decision variable, three more decision variables are used: τ_i—start time of service, z_i—remaining load capacity, and y_i—remaining battery capacity. Equations (4)–(6) ensure arc connectivity and only one visit to each customer, Equations (7)–(9) ensure travel time and time window feasibility, and Equations (10)–(13) ensure arc load and battery constraints. For a detailed description, the reader is referred to Schneider et al. [1].

$$x_{ij} \in \{0,1\}, \ \forall i \in V_0 \cup F', j \in V_{N+1} \cup F', i \neq j \tag{1}$$

$$\min \sum_{j \in V \cup F'} x_{0j} \tag{2}$$

$$\min \sum_{i \in V_0 \cup F'} \sum_{j \in V_{N+1} \cup F', i \neq j} d_{ij} x_{ij} \tag{3}$$

$$\sum_{j \in V_{N+1} \cup F', i \neq j} x_{ij} = 1, \ i \in V \tag{4}$$

$$\sum_{j \in V_{N+1} \cup F', i \neq j} x_{ij} \leq 1, \ i \in F' \tag{5}$$

$$\sum_{i \in V_{N+1} \cup F', i \neq j} x_{ji} - \sum_{i \in V_0 \cup F', i \neq j} x_{ij} = 0, \ j \in V \cup F' \tag{6}$$

$$\tau_i + (t_{ij} + s_i) x_{ij} - l_0 \cdot (1 - x_{ij}) \leq \tau_j, \forall i \in V_0, \forall j \in V_{N+1} \cup F', i \neq j \tag{7}$$

$$\tau_i + g(Q - y_i) + x_{ij} t_{ij} - (l_0 + gQ)(1 - x_{ij}) \leq \tau_j, \ \forall i \in F', \forall j \in V_{N+1} \cup F', i \neq j \tag{8}$$

$$e_j \leq \tau_j \leq l_j, \forall j \in V_{0,N+1} \cup F' \tag{9}$$

$$0 \leq z_j \leq z_i - x_{ij}(q_i + C) + C, \ \forall i \in V_0 \cup F', \forall j \in V_{N+1} \cup F', i \neq j \tag{10}$$

$$z_0 = C, \ y_0 = Q \tag{11}$$

$$0 \leq y_j \leq y_i - (e_{ij} + Q) x_{ij} + Q, \forall j \in V_{N+1} \cup F', \forall i \in V, i \neq j \tag{12}$$

$$0 \leq y_j \leq Q - e_{ij} x_{ij}, \forall j \in V_{N+1} \cup F', \forall i \in 0 \cup F', i \neq j \tag{13}$$

2.2. EVRPTW-PR

The EVRPTW-PR can be modeled as an extension of the MILP for the EVRPTW-FR [20]. Here, only the differences are highlighted. First of all, a new decision variable Y_i is added, which represents the remaining battery capacity on the departure from CS i, given by Equation (14). The remaining battery capacity after recharging Y_i is somewhere in between the remaining battery capacity with charging at previous CSs y_i and full battery capacity Q. The equations that changed are the ones for the ensurance of CSs' exit arcs travel time and remaining battery capacity (15) and (16). The only change is that instead of charging to full capacity, Q, the vehicle is charged up to the decision variable, Y_i.

$$y_i \leq Y_i \leq Q, \ \forall i \in 0 \cup F', \tag{14}$$

$$\tau_i + g(Y_i - y_i) + x_{ij} t_{ij} - (l_0 + gQ)(1 - x_{ij}) \leq \tau_j, \ \forall i \in 0 \cup F', \forall j \in V_{N+1} \cup F', i \neq j \tag{15}$$

$$0 \leq y_j \leq Y_i - (e_{ij} + Q)x_{ij} + Q, \forall i \in 0 \cup F', \forall j \in V_{N+1} \cup F', i \neq j \qquad (16)$$

2.3. Materials

The materials include EVRPTW benchmark instances which consist of: (i) 56 large instances, each with 100 customers and 21 CSs, and (ii) 36 small instances, with 5, 10, and 15 customers per instance. The example of three instances is presented in Figure 1, where each customer is presented as a filled circle with two attributes: (i) size, which represents the demand of a customer—the greater the demand, the larger the circle, and (ii) color, from red to green, with red representing customers that close sooner (need to be visited sooner) and green representing customers that close later. The depot is represented with the purple rectangle, while the CSs are represented with blue triangles. In all instances, the Euclidean distance is used for distance computation. Each customer in an instance is directly reachable from the depot regarding the time windows, while regarding the energy, at maximum, two CSs are needed to visit each customer. Depending on the geographical distribution of the customers, instances are divided into three groups: clustered customer distribution (C) (example in Figure 1a), random customer distribution (R) (example in Figure 1b), and a mixture of both (RC) (example in Figure 1c). Additionally, the instances are divided into two groups based on the scheduling horizon: short scheduling horizon with narrow time windows (1) (examples in Figure 1a,c), and long scheduling horizon with wide time windows (2) (example in Figure 1b). For example, in instance named R201, *R* stands for random customer distribution, 2 for wide time windows, and 01 represents the instance number.

(**a**) C101

(**b**) R201

(**c**) RC101
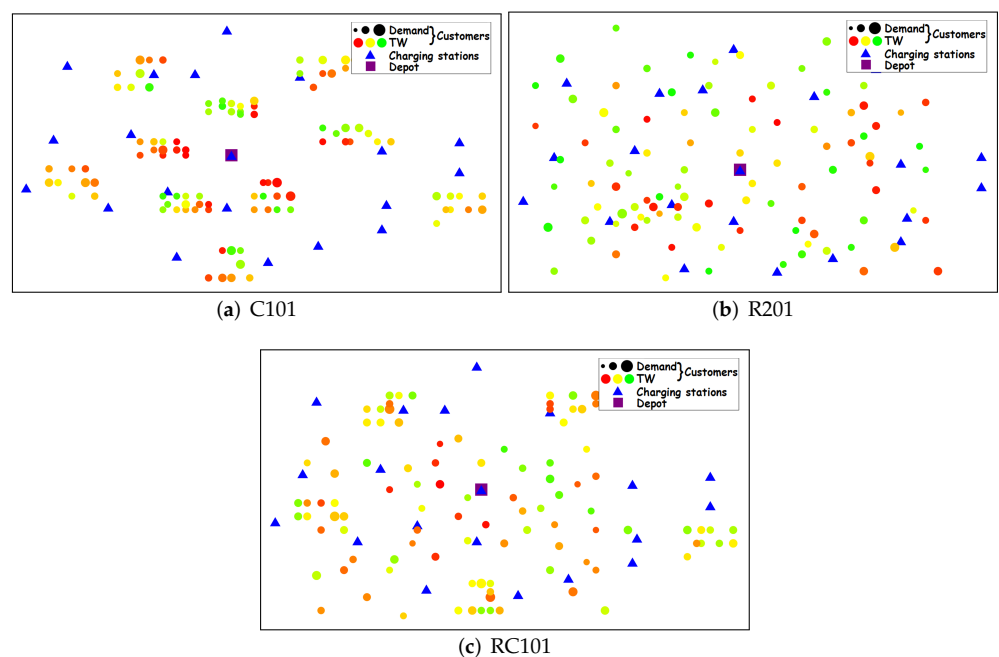
Figure 1. Examples of EVRPTW instances.

Vehicle battery capacity *Q*, vehicle load capacity *C*, energy consumption rate *r*, and recharge rate *g* per instance group are presented in Table 1. It can be seen that instance groups with wider time windows have larger vehicle load and battery capacities, which consequently results in a lower number of vehicles per instance.

Table 1. EVRPTW instances—group values.

	C1	C2	R1	R2	RC1	RC2
Q	79.69	[117.66, 118.31]	[62.14, 67.15]	[181.23, 267.18]	79.69	[159.68, 273.13]
C	200	700	200	1000	200	1000
r	1	1	1	1	1	1
g	3.39	[2.28, 2.29]	[0.45, 0.48]	[0.11, 0.17]	0.38	[0.11, 0.19]

3. Methodology

In this section, the ALNS metaheuristic is coupled with an exact procedure to solve the EVRPTW problems. Most of the algorithm framework is adopted from Schiffer and Walther [21] for the ELRPTW-PR and given by Algorithm 1. The basic idea of the method is to allow for searching in the infeasible search space. Therefore, the secondary objective function for solution s is given by Equation (17), as the sum of vehicle costs and penalties. $f_{cost}(v)$ represents the total cost of the vehicle v; $P_{load}(v)$, $P_{tw}(v)$, and $P_{batt}(v)$ represent penalty (violation) values, respectively, load capacity, time windows, and battery capacity; and α, β and γ represent penalty coefficients [1,21].

$$f(s) = \sum_{v \in s} (f_{cost}(v) + \alpha P_{load}(v) + \beta P_{tw}(v) + \gamma P_{batt}(v)) \tag{17}$$

Due to the infeasible search strategy, three solution instances were tracked during the search: (i) Temporal solution s_{temp} that stores good solution found in previous iterations (can be infeasible), (ii) the best solution s_{best} that stores the best solution so far (can be infeasible), and (iii) the best feasible solution s_{best_feas} so far. In each iteration, the current solution s is destroyed and repaired with selected operators. If the cost function of a new solution s is within Δ_{ls} percentage of the $f(s_{best})$, the LS procedure is performed. Additionally, if the cost function of new solution s after the LS procedure is within an Δ_{exact} percentage of the $f(s_{best})$, the exact procedure that determines optimal CS positions is performed. Afterwards, an update of the solution instances occurs: (i) if a new solution s is better than the current temporal solution s_{temp}, s is set as s_{temp}; (ii) if a new solution s is better than the best solution s_{best}, s is set as s_{best}; (iii) if a new solution s is feasible and better than the best feasible solution so far s_{best_feas}, s is set as s_{best_feas}. During the search, several criteria were checked: (i) The criterion for the application of the route removal algorithm (line 4) (not used in the original ALNS for ELRPTW-PR [21]), (ii) criterion for the update of values related to the used destroy-and-repair operators (line 24), (iii) criterion for the update of penalty coefficients (line 27), and (iv) overall method finish criterion (line 3). All criteria are expressed as the reached number of iterations. To intensify the search, LS and exact procedure are used, while to diversify the search, the destroy-repair strategy is used. The scoring and selection mechanisms for destroy-and-repair operators are adapted from Keskin et al. [20]. The control of intensification and diversification phases is done by the adaptive update of penalty coefficients: multiplied by ζ if in the last μ_{upw} iterations, a penalty in best solution s_{best} occurred, or divided by ζ if in the last μ_{upw} iterations, a penalty in the best solution s_{best} did not occur.

Algorithm 1 Adaptive large neighborhood search method.

```
 1: Generate initial solution s
 2: Copy instance s to instances s_temp, s_best, s_best_feas
 3: while finish criterion not met do
 4:     if route removal criterion met then
 5:         Perform route removal on s
 6:     end if
 7:     Perform destroy-and-repair operators on s
 8:     if f(s) − f(s_best) < Δ_ls f(s_best) then
 9:         Perform LS on s
10:         if f(s) − f(s_best) < Δ_exact f(s_best) then
11:             Perform exact CS placement procedure on s
12:         end if
13:     end if
14:     if f(s) < f(s_temp) then
15:         Set s as s_temp
16:         if f(s) < f(s_best) then
17:             Set s as s_best
18:             if s is feasible and f(s) < f(s_best_feas) then
19:                 Set s as s_best_feas
20:             end if
21:         end if
22:     end if
23:     Update scores of selected destroy-and-repair operators
24:     if destroy-and-repair update criterion met then
25:         Update destroy-and-repair operators scores, weights and probabilities
26:     end if
27:     if penalty coefficents update criterion met then
28:         Update penalty coefficients
29:     end if
30:     Set s_temp as s
31: end while
```

3.1. Initial Solution

The initial solution is constructed by a new Greedy Time-Oriented Nearest Neighbor Heuristic (GTONNH) given by Algorithm 2. Vehicle routes are constructed in a serial way by adding customers, one by one, that satisfy all problem constraints: vehicle load capacity, customer time window, depot time window, and vehicle battery capacity. If none of the unserved customers could be added to the current vehicle route, the current vehicle route is closed, added to the solution s, and a new vehicle route is opened. The proposed procedure is based on the procedure of Felipe et al. [23], but it has two different parts. First, instead of adding the nearest CS when a vehicle cannot visit a customer due to the violation of battery capacity, a Multiple Best CS Insertion (MBCSI) procedure is performed to insert multiple CSs in the route to make it completely feasible (including the last visit to the depot). The MBCSI is based on the Best Station Insertion (BSI) method [20] with a FR strategy at each CS, but it additionally takes multiple CSs insertions into account to make the solution completely feasible. Second, in each iteration, the best customer with possible CS insertions (move M) is determined. The cost of a move $C(M)$ is computed by Equation (18) [40], which takes into account an insertion distance increase (M_{dist}), total time increase M_{tot}, and spare time M_{st} (time between the arrival time at customer and late time window) with δ_1, δ_2 and δ_3 coefficients controlling the ratios between them. The result of the GTONNH heuristic is, if possible, is a completely feasible EVRPTW-FR solution.

$$C(M) = \delta_1 M_{dist} + \delta_2 M_{tot} + \delta_3 M_{st} \tag{18}$$
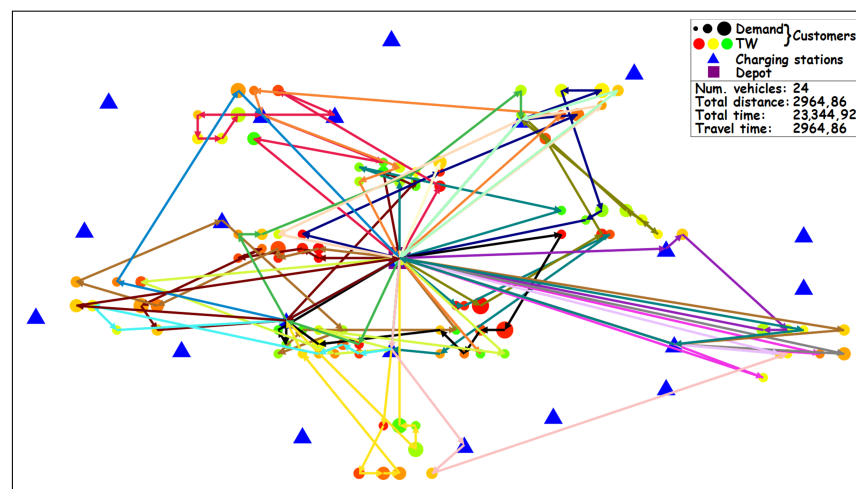
The example of the initial solution on instance C101 produced by GTONNH is presented in Figure 2. Each vehicle route is represented with different line colors. In total, 24 vehicles were produced, with a total traveled distance of 2964.86.

Algorithm 2 Greedy time-oriented nearest neighbor heuristic.

1: Create empty solution s
2: Open a new vehicle v
3: Set depot as the last visited user i
4: **while** all customers are not served **do**
5:     Set the best move M_{best} as empty and set $C(M_{best})$ to maximum value
6:     **for** each unserved customer j **do**
7:         **if** vehicle v has enough rest load capacity to accept j **and** j is reachable from i regarding time window **and** the depot is reachable from j regarding time window **then**
8:             **if** j is reachable from i regarding the energy **and** the depot is reachable from j regarding the energy **then**
9:                 Set move containing only customer j as M_{new}
10:                 **if** $C(M_{new}) < C(M_{best})$ **then**
11:                     Set move M_{new} as the best move M_{best}, and set j as i
12:                 **end if**
13:             **else**
14:                 Add depot to vehicle v
15:                 Perform MBCSI on vehicle v
16:                 **if** v is feasible **then**
17:                     Set move containing customer j and appropriate CS insertions as M_{new}
18:                     **if** $C(M_{new}) < C(M_{best})$ **then**
19:                         Set move M_{new} as the best move M_{best} and set customer j (or succeeding CS) as i
20:                     **end if**
21:                 **end if**
22:                 Remove the last depot from vehicle v
23:             **end if**
24:         **end if**
25:     **end for**
26:     **if** M_{best} is empty **then**
27:         Close the current vehicle v and add it to the solution s
28:         Open a new vehicle v
29:         Set depot as the last visited user i
30:     **else**
31:         Perform move M_{best} together with possible CSs insertions on the current vehicle v
32:         Set customer in move M_{best} as served
33:         Set customer (or succeeding CS) in move M_{best} as i
34:     **end if**
35: **end while**

Figure 2. GTONNH—Initial solution on instance C101.

3.2. Penalty Functions

As already mentioned, the penalties of the solution s are computed as the sum of penalties per each vehicle v in the solution s. Changes (moves) in the solution are frequently performed during the LS and destroy-repair phases, that is, to remove one customer from one place in the solution to another. The evaluation of these changes should be as fast as possible. The strategy of using concatenation operators for evaluation [41] is based on the principle that any route can be represented as the sum of its partial routes, meaning that a new route can be evaluated as the sum of two partial routes and a change of a customer schedule between them.

The load penalty is the easiest to compute, as it does not depend on the start time or the charging amount. If a vehicle route v is represented as a sequence of users $v = (u_0, \ldots, u_{N+1})$, the vehicle load penalty $LO(v)$ can be computed by Equation (19) as the amount of total vehicle overload. The evaluation of load penalty between partial routes $\phi_1 = (u_0, u_1, \ldots, x)$ and $\phi_2 = (y, \ldots, u_{N+1})$ can be done in $\mathcal{O}(1)$ for most of the basic move operators [42], given by the Equation (20). This requires tracking two forward variables for each user in route: rest load capacity z_i (without over-penalization) and load violation $\overrightarrow{LO}$.

$$LO(v) = \max\left(\sum_{u \in v} q_u - C, 0\right) \tag{19}$$

$$LO(\phi_1 \otimes \phi_2) = \overrightarrow{LO}(\phi_1) + \max(z_y - z_{N+1} + q_y - z_x, 0) \tag{20}$$

In this paper, concatenation operators and variables regarding time windows and battery capacity are presented for EVRPTW-FR and EVRPTW-PR, based on the ones for ELRPTW-PR [21]. As EVRPTW-PR can be considered as a special case of the ELRPTW-PR, only small changes in the variables for EVRPTW-PR were needed, that is, that do not consider charging while serving the customer, while for EVRPTW-FR, new forward variables were proposed. Additionally, variables to determine charging schedules in both variants were added. As the proposed original approach is complex and hard to follow, in this paper, an additional description is given to ease the understanding.

3.2.1. EVRPTW-PR

The first important aspect of the approach presented by Schiffer and Walther [21] is to express all variables in the unit of time [22]. The energy consumed on a path is expressed as time h_{ij} needed to recharge the consumed energy. The battery capacity Q is also expressed as the time $H = gQ$ needed to fully charge the battery. The second important aspect is the so-called shifting rules [37]—traveling back in time to the latest feasible point in time when a violation occurs. This helps to not over-penalize violations. The description of forward variables is given in Table 2 with corresponding expression given by Equations (21)–(27).

First of all, variables are propagated in forward fashion from the first depot in route to the last depot in route, that is, from user i to user j. The two basic variables are tracked: a_j^{min} and a_j^{max}, representing the earliest start time at user j by charging the minimum and maximum amount at preceding CSs. The shifted variants used to not over-penalize violations are expressed as $\tilde{a}_j^{min}$ and $\tilde{a}_j^{max}$. The goal is to, whenever possible, use slack (spare) time a_{ij}^{sl} for charging at previous CS. The spare time occurs when the vehicle arrives at a customer before the early time window and has to wait. The a_j^{rt} represents the time needed to charge the battery to the fullest with the assumption that minimum charging and spare charging was already performed. Sometimes, the whole amount of the spare time cannot be used for charging, because it would mean charging over the battery capacity Q. Therefore, the spare time is limited by the difference in time between charging minimum and maximum $\tilde{a}_j^{max} - \tilde{a}_j^{min}$. The next important variable is the minimum charging time a_{ij}^{add} that has to be added at previous CS to be able to reach user j without the battery capacity (fuel) violation. This is perhaps the most important variable that allows the

evaluation of basic moves in $\mathcal{O}(1)$. If all spare times are used for recharging, then the additional charging that needs to be performed will linearly increase the start time at user j. Of course, if this additional amount exceeds the vehicle battery capacity Q at some point, the violation will be accounted for, but still, the start time at the user will need to increase by the corresponding additional recharging time.

The variable a_{ij}^{add} represents the minimum additional amount that has to be added at previous CSs, but it does not determine the charging amount at each CS. If the partial route up to j is feasible, this means that multiple feasible CS schedules can possibly be determined, which would produce the same arrival time at user j. To know the exact charging amount, at each CS, the original approach is improved with variables that track the charging amount at CSs, given by Equations (28)–(32). The charging time at CS is computed as the sum of spare time a_{ij}^{sl} and additional charging time a_{ij}^{add}. By storing these values in the latest CS before user j, $j_{CS_{bef}}$, as slack charging time $a_{j_{CS_{bef}}}^{sl\_ct}$ and additional charging time $a_{j_{CS_{bef}}}^{add\_ct}$, the exact charging schedule can be determined. Consequently, the cumulative charging in CSs a_j^{cct} and the cumulative recharging time corresponding to the used fuel amount (energy consumed) a_j^{crt} can also be computed. In some cases, where large spare times occur as vehicles have to wait for the start of the service, the vehicle could charge more than what is needed for traversing the route with the smallest charging amount possible. This occurs when condition $H + a_{N+1}^{cct} > a_{N+1}^{crt}$ is satisfied, as the sum of the full battery capacity and recharge amount at CSs is larger than the charging amount needed for traversing the route. On such occasions, if needed, a post-optimization procedure could be applied to determine a different charging schedule that decreases the charging amount at CSs.

The computation of forward time window violation is given by Equation (33), as the amount of time after the late time window. The computation of fuel violation is given by Equation (34) as the difference between a_j^{min} and a_j^{max}. The violation occurs when the sum of additional charging time a_{ij}^{add} added to the latest CS in route is greater than the maximum possible charging time at the latest CS.

Initial values of all forward variables at the depot are set to 0, $a_0^{rt} = a_0^{min} = a_0^{max} = \tilde{a}_0^{min} = \tilde{a}_0^{max} = a_0^{cct} = a_0^{crt} = 0$.

Table 2. Forward variables—EVRPTW-PR.

Symbol	Description
a_j^{min}	Earliest start time at user j by charging the smallest amount possible before j
a_j^{max}	Earliest start time at user j by charging the maximum amount possible at preceding CS with the assumption that minimum charging was performed before preceding CS
a_j^{rt}	Time needed to charge to maximum at user j with the assumption that minimum charging was performed before j
a_{ij}^{sl}	Slack time at user j
a_{ij}^{add}	Additional charging time at user j that has to be added at preceding CS
$\tilde{a}_j^{min}$	Shifted earliest start time at user j when violation occurs by charging the smallest time possible before j
$\tilde{a}_j^{max}$	Shifted earliest start time at user j with charging maximum amount possible at preceding CS with an assumption that minimum charging was performed before preceding CS
$j_{CS_{bef}}$	An instance of the latest CS before user j
$a_{j_{CS_{bef}}}^{sl\_ct}$	Slack charging time at latest CS before user j
$a_{j_{CS_{bef}}}^{add\_ct}$	Additional charging time at latest CS before user j
a_j^{cct}	Cumulative charging time at CSs at user j
a_j^{crt}	Cumulative recharge time at user j corresponding to the used fuel amount

$$a_{ij}^{sl} = \max\left(0, e_j - \tilde{a}_i^{min} - t_{ij} - s_i\right) \tag{21}$$

$$a_j^{rt} = \begin{cases} \min\left(H, \max\left(0, a_i^{rt} - a_{ij}^{sl}\right) + h_{ij}\right) & i \in F' \\ \min\left(H, \max\left(0, a_i^{rt} - \min\left(a_{ij}^{sl}, \tilde{a}_j^{max} - \tilde{a}_j^{min}\right)\right) + h_{ij}\right) & else \end{cases} \tag{22}$$

$$a_{ij}^{add} = \begin{cases} \max\left(0, \max\left(0, a_i^{rt} - a_{ij}^{sl}\right) + h_{ij} - H\right) & i \in F' \\ \max\left(0, \max\left(0, a_i^{rt} - \min\left(a_{ij}^{sl}, \tilde{a}_j^{max} - \tilde{a}_j^{min}\right)\right) + h_{ij} - H\right) & else \end{cases} \tag{23}$$

$$a_j^{min} = \max\left(e_j, \tilde{a}_i^{min} + t_{ij} + s_i\right) + a_{ij}^{add} \tag{24}$$

$$a_j^{max} = \begin{cases} \max\left(e_j, \tilde{a}_i^{min} + a_i^{rt} + t_{ij}\right) & i \in F' \\ \max\left(e_j, \tilde{a}_i^{max} + t_{ij} + s_i\right) & else \end{cases} \tag{25}$$

$$\tilde{a}_j^{min} = \min(a_j^{min}, a_j^{max}, l_j) \tag{26}$$

$$\tilde{a}_j^{max} = \min(l_j, a_j^{max}) \tag{27}$$

$$j_{CS_{bef}} = \begin{cases} None & i = 0 \\ i & i \in F' \\ i_{CS_{bef}} & else \end{cases} \tag{28}$$

$$a_{j_{CS_{bef}}}^{sl\_ct} = \begin{cases} 0 & j_{CS_{bef}} = None \\ \min(a_{ij}^{sl}, a_i^{rt}) & i \in F' \\ a_{j_{CS_{bef}}}^{sl\_ct} + \min(a_{ij}^{sl}, \tilde{a}_j^{max} - \tilde{a}_j^{min}) & else \end{cases} \tag{29}$$

$$a_{j_{CS_{bef}}}^{add\_ct} = \begin{cases} 0 & j_{CS_{bef}} = None \\ a_{ij}^{add} & i \in F' \\ a_{j_{CS_{bef}}}^{add\_ct} + a_{ij}^{add} & else \end{cases} \tag{30}$$

$$a_j^{cct} = \begin{cases} a_i^{cct} + \min(a_{ij}^{sl}, a_i^{rt}) + a_{ij}^{add} & i \in F' \\ a_i^{cct} + \min(a_{ij}^{sl}, \tilde{a}_j^{max} - \tilde{a}_j^{min}) + a_{ij}^{add} & else \end{cases} \tag{31}$$

$$a_j^{crt} = a_i^{crt} + h_{ij} \tag{32}$$

$$\overrightarrow{TW}(\phi) = \sum_{u \in \phi} \max\left(\min\left(a_u^{min}, a_u^{max}\right) - l_u, 0\right) \tag{33}$$

$$\overrightarrow{FL}(\phi) = \sum_{u \in \phi} \max\left(a_u^{min} - a_u^{max}, 0\right) \tag{34}$$

To have an efficient move evaluation, beside the forward variables, the backward variables and penalties also have to be determined. The backward variables are presented in the Appendix A. The minimum functions are replaced by maximum functions, and e_i is swapped by l_i. The main idea is to propagate variables backwards, from the last depot in route to the first depot in route, that is, from user j to user i.

Concatenation operators for EVRPTW-PR are given by Equations (35)–(37). As these operators are not adequately described in the original paper, here, a more detailed explanation is given. For the concatenation of two partial routes $\phi_1 = (u_0, u_1, \ldots, x)$ and $\phi_2 = (y, \ldots, u_{N+1})$, first, the forward variables are propagated to user y, based on the forward variables of user x. Then, the time window of concatenated routes $TW(\phi_1 \otimes \phi_2)$ is computed by Equation (35) as the sum of forward time window violation $\overrightarrow{TW}(\phi_1)$ of partial route ϕ_1, backward time window violation $\overleftarrow{TW}(\phi_2)$ of partial route ϕ_2, and an additional two parts, noted with brackets (1) and (2). The first bracket (1) in Equation (35) represents the forward time window violation at user y consisting of the time window violation due to the late arrival at user y (bracket 1.1 in Equation (35)) reduced by the over-penalization value equal to the forward fuel penalty at user y (bracket 1.2 in Equation (35)). The second bracket (2) in Equation (35) represents the violation of the latest arrival time at user y, computed as the difference between the earliest feasible begin time (bracket 2.1 in Equation (35)) and the latest arrival time with minimum charging amount

b_y^{min}, further reduced by the over-penalization value equal to the backward fuel penalty at user y (bracket 2.2 in Equation (35)). As it can be seen, the general idea is to compute the time window violation in a typical way but to reduce it by both forward and backward fuel violations, to reduce the time window over-penalization.

The fuel violation is computed by Equation (36) as the sum of forward fuel violation $\overrightarrow{FL}(\phi_1)$ of route ϕ_1, backward fuel violation $\overleftarrow{FL}(\phi_2)$ of route ϕ_2, forward fuel violation at user y and additional value representing a case when the total route energy exceeds overall battery capacity. The value in bracket (3) in Equation (36) represents the additional charging time at user y if maximum charging is performed in both forward and backward directions. The value of D helps to reduce the over-penalization and differs depending on whether the user y is a CS, or not. In case the user y is not a CS, the overall difference is further reduced by possible spare charging time (bracket 4 in Equation (37)) at user y. The spare time for charging is computed as the sum of the time between a_y^{max} and a_y^{min} and the time between b_y^{min} and b_y^{max} (bracket 4.2 in Equation (37)), and limited by the total spare time between the latest minimum start time b_y^{min} and the earliest minimum start time a_y^{min} at user y (bracket 4.1 in Equation (37)). The computed spare time cannot be larger than H if user y is not a CS. If user y is a CS, then the spare time (bracket 5 in Equation (37)) that can be used for charging cannot be larger than the maximum charging time a_y^{rt} or the time violation between minimum forward and backward charging $a_y^{min} - b_y^{min}$ (bracket 6 in Equation (37)). The spare value (bracket 5 in Equation (37)) is computed as spare time between b_y^{min} and a_y^{min} reduced by the time window violation between maximum forward and backward charging $a_y^{max} - b_y^{max}$.

Although not discussed by Schiffer and Walther [21], it is important to note that the used procedure does not compute an exact value of the violation, but rather it approximates the sum of the time window and fuel violations. The approximated fuel and time window values can differ significantly from the corresponding exact values, but their sum is close to the exact value. The reason is that it is hard to distinguish whether the time window violation is caused by the travel time or too much charging at CS. The important part of the procedure is that it never underestimates evaluation, especially for feasible moves.

$$TW(\phi_1 \otimes \phi_2) = \overrightarrow{TW}(\phi_1) + \overleftarrow{TW}(\phi_2) + \overbrace{\max\left(0, \underbrace{a_y^{min} - l_y}_{(1.1)} - \underbrace{max(0, a_y^{min} - a_y^{max})}_{(1.2)}\right)}^{(1)} +$$

$$\overbrace{\max\left(0, \underbrace{\min\left(l_y, \max\left(e_y, a_y^{min}\right)\right)}_{(2.1)} - b_y^{min} - \underbrace{\max\left(b_y^{max} - b_y^{min}, 0\right)}_{(2.2)}\right)}^{(2)} \tag{35}$$

$$FL(\phi_1 \otimes \phi_2) = \overrightarrow{FL}(\phi_1) + \overleftarrow{FL}(\phi_2) + \max\left(0, a_y^{min} - a_y^{max}\right) + \max(0, \overbrace{a_y^{rt} + b_y^{rt} - H}^{(3)} - D) \tag{36}$$

$$D = \begin{cases} \overbrace{\min(H, \min(\underbrace{\max(0, b_y^{min} - a_y^{min})}_{(4.1)}, \underbrace{\max\left(0, a_y^{max} - a_y^{min}\right) + \max\left(0, b_y^{min} - b_y^{max}\right)}_{(4.2)})))}^{(4)} & y \notin F' \\ \min(a_y^{rt}, \underbrace{\max(0, b_y^{min} - a_y^{min} - \max\left(0, a_y^{max} - b_y^{max}\right))}_{(5)}, \underbrace{\max\left(0, a_y^{min} - b_y^{min}\right)}_{(6)})) & else \end{cases} \tag{37}$$

3.2.2. EVRPTW-FR

For EVRPTW-FR, a similar approach as in EVRPTW-PR is applied. All the variable names are similar to the ones for the EVRPTW-PR, but instead of PR, the FR at each CS is considered, that is, the whole value of a_j^{rt} is used. The new forward variables are computed by Equations (38)–(42). The a_{ij}^{add} variable accounts for the additional recharging time, which here directly corresponds to the fuel violation. The slack times are always zero, as there is no spare time to charge the vehicle any further. The forward shifted variables, time window penalties and fuel penalties remain the same as in EVRPTW-PR, as well as all backward variables and penalties. The backward variables remain the same because the solution to the FR strategy in the second partial route ϕ_2, is somewhere in between, the backward latest minimum and maximum arrival time $[b_y^{min}, b_y^{max}]$, depending on the charging amount in the forward direction. The time window concatenation operator remains the same, while the fuel concatenation operator is given by the Equation (43) without the value of D that helps to reduce the over-penalization in the PR variant. These equations represent the idea of concatenating two partial routes, the first one with the forward FR strategy and the second one with the backward PR strategy.

$$a_j^{rt} = \begin{cases} \min\left(H, h_{ij}\right) & i \in F' \\ \min\left(H, a_i^{rt} + h_{ij}\right) & else \end{cases} \tag{38}$$

$$a_{ij}^{add} = \begin{cases} \max\left(0, h_{ij} - H\right) & i \in F' \\ \max\left(0, a_i^{rt} + h_{ij} - H\right) & else \end{cases} \tag{39}$$

$$a_j^{min} = \begin{cases} \max\left(e_j, \tilde{a}_i^{min} + a_i^{rt} + t_{ij}\right) + a_{ij}^{add} & i \in F' \\ \max\left(e_j, \tilde{a}_i^{min} + t_{ij} + s_i\right) + a_{ij}^{add} & else \end{cases} \tag{40}$$

$$a_j^{max} = \begin{cases} \max\left(e_j, \tilde{a}_i^{min} + a_i^{rt} + t_{ij}\right) & i \in F' \\ \max\left(e_j, \tilde{a}_i^{min} + t_{ij} + s_i\right) & else \end{cases} \tag{41}$$

$$a_j^{cct} = \begin{cases} a_i^{cct} + a_i^{rt} & i \in F' \\ a_i^{cct} & else \end{cases} \tag{42}$$

$$FL(\phi_1 \otimes \phi_2) = \overrightarrow{FL}(\phi_1) + \overleftarrow{FL}(\phi_2) + \max\left(0, a_y^{min} - a_y^{max}\right) + \max(0, a_y^{rt} + b_y^{rt} - H) \tag{43}$$

As in EVRPTW-FR, the latest start time for FR strategy is somewhere in between $[b_y^{min}, b_y^{max}]$, the approximated violation can significantly differ from the exact one, but it still gives a good representation value to differ bad from good moves. Therefore, in this paper, we use a restricted move list that stores n_{RCL}^{ls} best moves during the search phase of each LS operator. At the end of each LS operator, n_{RCL}^{ls} moves are evaluated exactly with $\mathcal{O}(n)$ complexity, and the best one is selected.

3.3. Destroy-and-Repair Operators

The used destroy operators are the following: *Worst removal* [34], *Related removal* [43], *Shaw removal* [44], and *CS vicinity* [17]. The *Related removal* and *Shaw removal* operators consider removing customers that are close in some sense, while the *Worst removal* operator removes customers that incur the highest costs. *CS vicinity* operator removes a CS and all customers in a defined radial vicinity form it. The number of removed customers is selected at random in the $[\mu_{low}|N|, \mu_{high}|N|]$ interval, where $|N|$ is the number of customers in instance, and μ_{low} and μ_{high} are threshold percentage values. For the repair of the solution the Last-In First-Out (LIFO) strategy is used, meaning that the customers are inserted back in the solution based on their removal order [18,21]. In each iteration, four insertions are evaluated: (i) customer-only insertion; (ii) customer with preceding CS; (iii) customer with succeeding CS; and (iv) customer with both preceding and succeeding CS.

Additionally, a route removal algorithm is applied at certain points during the search. The algorithm removes one route at random and applies a similar ALNS framework as in Algorithm 1 to find a solution with a lower number of vehicles. The criteria for route removal tends to apply the route removal procedure more frequently at the beginning of the search and then gradually reduce its calls towards the end of the search.

3.4. Local Search and Optimal CS Placement

To improve the solution generated by destroy-and-repair operators, the following local search operators are used: *Intra relocate*, *Intra exchange*, *Or-opt*, *Intra station in*, *Intra station out*, *Inter relocate*, *Inter exchange*, *Inter cross exchange* ($k = 3$), and *Inter 2-Opt** [7,45–48].

As CS configuration significantly influences the solution quality, after the local search, sometimes the optimal CS placement is performed. The applied algorithm is based on the forward labeling algorithms for the elementary shortest path problem with resource constraints for both the FR and PR strategy [22]. The used variables and equations are presented by Desaulniers et al. [22]. The basic idea is to remove CS visits in the route and then generate a limited search tree by extending the fixed route with CSs (paths). To reduce the search space, the dominance rules are applied between paths to discard paths that are always worse than other paths (dominated by other paths). For the evaluation of paths and dominance comparison, the labeling technique is used.

3.5. Example of a Search Process

The example of a cost function during the search process when solving EVRPTW-FR on instance R101 is presented in Figure 3. The values of penalty coefficient α are plotted with black color and presented on the right y axis. On the left y axis are the values of the cost function $f(s)$ for the best solution s_{best} (blue line for the feasible part and red line for the infeasible part), and best feasible solution s_{best_feas} (green line). The first occurrence of the best overall feasible solution $s_{best\_feas}^{min}$ is represented with a green circle. The update value of penalties is $\zeta = 1.2$, and the penalties are updated every $\mu_{upw} = 20$ iterations. The cost value of the initial solution is 2680.61. It can be seen that the feasible solution is rapidly improved at the beginning of the search, with occasional spikes. These spikes represent that the feasible solution with a lower number of vehicles is found and that it is accepted although it has a worse cost function value. When the best solution s_{best} is infeasible but accepted due to the acceptance criteria (red lines), the penalty coefficient α increases, while the opposite occurs when the best solution s_{best} is feasible. In the zoomed part, it can be seen that allowing infeasible solutions can lead to better feasible solutions. Additionally, as the minimum best feasible solution $s_{best\_feas}^{min}$ is relatively quickly found ($i = 880$), in the rest of the search, the infeasible solutions are accepted, and penalties are updated to perhaps find a better solution.
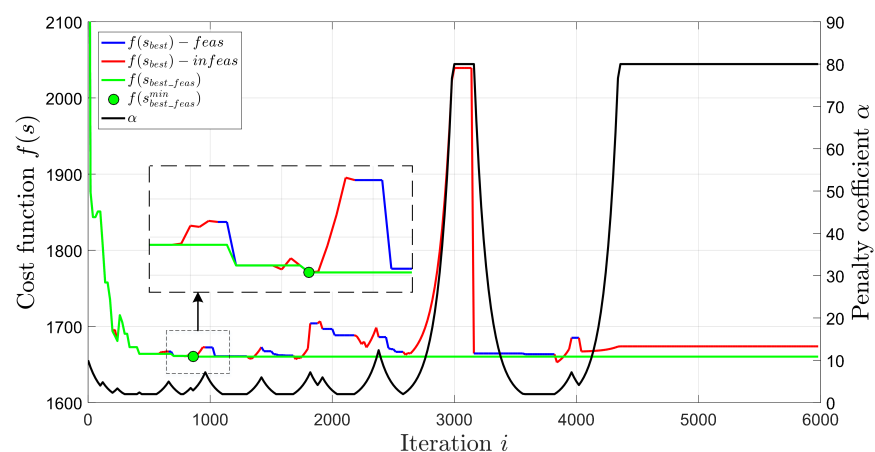

Figure 3. Cost function in search process—R101—EVRPTW-FR.

The example of applied destroy-and-repair operators on instance C101, together with the example of operators selection probabilities are presented in Appendix B.

4. Results

In this section, the proposed ALNS method is run on benchmark EVRPTW instances and compared to the BKSs from the literature. The ALNS is implemented as a single-thread code in the C# programming language. All tests were performed on a machine with an Intel E5 processor and 32 GB of RAM. The floating-point precision used for the comparison was set to 10^{-6}. Most of the ALNS parameters are similar to the one of Schiffer and Walther [21], except the following: (i) coefficients controlling ratios in GTONNH—$\delta_1 = 0.5$, $\delta_2 = 0.4$, $\delta_3 = 0.1$, (ii) total number of iterations $\mu_{max} = 10,000$, (iii) number of restricted LS moves $n_{RCL}^{ls} = 20$, and (iv) lower and upper percentage threshold for the number of customers removed $[\mu_{low}, \mu_{high}] = [0.1, 0.4]$. The text files containing results of ALNS on benchmark EVRPTW instances are available https://www.fpz.unizg.hr/zits/?page_id=1424 (accessed on 9 December 2021).

4.1. EVRPTW-PR

The results of distance minimization in EVRPTW-PR are presented in Table 3. The ALNS was run 10 times on each instance. The results are compared to the BKSs of Hierman et al. [19] (HIER), Keskin et al. [20] (KESK) and Schiffer and Walther [21] (SCHI). Here, we have to note that some BKSs reported by SCHI have a possibly wrong number of vehicles. The authors stated that they compared their results to the results of KESK [20] but did not use the BKS values presented in the paper; rather, they used some different reference values which are not available in the literature. Therefore, we tried to correct the number of vehicles reported by SCHI to make the comparison reliable. Further on, three columns are used to represent BKSs values: name N, vehicle number K, and total traveled distance d. For the ALNS, the following columns are presented: average vehicle number $\overline{K}$, best vehicle number K_{best}, the difference in best vehicle number ΔK between BKS and ALNS, average total traveled distance $\overline{d}$, best total traveled distance d_{best}, total traveled distance relative difference Δd and percentage relative difference $\Delta_p d$ between BKS and ALNS, total time of the best solution tot_{best}, recharging amount of the best solution rea_{best}, number of CSs in the best solution m, and average execution time $\overline{t}_e$ in minutes. The relative percentage difference was computed as $\Delta_p d = \frac{d_{best} - d_{BKS}}{d_{BKS}} \cdot 100$. It is important to note that in relative distance computation, Δd and $\Delta_p d$, only the solutions in which the ALNS produced the minimum number of vehicles were considered, because otherwise, with a higher number of vehicles, the total traveled distance decreases which then produces a wrong comparison. Lastly, the initial solution values produced by GTONNH are presented in the last two columns: initial vehicle number K^{ini}, and initial total traveled distance d^{ini}. The summary of the results is presented in the last two rows as averaged and summed values. All rows in which the ALNS produced a better solution regarding the vehicle number or the total traveled distance are bolded.

In 29 out of 56 instances, the ALNS found a better solution, with in total 9 vehicle less, and 0.52 average relative percentage difference in the total traveled distance, meaning that even if the lower number of vehicles is found, the ALNS finds a good user configuration in terms of distance minimization. The average value of the average vehicle number $\overline{K}$ has a lower value than the average of the BKSs vehicle number, indicating the good average performance of the ALNS. The sum of the total time in all instances consists of: travel time—20.48%, service time—67.02%, waiting times—7.21% and charging time—5.29%. As it can be seen, most of the total time is spent on service, while the least time is spent on charging and waiting. On average, 10 CSs are visited per instance and 1.31 per vehicle route. Additionally, on average, vehicle leaves the CS with the rest battery capacity equal to 72.26% of full battery capacity. In terms of an average execution time, the ALNS execution time of 8.93 min is comparable to HIER (9.96 min) and KESK (11.14 min), but worse than SCHI (4.13 min).

Table 3. EVRPTW-PR distance—results.

Inst.	BKS						ALNS								GTONNH	
	N	K	d	$\overline{K}$	K_{best}	ΔK	$\overline{d}$	d_{best}	Δd	$\Delta_p d$	tot_{best}	rea_{best}	m	$\overline{t}_e$	K^{ini}	d^{ini}
C101	SCHI	12	1043.38	12.00	12	0	1043.38	1043.38	0.00	0.00	12,647.62	177.34	8	3.67	24	2964.86
C102	**SCHI**	**11**	**1029.44**	**10.30**	**10**	**−1**	**1079.12**	**1058.67**	**29.23**	**2.84**	**11,619.45**	**295.44**	**12**	**3.17**	**20**	**2334.54**
C103	**SCHI**	**10**	**971.86**	**10.00**	**10**	**0**	**971.88**	**971.19**	**−0.67**	**−0.07**	**11,411.29**	**207.20**	**9**	**4.14**	**18**	**2423.24**
C104	SCHI	10	884.38	10.00	10	0	884.63	884.38	0.00	0.00	11,057.74	134.83	7	7.57	14	1827.17
C105	**KESK**	**11**	**1037.78**	**10.10**	**10**	**−1**	**1081.23**	**1064.66**	**26.88**	**2.59**	**11,436.30**	**296.02**	**11**	**2.39**	**21**	**2715.04**
C106	**SCHI**	**11**	**1010.56**	**10.10**	**10**	**−1**	**1070.88**	**1061.61**	**51.05**	**5.05**	**11,376.02**	**275.31**	**11**	**2.53**	**20**	**2546.45**
C107	SCHI	10	1010.91	10.00	10	0	1064.94	1046.50	35.59	3.52	11,254.18	283.82	12	2.64	19	2377.63
C108	**SCHI**	**10**	**1031.85**	**10.00**	**10**	**0**	**1025.16**	**1022.93**	**−8.92**	**−0.86**	**11,017.02**	**238.14**	**12**	**4.58**	**16**	**2157.54**
C109	SCHI	10	940.38	10.00	10	0	945.49	940.38	0.00	0.00	10,744.74	164.42	9	4.48	15	1802.63
C201	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,125.99	168.26	3	0.82	8	2197.72
C202	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,125.99	168.26	3	2.32	8	2135.94
C203	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,074.33	168.50	3	2.59	6	2042.13
C204	SCHI	4	628.91	4.00	4	0	628.91	628.91	0.00	0.00	10,343.46	168.92	3	5.17	5	1407.93
C205	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,020.61	169.85	3	1.06	7	1538.23
C206	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,021.16	170.09	3	1.29	6	1414.65
C207	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,021.43	170.21	3	1.61	7	2143.92
C208	KESK	4	629.95	4.00	4	0	629.95	629.95	0.00	0.00	10,021.43	170.21	3	1.31	6	1534.82
R101	**SCHI**	**18**	**1615.50**	**17.10**	**17**	**−1**	**1648.68**	**1624.89**	**9.39**	**0.58**	**3587.38**	**662.79**	**27**	**6.19**	**29**	**2680.61**
R102	**HIER**	**15**	**1521.32**	**15.00**	**15**	**0**	**1456.53**	**1454.53**	**−66.79**	**−4.39**	**3123.33**	**580.99**	**25**	**7.36**	**29**	**2609.90**
R103	**SCHI**	**13**	**1244.15**	**12.90**	**12**	**−1**	**1223.24**	**1304.24**	**60.09**	**4.83**	**2650.75**	**571.49**	**23**	**8.28**	**22**	**2164.09**
R104	**SCHI**	**11**	**1056.87**	**11.00**	**11**	**0**	**1055.36**	**1051.41**	**−5.46**	**−0.52**	**2386.27**	**345.83**	**12**	**11.77**	**18**	**1866.25**
R105	SCHI	14	1347.80	14.00	14	0	1347.80	1347.80	0.00	0.00	2903.37	509.99	21	5.40	24	2361.51
R106	**SCHI**	**13**	**1268.25**	**13.00**	**13**	**0**	**1265.18**	**1263.13**	**−5.12**	**−0.40**	**2722.36**	**481.85**	**19**	**8.50**	**22**	**2256.61**
R107	**SCHI**	**12**	**1110.95**	**11.00**	**11**	**−1**	**1106.56**	**1104.51**	**−6.44**	**−0.58**	**2391.69**	**391.89**	**17**	**10.59**	**19**	**2030.87**
R108	**SCHI**	**11**	**1020.52**	**10.60**	**10**	**−1**	**1030.40**	**1030.44**	**9.92**	**0.97**	**2239.85**	**389.84**	**16**	**11.60**	**17**	**1737.33**
R109	**SCHI**	**12**	**1186.99**	**12.00**	**12**	**0**	**1182.84**	**1176.69**	**−10.30**	**−0.87**	**2569.17**	**401.95**	**16**	**8.38**	**19**	**2048.53**
R110	**SCHI**	**11**	**1070.99**	**11.00**	**11**	**0**	**1070.49**	**1067.11**	**−3.88**	**−0.36**	**2395.60**	**340.31**	**15**	**8.25**	**18**	**1911.75**
R111	**HIER**	**11**	**1147.22**	**11.00**	**11**	**0**	**1085.18**	**1076.15**	**−71.07**	**−6.20**	**2338.33**	**355.71**	**14**	**8.15**	**19**	**1981.17**
R112	SCHI	11	1001.79	11.00	11	0	1001.79	1001.79	0.00	0.00	2322.52	303.04	13	7.47	16	1712.88
R201	SCHI	3	1255.81	3.00	3	0	1258.40	1255.81	0.00	0.00	2892.61	804.33	7	10.45	5	1985.37
R202	HIER	3	1051.46	3.00	3	0	1053.13	1052.52	1.06	0.10	2936.01	356.91	4	11.43	6	2155.57
R203	KESK	3	895.54	3.00	3	0	897.20	895.54	0.00	0.00	2930.01	500.04	4	19.07	4	1588.31
R204	SCHI	2	779.49	2.00	2	0	780.72	779.49	0.00	0.00	1987.21	314.15	3	22.09	5	1448.01
R205	**KESK**	**3**	**987.36**	**3.00**	**3**	**0**	**989.69**	**987.22**	**−0.14**	**−0.01**	**2718.76**	**449.59**	**4**	**9.31**	**4**	**1772.33**
R206	**KESK**	**3**	**922.70**	**3.00**	**3**	**0**	**929.39**	**922.08**	**−0.62**	**−0.07**	**2763.95**	**445.56**	**5**	**11.69**	**4**	**1658.96**
R207	SCHI	2	843.20	2.00	2	0	850.56	845.19	1.99	0.24	1993.88	372.47	3	16.01	4	1502.64
R208	KESK	2	736.12	2.00	2	0	743.32	736.46	0.34	0.05	1911.93	338.01	2	23.94	4	1489.43
R209	**SCHI**	**3**	**863.36**	**3.00**	**3**	**0**	**876.87**	**863.17**	**−0.19**	**−0.02**	**2726.92**	**425.69**	**4**	**11.14**	**4**	**1654.15**
R210	KESK	3	843.36	3.00	3	0	848.53	844.71	1.35	0.16	2785.50	416.14	5	13.92	4	1699.56
R211	**SCHI**	**2**	**827.29**	**2.00**	**2**	**0**	**831.66**	**825.25**	**−2.04**	**−0.25**	**1892.85**	**313.56**	**3**	**19.41**	**3**	**1550.22**
RC101	**HIER**	**15**	**1725.73**	**15.00**	**15**	**0**	**1677.28**	**1661.53**	**−64.20**	**−3.72**	**3193.93**	**529.16**	**22**	**5.88**	**28**	**3024.76**
RC102	KESK	14	1155.50	14.00	14	0	1517.43	1510.16	354.66	30.69	2951.51	436.57	19	7.69	23	2757.49
RC103	**HIER**	**12**	**1388.72**	**12.40**	**12**	**0**	**1353.28**	**1359.34**	**−29.38**	**−2.12**	**2705.80**	**432.85**	**15**	**7.59**	**20**	**2459.65**
RC104	**SCHI**	**11**	**1175.06**	**11.00**	**11**	**0**	**1179.20**	**1174.32**	**−0.74**	**−0.06**	**2416.92**	**362.04**	**13**	**9.59**	**17**	**2081.58**
RC105	**SCHI**	**14**	**1450.82**	**13.30**	**13**	**−1**	**1470.81**	**1471.80**	**20.98**	**1.45**	**2861.33**	**449.56**	**19**	**4.49**	**23**	**2679.28**
RC106	SCHI	13	1385.96	13.00	13	0	1397.63	1391.23	5.27	0.38	2832.90	374.79	16	5.56	23	2732.83
RC107	**SCHI**	**12**	**1250.30**	**11.20**	**11**	**−1**	**1255.91**	**1244.37**	**−5.93**	**−0.47**	**2448.38**	**386.12**	**15**	**6.14**	**19**	**2192.62**
RC108	SCHI	11	1154.14	11.00	11	0	1161.86	1154.14	0.00	0.00	2391.45	298.08	12	9.02	20	2347.57
RC201	**SCHI**	**4**	**1445.17**	**4.00**	**4**	**0**	**1446.60**	**1433.57**	**−11.60**	**−0.80**	**3650.43**	**653.52**	**7**	**7.90**	**6**	**2398.56**
RC202	**SCHI**	**3**	**1408.08**	**3.00**	**3**	**0**	**1413.90**	**1403.67**	**−4.41**	**−0.31**	**2838.67**	**590.55**	**6**	**14.14**	**6**	**2415.52**
RC203	**SCHI**	**3**	**1060.32**	**3.00**	**3**	**0**	**1062.00**	**1054.91**	**−5.41**	**−0.51**	**2708.69**	**492.39**	**7**	**21.07**	**5**	**2008.21**
RC204	SCHI	3	884.75	3.00	3	0	889.39	884.75	0.00	0.00	2679.15	458.49	5	20.62	5	1758.81
RC205	**SCHI**	**3**	**1259.69**	**3.00**	**3**	**0**	**1264.52**	**1238.46**	**−21.23**	**−1.69**	**2661.49**	**744.49**	**10**	**16.04**	**5**	**2118.31**
RC206	SCHI	3	1189.11	3.00	3	0	1207.98	1197.60	8.49	0.71	2693.07	558.52	6	10.66	5	1998.35
RC207	**HIER**	**3**	**985.67**	**3.00**	**3**	**0**	**1002.83**	**978.30**	**−7.37**	**−0.75**	**2515.67**	**342.55**	**5**	**14.87**	**4**	**1957.30**
RC208	SCHI	3	833.12	3.00	3	0	842.78	833.12	0.00	0.00	2425.03	336.23	6	16.90	4	1586.20
AVG		7.77	1041.95	7.66	7.61	−0.16	1051.47	1047.03	5.08	0.52	5115.42	374.02	10.00	8.93	13.07	2070.49
SUM		435	58,349.28	429.00	426	−9.00	58,882.22	58,633.65	284.37	29.13	286,463.42	20,944.92	560	499.91	732	115,947.54

As expected, GTONNH produced feasible initial solutions that are far from the best ones, as an initial feasible solution is hard to achieve. In the total sum of vehicle number, GTONNH produced 71.83% vehicle more, with also an increase in the total traveled distance by 97.75%.

The results also differ depending on the instance group. It can be seen that on instance groups with shorter time windows (*C*1, *R*1 and *RC*1), which also have a lower battery capacity *Q*, a larger number of vehicles is produced. The *RC*, and especially *R* instances, are harder to solve than *C* instances due to the larger feasible solution space. This is especially the case in *R*2 and *RC*2 instance groups, where the execution time significantly increases due to the larger number of customers per route and frequent application of LS and exact CS placement procedure.

To see the influence of integrating BEVs in the goods delivery, the results are compared to the BKSs on Solomon VRPTW instances, downloaded from https://www.sintef.no/projectweb/top/vrptw/solomon-benchmark/ page (accessed on 9 December 2021). In total, the BKSs on VRPTW instances contain 405 vehicles and produce the total traveled distance of 57,186.88. Although the direct comparison is not entirely correct as some customers in VRPTW instances have relaxed time windows in EVRPTW instances [1], still some general conclusion could be drawn, that even with the relaxed time windows, the EVRPTW-PR produced 5.18% more vehicles and increased total traveled distance by 2.53%.

The example of BKS on EVRPTW-PR C101 instance is presented in Figure 4a. In total, 12 vehicles are presented, each with different color, with the total traveled distance of 1043.38. To compare the recharge strategies, additionally user IDs are added above circles in the Figure 4.

4.2. EVRPTW-FR

The results of distance minimization in EVRPTW-FR are presented in Table 4. The ALNS was again run 10 times, and the same columns as in the EVRPTW-PR variant are used, except the columns for GTONNH, as it has the same values per instance as in the PR variant. The results are compared to the BKSs of: Schneider et al. [1] (SCHN), Goeke et al. [17] (GOEK), Hierman et al. [18] (HIE1), Hierman et al. [19] (HIE2), Keskin et al. [20] (KESK) and Schiffer et al. [21] (SCHI). It can be seen that ALNS was not able to reduce the number of vehicles on four instances: *R*105, *R*106, *R*110 and *RC*102. Nevertheless, nine new BKSs (bold font) are found for instances: *C*109, *R*104, *R*108, *R*211, *RC*105, *RC*107, *RC*201, *RC*205 and *RC*206. The total traveled distance is lower than in BKSs, due to the higher number of vehicles.

Compared to the FR strategy, it can be seen that PR strategy produced 18 vehicles less, with also decrease in the sum of the total traveled distance d_{best} (1.82%) and the sum of the total time tot_{best} (5.60%). As a result, a PR strategy uses 48 CSs more, but actually charges less, as a recharge amount is 30.39% lower than the one for the FR. The total time in all instances consists of the following percentages: travel time—19.75%, service time—63.46%, waiting times—8.24% and charging time—8.55%. Here again, it can be seen that the charging time with the FR strategy is greater than the charging time with the PR strategy. The average execution time $\bar{t}_e$ for FR strategy is 11.61 min. The produced time is slightly higher than the one in the PR strategy due to the additional evaluation of n_{LS} best moves exactly in the LS. Compared to the BKSs of VRPTW instances, the ALNS in EVRPTW-FR produced 39 vehicles more (9.63%) and increased total traveled distance by 4.43%.

The example of BKS on EVRPTW-FR C101 instance is presented in Figure 4b, with in total 12 vehicles and total traveled distance of 1053.83.

Table 4. EVRPTW-FR distance—results.

Inst.	BKS					ALNS								
	N	K	d	$\overline{K}$	K_{best}	ΔK	$\overline{d}$	d_{best}	Δd	$\Delta_p d$	tot_{best}	rea_{best}	m	$\overline{t}_e$
C101	SCHN	12	1053.83	12.00	12	0	1053.83	1053.83	0.00	0.00	129,04.61	396.61	8	2.65
C102	GOEK	11	1051.38	11.00	11	0	1055.18	1051.38	0.00	0.00	12,832.67	392.70	8	5.13
C103	GOEK	10	1034.86	10.00	10	0	1076.39	1034.86	0.00	0.00	11,818.14	397.53	9	3.55
C104	KESK	10	951.57	10.00	10	0	955.45	951.57	0.00	0.00	11,575.87	339.48	7	6.38
C105	SCHN	11	1075.37	11.00	11	0	1075.37	1075.37	0.00	0.00	12,039.54	459.49	9	3.73
C106	GOEK	11	1057.65	11.00	11	0	1057.65	1057.65	0.00	0.00	12,181.07	441.68	9	3.39
C107	SCHN	11	1031.56	11.00	11	0	1031.56	1031.56	0.00	0.00	12,167.43	444.15	9	5.07
C108	GOEK	10	1095.66	10.90	10	0	1026.70	1125.95	30.29	2.76	11,899.76	516.96	11	4.17
C109	**GOEK**	**10**	**1033.67**	**10.00**	**10**	**0**	**1032.55**	**1027.10**	**−6.57**	**−0.64**	**11,781.58**	**452.53**	**9**	**4.77**
C201	SCHN	4	645.16	4.00	4	0	645.16	645.16	0.00	0.00	10,288.23	258.90	4	2.13
C202	SCHN	4	645.16	4.00	4	0	645.16	645.16	0.00	0.00	10,288.23	258.90	4	7.57
C203	SCHN	4	644.98	4.00	4	0	644.98	644.98	0.00	0.00	11,114.16	293.99	4	6.11
C204	SCHN	4	636.43	4.00	4	0	636.43	636.43	0.00	0.00	10,770.32	291.50	4	12.12
C205	SCHN	4	641.13	4.00	4	0	641.13	641.13	0.00	0.00	10,289.50	282.39	3	2.81
C206	SCHN	4	638.17	4.00	4	0	638.17	638.17	0.00	0.00	11,506.96	352.47	4	3.48
C207	SCHN	4	638.17	4.00	4	0	638.17	638.17	0.00	0.00	11,360.96	352.47	4	4.17
C208	SCHN	4	638.17	4.00	4	0	638.17	638.17	0.00	0.00	11,382.96	352.47	4	3.61
R101	HIE2	18	1663.04	18.00	18	0	1667.72	1663.04	0.00	0.00	3820.37	839.73	22	7.00
R102	HIE2	16	1484.57	16.50	16	0	1489.33	1490.40	5.83	0.39	3301.68	811.83	23	11.55
R103	HIE2	13	1268.88	13.20	13	0	1297.50	1279.53	10.65	0.84	2817.76	669.64	18	10.90
R104	**SCHN**	**11**	**1088.43**	**11.00**	**11**	**0**	**1093.11**	**1088.09**	**−0.34**	**−0.03**	**2429.12**	**492.75**	**13**	**12.89**
R105	GOEK	14	1442.35	15.00	15	1	1390.84	1383.29	−59.06	−4.09	3109.09	700.23	19	6.35
R106	GOEK	13	1324.10	14.00	14	1	1283.53	1280.14	−43.96	−3.32	2812.46	639.57	17	11.96
R107	HIE2	12	1148.38	12.00	12	0	1152.32	1148.38	0.00	0.00	2610.63	628.05	14	10.87
R108	**HIE2**	**11**	**1049.12**	**11.00**	**11**	**0**	**1046.54**	**1042.63**	**−6.49**	**−0.62**	**2388.86**	**565.50**	**15**	**18.80**
R109	GOEK	12	1261.31	12.90	12	0	1228.20	1281.07	19.76	1.57	2634.99	618.87	18	10.61
R110	GOEK	11	1119.50	12.00	12	1	1098.36	1094.35	−25.15	−2.25	2610.55	575.46	14	14.07
R111	HIE2	12	1099.53	12.00	12	0	1107.58	1102.18	2.65	0.24	2618.52	577.83	15	12.24
R112	GOEK	11	1016.63	11.00	11	0	1017.42	1016.63	0.00	0.00	2411.74	497.44	13	11.05
R201	SCHI	3	1264.32	3.00	3	0	1268.08	1264.37	0.05	0.00	2912.47	865.41	7	7.78
R202	SCHN	3	1052.32	3.00	3	0	1053.59	1052.32	0.00	0.00	2882.20	497.94	3	11.98
R203	GOEK	3	895.54	3.00	3	0	896.08	895.54	0.00	0.00	2930.01	500.04	4	22.22
R204	GOEK	2	779.49	2.00	2	0	781.05	779.49	0.00	0.00	1988.76	327.10	3	30.69
R205	GOEK	3	987.36	3.00	3	0	994.05	989.03	1.67	0.17	2695.55	611.30	4	11.86
R206	GOEK	3	922.19	3.00	3	0	927.26	922.19	0.00	0.00	2782.55	554.32	4	19.39
R207	SCHI	2	843.20	2.00	2	0	858.15	848.67	5.47	0.65	1986.82	408.39	2	15.51
R208	GOEK	2	736.12	2.00	2	0	740.59	736.12	0.00	0.00	1875.73	306.38	2	21.78
R209	GOEK	3	867.05	3.00	3	0	876.72	871.23	4.18	0.48	2584.27	389.07	3	9.35
R210	KESK	3	843.65	3.00	3	0	850.29	845.83	2.18	0.26	2788.83	534.42	5	9.79
R211	**GOEK**	**2**	**827.89**	**2.00**	**2**	**0**	**846.00**	**827.29**	**−0.60**	**−0.07**	**1901.20**	**322.57**	**3**	**15.83**
RC101	HIE2	16	1723.79	16.00	16	0	1730.84	1723.79	0.00	0.00	3445.33	816.58	17	6.50
RC102	HIE1	14	1659.53	15.00	15	1	1553.86	1552.55	−106.98	−6.45	3243.74	858.32	16	12.65
RC103	GOEK	13	1350.09	13.00	13	0	1351.01	1350.55	0.46	0.03	2821.98	521.02	14	16.47
RC104	GOEK	11	1227.25	11.70	11	0	1229.92	1228.69	1.44	0.12	2495.46	517.94	13	14.70
RC105	**HIE2**	**14**	**1473.24**	**14.00**	**14**	**0**	**1474.24**	**1471.87**	**−1.37**	**−0.09**	**3078.57**	**814.57**	**16**	**6.00**
RC106	HIE2	13	1423.27	13.00	13	0	1433.94	1423.27	0.00	0.00	2906.45	795.31	16	7.33
RC107	**HIE2**	**12**	**1274.41**	**12.00**	**12**	**0**	**1276.14**	**1274.25**	**−0.16**	**−0.01**	**2691.41**	**673.54**	**13**	**12.84**
RC108	HIE2	11	1197.41	11.00	11	0	1200.34	1197.41	0.00	0.00	2507.67	670.42	14	12.38
RC201	**SCHN**	**4**	**1444.94**	**4.00**	**4**	**0**	**1446.08**	**1441.47**	**−3.47**	**−0.24**	**3649.46**	**695.66**	**5**	**7.84**
RC202	SCHI	3	1408.16	3.00	3	0	1420.07	1414.97	6.81	0.48	2830.46	694.64	5	18.63
RC203	GOEK	3	1055.19	3.00	3	0	1059.59	1055.19	0.00	0.00	2728.63	659.24	5	25.51
RC204	SCHI	3	884.53	3.00	3	0	888.61	884.75	0.22	0.03	2679.81	461.96	5	36.27
RC205	**SCHI**	**3**	**1262.38**	**3.00**	**3**	**0**	**1267.20**	**1256.09**	**−6.29**	**−0.50**	**2701.78**	**951.80**	**9**	**21.37**
RC206	**GOEK**	**3**	**1188.63**	**3.00**	**3**	**0**	**1197.83**	**1187.72**	**−0.91**	**−0.08**	**2605.25**	**626.11**	**4**	**11.90**
RC207	GOEK	3	985.03	3.00	3	0	998.64	985.03	0.00	0.00	2572.44	551.65	4	21.33
RC208	GOEK	3	836.29	3.00	3	0	836.95	836.29	0.00	0.00	2456.21	560.86	5	23.29
AVG		**7.86**	**1069.50**	**7.99**	**7.93**	**0.07**	**1068.24**	**1066.47**	**−3.03**	**−0.19**	**5401.98**	**537.28**	**9.14**	**11.61**
SUM		**440**	**59,892.03**	**447.20**	**444**	**4.00**	**59,821.63**	**59,722.33**	**−169.70**	**−10.36**	**302,510.77**	**30,087.68**	**512**	**650.31**

To show the difference between the FR and PR strategies, one route is selected for the comparison. It is important to note, that the optimization procedure in most of the cases produces different customer and CS schedules for routes in different EVRPTW variants, which makes the direct comparison between routes non applicable. But here, there are some routes

that have similar customer schedule but differ in CS schedule. This happened as routes with FR and PR strategies on clustered instances overlap in grater extent than, for example, routes on random instances. The comparison is conducted for route 1—dark green route in EVRPTW-PR (Figure 4a) and black route in EVRPTW-FR (Figure 4b). The following rows are presented in Tables 5 and 6: user ID (ID), time window (TW), service time (ST), remaining load capacity (RL), arrival distance (AD), arrival time (AT), start time of service (BT), remaining battery capacity (RB), charging time (CT) and recharged capacity (RC). Color of the user's ID matches the color of the user's circle in Figure 4. It can be seen that the PR strategy has two visits to the CS 16, while the FR strategy has only one visit to CS 17. The fuel and time related variables differ between the variants, while the remaining load capacity variable has the same values. The arrival distance at the depot (vehicle total traveled distance) is lower for PR than for the FR strategy, as well as the total recharge amount ($19.03 + 3.22 < 61.71$). Although less time is spent on charging, the arrival time at the depot (vehicle total time) is actually higher for the PR strategy than the FR strategy. Usually, this is not the case, but here it happened as vehicle visits a CS immediately before the depot. To further show the difference PR and FR strategies, the route with the PR strategy is evaluated as a route with a FR strategy. The rows are presented at the end of the Table 6. It can be seen that there is no change at the first CS, as the vehicle is recharged to full capacity in both strategies. The only difference is in the second visited CS, where in the PR strategy, the vehicle is recharged by only a small amount needed to finish the route, while in the FR strategy, the vehicle is recharged to the fullest (80.17% of the full battery capacity), resulting in an infeasible arrival time at the depot (violation of the depot late time window). The evaluation of a route with a FR strategy as a route with PR strategy would produce the same solution as the FR strategy, as the almost whole spare time at CS 17 ($810 - (543.21 + 6.08) = 260.71 > 209.19$) can be used for charging to the fullest.

Table 5. Values in route 1—FR.

ID	0	80	81	79	77	74	17	75	76	78	0
TW	[0, 1236]	[66, 124]	[167, 223]	[256, 320]	[350, 410]	[437, 511]	[0, 1236]	[810, 868]	[905, 963]	[989, 1063]	[0, 1236]
ST	0	90	90	90	90	90	0	90	90	90	0
RL	200	190	170	140	110	90	90	50	40	0	0
AD	0	35.06	45.50	48.50	50.50	54.50	61.71	67.79	72.79	74.79	109.79
AT	0	35.06	166.44	260.00	352.00	446.00	543.21	758.49	905.00	997.00	1122.00
BT	0	66.00	167.00	260.00	352.00	446.00	543.21	810.00	905.00	997.00	1122.00
RB	79.69	44.63	34.19	31.19	29.19	25.19	17.98	73.61	68.61	66.61	31.61
CT	–	–	–	–	–	–	209.19	–	–	–	–
RC	–	–	–	–	–	–	61.71	–	–	–	–

Table 6. Values in route 1—PR.

| ID	0	16	80	81	79	77	74	75	76	78	16	0
TW	[0, 1236]	[0, 1236]	[66, 124]	[167, 223]	[256, 320]	[350, 410]	[437, 511]	[810, 868]	[905, 963]	[989, 1063]	[0, 1236]	[0, 1236]
ST	0	0	90	90	90	90	90	90	90	90	0	0
RL	200	200	190	170	140	110	90	50	40	0	0	0
AD	0	19.03	35.06	45.50	48.50	50.50	54.50	59.88	64.88	66.88	82.91	101.94
AT	0	19.03	99.56	200.00	293.00	385.00	479.00	574.38	905.00	997.00	1103.03	1132.99
BT	0	19.03	99.56	200.00	293.00	385.00	479.00	810.00	905.00	997.00	1103.03	1132.99
RB	79.69	60.66	63.66	53.22	50.22	48.22	44.22	38.83	33.83	31.83	15.80	0.00
CT	–	64.50	–	–	–	–	–	–	–	–	10.93	–
RC	–	19.03	–	–	–	–	–	–	–	–	3.22	–
						Evaluated as FR					
AT	0	19.03	99.56	200.00	293.00	385.00	479.00	574.38	905.00	997.00	1103.03	1338.64
BT	0	19.03	99.56	200.00	293.00	385.00	479.00	810.00	905.00	997.00	1103.03	1338.64
RB	79.69	60.66	63.66	53.22	50.22	48.22	44.22	38.83	33.83	31.83	15.80	60.66
CT	–	64.50	–	–	–	–	–	–	–	–	216.58	–
RC	–	19.03	–	–	–	–	–	–	–	–	63.89	–

(**a**) EVRPTW-PR

(**b**) EVRPTW-FR
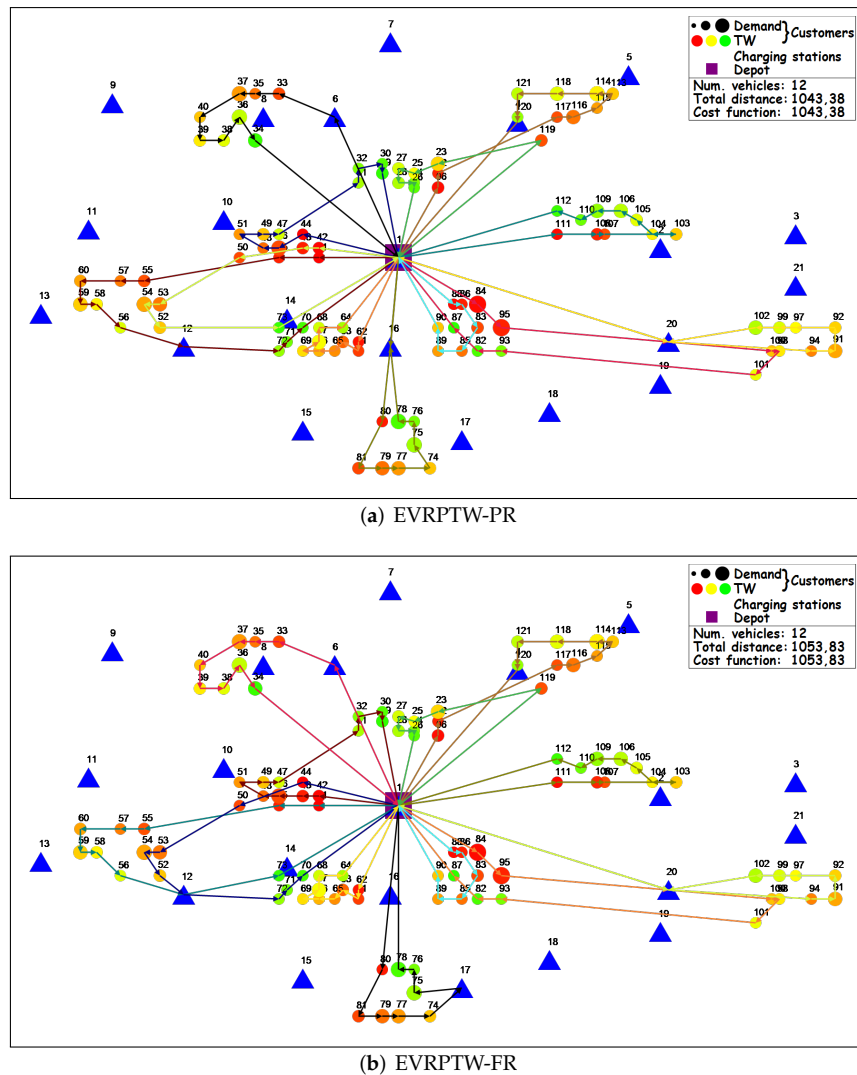
Figure 4. C101: BKSs.

4.3. Evaluation Test

As already highlighted, the evaluation of a change in the route with FR and PR strategy can be done in $\mathcal{O}(1)$ for most of the basic operators, but FR requires additional exact evaluation. As in the original research for ELRPTW-PR, the difference between exact and approximated violation values was not reported; here, a small test was conducted.

To validate the evaluation variables and operators, the testing was performed on instance *R*107 [20] with the total number of iterations set to $\mu_{max} = 500$. The results are presented in Table 7 as a difference between approximated $\mathcal{O}(1)$ evaluation and exact $\mathcal{O}(n)$ evaluation. The positive values represent an overestimation of penalties, while the negative values represent underestimation. Column *Feas/Infeas* represents either a completely feasible move or an infeasible move (evaluated exactly). It can be seen that operators for the PR strategy never underestimate violations as column *Min* in *Total* is zero in all cases. Although the time window and fuel by itself can be underestimated, their sum is not, meaning that the underestimation of time windows is compensated with fuel violation. On average, the penalty violations are overestimated by the value of 3.01, even for the completely feasible moves where the value is 1.56.

For the FR strategy, the mean total violations are even lower than in PR strategy, with the value of 1.89. For completely feasible moves, there was not an underestimation of total violation, and the mean value of violation is 2.69. As it can be seen, in the FR strategy, the underestimation of total violations occur up to -30.28, and also the overestimation

values are greater, almost doubled compared to the PR strategy. This is the main reason why the latter exact evaluation of the n_{LS} best moves was used in the LS procedure. Interestingly, the fuel violation is never underestimated.

Table 7. Evaluation.

Strategy	Feas/Infeas	Total			Time Window			Fuel		
		Mean	Min	Max	Mean	Min	Max	Mean	Min	Max
PR	Both	3.01	0.00	144.35	−0.25	−41.52	136.89	3.26	−25.43	48.24
	Feas	1.56	0.00	39.58	0.00	0.00	23.41	1.56	0.00	39.58
	Infeas	3.35	0.00	144.35	−0.31	−41.52	136.89	3.66	−25.43	48.25
FR	Both	1.89	−30.28	253.25	−4.44	−65.99	238.05	6.34	0.00	71.82
	Feas	2.69	0.00	55.16	0.20	0.00	32.35	2.49	0.00	42.21
	Infeas	1.78	−30.28	253.25	−5.09	−65.99	238.05	6.87	0.00	71.82

5. Discussion

The selection of the recharging strategy depends on the delivery problem characteristics and decision-makers. In the following paragraphs, the results of the paper regarding the used recharging strategy are summarized.

- With the PR strategy, greater savings can be achieved, but in real-life application, it is harder to implement as drivers need to visit CSs more frequently and perform time-precise charging with a high possibility of leaving a CS without a fully charged battery.
- Maintaining the energy reserve in the PR strategy to overcome the so-called range anxiety [15,49]—a fear that the vehicle range will not be enough to complete the designated route. The reason is that the amount of the energy recharged at the CS is usually only enough to reach the next CS or depot, which is not flexible enough for the application in real-life where the travel time is time-dependent [50], as well as waiting times that can occur at CSs which prolong delivery time [30].
- On the other hand, significant savings can be achieved by the PR strategy, as the total number of vehicles decreased by 4.05% and the total traveled distance decreased by 1.82% compared to the FR strategy. Additional savings could also be achieved by PR strategy as the amount of energy recharged during the day at public or private CSs is almost 30% less than the amount of energy recharged with the FR strategy. In the PR strategy, the minimal amount of energy could be recharged during the day when electricity costs are higher, and the rest of the energy could be recharged during the night.
- The PR strategy also helps to reduce the battery degradation and to prolong the battery life [51] as it escapes charging at a CSs with high SoC values (in numerical tests, it was shown that on average, the vehicle leaves a CS with the SoC value of 72.26%. Additionally, if a more realistic non-linear charging process that prolongs charging time after approx. 80% of the SoC value is considered [25], the FR strategy would lead to a higher number of infeasible solutions than the PR strategy.
- Comparison between BKSs on VRPTW instances and the best ALNS solutions for FR and PR variants indicates that integration of BEVs in the delivery processes increases the total vehicle number by at least 5.18% and the total traveled distance by at least 2.53%. Additional analysis should be conducted to validate if the reduction of transportation costs by BEVs can compensate for the increase in the vehicle number and total traveled distance.

There are several research directions left for future research. First, the evaluation of moves in the EVRPTW-FR could perhaps be done in $\mathcal{O}(1)$ without the need for additional $\mathcal{O}(n)$ evaluation, by the use of a bidirectional exact procedure [22]. The second direction is the extension of the proposed methodology on the EVRPTW problem with different charger types at CSs [16]. The third direction is the consideration of dynamic traffic conditions on the road network in terms of the time-dependent travel time [52], as well as different

secondary objectives: total time, travel time, recharging costs, recharging amount, and so forth.

6. Conclusions

In this paper, the delivery of goods with electric vehicles was observed through the solution of the EVRPTW problem, which aims to determine the minimum number of vehicles and the minimum total distance traveled needed to perform the delivery. Two recharging strategies were considered: FR and PR. For the creation of the initial solution, a new GTONNH heuristic is proposed, which produces a feasible solution with a relatively high number of vehicles due to the narrow feasible solution space. The route removal, local search and exact procedures were integrated within the ALNS framework to further intensify the search and lead to better solutions. The proposed ALNS metaheuristic uses the approximation of changes in the solution, which results in $\mathcal{O}(1)$ evaluation of moves for the PR strategy. The FR strategy also uses $\mathcal{O}(1)$ evaluation of moves, but additional exact $\mathcal{O}(n)$ evaluation is performed on n_{LS} best moves.

On 56 benchmark EVRPTW instances, the proposed ALNS metaheuristic was able to find 29 new BKSs for EVRPTW-PR and 9 BKSs for EVRPTW-FR. The PR strategy was able to reduce the total vehicle number, total distance traveled, total time and total recharging amount compared to the FR strategy, at the expense of the increase in the number visited CSs. The PR strategy turned out to be more robust regarding the battery degradation costs, non-linear charging function, and daily charging costs, but it has to have a high compliance rate to be efficient in the real-life application. Compared to the BKSs on VRPTW instances, which do not consider delivery with BEVs, the total number of vehicles increased by at least 5% in EVRPTW problems, and the additional cost-effectiveness of BEVs application for goods delivery should be investigated.

Author Contributions: Conceptualization, T.E. and T.C.; methodology, T.E.; software, T.E.; validation, T.E.; resources, T.C.; writing—original draft preparation, T.E.; writing—review and editing, T.E.; visualization, T.E.; supervision, T.C.; project administration, T.C.; funding acquisition, T.C. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: EVRPTW benchmark instances: https://data.mendeley.com/datasets/h3mrm5dhxw/1 (accessed on 9 December 2021); EVRPTW-FR/PR BKSs values found by the proposed ALNS metaheuristic https://www.fpz.unizg.hr/zits/?page_id=1424 (accessed on 9 December 2021).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

BEV	Battery electric vehicle
BKS	Best known solution
BSI	Best station insertion
CS	Charging station
ELRPTW-PR	Electric location routing problem with time windows and partial recharging
EV	Electric vehicle
EVRP	Electric vehicle routing problem
EVRPTW	Electric vehicle routing problem with time windows
EVRPTW-FR	Electric vehicle routing problem with time windows and full recharge
EVRPTW-PR	Electric vehicle routing problem with time windows and partial recharge
FR	Full recharge
GHG	Greenhouse gas
GTONNH	Greedy time-oriented nearest neighbor heuristic
ICEV	Internal combustion engine vehicle
LIFO	Last-in first-out
LS	Local search
MBCSI	Multiple best charging station insertion
MILP	Mixed integer linear program
NP	Non-deterministic polynomial
PHEV	Plug-in hybrid electric vehicle
PR	Partial recharge
SoC	State of charge
VRP	Vehicle routing problem
VRPTW	Vehicle routing problem with time windows

Appendix A

The used backward variables are presented in Table A1 and are computed by Equations (A1)–(A7). The backward time window and fuel violations are given by Equations (A8) and (A9). The initial values for backward variables related to the time of the last user in route u_{N+1} are set to the late time window, $\tilde{b}_{N+1}^{min} = \tilde{b}_{N+1}^{max} = b_{N+1}^{min} = b_{N+1}^{max} = l_{N+1}$, while the fuel related variable is set to zero, $b_{N+1}^{rt} = 0$.

Table A1. Backward variables—EVRPTW-PR.

Symbol	Description
b_i^{min}	Latest start time at user i with charging minimum amount as possible before i
b_i^{max}	Latest start time at user i with charging maximum amount possible at preceding CSs with the assumption that minimum charging was performed before preceding CS
b_i^{rt}	Time needed to charge to maximum at user i with charging minimum amount as possible before i
b_{ij}^{sl}	Slack time at user i
b_{ij}^{add}	Additional charging time at user i that has to be added at preceding CSs
$\tilde{b}_i^{min}$	Shifted latest start time at user i when violation occurs with charging minimum amount as possible before i
$\tilde{b}_i^{max}$	Shifted latest start time at user i when violation occurs with charging maximum amount possible at preceding CS with assumption that minimum charging was performed before preceding CS

$$b_{ij}^{sl} = \max\left(0, \tilde{b}_j^{min} - t_{ij} - s_i - l_i\right) \tag{A1}$$

$$b_i^{rt} = \begin{cases} \min\left(H, \max\left(0, b_j^{rt} - b_{ij}^{sl}\right) + h_{ij}\right) & j \in F' \\ \min\left(H, \max\left(0, b_j^{rt} - \min\left(b_{ij}^{sl}, \tilde{b}_j^{min} - \tilde{b}_j^{max}\right)\right) + h_{ij}\right) & else \end{cases} \tag{A2}$$

$$b_{ij}^{add} = \begin{cases} \max\left(0, \max\left(0, b_j^{rt} - b_{ij}^{sl}\right) + h_{ij} - H\right) & j \in F' \\ \max\left(0, \max\left(0, b_j^{rt} - \min\left(b_{ij}^{sl}, \tilde{b}_j^{min} - \tilde{b}_j^{max}\right)\right) + h_{ij} - H\right) & else \end{cases} \tag{A3}$$

$$b_i^{min} = \min\left(l_i, \tilde{b}_j^{min} - t_{ij} - s_i\right) - b_{ij}^{add} \tag{A4}$$

$$b_i^{max} = \begin{cases} \min\left(l_i, \tilde{b}_j^{min} - b_j^{rt} - t_{ij} - s_i\right) & j \in F' \\ \min\left(l_i, \tilde{b}_j^{max} - t_{ij} - s_i\right) & else \end{cases} \tag{A5}$$

$$\tilde{b}_i^{min} = \max(b_i^{min}, b_i^{max}, e_i) \tag{A6}$$

$$\tilde{b}_j^{max} = \max(e_j, b_j^{max}) \tag{A7}$$

$$\overleftarrow{TW}(\phi) = \sum_{u \in \phi} \max\left(e_u - \max\left(b_u^{min}, b_u^{max}\right), 0\right) \tag{A8}$$

$$\overleftarrow{FL}(\phi) = \sum_{u \in \phi} \max\left(b_u^{max} - b_u^{min}, 0\right) \tag{A9}$$

Appendix B

The selection probabilities of the operators change depending on their performance. The example of how probabilities change for used operators in EVRPTW-PR on instance C101 is presented in Figure A1. The update of operator probabilities occurs every $\mu_{uop} = 50$ iterations. As it can be seen, different operators are preferred in different stages of the search. The related and CS vicinity destroy operators show lower probabilities than worst and Shaw destroy operators. The probability of related operator increased in the last phase, showing its benefits in the cases when the search is near the best found local optima. The sequential insertion operator is preferred in the early stages of the search when more frequently, better solutions are found. But later on, when the search gets stuck in the local optima, the perturbed variant shows its benefits.

The example of the ALNS heuristic on the C101 instance and EVRPTW-FR problem is presented in Figure A2. The vehicle number remained the same, while the distance traveled decreased from 1085.55 before removal to 1053.88 after insertion. The removed customers are presented with red dashed circles.
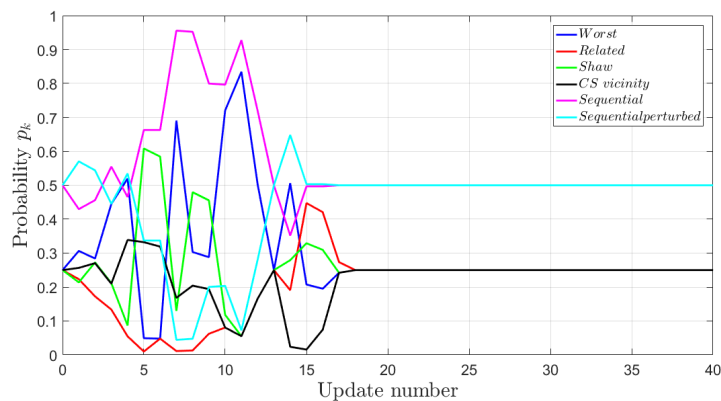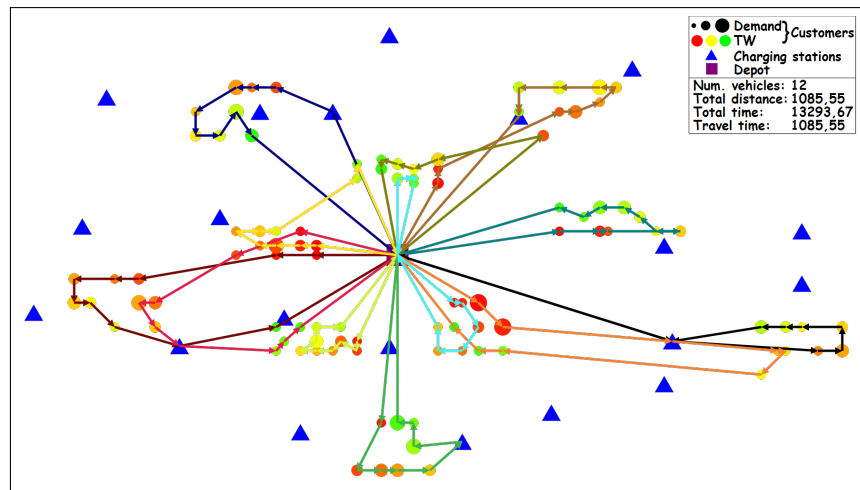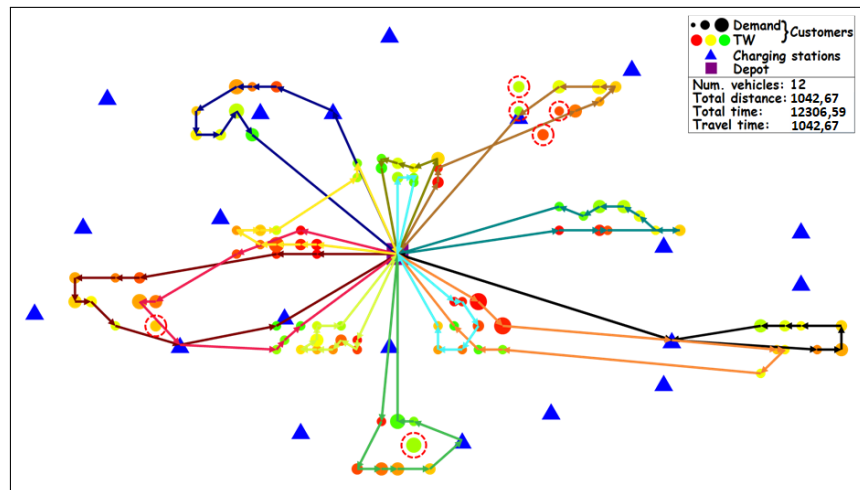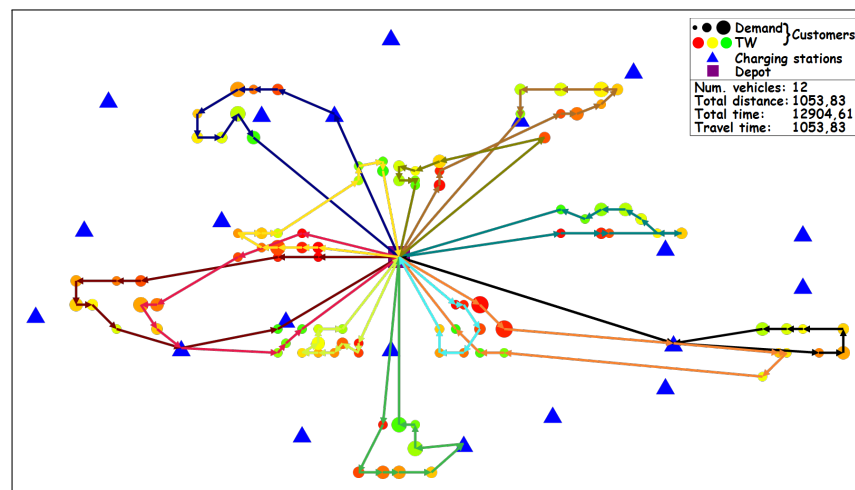

Figure A1. Example of operator probabilities: C101 -EVRPTW-PR.

(**a**) Before removal

(**b**) After removal

(**c**) After insertion

Figure A2. ALNS example: C101—EVRPTW-FR.

References

1. Schneider, M.; Stenger, A.; Goeke, D. The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations. *Transp. Sci.* **2014**, *48*, 500–520. [CrossRef]
2. Vidal, T.; Crainic, T.G.; Gendreau, M.; Prins, C. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *Eur. J. Oper. Res.* **2013**, *231*, 1–21. [CrossRef]
3. Toth, P.; Vigo, D. (Eds.) *The Vehicle Routing Problem*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2001.
4. WEEA. Greenhouse Gas Data—Emissions Share by Sector in EU28. 2015. Available online: http://www.eea.europa.eu/data-and-maps/data/data-viewers/greenhouse-gases-viewer (accessed on 31 August 2020).
5. European Commission. Effort Sharing: Member States' Emission Targets. 2009. Available online: https://ec.europa.eu/clima/policies/effort_en (accessed on 7 February 2021).
6. Davis, B.A.; Figliozzi, M.A. A methodology to evaluate the competitiveness of electric delivery trucks. *Transp. Res. Part E Logist. Transp. Rev.* **2013**, *49*, 8–23. [CrossRef]
7. Erdelić, T.; Carić, T. A Survey on the Electric Vehicle Routing Problem: Variants and Solution Approaches. *J. Adv. Transp.* **2019**, *2019*, 1–48. [CrossRef]
8. Moon, S.; Lee, D.J. An optimal electric vehicle investment model for consumers using total cost of ownership: A real option approach. *Appl. Energy* **2019**, *253*, 113494. [CrossRef]
9. Feng, W.; Figliozzi, M. An economic and technological analysis of the key factors affecting the competitiveness of electric commercial vehicles: A case study from the USA market. *Transp. Res. Part C Emerg. Technol.* **2013**, *26*, 135–145. [CrossRef]
10. Young, K.; Wang, C.; Wang, L.Y.; Strunz, K. Electric Vehicle Battery Technologies. In *Electric Vehicle Integration into Modern Power Networks*; Garcia-Valle, R., Peças Lopes, J.A., Eds.; Springer: New York, NY, USA, 2013; pp. 15–56._2. [CrossRef]
11. Wróblewski, P.; Lewicki, W. A Method of Analyzing the Residual Values of Low-Emission Vehicles Based on a Selected Expert Method Taking into Account Stochastic Operational Parameters. *Energies* **2021**, *14*, 6859. [CrossRef]
12. Hao, X.; Lin, Z.; Wang, H.; Ou, S.; Ouyang, M. Range cost-effectiveness of plug-in electric vehicle for heterogeneous consumers: An expanded total ownership cost approach. *Appl. Energy* **2020**, *275*, 115394. [CrossRef]
13. Bräysy, O.; Gendreau, M. Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms. *Transp. Sci.* **2005**, *39*, 104–118. [CrossRef]
14. Macrina, G.; Pugliese, L.D.P.; Guerriero, F.; Laporte, G. The green mixed fleet vehicle routing problem with partial battery recharging and time windows. *Comput. Oper. Res.* **2019**, *101*, 183–199. [CrossRef]
15. Barco, J.; Guerra, A.; Muñoz, L.; Quijano, N. Optimal Routing and Scheduling of Charge for Electric Vehicles: Case Study. *Math. Probl. Eng.* **2013**, *2017*, 8509783. [CrossRef]
16. Keskin, M.; Çatay, B. A matheuristic method for the electric vehicle routing problem with time windows and fast chargers. *Comput. Oper. Res.* **2018**, *100*, 172–188. [CrossRef]
17. Goeke, D.; Schneider, M. Routing a mixed fleet of electric and conventional vehicles. *Eur. J. Oper. Res.* **2015**, *245*, 81–99. [CrossRef]
18. Hiermann, G.; Puchinger, J.; Ropke, S.; Hartl, R.F. The Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations. *Eur. J. Oper. Res.* **2016**, *252*, 995–1018. [CrossRef]
19. Hiermann, G.; Hartl, R.F.; Puchinger, J.; Vidal, T. Routing a mix of conventional, plug-in hybrid, and electric vehicles. *Eur. J. Oper. Res.* **2019**, *272*, 235–248. [CrossRef]
20. Keskin, M.; Çatay, B. Partial recharge strategies for the electric vehicle routing problem with time windows. *Transp. Res. Part C Emerg. Technol.* **2016**, *65*, 111–127. [CrossRef]
21. Schiffer, M.; Walther, G. An Adaptive Large Neighborhood Search for the Location-routing Problem with Intra-route Facilities. *Transp. Sci.* **2018**, *52*, 331–352. [CrossRef]
22. Desaulniers, G.; Errico, F.; Irnich, S.; Schneider, M. Exact Algorithms for Electric Vehicle-Routing Problems with Time Windows. *Oper. Res.* **2016**, *64*, 1388–1405. [CrossRef]
23. Felipe, Á.; Ortuño, M.T.; Righini, G.; Tirado, G. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transp. Res. Part E Logist. Transp. Rev.* **2014**, *71*, 111–128. [CrossRef]
24. Masmoudi, M.A.; Hosny, M.; Demir, E.; Genikomsakis, K.N.; Cheikhrouhou, N. The dial-a-ride problem with electric vehicles and battery swapping stations. *Transp. Res. Part E Logist. Transp. Rev.* **2018**, *118*, 392–420. [CrossRef]
25. Montoya, A.; Guéret, C.; Mendoza, J.E.; Villegas, J.G. The electric vehicle routing problem with nonlinear charging function. *Transp. Res. Part B Methodol.* **2017**, *103*, 87–110. [CrossRef]
26. Froger, A.; Mendoza, J.E.; Jabali, O.; Laporte, G. Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions. *Comput. Oper. Res.* **2019**, *104*, 256–294. [CrossRef]
27. Yang, J.; Sun, H. Battery swap station location-routing problem with capacitated electric vehicles. *Comput. Oper. Res.* **2015**, *55*, 217–232. [CrossRef]
28. Hof, J.; Schneider, M.; Goeke, D. Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops. *Transp. Res. Part B Methodol.* **2017**, *97*, 102–112. [CrossRef]
29. Schiffer, M.; Walther, G. The electric location routing problem with time windows and partial recharging. *Eur. J. Oper. Res.* **2017**, *260*, 995–1013. [CrossRef]

30. Keskin, M.; Laporte, G.; Çatay, B. Electric Vehicle Routing Problem with Time-Dependent Waiting Times at Recharging Stations. *Comput. Oper. Res.* **2019**, *107*, 77–94. [CrossRef]

31. Kullman, N.D.; Goodson, J.; Mendoza, J.E. Dynamic Electric Vehicle Routing: Heuristics and Dual Bounds; HAL. 2018. Working Paper or Preprint. Available online: https://hal.archives-ouvertes.fr/hal-01928730 (accessed on 20 December 2021).

32. Schiffer, M.; Schneider, M.; Walther, G.; Laporte, G. Vehicle Routing and Location Routing with Intermediate Stops: A Review. *Transp. Sci.* **2019**, *53*, 319–343. [CrossRef]

33. Demir, E.; Bektaş, T.; Laporte, G. An adaptive large neighborhood search heuristic for the Pollution-Routing Problem. *Eur. J. Oper. Res.* **2012**, *223*, 346–359. [CrossRef]

34. Ropke, S.; Pisinger, D. An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transp. Sci.* **2006**, *40*, 455–472. [CrossRef]

35. Sassi, O.; Cherif-Khettaf, W.R.; Oulamara, A. Iterated Tabu Search for the Mix Fleet Vehicle Routing Problem with Heterogenous Electric Vehicles. In *Modelling, Computation and Optimization in Information Systems and Management Sciences*; Le Thi, H.A., Pham Dinh, T., Nguyen, N.T., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 57–68.

36. Schiffer, M.; Walther, G. Strategic planning of electric logistics fleet networks: A robust location-routing approach. *Omega* **2018**, *80*, 31–42. [CrossRef]

37. Nagata, Y.; Bräysy, O.; Dullaert, W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. *Comput. Oper. Res.* **2010**, *37*, 724–737. [CrossRef]

38. Schneider, M.; Sand, B.; Stenger, A. A note on the time travel approach for handling time windows in vehicle routing problems. *Comput. Oper. Res.* **2013**, *40*, 2564–2568. [CrossRef]

39. Erdoğan, S.; Miller-Hooks, E. A Green Vehicle Routing Problem. *Transp. Res. Part E Logist. Transp. Rev.* **2012**, *48*, 100–114. [CrossRef]

40. Solomon, M.M. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Oper. Res.* **1987**, *35*, 254–265. [CrossRef]

41. Vidal, T.; Crainic, T.G.; Gendreau, M.; Prins, C. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Comput. Oper. Res.* **2013**, *40*, 475–489. [CrossRef]

42. Kindervater, G.A.P.; Savelsbergh, M.W.P. Vehicle routing: Handling edge exchanges. In *Local Search in Combinatorial Optimization*; Aarts, E., Lenstra, J.K., Eds.; Princeton University Press: Princeton, NJ, USA, 2018; pp. 337–360. [CrossRef]

43. Pisinger, D.; Ropke, S. A general heuristic for vehicle routing problems. *Comput. Oper. Res.* **2007**, *34*, 2403–2435. [CrossRef]

44. Shaw, P. Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In *Principles and Practice of Constraint Programming—CP98*; Maher, M., Puget, J.F., Eds.; Springer: Berlin/Heidelberg, Germany, 1998; pp. 417–431.

45. Osman, I.H. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Ann. Oper. Res.* **1993**, *41*, 421–451. [CrossRef]

46. Savelsbergh, M.W.P. The Vehicle Routing Problem with Time Windows: Minimizing Route Duration. *ORSA J. Comput.* **1992**, *4*, 146–154. [CrossRef]

47. Or, I. *Traveling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Regional Blood Banking*; Northwestern University: Evanston, IL, USA, 1976.

48. Potvin, J.Y.; Rousseau, J.M. An Exchange Heuristic for Routeing Problems with Time Windows. *J. Oper. Res. Soc.* **1995**, *46*, 1433–1446. [CrossRef]

49. Sweda, T.M.; Dolinskaya, I.S.; Klabjan, D. Adaptive Routing and Recharging Policies for Electric Vehicles. *Transp. Sci.* **2017**, *51*, 1326–1348. [CrossRef]

50. Figliozzi, A.M. The time dependent vehicle routing problem with time windows: Benchmark problems, an efficient solution algorithm, and solution characteristics. *Transp. Res. Part E Logist. Transp. Rev.* **2012**, *48*, 616–636. [CrossRef]

51. Pelletier, S.; Jabali, O.; Laporte, G.; Veneroni, M. Battery degradation and behaviour for electric vehicles: Review and numerical analyses of several models. *Transp. Res. Part B Methodol.* **2017**, *103*, 158–187. [CrossRef]

52. Erdelić, T.; Carić, T.; Erdelić, M.; Tišljarić, L.; Turković, A.; Jelušić, N. Estimating congestion zones and travel time indexes based on the floating car data. *Comput. Environ. Urban Syst.* **2021**, *87*, 101604. [CrossRef]