Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 2545 – 2560

doi:10.3906/elk-2011-124

# A new model for minimizing the electric vehicle battery capacity in electric travelling salesman problem with time windows

**Kazım ERDOĞDU***, **Korhan KARABULUT**
Department of Software Engineering, Faculty of Engineering, Yaşar University, İzmir, Turkey

**Received:** 25.02.2021 • **Accepted/Published Online:** 16.06.2021 • **Final Version:** 23.09.2021

**Abstract:** The growing pollution in the environment and the negative shift in the global climate compel authorities to take action to protect the environment and human health. Transportation is one of the major contributors to this environmental decay. The harmful gases released to the air by the vehicles using petroleum fuel increase each day. One of the solutions is to make a gradual transition to electric vehicles. A major part of manufacturing an electric vehicle is to produce an efficient electric motor and battery for it. Reducing the manufacturing and operating costs of these components will result in reducing the overall costs of electric vehicles. In this study, a new variant of the electric travelling salesman problem with time windows (E-TSPTW) was proposed. The objective function of the problem is to minimize the required initial battery capacity of the electric vehicle. For this goal, a new energy consumption model considering the load of the vehicle was proposed with three scenarios. The proposed model was solved with a hybrid simulated annealing algorithm for all these scenarios. The performance of the proposed method was compared to the solutions found by a mixed integer linear programming model. The experimental results on the benchmark instances show that up to a 35% reduction in initial battery capacity, hence reduction in its cost is possible.

**Key words:** Electric travelling salesman problem with time windows, energy consumption, battery capacity, mixed integer linear programming, simulated annealing

## 1. Introduction

The growth in environmental pollution and climate change continues to be a threat to nature and human life. One of the major contributors to that pollution is the vehicles using petroleum fuel. The $CO_2$ released by these internal combustion engine vehicles (ICEVs) increases the greenhouse gas (GHG) emissions each year, which, in turn, damages the nature and human health. The growth of the light- and heavy-duty trucks used in freight transport globally increases the amount of GHGs released to the air each day. For this reason, many developed countries have begun to provide considerable economic subsidies to vehicle manufacturers for developing electric vehicles (EVs) [1]. According to the 2020 technical report of the International Energy Agency (IEA)[1], there were 7.2 million EVs in the world in 2019, including battery electric vehicles (BEVs) and plug-in hybrid electric vehicles (PHEVs) together. This number constituted 1% of the total vehicles in the world in 2019. Depending on the gradual increase in EV sales and the developed countries' subsidies for EVs, IEA estimates that the number of EVs will be 7% of the total number of vehicles around the world by 2030. It is seen that there is a growing inclination towards including EVs in transportation for environmental purposes.

*Correspondence: kazim.erdogdu@yasar.edu.tr

[1]International Electric Acency (2020). Global EV Outlook 2020: Entering the decade of electric drive? [online]. Website https://www.oecd-ilibrary.org/content/publication/d394399e-en [accessed 12 August 2020].

Despite the various benefits of using EVs in transportation, certain issues impact the cost and performance of the EVs. EV batteries' manufacturing costs, capacities, weights, lifespans, driving ranges, recharging speeds and infrastructure, size and weight of the EV, ambient temperature, etc. all need to be taken into consideration in producing and using EVs [2–5]. Among all these factors, the procurement of the EV batteries shares a major part in manufacturing and operating costs. According to Zhu et al. [1], EV batteries comprise 40% of the total manufacturing cost of EVs. For this reason, EV manufacturers are challenged to invest in developing low-cost high-performance EV batteries. As the capacity of the EV batteries increase, the driving range of the EV also increases. This increment, however, increases the manufacturing costs, recharging time, and total weight of the vehicle, which, in turn, increases the total energy consumption (EC). Deciding on the optimum EV battery capacity (BC) is one of the issues in lowering the cost of EV manufacturing. In this paper, an electric travelling salesman problem with time windows (E-TSPTW) was studied. Unlike classical E-TSPTW studies, minimizing the BC was used as the objective function. A new mathematical model was proposed. In this model, three scenarios were considered, and these scenarios were applied to the two well-known benchmark sets. As a solution method, a hybrid simulated annealing (SA) method was used and its results for the smaller problem instances were compared with the results of the mixed integer linear programming model (MILP).

### 1.1. Literature review

Few studies have been done in E-TSPTW compared to TSPTW and electric vehicle routing problem with time windows (E-VRPTW) in literature. The pioneering study for E-TSPTW was done by Roberti and Wen [6]. In their study, they proposed a MILP model in which they took into consideration both partial and full recharging battery policies of the EVs. They aimed to minimize the total distance traveled by the EV. They added a BC constraint for the EV and used a battery consumption function that was proportionate to the traveling distance of the EV. They developed their three-phase heuristic algorithm and applied it on their benchmark set with two different scenarios, one having five charging stations and another having ten charging stations. Their charging stations were placed separately from the customer locations and no recharging was done at customer locations. They concluded that their solution algorithm provided successful results on their benchmark instances with the number of customers varying from 20 to 200.

Küçükoğlu et al. [7] proposed the E-TSPTW with mixed charging rates (E-TSPTW-MCR). They based their model on the model of Roberti and Wen [6] and used the same objective function, BC constraints, and recharging scenarios. They, however, allowed the EV in their problem to recharge at customer locations as well as charging stations. They applied their heuristic algorithm, which is a hybridization of SA and Tabu Search (TS), on the same benchmark instances of [6]. The authors concluded that they either found better solutions or obtained the best-known results in less computation time for these problem instances.

Baek et al. [8] studied a variation of E-TSP in which they proposed an EC model including the weight of the EV. They used an electric truck simulator and analyzed the results of different solution methods. Unlike the previous studies, they explored the impact of the payload of the electric truck on EC. They concluded that the total weight of the vehicle also impacts the EC. Hence, they formulated their objective function to include both the distances and the payload of the EV during the travel. They applied their proposed greedy algorithm and some traditional heuristic routing algorithms on the problem instances generated by the authors.

Erdoğdu and Karabulut [9] studied an E-TSPTW with two objectives: minimizing total distance and minimizing EC. They included the payload of the EV in their EC model. They proposed a hybrid SA heuristic algorithm and applied it to the well-known benchmark instances of Potvin and Bengio [10] separately for two objective functions. According to their experimental results, they concluded that these two objectives are semiconflicting, that is, minimizing total distance does not necessarily minimize the total EC when the load of the EV is taken account in the objective function. They obtained the optimum results for distance minimization objective in the literature and provided benchmark results for the energy minimization objective in their model.

Doppstadt et al. [11] proposed the hybrid electric vehicle-TSPTW (HEV-TSPTW). They introduced four modes of operation: pure combustion mode, pure electric mode, charging while driving in combustion mode and boost mode which is the combination of combustion and electric modes. They modeled their objective function as the linear combination of the costs of each mode and aimed to minimize it. They generated 216 problem instances for their problem. They solved their problem with IBM CPLEX for the smaller size instances and applied a variable neighborhood search (VNS) for the rest of their instances.

There have also been various studies in E-VRPTW in literature. An extensive survey for E-VRPs can be found in the work of Pelletier et al. [3]. Since the problem studied in this paper is a version of E-TSPTW, the detailed literature on E-VRP was not mentioned. On the other hand, the related state-of-the-art studies in E-VRPTW with regards to EC models are mentioned as follows.

Goeke and Schneider [12] formulated a realistic EC model in their E-VRPTW study that uses the EV speed, the slope of the arcs, and the payload of the EVs. Keskin and Çatay [13] studied E-VRPTW with partial recharging in which the EC and EV battery levels were calculated as ratios directly proportionate to the distance traversed by the EVs. Cortés-Murcia et al. [14] studied E-VRPTW and satellite customers (E-VRPTWsc) in which they aimed to minimize the total recharging time. They used an EC rate that is proportional to the traveling distance of the EVs. Xiao et al. studied E-VRPTW considering energy/electricity consumption rate (E-VRPTW-ECR) [16]. Their EC rate consists of a nonlinear function of the speed which is assumed to be continuous and the linear effect of the load. Although Macrina et al. [17] studied a version of green VRPTW, the fleet in their problem contains both BEVs and ICEVs. The EC model in their study is comprehensive and contains speed, acceleration, deceleration, cargo load, and gradients.

In this paper, a new E-TSPTW model is proposed. The objective function is to minimize the needed BC for the EV to finish its tour. The EC model used in the BC model is directly proportional to the load and the traveled distance of the EV. To the best of our knowledge, the BC model proposed for E-TSPTW in this study is novel. Three scenarios were considered in the problem. In scenario 1, no recharging takes place. In scenario 2, the EV is partially recharged only during the service times at customer locations. In scenario 3, the EV is partially recharged both during the service times and the waiting times at customer locations. All the recharging of the EV occurs at customer locations and no extra charging stations were added to the problem. As a solution method, the well-known SA heuristic method was hybridized with initialization and local search heuristics and applied to the two widely considered TSPTW benchmark sets for each scenario. The same problem was solved using IBM CPLEX for a subset of these problem instances and optimal results were obtained for these instances. The hybrid SA's results were compared with the CPLEX results. After the verification of the success of the hybrid SA, it was applied the rest of the problem instances that were used in this study.

## 2. Problem definition

Similar to TSPTW, in E-TSPTW, there is a traveling salesperson who delivers the demands of his or her customers by visiting them exactly once during their available time windows and returning to the starting point at the end of the tour. The initial and final locations for the salesperson are set to the depot. In conventional E-

TSPTW, hence the name, the vehicle being used is an EV. As a result, the EC, BC, battery recharging, etc. must be taken into consideration so that the traveling salesperson can accomplish his or her delivering task efficiently and under the given constraints. In the literature, there is a vast amount of studies being done on modeling the EC and measuring the cost of EVs [18]. Some of these studies are already mentioned in the literature review of this paper. Those studies that include more real-life parameters such as vehicle mass, speed, payload, traffic congestion, battery type, ambient temperature, air resistance, road frictions and slopes, etc. require detailed and specific data for the problem. Besides, the values of some of these parameters (e.g. ambient temperature, vehicle speed, traffic congestion, etc.) momentarily change depending on the circumstances so that they cannot be known priorly. For this reason, even in these more realistic models, these values are fixed with constants estimated via the experimental studies. Although the more realistic ECs for EVs converge to the actual EC values, they get more problem-specific and still contain fixed values for dynamic variables.

The objective function of the E-TSPTW studied in this paper contains a more generic EC model (Eq. (2)) so that it can be applied to any existing TSPTW or VRPTW benchmark problem instance in the literature. In the model, EC includes only the total load and the traversing distance of the EV. The full capacity of the EV was taken as the sum of demands of the customers. As in the study of [6], the EC was taken proportionally linear to the traveling distance and yet a little modification was made. The impacts of the vehicle mass and payload of the EV were also included in the model. The impact of the vehicle mass was taken as 1, while the impact of the payload was taken as the ratio of the payload over the EV capacity.

Recharging the EV battery could be done only at customer locations. No separate charging station was added to the problem instances. Since the EC rate and the recharging rate are not the same, the EC and recharging values were associated with two coefficients, $h$, and $g$, respectively. The values of these coefficients were taken as $h = 1$ and $g = 0.25$, same as in Roberti and Wen [6]. The same coefficient values were used in all three scenarios.

### 2.1. Mathematical model

The E-TSPTW studied in this paper is defined as a complete undirected graph $G = (V, A)$. Here, $V$ represents the set of vertices and is the union of the customers set $N = \{1, 2, , n\}$ and the depot with index 0, $V = N \cup \{0\}$. The arcs between the vertices in $V$ are represented with the set $A = \{(i, j) \in V \times V \mid i, j \in V; i \neq j\}$. The parameters and the decision variables of the mathematical model are given in Table 1.

**Table 1**. Variables and parameters used in the mathematical model of E-TSPTW.

| Notation | Description |
|---|---|
| $d_{ij}$ | Euclidean distance between the customers $i$ and $j$. |
| $Q$ | Full payload weight of the EV. |
| $q_{ij}$ | Payload weight of the EV while traveling from customer $i$ to customer $j$. |
| $a_i$ | The earliest allowed arrival time at customer $i$. (i.e. time window beginning) |
| $b_i$ | The latest allowed arrival time at customer $i$. (i.e. time window ending) |
| $t_{ij}$ | Travel time from customer $i$ to customer $j$. |
| $T_i$ | Arrival time of the EV at customer $i$. |
| $s_i$ | Service time at customer $i$. |
| $h$ | Energy consumption coefficient. Its value is 1. |
| $g$ | Recharging coefficient. Its value is 0.25. |
| $R_k$ | Recharging formula for scenario $k$. |
| $x_{ij}$ | Binary decision variable indicating whether the arc $(i, j)$ is traveled by the EV. |

The objective function is defined as follows:

$$min\lceil h * ec - g * R_k \rceil, \tag{1}$$

where

$$ec = \sum_{\substack{i,j \in V \\ i \neq j}} \left(1 + \frac{q_{ij}}{Q}\right) d_{ij} x_{ij} \tag{2}$$

$$R_k = \begin{cases} 0 & \text{if } k = 1 \\ \sum\limits_{i \in N} s_i & \text{if } k = 2 \\ \sum\limits_{i \in N} (s_i + max(0, a_i - T_i)) & \text{if } k = 3 \end{cases} \tag{3}$$

Subject to:

$$\sum_{i \in N} x_{ij} = 1, \qquad \forall j \in N \tag{4}$$

$$\sum_{j \in N} x_{ij} = 1, \qquad \forall i \in N \tag{5}$$

$$\sum_{j \in N} x_{0j} = 1 \tag{6}$$

$$\sum_{i \in N} x_{i0} = 1 \tag{7}$$

$$\sum_{i,j \in V} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, \quad S \neq \varnothing \tag{8}$$

$$T_j = max(a_j, T_i + s_i + t_{ij}), \qquad \forall i, j \in V \tag{9}$$

$$a_i \leq T_j \leq b_j, \qquad \forall j \in V \tag{10}$$

$$q_{ij} \leq Q, \qquad \forall i, j \in V \tag{11}$$

$$x_{ij} \in 0, 1, \qquad \forall i, j \in V \tag{12}$$

The objective function in (1) minimizes the needed BC for the EV to be able to finish its tour. The EC function was defined in (2). The scenarios used in the problem have different impacts on the objective function. These different impacts are defined in (3), where each $k$ represents the scenarios of 1, 2, and 3. In scenario 1, there is no recharging, while in scenario 2, the EV's battery is recharged during the service time. In scenario 3, the EV's battery gets charged both during the service time and the waiting time. If the EV arrives its next customer before the customer's earliest available time, then it must wait. The EV battery can be recharged at the customer's location during this waiting time.

Constraints (4) and (5) ensure that each customer is visited exactly once. Constraints (6) and (7) guarantee that the tour begins and ends at the depot. Constraint (8) does not allow subtours on the tour. Constraints (9) and (10) define and make sure that the EV delivers the customers' demands during their available time windows. Constraint (11) ensures that the current load of the EV during the travel does not exceed the vehicle's capacity. Constraint (12) is the decision variable indicating which arcs are being selected on the tour.

## 3. Solution methods

Since TSPTW is an NP-hard problem [19], and so is E-TSPTW, a heuristic method was used as the main solution method in this study: the simulated annealing (SA) algorithm [20]. The SA was hybridized with a constructive heuristic and a local search procedure. The initial solution was generated by the constructive heuristic. In the constructive heuristic, the tour permutation in the initial solution was constructed by sorting the customers in ascending order of their time window beginnings. The purpose of this constructive heuristic is to minimize the time window violations by minimization of waiting times at the customers so that it is less likely to start with an infeasible solution at the beginning of the SA process.

The perturbation in the SA operator is the swapping of the two randomly selected customers on the tour. In the local search phase, each customer in the current solution is removed from its current location and reinserted in all other possible locations. The first improvement pivoting rule was used in the selection process of local search. The local search ended whenever it found an improvement in the fitness value of the current feasible solution. The superiority of feasibility method [21] was used when updating the best and current solutions in the hybrid SA and selecting the better neighbor in local search. When two solutions (i.e. current and neighbor solutions) were compared, the feasible one was preferred to the infeasible one. In the case of two feasible solutions, the one having better fitness was selected. If none of the two solutions were feasible, then the one having a smaller number of time window violations was selected. This procedure of superiority of feasibility is given in Algorithm 1.

**Algorithm 1: superiority of feasibility**

**Input:** Two solutions: $\pi_1$ and $\pi_2$
**Output:** True if $\pi_1$ is better than $\pi_2$, False otherwise
**Procedure** isBetter($\pi_1, \pi_2$):
  **if** (isFeasible($\pi_1$) and isFeasible($\pi_2$))
    **if** (fitness($\pi_1$) $<$ fitness($\pi_2$))
      **return True**
  **else if** (**not** isFeasible($\pi_1$) and **not** isFeasible($\pi_2$))
    **if** (nbOfViolations($\pi_1$) $<$ nbOfViolations($\pi_2$))
      **return True**
  **else**
    **if** (isFeasible($\pi_1$))
      **return True**
  **end if**
  **return False**
**End Procedure**

The pseudocode of the hybrid SA is given in Algorithm 2. The initial temperature value was taken as 100 and decreased by 1 at each iteration. At each iteration, 10 perturbations are made to the current solution. The best solution of the SA was recorded during the whole process and was returned as the result when the algorithm terminates.

**Algorithm 2: the hybrid SA**

```
Output: Best solution found, i.e. π_best
Procedure Hybrid_SA():
  π_current ⟵ constructiveHeuristic()
  insertionLocalSearch(π_current)
  π_best ⟵ π_current
  temperature ⟵ 100
  while (temperature ≥ 1)
    for i ⟵ 1 to 10 do // number of perturbations
      π_neighbor ⟵ randomSwap (π_current)
      insertionLocalSearch (π_neighbor)
      if (isBetter(π_neighbor, π_current))
        π_current ⟵ π_neighbor
        if (isBetter(π_neighbor, π_best))
          π_best ⟵ π_neighbor
      else if (random() < e^(fitness(π_neighbor) − fitness(π_current)))
        π_current ⟵ π_neighbor
    end for
    temperature ⟵ temperature - 1
  end while
  return π_best
End Procedure
```

## 4. Experimental results

The hybrid SA algorithm designed for the solution of the E-TSPTW in this study was coded in C++ and executed on a computer with an Intel Core i5 processor at 2.50 GHz and 8 GBs of RAM. Two different benchmark sets were used in the experimental studies: Gendreau–Dumas–extended (GDE) [22] and Solomon–Potvin–Bengio (SPB) instances [10]. The former benchmark set contains 250 problem instances grouped with regards to the number of customers 20, 40, 60, 80, and 100. In each group, there are ten different time window subgroups (i.e. 20, 40, 60, 80, 100, 120, 140, 160, 180, and 200), and each of these subgroups contain five different problem instance sets. GDE instances do not contain the demands of the customers since they are originally for TSPTW. We generated demands for each problem instance according to a discrete uniform distribution in the range [1, 40].[2]. The latter benchmark set contains 30 problem instances in which the number of customers range from 3 to 44 and demands of the customers exist in the original problem instances.

The proposed E-TSPTW model was first modeled as a MILP and solved by IBM CPLEX version 12.10.0.0 implemented in the Python programming language with version 3.7.8 on the same computer for all 50 GDE instances having 20 customers. Since there is no service time in GDE instances, only scenarios 1 and 3 were taken into consideration. It is because when there is no service time, then scenarios 1 and 2 are equivalent (Eq. (3)). New service times for the customers were not generated for these instances in order not to affect the

[2]These generated demands and the detailed results of each instance can be found in the supplementary material provided at https://kkarabulut.yasar.edu.tr/etsptw/

feasibility of the solutions. Then, the hybrid SA was applied to all the instances used in this study. For each instance, the hybrid SA was executed for 10 different runs and the best of these 10 runs were recorded as the solution of the hybrid SA. The results of CPLEX and the hybrid SA for the related instances are presented in Tables 2 and 3. In these tables, only the first instances in each n20w* GDE instance groups are given. The rest of the n20w* GDE instances are presented in the SM of this study.

In all the tables of this paper, BC, EC, D, BB, and TT stand for battery capacity, energy consumption, distance, the time (in seconds) of the algorithm in finding the best result, and the total time that the hybrid SA ran for the related instance, respectively. The * in the BC columns of Tables 2 and 3 indicates that CPLEX found the optimum result for that instance in the related scenario. Since the number of customers and the time windows of customers are already specified in the name of the GDE instances, an extra column in Tables **??** was not added for showing the number of customers. In each GDE instance file name, the number following the letter n gives the number of customers and the number following w gives the time window width of the customers in the instance set. The number following the "." is the id of the different samples of the instances containing the same customer number and time window.

**Table 2**. Comparison of the results of CPLEX and the hybrid SA on the first group of GDE instances for scenario 1.

| | CPLEX | | | | The hybrid SA | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Instance | BC | EC | D | BT | BC | EC | D | BT | TT |
| n20w20.001 | 513* | 512.8 | 378 | 0.16 | 513 | 512.8 | 378 | 0.002 | 8.365 |
| n20w40.001 | 356* | 355.68 | 254 | 1.69 | 356 | 355.68 | 254 | 0.001 | 8.11 |
| n20w60.001 | 434* | 433.62 | 335 | 20.42 | 434 | 433.62 | 335 | 0.02 | 8.18 |
| n20w80.001 | 462* | 461.63 | 331 | 18.42 | 462 | 461.74 | 330 | 0.006 | 8.15 |
| n20w100.001 | 325* | 324.53 | 237 | 245.45 | 325 | 324.53 | 237 | 0.002 | 8.285 |
| n20w120.001 | 362* | 361.5 | 267 | 37.83 | 362 | 361.5 | 267 | 0.004 | 7.736 |
| n20w140.001 | 267 | 266.7 | 176 | 1000.02 | 267 | 266.7 | 176 | 0.074 | 7.62 |
| n20w160.001 | 327 | 326.33 | 242 | 1000.12 | 327 | 326.33 | 242 | 0.034 | 8.157 |
| n20w180.001 | 366 | 365.63 | 254 | 1000.03 | 366 | 365.63 | 254 | 0.015 | 8.29 |
| n20w200.001 | 321 | 320.79 | 233 | 1000.02 | 320 | 319.75 | 234 | 0.005 | 7.449 |

The running time of CPLEX was limited to 1000 s for each problem instance. As the width of the time window increases, the running time of CPLEX increases exponentially. Nevertheless, CPLEX found optimum values for 26 instances and upper bounds for 24 instances in scenario 1, while finding optimum values for 20 instances and upper bounds for 30 instances in scenario 3. The hybrid SA has provided good results for BC compared to the CPLEX results. The hybrid SA obtained the same results for the optimum solutions of CPLEX and better results for the nonoptimum solutions of CPLEX. In Tables 2 and 3, the EC results were also provided to underline the BC savings between the two scenarios even when ECs are the same in the corresponding instances.

After this verification of the hybrid SA's performance, the rest of the GDE instances and all the SPB instances were solved with the hybrid SA algorithm. There are 250 GDE instances in total. The average results of the hybrid SA for each GDE instance group were given in Table 4. The detailed results for all GDE instances were provided in the supplementary material. The results of the hybrid SA for SPB instances are given in Table 5. In Tables 4 and 5, S1, S2, and S3 are the abbreviations of scenarios 1, 2, and 3, respectively.

**Table 3**. Comparison of the results of CPLEX and the hybrid SA on the first group of GDE instances for scenario 3.

| | CPLEX | | | | The hybrid SA | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Instance | BC | EC | D | BT | BC | EC | D | BT | TT |
| n20w20.001 | 511* | 512.8 | 378 | 0.12 | 511 | 512.8 | 378 | 0.002 | 8.398 |
| n20w40.001 | 349* | 355.68 | 254 | 10.16 | 349 | 355.68 | 254 | 0.001 | 8.196 |
| n20w60.001 | 418* | 433.62 | 335 | 82.69 | 418 | 433.62 | 335 | 0.02 | 8.217 |
| n20w80.001 | 444* | 461.74 | 330 | 56.53 | 444 | 462.34 | 329 | 0.003 | 8.208 |
| n20w100.001 | 306 | 324.53 | 237 | 1000.02 | 306 | 324.53 | 237 | 0.003 | 7.978 |
| n20w120.001 | 353 | 361.5 | 267 | 1000.05 | 342 | 361.5 | 267 | 0.044 | 7.95 |
| n20w140.001 | 268 | 280.87 | 184 | 1006.83 | 252 | 266.7 | 176 | 0.067 | 7.932 |
| n20w160.001 | 310 | 327.82 | 243 | 1002.69 | 301 | 326.33 | 242 | 0.041 | 8.398 |
| n20w180.001 | 360 | 389.46 | 268 | 1000.09 | 333 | 366.23 | 253 | 0.009 | 8.72 |
| n20w200.001 | 330 | 342.09 | 252 | 1003.67 | 313 | 320.79 | 233 | 0.003 | 8.013 |

**Table 4.** Average results of the hybrid SA for the GDE groups instances for scenarios 1 and 3.

| | Scenario 1 | | | | Scenario 3 | | | | Avg BC |
|---|---|---|---|---|---|---|---|---|---|
| Instance groups | Avg BC | Avg EC | Avg D | Avg BT | Avg BC | Avg EC | Avg D | Avg BT | savings S1 to S3 |
| n20w20 | 489.2 | 488.71 | 361.2 | 0.001 | 487 | 488.71 | 361.2 | 0.0006 | 0.45 % |
| n20w40 | 431 | 430.55 | 271 | 0.006 | 423.8 | 430.52 | 316 | 0.006 | 1.67 % |
| n20w60 | 417 | 416.46 | 337.5 | 0.012 | 401.6 | 416.46 | 310.2 | 0.0124 | 3.69 % |
| n20w80 | 423.4 | 422.87 | 297 | 0.0397 | 408.2 | 422.99 | 311 | 0.008 | 3.59 % |
| n20w100 | 381.6 | 380.98 | 247.5 | 0.0096 | 365.8 | 380.98 | 275.2 | 0.011 | 4.14 % |
| n20w120 | 364.6 | 363.96 | 253.5 | 0.0074 | 348 | 364.95 | 266.4 | 0.0158 | 4.55 % |
| n20w140 | 334.2 | 333.57 | 202 | 0.1502 | 314.4 | 334.48 | 235.4 | 0.0498 | 5.92 % |
| n20w160 | 307.8 | 307.19 | 246 | 0.0338 | 282.2 | 308.07 | 220 | 0.0696 | 8.32 % |
| n20w180 | 333.2 | 332.78 | 223.5 | 0.016 | 308.2 | 332.9 | 237.4 | 0.0176 | 7.50 % |
| n20w200 | 336.2 | 335.62 | 230.5 | 0.3698 | 319.2 | 335.83 | 242.4 | 0.0512 | 5.06 % |
| n40w20 | 656.2 | 655.8 | 499.5 | 0.0964 | 647.2 | 655.69 | 486.6 | 0.1774 | 1.37 % |
| n40w40 | 616 | 615.47 | 459 | 0.0432 | 603.2 | 615.45 | 461 | 0.0444 | 2.08 % |
| n40w60 | 556.8 | 556.4 | 411 | 0.6146 | 539.2 | 556.4 | 416.4 | 0.5352 | 3.16 % |
| n40w80 | 558.2 | 557.81 | 369.5 | 2.996 | 535.2 | 558.25 | 400.4 | 0.5654 | 4.12 % |
| n40w100 | 516.6 | 516.33 | 403.5 | 1.2264 | 490.8 | 517.49 | 379.4 | 2.1754 | 4.99 % |
| n40w120 | 521.6 | 521.16 | 392 | 1.6632 | 490.4 | 521.18 | 378 | 0.7696 | 5.98 % |
| n40w140 | 498.2 | 497.81 | 349.5 | 3.4 | 468.6 | 497.85 | 364.4 | 3.3938 | 5.94 % |
| n40w160 | 451.2 | 450.72 | 331.5 | 1.6312 | 420.8 | 450.79 | 327.2 | 2.141 | 6.74 % |
| n40w180 | 475 | 474.84 | 338.5 | 30.1172 | 446.2 | 477.54 | 333.6 | 16.1702 | 6.06 % |
| n40w200 | 439.8 | 439.45 | 316.5 | 1.9738 | 409.2 | 439.14 | 316.4 | 19.6862 | 6.96 % |
| n60w20 | 781.2 | 780.85 | 577 | 3.9366 | 770 | 780.9 | 581.6 | 3.5102 | 1.43 % |
| n60w40 | 796.6 | 796.12 | 565 | 3.4072 | 780.4 | 796.11 | 590.2 | 2.4788 | 2.03 % |

**Table 4.** (Continued).

| | Scenario 1 | | | | Scenario 3 | | | | Avg BC |
|---|---|---|---|---|---|---|---|---|---|
| Instance groups | Avg BC | Avg EC | Avg D | Avg BT | Avg BC | Avg EC | Avg D | Avg BT | savings S1 to S3 |
| n60w60 | 744 | 743.38 | 589 | 24.127 | 715.2 | 743.56 | 560.2 | 42.649 | 3.87 % |
| n60w80 | 693.4 | 692.83 | 463 | 4.9108 | 661.6 | 692.99 | 508.6 | 8.5876 | 4.59 % |
| n60w100 | 684.8 | 684.3 | 483 | 104.313 | 654.8 | 684.81 | 515.4 | 108.403 | 4.38 % |
| n60w120 | 619.4 | 619.19 | 466 | 28.043 | 584.4 | 619.27 | 451.6 | 51.8614 | 5.65 % |
| n60w140 | 636.6 | 636.32 | 442.5 | 110.199 | 596.4 | 637.9 | 453.8 | 118.962 | 6.31 % |
| n60w160 | 629.6 | 629.06 | 532.5 | 28.756 | 582.8 | 629.11 | 464.6 | 32.931 | 7.43 % |
| n60w180 | 585 | 584.56 | 404 | 42.0598 | 537 | 584.57 | 421.6 | 96.6666 | 8.21 % |
| n60w200 | 583.2 | 582.8 | 419.5 | 78.1908 | 543.6 | 583.05 | 427.8 | 57.9898 | 6.79 % |
| n80w20 | 904.6 | 904.19 | 682 | 9.798 | 886.4 | 904.1 | 676.6 | 164.2548 | 2.01 % |
| n80w40 | 850.4 | 849.99 | 650.5 | 94.839 | 826.4 | 849.98 | 630 | 65.1682 | 2.82 % |
| n80w60 | 813.6 | 812.99 | 564.5 | 254.1122 | 785.6 | 813.03 | 606.4 | 424.661 | 3.44 % |
| n80w80 | 802.4 | 802.08 | 597.5 | 144.0218 | 770.8 | 803.21 | 595.2 | 196.3636 | 3.94 % |
| n80w100 | 769 | 768.61 | 548.5 | 261.8108 | 733.6 | 768.74 | 579.4 | 135.0048 | 4.60 % |
| n80w120 | 736 | 735.65 | 544.5 | 352.3612 | 688 | 735.79 | 541.6 | 233.1364 | 6.52 % |
| n80w140 | 699.8 | 699.06 | 528.5 | 371.4204 | 652.4 | 700.08 | 509.4 | 514.6308 | 6.77 % |
| n80w160 | 697.8 | 697.32 | 472.5 | 91.9878 | 653.6 | 698.99 | 507.4 | 478.0394 | 6.33 % |
| n80w180 | 691 | 690.45 | 510.5 | 284.0672 | 643 | 690.58 | 501.4 | 201.2788 | 6.95 % |
| n80w200 | 659.6 | 658.94 | 470 | 123.4964 | 611.6 | 660.56 | 485 | 196.7204 | 7.28 % |
| n100w20 | 1026.8 | 1026.34 | 756 | 483.6986 | 1010.6 | 1026.35 | 757.6 | 222.2554 | 1.58 % |
| n100w40 | 940.2 | 939.92 | 734.5 | 257.7136 | 908.8 | 939.86 | 701.8 | 233.712 | 3.34 % |
| n100w60 | 931.4 | 931.18 | 658 | 179 | 901.2 | 931.25 | 696.6 | 252.0832 | 3.24 % |
| n100w80 | 887 | 886.78 | 636.5 | 233.6006 | 850.8 | 886.66 | 666.4 | 787.624 | 4.08 % |
| n100w100 | 857.8 | 857.4 | 608 | 1097.947 | 815.4 | 857.5 | 643 | 1462.868 | 4.94 % |
| n100w120 | 822.2 | 821.79 | 587 | 713.4136 | 743.8 | 822.21 | 601.2 | 802.3502 | 9.54 % |
| n100w140 | 765.6 | 765.03 | 557.5 | 1318.271 | 671 | 764.33 | 550 | 549.4516 | 12.36 % |
| n100w160 | 770.6 | 770.24 | 586 | 1346.483 | 690 | 770.53 | 555.6 | 512.3078 | 10.46 % |
| n100w180 | 774.2 | 773.75 | 564 | 987.1712 | 722.6 | 775.92 | 566 | 1009.613 | 6.66 % |
| n100w200 | 763.6 | 763.17 | 553.5 | 692.9886 | 711.6 | 765.61 | 559.6 | 826.071 | 6.81 % |

**Table 5**. The results of the hybrid SA for the SBP instances for scenarios 1–3.

| | | Scenario 1 | | | | Scenario 2 | | | | Scenario 3 | | | | BC Savings | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Instance | n | BC | EC | D | BT | BC | EC | D | BT | BC | EC | D | BT | S1 to S2 | S2 to S3 | S1 to S3 |
| rc201.1 | 19 | 383 | 382.01 | 248 | 0.002 | 335 | 382.01 | 248 | 0.002 | 297 | 382.01 | 248 | 0.002 | 12.53 % | 11.34 % | 22.45 % |
| rc201.2 | 25 | 683 | 682.58 | 455 | 0.021 | 620 | 682.06 | 454 | 0.004 | 579 | 682.06 | 454 | 0.005 | 9.22 % | 6.61 % | 15.23 % |
| rc201.3 | 31 | 711 | 710.93 | 469 | 1.368 | 633 | 710.46 | 469 | 3.156 | 615 | 710.93 | 469 | 1.332 | 10.97 % | 2.84 % | 13.5 % |
| rc201.4 | 25 | 796 | 795.25 | 538 | 0.027 | 733 | 795.25 | 538 | 0.007 | 708 | 795.25 | 538 | 0.007 | 7.91 % | 3.41 % | 11.06 % |
| rc202.1 | 32 | 622 | 621.62 | 441 | 0.324 | 542 | 621.62 | 441 | 0.295 | 520 | 621.62 | 441 | 0.556 | 12.86 % | 4.06 % | 16.4 % |
| rc202.2 | 13 | 258 | 257.98 | 171 | 1.413 | 226 | 257.98 | 171 | 0.781 | 204 | 258.98 | 173 | 0.036 | 12.4 % | 9.73 % | 20.93 % |
| rc202.3 | 28 | 808 | 807.59 | 546 | 0.031 | 738 | 807.59 | 546 | 0.031 | 719 | 807.59 | 546 | 0.031 | 8.66 % | 2.57 % | 11.01 % |
| rc202.4 | 27 | 765 | 764.76 | 508 | 0.378 | 698 | 764.76 | 508 | 0.321 | 678 | 764.76 | 508 | 0.912 | 8.76 % | 2.87 % | 11.37 % |
| rc203.1 | 18 | 384 | 383.36 | 266 | 0.793 | 339 | 383.36 | 266 | 0.78 | 327 | 388.69 | 270 | 0.719 | 11.72 % | 3.54 % | 14.84 % |
| rc203.2 | 32 | 663 | 662.88 | 456 | 1.192 | 583 | 662.88 | 456 | 0.989 | 564 | 662.97 | 456 | 0.509 | 12.07 % | 3.26 % | 14.93 % |
| rc203.3 | 36 | 626 | 625.88 | 442 | 18.63 | 536 | 625.88 | 442 | 18.341 | 498 | 625.88 | 442 | 3.082 | 14.38 % | 7.09 % | 20.45 % |
| rc203.4 | 14 | 264 | 264 | 169 | 0.001 | 229 | 264 | 169 | 0.001 | 219 | 264 | 169 | 0.001 | 13.26 % | 4.37 % | 17.05 % |
| rc204.1 | 45 | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | $\infty$ | NA | NA | NA |
| rc204.2 | 32 | 477 | 476.87 | 339 | 3.47 | 397 | 476.87 | 339 | 3.497 | 368 | 507.71 | 355 | 2.084 | 16.77 % | 7.3 % | 22.85 % |
| rc204.3 | 23 | 327 | 326.69 | 232 | 24.467 | 270 | 326.69 | 232 | 6.504 | 234 | 345.05 | 229 | 0.075 | 17.43 % | 13.33 % | 28.44 % |
| rc205.1 | 13 | 326 | 325.61 | 208 | 0.001 | 294 | 325.61 | 208 | 0.001 | 264 | 325.61 | 208 | 0 | 9.82 % | 10.2 % | 19.02 % |
| rc205.2 | 26 | 747 | 746.79 | 486 | 0.4 | 682 | 746.79 | 486 | 0.406 | 665 | 746.87 | 486 | 0.652 | 8.7 % | 2.49 % | 10.98 % |
| rc205.3 | 34 | 659 | 658.26 | 470 | 1.752 | 574 | 658.26 | 470 | 1.787 | 537 | 658.26 | 470 | 6.234 | 12.9 % | 6.45 % | 18.51 % |
| rc205.4 | 27 | 742 | 741.97 | 476 | 1.467 | 675 | 741.97 | 476 | 1.434 | 645 | 741.97 | 476 | 0.551 | 9.03 % | 4.44 % | 13.07 % |
| rc206.1 | 3 | 130 | 129.83 | 88 | 0 | 123 | 129.83 | 88 | 0 | 123 | 129.83 | 88 | 0 | 5.38 % | 0 % | 5.38 % |
| rc206.2 | 36 | 681 | 680.75 | 461 | 25.833 | 591 | 680.75 | 461 | 26.178 | 568 | 680.75 | 461 | 33.176 | 13.22 % | 3.89 % | 16.59 % |
| rc206.3 | 24 | 511 | 510.13 | 328 | 0.251 | 451 | 510.13 | 328 | 0.257 | 422 | 511.24 | 328 | 0.035 | 11.74 % | 6.43 % | 17.42 % |
| rc206.4 | 37 | 650 | 649.35 | 456 | 36.891 | 560 | 652.09 | 447 | 4.381 | 533 | 652.09 | 447 | 2.084 | 13.85 % | 4.82 % | 18 % |
| rc207.1 | 33 | 556 | 555.46 | 390 | 0.292 | 473 | 555.32 | 390 | 0.662 | 437 | 555.32 | 390 | 2.007 | 14.93 % | 7.61 % | 21.4 % |
| rc207.2 | 30 | 593 | 592.71 | 390 | 1.566 | 518 | 592.71 | 390 | 1.593 | 508 | 592.71 | 390 | 1.766 | 12.65 % | 1.93 % | 14.33 % |
| rc207.3 | 32 | 516 | 515.75 | 350 | 1.776 | 436 | 515.75 | 350 | 1.822 | 410 | 519.47 | 359 | 63.645 | 15.5 % | 5.96 % | 20.54 % |
| rc207.4 | 5 | 102 | 101.89 | 71 | 0 | 90 | 102.43 | 73 | 0.001 | 66 | 101.92 | 68 | 0.001 | 11.76 % | 26.67 % | 35.29 % |
| rc208.1 | 37 | 622 | 621.74 | 421 | 9.821 | 521 | 612.6 | 413 | 30.619 | 507 | 612.6 | 413 | 33.144 | 16.24 % | 2.69 % | 18.49 % |
| rc208.2 | 28 | 392 | 391.12 | 247 | 26.411 | 322 | 391.12 | 247 | 27.11 | 300 | 401.98 | 258 | 33.142 | 17.86 % | 6.83 % | 23.47 % |
| rc208.3 | 35 | 420 | 419.21 | 282 | 12.749 | 332 | 419.31 | 282 | 60.448 | 293 | 426.95 | 289 | 19.407 | 20.95 % | 11.75 % | 30.24 % |

Table 4 shows the results of the hybrid SA on GDE instance groups only for scenarios 1 and 3. Since there is no service time in GDE instances, so scenarios 1 and 2 become identical. The results in Table 4 show that scenario 3 provides a better BC option for the EVs in the E-TSPTW model presented in this paper. This is an expected result since there is no recharging in scenario 1 but partial recharging is included in scenario 3. When there is no recharging (i.e. scenario 1), the full BC of the EV should be greater than or equal to the EC throughout the tour. Otherwise, the EV cannot finish its tour. Recharging the EV battery at the customer locations during waiting and service time (i.e. scenario 3), on the other hand, reduces the needed initial BC for the EV. In the worst-case situation in scenario 3 (i.e. no service or waiting time), the maximum needed capacity of the EV is equal to the EC of the EV. In other words, EC gives the upper bound for the EV in scenario 3, while it becomes the lower bound for the EV in scenario 1.

Another observation in Table 4 is that the savings on the needed BCs are directly proportional to the width of the time window in the problem set, in general. As the time window width increases, the reduction on the needed BC also increases. As a result, the cost of the BC of the EV decreases. The BC savings in both Tables 4 and 5 were calculated as in Eq. (13).

$$BC\ Savings\ of\ S_i\ to\ S_j = \frac{BC\ of\ S_i - BC\ of\ S_j}{BC\ of\ S_i} * 100, \quad where\ i, j \in \{1, 2, 3\} \tag{13}$$

Although minimizing the total distance will contribute to finding a better BC for both scenarios, there are certain cases where the distance increases while the corresponding BC decreases from scenarios 1 to 3. This fact can be seen in 32 instances of GDE and 6 instances of SBP in the supplementary material. It is because, in some of the feasible tour permutations, more waiting occurs at the customer locations although the total distance of the EV gets worse. As the waiting time increases, the recharging amount also increases which leads to smaller BC. This fact shows that minimizing the BC of the EV does not necessarily minimize the total distance in all cases, hence they are two different objectives.

These same observations are also seen on the results of the hybrid SA for SPB instances in Table 5. Unlike in GDE instances, the number of customers in SBP instances cannot be derived from the names of the instances. For this reason, a column called “n” representing the number of customers in the instance was added in Table 5. Different than the GDE instances, the SBP instances do not contain a fixed time window width for each customer in them. Therefore, it is impossible to comment on the relationship between the time window width and the change of BC savings for SBP instances. Another difference between the GDE and SBP instances is that the latter one contains service times. Therefore, the hybrid SA provided results for scenario 2 for SBP instances. No infeasible solution for rc204.1 could be found by the hybrid SA for any scenario, so the related row in Table 5 was shown as $\infty$ and NA. Similar to the results for GDE, it is seen that recharging the EV during service time and/or waiting time at the customer locations reduces the needed initial BC for the EV. As the number of customers in the problem increases, in general, the ratio of BC savings also increases because the EV gets charged more during service and waiting times due to the number of customers.

## 5. Discussion

According to the experimental results for the proposed E-TSPTW in this study, it was observed that recharging the EV during the service time and waiting time at customer locations contribute to reducing the initially needed BC of the EV. Both scenarios 2 and 3 provide savings on the BC but inherently scenario 3 makes the most contribution. In GDE instance groups, the initial BC can be reduced up to 12.36% in scenario 3. Note that GDE instances do not have service times, if service times were included, this reduction amount would be larger. In SPB instances this saving can increase to 35.29% in scenario 3, as they include service times. Since the BC manufacturing cost consists of a big portion of the EV costs [1], these reductions in the initial BC will reduce the EV manufacturing and operating costs.

Time window width is one of the major factors that impact the BC savings. It is seen from the experimental results on the GDE instances that as the time window width expands, the BC savings increase. It is because as the time window width gets broader, then two things happen. First, waiting times increase, so that the recharging of the EV battery during the waiting time increases which, in turn, assists in BC savings. Second, the number of feasible solutions also increases. This way, the probability of finding better feasible solutions increases. Unfortunately, the running time of the algorithm increases, as well. This fact can be observed in the results of both CPLEX and the hybrid SA presented in all tables in this paper and the supplementary material. In CPLEX, the expansion of the time window width produces more feasible solutions. Therefore, the program either finds the optimum solution for a longer running time for the same number of customers, or cannot find the optimum solution within the running time limit although it provides better BC savings between scenarios 1 and 3. In case of the hybrid SA, the same observation can be made for the results for both GDE and SBP instances. As the time window expands, the BC savings get better between scenarios 1, 2 and 3, respectively, and yet feasible searching space also expands. This, in turn, makes isBetter($\pi_{neighbor}, \pi_{best}$) function to be called more often in Algorithm 2. As a result, the hybrid SA makes more comparisons and fitness evaluations, thus, this process extends the running time of the hybrid SA algorithm.

In addition, the experimental studies show the efficiency of the proposed hybrid SA method. The proposed hybrid SA performed well on the E-TSPTW model in this paper. Compared to the CPLEX results, it both achieved the optimum results found by CPLEX and provided the same or better results with much shorter running time. Besides, the hybrid SA found promising and feasible solutions on the instances that CPLEX could not be applied due to the size of these instances.

In real-life cases, the EC and BC calculations are more complex and require more instantaneous data. Even doing the same experimental study on the same problem set may produce different results due to the dynamic nature of certain parameters (e.g., speed, temperature, etc.) in real-life cases. In this study, as in many other studies in literature, many parameters were either fixed to a constant or ignored in the model. Since the purpose of this study is not to provide the most realistic EC and BC formulations, some of these real-life parameters were ignored. The purpose of this study is to provide an efficient model that works under different cases with modifications.

Nevertheless, this proposed model can easily be adapted by necessary changes that can occur in real-life cases. This can be done by including certain real-life parameters in the energy consumption model (Eq. 2) and recharging coefficient $g$ in (Eq. 1). For example, the energy consumption model proposed by Goeke and Schneider [12] provides such an example (Eqs. 14 and 15).

$$P_M = (m.a \ + \ \frac{1}{2}.c_d.\rho.A.v^2 \ + \ m.g.sin(\alpha) \ + \ c_r.m.g.cos(\alpha)).v \tag{14}$$

$$P_E = \phi.P_M \tag{15}$$

In these equations, $P_M$ is the mechanical power which an EV needs to overcome the rolling and aerodynamic resistance during its traverse. $m$ is EV's total mass, $a$ is acceleration, $c_d$ is aerodynamic drag coefficient, $\rho$ is air density, $A$ is frontal area of EV, $v$ is the speed of EV, $g$ is the gravitational constant, $\alpha$ is the gradient angle of the road, $c_r$ is the rolling friction coefficient depending on the factors such as tire and road surface conditions, and $\phi$ is the regression coefficient. $P_E$ is the energy consumption based on this mechanical power and regression coefficient.

Battery recharging rate highly depends on the brand, quality, and quantity of the materials used in EV battery. The exact information about the recharging rate depending on these factors can be obtained from the manufacturer. Thus, this rate can easily be replaced by the recharging coefficient $g$ in (Eq. 1) in this study.

Once this specific information is obtained, the proposed model in this study can be modified with these real-life parameters and applied on a specific problem. Due to the dynamic nature of most of these real-life parameters and the problem's dependency on the environmental settings, a single run of the proposed solution method in this study may not be enough. Since one size does not fit all, the solution methods can be applied to the same real-life problem instances more than once, and the average results can be obtained. Then, the average or worst-case situation results may provide the desired BCs for the real EVs.

## 6. Conclusion

In this paper, a new E-TSPTW model was proposed. The objective function in this model is to minimize the required initial BC of the EV. Three scenarios were taken into consideration in the problem: no recharging of the EV (scenario 1), recharging the EV during service time at customer locations (scenario 2), recharging the EV both during the service time and waiting time at customer locations (scenario 3). The impacts of recharging during service time and waiting time on the BC of the EV were observed. For keeping the originality of the instance sets, no extra charging station was added to the problem. Recharging of the EV was allowed to be done only at customer locations. The proposed model was solved with a hybrid SA algorithm and its efficiency was tested with IBM CPLEX for the smaller size instances used in the problem. The comparison results verify the efficiency of the proposed hybrid SA. The hybrid SA algorithm provided results for the two well-known TSPTW benchmark sets in the literature for the E-TSPTW model in this paper. The major contributions of this paper are providing a new E-TSPTW model based on BC minimization on three scenarios and the results for benchmark sets for the two well-known TSPTW instances for the problem being studied.

The EC used in the model requires only the distance information between the customers, demands of the customers, and the total capacity of the EV. It was aimed to provide a general EC model for the problem so that it can be applied to a greater number of instances in the literature. The model can be easily adapted to real-life situations by using additional parameters (e.g., vehicle speed, air resistance, road frictions, ambient temperature, etc.). Multiple test runs using different/additional parameter values on the same real-life instance can be made and evaluation can be done for the actual needed BC of the EV in real-life situations. The proposed model can efficiently be used for these purposes.

As a future study, this model can be modified for E-VRPTWs and applied to VRPTW instances. Besides, the relation between the distance, demands, EC, and BC can be analyzed in a multiobjective optimization problem structure. This way, a set of solutions can be offered from which the user can make his or her own decision.

## References

[1] Zhu M, Liu Z, Li J, Zhu SX. Electric vehicle battery capacity allocation and recycling with downstream competition. European Journal of Operational Research 2020; 283 (1): 365-379. doi: 10.1016/j.ejor.2019.10.040

[2] Juan AA, Mendez CA, Faulin J, De Armas J, Grasman SE. Electric vehicles in logistics and transportation: A survey on emerging environmental, strategic, and operational challenges. Energies 2016; 9 (2): 86. doi: 10.3390/en9020086

[3] Pelletier S, Jabali O, Laporte G. 50th anniversary invited article—goods distribution with electric vehicles: review and research perspectives. Transportation Science 2015; 50 (1): 3-22. doi: 10.1287/trsc.2015.0646

[4] Shaukat N, Khan B, Ali SM, Mehmood CA, Khan J et al. a survey on electric vehicle transportation within smart grid system. Renewable and Sustainable Energy Reviews 2018; 81: 1329–1349. doi: 10.1016/j.rser.2017.05.092

[5] Hu S, Chen P, Xin F, Xie C. Exploring the effect of battery capacity on electric vehicle sharing programs using a simulation approach. Transportation Research Part D: Transport and Environment 2019; 77: 164-177. doi: 10.1016/j.trd.2019.10.013

[6] Roberti R, Wen M. The electric traveling salesman problem with time windows. Transportation Research Part E: Logistics and Transportation Review 2016; 89: 32-52. doi: 10.1016/j.tre.2016.01.010

[7] Küçükoğlu İ, Dewil R, Cattrysse D. Hybrid simulated annealing and tabu search method for the electric travelling salesman problem with time windows and mixed charging rates. Expert Systems with Applications 2019; 134: 279-303. doi: 10.1016/j.eswa.2019.05.037

[8] Baek D, Chen Y, Macii E, Poncino M, Chang N. Battery-aware electric truck delivery route planner. In: IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED); Lausanne, Switzerland; 2019. pp. 1-6. doi: 10.1109/ISLPED.2019.8824835

[9] Erdogdu K, Karabulut K. Distance and energy consumption minimization in electric traveling salesman problem with time windows. In: 7th International Conference on Electrical and Electronics Engineering (ICEEE); Antalya, Turkey; 2020. pp. 160-164. doi: 10.1109/ICEEE49618.2020.9102602

[10] Potvin JY, Bengio S. The vehicle routing problem with time windows part II: genetic search. INFORMS Journal on Computing 1996; 8 (2): 165-172. doi: 10.1287/ijoc.8.2.165

[11] Doppstadt C, Koberstein A, Vigo D. The hybrid electric vehicle—traveling salesman problem with time windows. European Journal of Operational Research 2020; 284 (2): 675-692. doi: 10.1016/j.ejor.2019.12.031

[12] Goeke D, Schneider M. Routing a mixed fleet of electric and conventional vehicles. European Journal of Operational Research 2015; 245 (1): 81-99. doi: 10.1016/j.ejor.2015.01.049

[13] Keskin M, Çatay B. Partial recharge strategies for the electric vehicle routing problem with time windows. Transportation Research Part C: Emerging Technologies 2016; 65: 111-127. doi: 10.1016/j.trc.2016.01.013

[14] Cortés-Murcia DL, Prodhon C, Afsar HM. The electric vehicle routing problem with time windows, partial recharges and satellite customers. Transportation Research Part E: Logistics and Transportation Review 2019; 130: 184-206. doi: 10.1016/j.tre.2019.08.015

[15] Solomon MM. Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research 1987; 35 (2): 254-265. doi: 10.1287/opre.35.2.254

[16] Xiao Y, Zuo X, Kaku I, Zhou S, Pan X. Development of energy consumption optimization model for the electric vehicle routing problem with time windows. Journal of Cleaner Production 2019; 225: 647-663. doi: 10.1016/j.jclepro.2019.03.323

[17] Macrina G, Laporte G, Guerriero F, Pugliese LDP. An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. European Journal of Operational Research 2019; 276 (3): 971-982. doi: 10.1016/j.ejor.2019.01.067

[18] Markel T, Simpson A. Cost-benefit analysis of plug-in hybrid electric vehicle technology. World Electric Vehicle Journal 2007; 1 (1): 294-301. doi: 10.3390/wevj1010294

[19] Savelsbergh MWP. Local search in routing problems with time windows. Annals of Operations Research 1985; 4 (1): 285-305. doi: 10.1007/BF02022044

[20] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by Simulated Annealing. Science 1983; 220 (4598): 671-680. doi: 10.1126/science.220.4598.671

[21] Kalyanmoy D. An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering 2000; 186 (2): 311-338. doi: 10.1016/S0045-7825(99)00389-8

[22] Gendreau M, Hertz A, Laporte G, Stan M. A generalized insertion heuristic for the traveling salesman problem with time windows. Operations Research 1998; 46 (3): 330-346. doi: 10.1287/opre.46.3.330