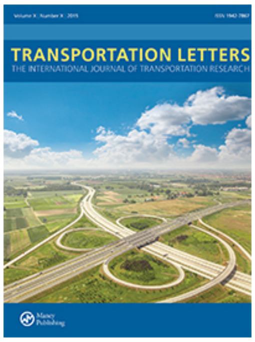

Transportation Letters

The International Journal of Transportation Research

ISSN: 1942-7867 (Print) 1942-7875 (Online) Journal homepage: https://www.tandfonline.com/loi/ytrl20

Electric vehicle routing with flexible time windows: a column generation solution approach

Duygu Taş

TRANSPORTATION LETTERS
https://doi.org/10.1080/19427867.2020.1711581

Taylor & Francis
Taylor & Francis Group

Electric vehicle routing with flexible time windows: a column generation solution approach

Duygu Taş

MEF University, Istanbul, Turkey

ABSTRACT
In this paper, we introduce the Electric Vehicle Routing Problem with Flexible Time Windows (EVRPFTW) in which vehicles are allowed to serve customers before and after the earliest and latest time window bounds, respectively. The objective of this problem is to assign electric vehicles to feasible routes and make schedules with minimum total cost that includes the traveling costs, the costs of using electric vehicles and the penalty costs incurred for earliness and lateness. The proposed mathematical model is solved by a column generation procedure. To generate an integer solution, we solve an integer programming problem using the routes constructed by the column generation algorithm. We further develop a linear programming model to compute the optimal times to start service at each customer for the selected routes. A number of well-known benchmark instances is solved by our solution procedure to evaluate the operational gains obtained by employing flexible time windows.

KEYWORDS
Routing; electric vehicles; time windows; column generation

Introduction

The Electric Vehicle Routing Problem with Time Windows (EVRPTW) is introduced by Schneider, Stenger, and Goeke (2014) as an extension of the classical Vehicle Routing Problem (VRP). The VRP aims to find a set of least cost routes such that each vehicle starts and ends its route at the depot, each customer is served exactly once by one vehicle, and the total demand of the customers visited by a vehicle cannot exceed the vehicle capacity. Several versions of the VRP have been studied in the literature, and some recent extensions focus on including a number of depots with heterogeneous fleet of vehicles (Azadeh and Farrokhi-Asl 2018), employing reliable link travel times (Musolino, Polimeni, and Vitetta 2018), and using mixed load plan specifically for school bus routing (Yao et al. 2016). Carrier companies daily deal with the transportation problem of delivering goods to customers in a cost-effective manner, where they still need to pay sufficient attention to customer service requirements. In such environments, customer service requirements are mainly reflected by the Vehicle Routing Problem with Time Windows (VRPTW). In the VRPTW, each customer has one single time interval that restricts the delivery time. This problem is usually formulated as a multicommodity network flow problem with capacity and time window constraints (Cordeau et al. 2002). The EVRPTW extends this classical problem by using a fleet of Electric Vehicles (EVs) to serve customers. The EVs have short driving ranges and thus may require to visit stations to recharge their batteries along the route. The recharging time depends on the battery level on arrival at a station and it is assumed that the battery is full after the recharge takes place.

In the classical VRPTW and EVRPTW, time windows are defined as hard constraints. However, in real-life applications, these constraints can be violated to a certain extent. The relaxation of the time window constraints may lead to operational gains such as reduction in the number of EVs used and/or reduction in the total distance traveled. In this paper, we evaluate such gains considering the Electric Vehicle Routing Problem with Flexible Time Windows (EVRPFTW). The interested reader is referred to Taş, Jabali, and van Woensel (2014b) for a vehicle routing problem with flexible time windows and for a solution approach based on tabu search method. Flexible time windows allow vehicles to serve customers outside their time windows with respect to a given tolerance. However, early and late services bring penalty costs since they have an impact on the customer satisfaction. Therefore, the objective of the EVRPFTW is to minimize the total cost which consists of the classical routing costs plus the penalty costs incurred due to time window violations. The main contributions of this paper are threefold:

(1) First, we introduce and model the EVRPFTW. To the best of our knowledge, no research has addressed this problem.
(2) Second, we propose a solution approach in which routes are obtained by a column generation procedure and further improved by a linear programming model which optimizes the penalty costs of given routes. This solution approach is combined with an integer programming model to generate integer solutions.
(3) Third, we solve a number of problem instances by our solution approach to evaluate the effects of employing flexible time windows.

The remainder of this paper is organized as follows. The problem and the formulation proposed in this paper are presented in Section 2. The column generation procedure is explained in Section 3 and the pricing subproblem with a dominance relation is described in Section 4. The scheduling method obtaining the optimal start time of service at each customer for each route is provided in Section 5. This is followed by computational results presented in Section 6. Finally, we end the paper with conclusions and with suggestions for future research.

CONTACT Duygu Taş duygu.tas@mef.edu.tr MEF University, Huzur Mah. Maslak Ayazağa Cad. No.4, Saryer, Istanbul, Turkey

Problem description and formulation

Let $V = \{1, ..., n\}$ be the set of customers, F' be the set of dummy nodes created to allow several visits to each node in the set F of recharging stations, and $V' = V \cup F'$. Nodes 0 and $n+1$ correspond to the central depot, and each vehicle route originates at node 0 and ends at node $n+1$. In case a set includes the respective instance of the depot, it is then subscripted with the corresponding index, e.g., $V'_0 = V' \cup \{0\}$. The EVRPFTW can be defined on a connected digraph $G = (V'_{0,n+1}, A)$ where $A = \{(i,j)|i,j \in V'_{0,n+1}, i{\neq}j\}$ is the set of arcs. A distance d_{ij} and a travel time t_{ij} are defined for each arc $(i,j) \in A$. It is assumed that traversing the arc (i,j) consumes $h.d_{ij}$ of the remaining battery charge where h corresponds to the battery charge consumption rate. With each node $i \in V'_{0,n+1}$, is associated a demand q_i, a service time s_i, a hard time window $[l_i, u_i]$, and a fraction f_i which represents the maximum allowed time window violation. Note that the parameters defined for each node are non-negative ($q_i, s_i, l_i, u_i, f_i \geq 0$). For each node i, a flexible time window $[l'_i, u'_i]$ is then generated where $l'_i = l_i - f_i(u_i - l_i)$, $u'_i = u_i + f_i(u_i - l_i)$, and $l'_i, u'_i \geq 0$. Moreover, a homogeneous fleet of EVs of equal load capacity (C) and equal battery capacity (Q) is located at the depot. At a recharging station, the battery of each EV is recharged with a rate of g. In other words, the required recharging time depends on the remaining charge level of the EV upon arrival at the recharging station.

The coefficients c_e and c_d are the penalty costs paid for one unit of earliness and for one unit of delay, respectively. More specifically, if an EV serves a customer between l'_i and l_i (early service) or between u_i and u'_i (late service), then a penalty proportional to the time window violation is accounted for in the total cost. EVs wait at customers at least until the flexible time window is reached if they arrive early (with no cost) and they cannot serve after the customer flexible time window closes. Moreover, a fixed cost c_f is incurred for using an EV.

In the following formulation, τ_i denotes the start time of service at node i, and z_i and y_i specify the remaining load capacity and remaining charge level on arrival at node i, respectively. The decision variable x_{ij} is equal to 1 if node j is visited immediately after node i. Furthermore, e_i and k_i represent the earliness and delay at customer i, respectively.

Based on the above definition, the EVRPFTW is modeled as the formulation (1)–(16). In this model, the objective function (1) minimizes the total cost that consists of fixed costs of vehicles used, traveling costs, and penalty costs incurred for early and late service. Constraints (2) ensure that each customer is visited exactly once. Constraints (3) handle the connectivity of visits to recharging stations. Constraints (4) guarantee the conservation of flow at each customer and at each recharging station. Constraints (5) and (6) describe the relationship between the departure time of vehicle from a node and the starting time of service at its successor. More specifically, customers are considered in constraints (5), and recharging stations are considered in constraints (6). Constraints (7) ensure that each customer is served within its flexible time window. Note that, constraints (5)–(7) further eliminate subtours. Constraints (8) and (9) satisfy the demand of each customer with respect to the vehicle load capacity. Constraints (10) and (11) ensure that the battery charge level never takes negative values. Constraints (12) link the start time of service and the earliness, and constraints (13) link the start time of service and the delay. Constraints (14) and (15) ensure that earliness and delay take non-negative values. Finally, constraints (16) indicate that partial service is not allowed.

$$\min \quad c_f \sum_{j \in V'} x_{0j} + \sum_{i \in V'_0, j \in V'_{n+1}, i \neq j} d_{ij} x_{ij} + c_e \sum_{i \in V'_{0,n+1}} e_i + c_d \sum_{i \in V'_{0,n+1}} k_i \tag{1}$$

$$\text{s.t.} \sum_{j \in V'_{n+1}, i \neq j} x_{ij} = 1, \qquad i \in V, \tag{2}$$

$$\sum_{j \in V'_{n+1}, i \neq j} x_{ij} \leq 1, \qquad i \in F', \tag{3}$$

$$\sum_{i \in V'_{n+1}, i \neq j} x_{ji} - \sum_{i \in V'_0, i \neq j} x_{ij} = 0, \qquad j \in V', \tag{4}$$

$$\tau_i + (t_{ij} + s_i) x_{ij} - u'_0 (1 - x_{ij}) \leq \tau_j, \quad i \in V_0, j \in V'_{n+1}, i{\neq}j, \tag{5}$$

$$\tau_i + t_{ij} x_{ij} + g(Q - y_i) - (u'_0 + gQ)(1 - x_{ij}) \leq \tau_j, \quad i \in F', j \in V_n + 1', i{\neq}j, \tag{6}$$

$$l'_i \leq \tau_i \leq u'_i, \qquad i \in V'_{0,n+1}, \tag{7}$$

$$0 \leq z_j \leq z_i - q_i x_{ij} + C(1 - x_{ij}), \qquad i \in V'_0, j \in V'_{n+1}, i{\neq}j, \tag{8}$$

$$0 \leq z_0 \leq C, \tag{9}$$

$$0 \leq y_j \leq y_i - (h.d_{ij}) x_{ij} + Q(1 - x_{ij}), \quad i \in V, j \in V_n + 1', i{\neq}j, \tag{10}$$

$$0 \leq y_j \leq Q - (h.d_{ij}) x_{ij}, \qquad i \in F'_0, j \in V'_{n+1}, i{\neq}j, \tag{11}$$

$$e_i \geq l_i - \tau_i, \qquad i \in V'_{0,n+1}, \tag{12}$$

$$k_i \geq \tau_i - u_i, \qquad i \in V'_{0,n+1}, \tag{13}$$

$$e_i \geq 0, \qquad i \in V'_{0,n+1}, \tag{14}$$

$$k_i \geq 0, \qquad i \in V'_{0,n+1}, \tag{15}$$

$$x_{ij} \in \{0, 1\}, \qquad i \in V'_0, j \in V'_{n+1}, i{\neq}j. \tag{16}$$

Column generation

For EVRPTW extensions, the interested reader is referred to Desaulniers et al. (2016) for exact algorithms considering a number of full and partial recharging strategies, to Keskin and Çatay (2016) for an Adaptive Large Neighborhood Search (ALNS) method allowing partial recharging, and to Bruglieri et al. (2017) for a three-phase matheuristic method developed for a similar problem with partial recharging. Recently, Keskin and Çatay (2018) extend the EVRPTW with partial recharging by considering the fast charging option, and propose a solution methodology combining an

ALNS algorithm with an exact method. Montoya et al. (2017) study nonlinear recharging functions which are approximated by piece-wise linear modeling, and develop a hybrid metaheuristic to effectively solve this problem.

As presented above, most research given in the literature for the EVRPTW has focused on the recharging options and the resulting effects. In this paper, we consider a new version of the EVRPTW, which is formally described in Section 2. In this version, time windows are relaxed by a given tolerance and vehicles are fully charged when a recharging station is visited. The latter assumption allows us to evaluate the operational advantages of the EVRPFTW, such as reduction in the number of EVs activated or reduction in the total distance traveled, compared to the original problem. To solve the EVRPFTW, we develop a heuristic solution procedure based on the column generation algorithm. In the following, we present the master problem and the pricing subproblem of the column generation method proposed for the formulation (1)–(16).

The master problem

The master problem corresponding to constraints (2) can be formulated as a set partitioning problem as follows:

$$\min \sum_{p\in P} c_p y_p \tag{17}$$

$$\text{subject to} \quad \sum_{p\in P} a_{ip} y_p = 1, \qquad i \in V, \tag{18}$$

$$y_p \in \{0,1\}, \qquad p \in P. \tag{19}$$

In the above model, P denotes the set of all feasible routes starting from node 0 and ending at node $n+1$. c_p is the total cost of route p including the fixed cost of the EV used, the total traveling cost, and the total penalty cost incurred for early and late service along that route. a_{ip} is equal to 1 if customer i is visited by route p and 0, otherwise. The decision variable y_p takes the value 1 if route p is selected by the solution and 0, otherwise.

The pricing subproblem

The pricing subproblem for each EV corresponds to constraints (3)–(16) in the formulation given for the EVRPFTW. This subproblem is an Elementary Shortest Path Problem with Resource Constraints (ESPPRC) and formulated as follows:

$$\min \quad \bar{c}_p \tag{20}$$

$$\text{subject to } (3) - (16). \tag{21}$$

In the above model, $\bar{c}_p$ denotes the reduced cost of route p and it is computed by:

$$\bar{c}_p = c_p - \sum_{i\in V} a_{ip} v_i, \tag{22}$$

where v_i, $i \in V$ is the dual value corresponding to the set of constraints (18).

The column generation algorithm starts with solving a Restricted Linear Programming Master Problem (RLPMP). More specifically, constraints (19) are relaxed in the RLPMP and a subset of feasible vehicle routes is considered. The optimal dual values obtained by solving the RLPMP are then employed in the pricing subproblem, and the columns with negative reduced costs are added to the RLPMP. Next, the RLPMP is reoptimized to obtain new optimal dual values. The column generation algorithm terminates in case all columns generated by the pricing subproblem have non-negative reduced costs, which indicates that the optimal solution of the linear programming relaxation of the formulation (17)–(19) is obtained.

To generate feasible routes for the first step of the algorithm, we employ an initialization procedure based on the time-oriented nearest neighbor heuristic presented by Solomon (1987). This method searches the set of feasible customers that are not visited yet to find the closest customer to the last node in the current route. The closeness is defined by a function using three measures. The interested reader is referred to Solomon (1987) for the details about these metrics. The feasibility is checked with respect to load and battery capacities of EVs, and the flexible time windows at customers and at the ending depot. In case the battery level is not sufficient to arrive at the ending depot after serving the selected customer i, the station j^*, where $d_{ij^*} = \min_{j\in F}\{d_{ij}\}$, is then visited to recharge the battery. Note that the customer is considered as feasible (in terms of the battery capacity) if the remaining capacity after serving that customer is enough to visit its closest station. In case the procedure cannot find a feasible customer for the current route, a new partial route is then started by visiting the customer that is closest to the starting depot. This procedure continues until all customers are assigned to an EV.

Elementary shortest path problem with resource constraints

To effectively solve the pricing subproblem of the column generation, we apply the algorithm of Feillet et al. (2004) with the decremental state space relaxation method of Boland, Dethridge, and Dumitrescu (2006), and Righini and Salani (2008). The algorithm developed by Feillet et al. (2004) is based on the label correcting reaching method of Desrochers (1988), which has been proposed for the non-elementary shortest path problem with resource constraints. This method employs labels to represent the paths on each node. In other words, each label on a node denotes a partial path originating from the starting depot and arriving at that node, and specifies the cost of the path with the consumption of the resources along the path. The label correcting reaching algorithm iteratively extends each new label to each feasible successor node, and terminates when new labels cannot be generated. To adjust this algorithm to ESSPRC, Feillet et al. (2004) include a binary resource for each node and further define unreachable nodes since the problem is solved on a full-dimensional state space. More specifically, a node is unreachable for a path since either it has already been served by that path or it cannot be visited due to the violation of at least one resource.

The decremental state space augmentation technique forbids multiple visits only for the customers in a set S, where $S \subseteq V$. If the optimal solution of this relaxed problem is elementary, it is also optimal for the ESPPRC. Otherwise, some of the customers which have been visited several times in the optimal solution of the relaxed problem is added into the set S, and the problem is solved again.

In our pricing subproblem, a partial path p from node 0 to node i is represented by a label $L_p = (\bar{c}_p, W_p^l, W_p^r, v_p, \tau_p, a_p^S, \mathbf{V}_p^S)$, where the label components are given as follows:

- $\bar{c}_p$: reduced cost of path p,
- W_p^l: consumption of the load capacity resource along path p (total demand of the customers visited by path p),
- W_p^r: consumption of the battery capacity resource since the last recharge along path p,

- v_p: the arrival time of path p at node i,
- τ_p: the start time of service of path p at node i,
- a_p^S: the number of nodes in S that are unreachable by path p,
- $\mathbf{V}_p^S$: the vector of unreachable nodes in S, which is defined by $V_p^b = 1$ if node $b \in S$ is unreachable by path p and 0, otherwise.

Note that v_p is needed both to check the time window constraints and to compute the start time of service at node i. The latter, τ_p, is then employed to obtain the earliness and lateness penalties, which are included by $\bar{c}_p$. τ_p is also used to compute the arrival time of the path at a successor node j when it is extended (see Algorithm 1).

Let p^* and p be two distinct paths from node 0 to node i. The dominance relation is then defined as follows. With respect to this relation, dominated labels are discarded and a path is called efficient if its corresponding label is non-dominated.

Algorithm 1. Extend(i, L_p, j)

```
if (W^l_p + q_j > C) or (W^r_p + h.d_ij > Q) or (τ_p + t_ij + s_i > u'_j) then
|  return FALSE
else
|  compute W^l_p', W^r_p', v_p, τ_p' and c̄_p'
|  a^S_p' ← a^S_p
|  V^S_p' ← V^S_p
|  if j ∈ S then
|  |  a^S_p' ← a^S_p' + 1
|  |  V^j_p' ← 1
|  end
|  foreach o ∈ S and (j, o) ∈ A such that (W^l_p' + q_o > C) or
|  (W^r_p' + h.d_jo > Q) or (τ_p' + t_jo + s_j > u'_o) do
|  |  a^S_p' ← a^S_p' + 1
|  |  V^o_p' ← 1
|  end
|  return L_p'
end
```

Definition 4.1. *If p^* and p are two distinct paths from node 0 to node i with labels L_{p^*} and L_p, then path p^* dominates path p if and only if $\bar{c}_{p^*} \le \bar{c}_p$, $W^l_{p^*} \le W^l_p$, $W^r_{p^*} \le W^r_p$, $a^S_{p^*} \le a^S_p$ and $V^o_{p^*} \le V^o_p$ for all $o \in S$, and at least one of these inequalities is strict.*

The above definition indicates that path p^* dominates path p if (i) it is less costly, (ii) it consumes fewer resources for each resource, and (iii) each unreachable customer is also unreachable for path p.

The extension of a label at node i to node j is defined in Algorithm 1. In this algorithm, $L_{p'}$ represents the resulting label obtained by extending label L_p from node i to node j. This algorithm first updates the consumption of the vehicle capacity and the consumption of the battery capacity. If node j is a recharging station, $W^r_{p'}$ takes the value of 0 since vehicles are fully charged at stations. The required recharging time is then considered in the computation of the arrival time, and thus in the computation of the start time of service at successor nodes. If all resource constraints (load capacity, battery capacity and time windows) are satisfied, the reduced cost of the resulting label is then computed. Furthermore, the number of unreachable nodes and the vector of unreachable nodes are updated with respect to node j and its possible successor nodes.

In our pricing subproblem, when efficient paths on the ending depot are determined, the elementary ones with non-negative reduced costs and dominated ones with negative or non-negative reduced costs are stored in a column pool. After the RLPMP is reoptimized and new optimal dual values are obtained, we first search the column pool to find columns with negative reduced costs. We solve the ESPPRC if such columns cannot be found. At each iteration, we check the size of the column pool to make the search as efficiently as possible. More specifically, if the number of columns in the pool is larger than a predetermined value, the columns that have been kept for more than a threshold number of iterations are then removed from the column pool. As an accelerating technique, we further terminate the algorithm in case the number of efficient elementary paths with negative reduced costs on the ending depot is larger than a predetermined value (see Feillet et al. 2004; Feillet, Gendreau, and Rousseau 2007; Taş et al. 2014a, for similar implementations of this technique). If column generation provides us with a fractional solution at termination, we then obtain an integer feasible solution by solving an integer programming problem over all columns included by the final RLPMP.

Scheduling method

In the column generation algorithm, service takes place immediately if a vehicle arrives at a customer within its flexible time window. For each route in the final solution obtained by the column generation algorithm, we solve the following linear programming model to find the optimal start time of service at each customer. In the following model, $N' \subseteq V'$ is the set of nodes visited in the route of the considered EV. Moreover, the parameter b_{ij} takes the value 1 if arc (i, j) is traversed by that EV and 0, otherwise.

$$\min \quad c_e \sum_{i \in N'_{0,n+1}} e_i + c_d \sum_{i \in N'_{0,n+1}} k_i \tag{23}$$

$$\text{s.t.} \quad \tau_i + (t_{ij} + s_i) b_{ij} - u'_0 (1 - b_{ij}) \le \tau_j, \quad i \in N_0, j \in N'_{n+1}, i \neq j, \tag{24}$$

$$\tau_i + t_{ij} b_{ij} + g(Q - y_i) - (u'_0 + gQ)(1 - b_{ij}) \le \tau_j, \quad i \in F', j \in N'_{n+1}, i \neq j, \tag{25}$$

$$l'_i \le \tau_i \le u'_i, \qquad i \in N'_{0,n+1}, \tag{26}$$

$$e_i \ge l_i - \tau_i, \qquad i \in N'_{0,n+1}, \tag{27}$$

$$k_i \ge \tau_i - u_i, \qquad i \in N'_{0,n+1}, \tag{28}$$

$$e_i \ge 0, \qquad i \in N'_{0,n+1}, \tag{29}$$

$$k_i \ge 0, \qquad i \in N'_{0,n+1}, \tag{30}$$

where the objective is to minimize the total penalty cost (incurred for earliness and delay) of the route operated by the considered EV.

Numerical results and insights

To evaluate the operational gains obtained by employing flexible time windows, we use Schneider's problem instances (see Schneider, Stenger, and Goeke 2014), and we consider 18 problem instances with tight time windows including 5, 10, and 15 customers. We set the cost coefficients c_e and c_d to 0.10 and 0.20, respectively (see Taş et al. 2014a,b). Moreover, c_f is set to $2.n.\max_{i \in V'_0, j \in V'_{n+1}, i \neq j}\{d_{ij}\}$ (following Schneider, Stenger, and Goeke 2014) to have a hierarchical objective function which first minimizes the number of vehicles used for service before minimizing the traveled distance and the time window deviations. In case the number of columns in the column pool is larger than 150, all columns that have been kept for more than 15 iterations are removed from the pool. Moreover, the ESPPRC is terminated if the number of efficient elementary paths with negative reduced costs on the ending depot is larger than 5 ($eff = 5$). The algorithms proposed in our solution procedure are coded in C++ and each model (master problem, integer programming problem, and linear programming problem) is solved using IBM ILOG CPLEX 12.6 (IBM 2018). All experiments are conducted on MacBook Pro with a 2.7-GHz Intel Core i5 processor and 8 GB of memory.

Next, we evaluate the performance of our solution procedure by solving the classical EVRPTW. In other words, f_i is set to 0 for each node $i \in V'_{0,n+1}$, and the obtained solutions are compared to the optimal/best-known solutions of the EVRPTW. We also analyze the effects of considering different threshold values to terminate the ESSPRC and then evaluate the operational benefits gained by considering flexible time windows.

Table 2. Details of the solutions obtained by our solution procedure to evaluate the effects of *eff*.

	EVRPFTW, $f_i = 0$, $eff = 10$			EVRPFTW, $f_i = 0$, $eff = 15$		
Ins.	#Veh.	Dist.	CPU	#Veh.	Dist.	CPU
c101-5	2	257.75	0.02	2	257.75	0.01
c103-5	1	184.50	0.18	1	184.50	0.18
r104-5	2	139.95	0.09	2	139.95	0.09
r105-5	2	156.08	0.01	2	156.08	0.01
rc105-5	2	241.30	0.02	2	241.30	0.02
rc108-5	2	253.93	0.04	2	253.93	0.04
c101-10	3	397.16	0.13	3	401.09	0.08
c104-10	2	300.82	18.20	2	307.89	22.19
r102-10	4	262.93	0.07	4	262.93	0.07
r103-10	2	212.81	0.69	2	212.81	0.84
rc102-10	4	423.51	0.02	4	423.51	0.03
rc108-10	3	352.40	1.47	3	352.40	1.97
c103-15	3	413.94	8.59	3	420.42	10.86
c106-15	3	313.14	0.72	3	298.19	1.14
r102-15	5	413.93	0.28	5	413.93	0.25
r105-15	4	366.72	0.26	5	364.47	0.28
rc103-15	4	400.01	2.46	4	397.67	2.29
rc108-15	3	370.25	11.58	3	370.25	5.42
Avg.	2.83	303.40	2.49	2.89	303.28	2.54

Validating the proposed solution procedure

In the first part of the numerical experiments, we solve the problem instances by considering hard time window constraints to validate the performance of our solution procedure based on column generation. Table 1 presents the optimal/best-known EVRPTW solutions (Opt./BKS EVRPTW) reported in the literature and the solutions generated by our solution procedure for the EVRPFTW where $f_i = 0$ for all $i \in V'_{0,n+1}$ (corresponding to the EVRPTW). In this table, we present the number of EVs activated for service (#Veh.), the total distance (Dist.), and the computation times in seconds (CPU). Note that in these experiments, *eff*, which is the threshold value used to terminate the ESPPRC, is equal to 5.

Results given in Table 1 show that our solution procedure performs well with respect to the optimal/best-known solutions of the EVRPTW. More specifically, the proposed methodology obtains the optimal/best-known solutions (with the same number of vehicles and the same total distance) for nine instances and obtains the same total distance for one instance (rc108-5). The solutions obtained by our solution procedure achieve a 4.04% gap in the average number of vehicles and a 3.42% gap in the average total distance. Furthermore, these solutions are obtained in 2.57 seconds on average, leading to an effective methodology.

Table 1. Comparison of the classical EVRPTW solutions with the solutions obtained by our solution procedure.

	Opt./BKS EVRPTW		EVRPFTW, $f_i = 0$, $eff = 5$		
Ins.	#Veh.	Dist.	#Veh.	Dist.	CPU
c101-5	2	257.75	2	257.75	0.03
c103-5	1	176.05	1	176.05	0.15
r104-5	2	136.69	2	136.69	0.09
r105-5	2	156.08	2	156.08	0.01
rc105-5	2	241.30	2	241.30	0.03
rc108-5	1	253.93	2	253.93	0.03
c101-10	3	393.76	3	397.16	0.14
c104-10	2	273.93	2	299.35	22.02
r102-10	3	249.19	4	262.93	0.08
r103-10	2	207.05	2	212.81	0.90
rc102-10	4	423.51	4	423.51	0.04
rc108-10	3	345.93	3	345.93	1.02
c103-15	3	384.29	3	409.40	10.94
c106-15	3	275.13	3	343.86	1.02
r102-15	5	413.93	5	413.93	0.51
r105-15	4	336.15	4	371.33	0.48
rc103-15	4	397.67	4	401.32	3.01
rc108-15	3	370.25	3	370.25	5.66
Avg.	2.72	294.03	2.83	304.09	2.57

We now evaluate the effects of *eff* on the performance of our solution procedure. Table 2 provides the solutions obtained by employing higher threshold values to prematurely terminate the ESPPRC. In other words, we stop the ESPPRC if the number of efficient elementary columns with negative reduced costs on the ending depot is larger than (i) 10 and (ii) 15. Compared to the solutions presented in Table 1, the results obtained by setting *eff* to 10 indicate that the average total distance decreases while the average number of vehicles stays the same (even though the number of instances solved to optimality decreases). We also observe that the average computation time slightly decreases. When *eff* is equal to 15, the average total distance further decreases; however, the average number of vehicles increases. In the following experiments *eff* is set to 10 since the solution procedure using this intermediate value performs well in terms of both the solution quality and the computation time.

EVRPFTW versus EVRPTW

The aim of this subsection is to evaluate the benefits obtained by employing flexible time windows compared to hard time windows. Table 3 represents the solutions of the EVRPFTW with (i) $f_i = 0.15$, $\forall i \in V'_{0,n+1}$, (ii) $f_i = 0.20$, $\forall i \in V'_{0,n+1}$, and (iii) $f_i = 0.25$, $\forall i \in V'_{0,n+1}$. In this table, we also report the total delay (Del.) and the total earliness (Earl.) of each solution. Results given in this table indicate that as the value of f_i increases, the number of EVs decreases and the total distance traveled increases. Compared to the optimal/best-known EVRPTW solutions, the intermediate flexibility fraction (0.20) reduces the number of EVs by 4.04% on average with a 2.18% increase in the total distance on average. Compared to the solutions obtained by the proposed procedure for the EVRPFTW with $f_i = 0$, we observe a 7.77% decrease in the number of EVs and a 1.19% decrease in the total distance on average. In other words, we observe that the case with $f_i = 0.20$, $\forall i \in V'_{0,n+1}$ provides very good solutions in small computation times with small violations in time windows.

We further examine the solutions obtained by employing the intermediate flexibility fraction. More specifically, Table 4 provides the analysis of the solutions obtained for the EVRPFTW with $f_i = 0.20$ for all nodes with respect to the EVRPTW solutions, where we

Table 3. Details of the solutions obtained by our solution procedure to evaluate the benefits gained by flexible time windows.

	EVRPFTW, $f_i = 0.15$					EVRPFTW, $f_i = 0.20$					EVRPFTW, $f_i = 0.25$				
Ins.	#Veh.	Dist.	Del.	Earl.	CPU	#Veh.	Dist.	Del.	Earl.	CPU	#Veh.	Dist.	Del.	Earl.	CPU
c101-5	2	257.75	0.00	0.00	0.01	2	257.75	0.00	0.00	0.01	2	257.75	0.00	0.00	0.02
c103-5	1	176.05	0.00	0.00	0.26	1	176.05	0.00	0.00	0.26	1	176.05	0.00	0.00	0.26
r104-5	1	136.69	1.50	16.34	0.07	1	136.69	2.00	15.34	0.08	1	136.69	2.50	14.53	0.15
r105-5	2	156.08	0.00	0.00	0.02	2	156.08	0.00	0.00	0.03	2	156.08	0.00	0.00	0.03
rc105-5	2	232.00	15.75	16.62	0.03	2	232.00	21.00	8.62	0.03	2	232.00	25.15	2.73	0.04
rc108-5	2	253.93	0.00	0.00	0.17	2	253.93	0.00	0.00	0.17	2	253.93	0.00	0.00	0.20
c101-10	3	401.09	0.00	0.00	0.12	3	409.45	9.6	2.07	0.16	3	398.32	5.30	0	0.14
c104-10	2	309.38	0.00	0.00	85.68	2	301.30	0.00	0.00	50.81	2	299.35	0.00	0.00	75.53
r102-10	3	249.19	0.00	0.00	0.17	3	249.19	0.00	0.00	0.12	3	288.30	5	49.59	0.07
r103-10	2	209.49	1.50	3.62	0.82	2	188.04	3.10	1.65	6.67	2	177.84	4.10	1.15	13.61
rc102-10	4	423.51	0.00	0.00	0.06	4	423.51	0.00	0.00	0.05	4	423.51	0.00	0.00	0.05
rc108-10	3	352.40	0.00	0.00	3.30	3	345.53	5.80	27.51	3.59	3	352.40	0.00	0.00	6.10
c103-15	3	430.84	19.20	107.90	5.51	3	432.78	7.25	0	3.59	3	444.29	17.50	14.09	3.83
c106-15	3	275.13	0.00	0.00	4.20	3	275.13	0.00	0.00	5.81	2	316.42	102.93	125.82	4.00
r102-15	5	413.46	0.40	0.00	0.41	4	435.43	2.40	1.98	0.51	4	435.43	2.90	1.48	0.57
r105-15	4	336.94	1.04	3.79	0.77	3	369.00	10.14	8.22	1.63	3	348.51	9.69	4.01	2.98
rc103-15	4	400.53	0.00	0.00	7.45	4	400.53	0.00	0.00	18.70	3	458.89	17.91	63.24	11.19
rc108-15	3	378.23	3.14	0	28.02	3	365.89	0.10	23.15	12.97	3	370.589	0.10	23.98	46.93
Avg.	2.72	299.59	2.36	8.24	7.62	2.61	300.46	3.41	4.92	5.84	2.50	307.02	10.73	16.70	9.21

Table 4. EVRPFTW with $f_i = 0.20\ \forall i \in V'_{0,n+1}$ versus EVRPTW with the optimal/best-known solutions and the solutions obtained by our solution procedure.

	EVRPFTW, $p_i = 0$	EVRPTW, Opt./BKS
Dist.(↓), #Veh.(↓)	2	0
Dist.(↓), #Veh.(↔)	6	4
Dist.(↑), #Veh.(↓)	2	2
Dist.(↔), #Veh.(↔)	4	6
Dist.(↑), #Veh.(↔)	4	4
Dist.(↔), #Veh.(↑)	0	1
Dist.(↔), #Veh.(↓)	0	1

consider both (i) the solutions generated by our solution procedure, and (ii) the optimal/best-known EVRPTW solutions. Compared to the solutions obtained by our solution procedure for the classical EVRPTW, results obtained for the EVRFTW with $f_i = 0.20$, $\forall i \in V'_{0,n+1}$ show that both the total distance and the number of EVs are reduced for two instances (r104-5, r102-10). We observe a reduction in the total distance with the same number of the EVs as the one obtained by our solution procedure for the classical EVRPTW for six problem instances (c103-5, rc105-5, r103-10, rc108-10, c106-15, rc108-15). For two instances (r102-15, r105-15), EVRFTW provides a reduction in the number of EVs; however, this brings an increase in the total distance. For four problem instances (c101-5, r105-5, rc108-5, rc102-10), the EVRPFTW obtains the same solutions as the ones generated by our procedure for the EVRPTW. For the remaining four problem instances (c101-10, c104-10, c103-15, rc103-15), EVRFTW obtains solutions with higher total distance whereas the number of EVs stays the same.

Compared to the optimal/best-known solutions of the EVRPTW, results obtained for the EVRFTW with $f_i = 0.20$, $\forall i \in V'_{0,n+1}$ show that the total distance is reduced with the same number of EVs for four instances (rc105-5, r103-10, rc108-10, rc108-15). For two instances (r102-15, r105-15), the EVRFTW provides a reduction in the number of EVs; however, this brings an increase in the total distance. For six problem instances (c101-5, c103-5, r105-5, r102-10, rc102-10, c106-15), the EVRPFTW obtains the same solutions as the classical EVRPTW. For four problem instances (c101-10, c104-10, c103-15, rc103-15), the EVRFTW obtains solutions with higher total distance whereas the number of EVs stays the same. For one problem instance (rc108-5), the EVRPFTW obtains a solution with the same total distance where the number of EVs is increased by one. For the remaining problem instance (r104-5), the EVRPFTW yields a reduction in the number of EVs with the same total distance as the one given by the optimal/best-known EVRPTW solutions.

Results given in Tables 3 and 4 show that the carrier companies can benefit from the operational gains provided by the flexible time windows. These gains are observed as traversing less distance or employing fewer vehicles by delivering the goods to customers with a small violation in time windows. In other words, flexible time windows yield a reduction in the total distance traveled or in the number of EVs activated, which are the two basic components of the cost function considered in the classical EVRPTW.

Conclusions

In this paper, we introduce the EVRPFTW in which vehicles are allowed to serve customers outside their original time windows. More specifically, time window boundaries are relaxed with respect to a given tolerance which results in penalty costs since customer satisfaction is negatively affected.

We propose a solution procedure based on column generation. In the initialization phase, we employ the time oriented-nearest neighbor heuristic. The routing component is handled via column generation algorithm. In case the obtained solution is fractional, an integer programming problem is used to generate an integer solution. We then solve a linear programming model to find the optimal start time of service at each customer for each route in the generated solution. We validate our solution procedure with various threshold values used to prematurely terminate the pricing subproblem. Moreover, we compare the solutions of the EVRPTW with those of the EVRPFTW and observe that EVRPFTW yields a reduction in the total distance or a reduction in the number of EVs or a reduction both in the total distance and in the number of EVs.

In this paper, a quite practical problem is modeled and a solution approach is developed to analyze the operational gains obtained by the flexible time windows. Further research may focus on developing a methodology to solve larger problem instances and on considering uncertainties in recharging and travel times.

Disclosure statement

No potential conflict of interest was reported by the author.

Funding

This research was partially supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) through Grant #118M412. This support is gratefully acknowledged.

ORCID

Duygu Taş http://orcid.org/0000-0002-3579-4600

References

Azadeh, A., and H. Farrokhi-Asl. 2018. “The Close-open Mixed Multi Depot Vehicle Routing Problem considering Internal and External Fleet of Vehicles.” *Transportation Letters* 11 (2): 78–92. doi:10.1080/19427867.2016.1274468.

Boland, N., J. Dethridge, and I. Dumitrescu. 2006. “Accelerated Label Setting Algorithms for the Elementary Resource Constrained Shortest Path Problem.” *Operations Research Letters* 34: 58–68. doi:10.1016/j.orl.2004.11.011.

Bruglieri, M., S. Mancini, F. Pezzella, O. Pisacane, and S. Suraci. 2017. “A Three-phase Matheuristic for the Time-effective Electric Vehicle Routing Problem with Partial Recharges.” *Electronic Notes in Discrete Mathematics* 58: 95–102. doi:10.1016/j.endm.2017.03.013.

Cordeau, J.-F., G. Desaulniers, J. Desrosiers, M. M. Solomon, and F. Soumis. 2002. “VRP with Time Windows.” In *The Vehicle Routing Problem*, edited by P. Toth and D. Vigo, 156–193. Philadelphia: SIAM Monographs on discrete mathematics and applications.

Desaulniers, G., F. Errico, S. Irnich, and M. Schneider. 2016. “Exact Algorithms for Electric Vehicle-routing Problems with Time Windows.” *Operations Research* 64 (6): 1388–1405. doi:10.1287/opre.2016.1535.

Desrochers, M. “An Algorithm for the Shortest Path Problem with Resource Constraints.” Technical Report G-88-27; GERAD; 1988. doi: 10.3168/jds.S0022-0302(88)79586-7

Feillet, D., P. Dejax, M. Gendreau, and C. Gueguen. 2004. “An Exact Algorithm for the Elementary Shortest Path Problem with Resource Constraints: Application to Some Vehicle Routing Problems.” *Networks* 44: 216–229. doi:10.1002/(ISSN)1097-0037.

Feillet, D., M. Gendreau, and L.-M. Rousseau. 2007. “New Refinements for the Solution of Vehicle Routing Problems with Branch and Price.” *INFOR* 45 (4): 239–256.

IBM . ILOG CPLEX Optimizer 12.6. https://www.ibm.com/analytics/cplex-optimizer; 2018.

Keskin, M., and B. Çatay. 2016. “Partial Recharge Strategies for the Electric Vehicle Routing Problem with Time Windows.” *Transportation Research Part C* 65: 111–127. doi:10.1016/j.trc.2016.01.013.

Keskin, M., and B. Çatay. 2018. “A Matheuristic Method for the Electric Vehicle Routing Problem with Time Windows and Fast Chargers.” *Computers & Operations Research* 100: 172–188. doi:10.1016/j.cor.2018.06.019.

Montoya, A., C. Guéret, J. E. Mendoza, and J. G. Villegas. 2017. “The Electric Vehicle Routing Problem with Nonlinear Charging Function.” *Transportation Research Part B* 103: 87–110. doi:10.1016/j.trb.2017.02.004.

Musolino, G., A. Polimeni, and A. Vitetta. 2018. “Freight Vehicle Routing with Reliable Link Travel Times: A Method Based on Network Fundamental Diagram.” *Transportation Letters* 10 (3): 159–171. doi:10.1080/19427867.2016.1241040.

Righini, G., and M. Salani. 2008. “New Dynamic Programming Algorithms for the Resource Constrained Elementary Shortest Path Problem.” *Networks* 51 (3): 155–170. doi:10.1002/(ISSN)1097-0037.

Schneider, M., A. Stenger, and D. Goeke. 2014. “The Electric Vehicle-routing Problem with Time Windows and Recharging Stations.” *Transportation Science* 48 (4): 500–520. doi:10.1287/trsc.2013.0490.

Solomon, M. M. 1987. “Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints.” *Operations Research* 35 (2): 254–265. doi:10.1287/opre.35.2.254.

Taş, D., M. Gendreau, N. Dellaert, T. van Woensel, and A. G. de Kok. 2014a. “Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation and Branch-and-price Solution Approach.” *European Journal of Operational Research* 236 (3): 789–799. doi:10.1016/j.ejor.2013.05.024.

Taş, D., O. Jabali, and T. van Woensel. 2014b. “A Vehicle Routing Problem with Flexible Time Windows.” *Computers & Operations Research* 52: 39–54. doi:10.1016/j.cor.2014.07.005.

Yao, B., Q. Cao, Z. Wang, P. Hu, M. Zhang, and B. Yu. 2016. “A Two-stage Heuristic Algorithm for the School Bus Routing Problem with Mixed Load Plan.” *Transportation Letters* 8 (4): 205–219. doi:10.1080/19427867.2015.1110953.