Modeling a Periodic Electric Vehicle Routing Problem Considering Delivery Due Date and Mixed Charging Rates using Metaheuristic method

Maryam Elahi[1], Soroush Avakh Darestani[2, *]

[1] Department of Industrial Engineering, Faculty of Industrial and Mechanical Engineering, Islamic Azad University, Qazvin Branch, Qazvin, Iran, maryam.elahi7171@gmail.com

[2] Guidhall School of Business and Law, London Metropolitan University, London, United Kingdom, s.avakhdarestani@londonmet.ac.uk

Abstract

The coupling of ever-increasing consumption of fossil fuels around the globe with the decrease in the availability of fossil fuel supplies has led to increased cost of energy commodities, which together with ever-expanding requirements for reducing the level of environmental pollutions has resulted in an ever-increasing deal of attention to alternative transportation schemes such as electric vehicles (EVs). Since decades ago, national governments and environmental activists have initiated various efforts towards reducing atmospheric pollutions. A part of such effort has been focused on reducing the use of internal combustion vehicles and rather replacing them with EVs. In this research, we attempt to fill in this research gap by presenting a mathematical model for minimizing the sum of travelled distance and recharging cost of EVs per a given period and then solving it by simulated annealing (SA) algorithm. Results of the proposed algorithm were then compared to those of coding in GAMS for 30 different sample problems with different counts of customers, EVs, and charging stations. Numerical results indicated good efficiency of the metaheuristic algorithm in terms of processing time and solution quality. Indeed, with the SA algorithm, the processing time was seen to increase gradually with increasing the problem complexity, while the rate of increase in processing time was much steeper with the GAMS.

Keywords: electric vehicle (EV), green routing, delivery due date, mathematical programming, simulated annealing (SA).

1. Introduction

Global warming has become a serios threat to the humans and the nature. After adopting the so-called Paris Accordance by the United Nations Framework Convention on Climate Change in 2015, global efforts toward rapid reduction of CO_2 emissions have been accelerated. Fossil energy is the largest source of CO_2 emission in the world. Introduction of green transportation (*e.g.*, hybrid and electric vehicles or EVs) represents an excellent example of the efforts toward using non-fossil fuels. Nevertheless, green transportation suffers from difficulties in scheduling and routing for complicated product distribution systems. Product distribution scheme has long been a challenging issue. In the past, shipping (consider a letter, for example) was aimed at nothing but to deliver an envelope/parcel from the sender to the receiver at the destination – a presumption that ruled out the necessity o accurate scheduling. Today, however, product delivery has become crucially important because of various factors including air pollution, urban traffic, and possible savings in terms of time and money. Meanwhile, the demand time is almost known for particular products such as diaries for which customers demand on a relatively regular basis. Accordingly, shipping to this group of customers must be scheduled in such a way that shipping method is well determined for different periods. Considering such situations in the field of operational research following the general view to the product distribution, they usually represent periodic vehicle routing problems (PVRP). This means that the customers are not visited on a daily basis but rather on a periodic basis to satisfy their demand timely. Periodic decision-making imposes a substantial impact on the decisions adopted in various days. The fact that not every customer needs a delivery (product shipping) makes this problem a bit more complicated than the conventional vehicle routing problem. The reason of this increased complexity is that a conventional vehicle routing problem tries to address the routing for a particular day, while a periodic routing problem begins with assigning different customers to the corresponding visit days and then *T* vehicle routing problem is developed that must be solved. Therefore, green vehicle routing and scheduling are complex tasks in product distribution systems (Revesz *et al.*, 2014).

With the development of new technologies, various energies have been invented to power vehicles. Critical decrease in the availability of fossil fuel supplies, high cost of such fuels, and significance of environmental issues and air pollution have led to extensive development and rapid growth of electrical vehicles (EVs). A routing problem in which environmental concerns are further taken into consideration is referred to as a green vehicle routing problem. In this respect, the present research can be categorized in the field of green routing as it focuses on the EVs to lower environmental pollution.

Numerous research works have been reported on periodic routing for vehicles, but only a few of them have specifically focused on EVs. The fact that we further consider the delivery due date can add to the novelty and attractiveness of the present research to practitioners. Considering various factors related to EVs in the modeling for periodic vehicle routing, the present work represents an attempt to not only fill in the research gap but also get the problem closer to reality to improve its practical applicability. Goeke and Schneider (2015) presented an EV routing problem considering delivery time windows and a mixed transportation fleet including commercial EVs and conventional vehicles. In this model, the researchers assumed that the fuel consumption is a linear function of traveled distance, with equations of fuel consumption, velocity, slope, and parcel weight applied as constraints. They employed neighborhood search (NS) algorithm to solve the developed problem. In another piece of research, Keskin and Catay (2016) relaxed the full charging constraint to let the EV to be partially recharged – a relaxation that pushed the problem closer to reality while reducing the recharging time. They then presented an integer programming model and utilized adaptive neighborhood search (ANS) to have it solved. Their modeling outcomes indicated that partial recharging can improve the routing decisions and that the proposed methodology could outperform similar methods in converging to high-quality solutions. Raeesi and Zografos (2020) presented a capacity-constrained EV routing problem where the EVs could be recharged by mobile charger vehicles. They used large neighborhood search (LNS) algorithm for solving the problem. Based on the results, they further investigated differences between the proposed model and one where recharging was done at charge stations.

In a PVRP, the aim is to design routes for a vehicle for some particular period. The PVRP was first introduced by Beltrami and Bodin (1974) for addressing the assignment of municipal waste compactor trucks in New York. Francis and Smilowitz (2006) considered an PVRP into which service selection was further taken into account. The objective was to find a tour for each vehicle along the time horizon, where traveled distance minus service profit was taken as the objective function and vehicle capacity and minimum visiting frequency were seen as constraints. In this problem, a required visiting frequency was defined for each customer and a constraint was declared by which the visiting frequency to each customer was surely higher than that minimum required visiting frequency. Finally, they used an exact method based on lower bound and Lagrangian method for solving small- and medium-scale problems and presented a heuristic method for addressing large-scale problems. Baldacci and Bartolini (2011) formulated an exact method for solving PVRP. In this respect, they presented an integer programming model and then solved it using the exact solution method presented by Baldacci and Boschetti (2008) to come up with the lower bound to the problem. This methodology includes the calculation of a near-optimal dual solution that is obtained by relaxing inequality constraints. They then utilized the dual solution to generate a reduced integer problem including the exact solutions. Finally, the reduced problem was solved using an integer programming solver.

Optimization models can represent a physically existing phenomenon and, with help of rational mathematical relationships, produce the best possible mix of multiple variables for the sake of a particular objective, such as minimization of routing costs for EVs during a particular period of time. Nevertheless, unique features of optimization models have made them widely popular for routing problems. Noteworthily, an optimization model may not necessarily lead to actual optimality due to the special assumptions made to the problem complexity and/or assumptions made to build/run the model. Should the problem at hand is so large, evolutionary algorithms can be devised to avoid local optima and rather converge to universal optimum point. The evolutionary algorithms can identify the universal optimum point and have demonstrated promising potentials for solving real-world problems (Heidari *et al.*, 2019; Mirjalili *et al.*, 2016; Yang, 2010). So far, various researchers have been proposed numerous algorithms for solving optimization problems in the field of conjunctive utilization, of which one may refer to simulated annealing (SA) as an example. High efficiency of this algorithm for solving a wide spectrum of different optimization problems has convinced researchers toward using it for different applications. Accordingly, the present research, we use SA for minimizing the sum of distance traveled and recharging cost.

Table 1 summarizes the literature on the use of EVs. An overview of the relevant research shows that majority of previous researchers have sought to minimize distance traveled as well as the costs of recharging and energy as objective function. In Late 2019, researchers started to consider simultaneous impacts of C (Capacity of vehicle), TW (Time Window), MIX (Mix fleet), LRP (Location routing problem), F (fixed charging time), L (Liner charging time), NL (Non-liner charging time), and PR (Partial charging). Only few studies have been published during the indicated period, where researchers have merely considered linear charging (L) (Poonthalir and Nadarajan, 2018; Wang *et al.*, 2018). Hiermann *et al.* (2019) introduced a special VRP by combining conventional vehicles with EVs. EVs can avoid charging stations by using fossil fuels. This flexibility, however, leads to different decisions as one should identify the amounts of fossil fuel and electrical energy consumed by a hybrid vehicle as it goes through a particular route, and whether it needs to recharge its electrical batteries or not. In order to solve the problem, the authors designed a metaheuristic algorithm that was indeed a combination of genetic algorithm (GA) with LNS.

Table 1. A summary of literature review on EVs.

Objective function*	H	TD	BS	DFC	PR	NL	L	F	LRP	MIX	TW	C	Problem type**	Author & year
Minimizing the traveled distance					✓						✓		EVRPTW	Keskin and Çatay (2016)
Minimizing fuel and routing costs	✓												HVRP	vincent (2017)
Total cost of Traveled distance		✓		✓									EVRP	montaya(2017)
Operating costs of a taxi company		✓						✓		✓		✓	EVRP	Lu et al (2018)
Distance minimization			✓							✓	✓	✓	DARP-EV	Masmoudi et al (2018)
Distance minimization			✓		✓		✓		✓		✓	✓	MDEVLRP TW	Paz et al. (2018)
Sum of recharging costs				✓	✓	✓							EFV-CSP	Pelletier et al. (2018)
Fuel consumption and rout cost							✓						F-GVRP	Poonthalir & Nadarajan (2018)
Vehicle maintenance cost and facilities					✓		✓		✓		✓	✓	LRPIF	Schiffer & Walther (2018)
Travel time. Recharging cost, and charge consumption							✓						BEV routing	Wang et al. (2018)
Energy consumption												✓	E-VRP	Zhang et al. (2018)
Energy consumption				✓			✓				✓	✓	2sEVRP	Basso et al. (2018)
Count of vehicles and total traveled distance								✓					GVRP	Breunig et al. (2019)
Combination of travel time and recharging time				✓	✓	✓							E-VRP-NL	Froger et al. (2019)
Sum of fixed and variable costs	✓				✓		✓				✓	✓	H2E-FTW	Hiermann et al. (2019)
Route and battery replacement costs			✓									✓	2E-EVRP-BSS	Jie et al. (2019)
Travelling cost				✓	✓		✓	✓		✓	✓		MGVRP	Koyuncu & Yavuz (2019)
Energy costs per km of route		✓			✓	✓		✓		✓	✓	✓	GMFVRP PRTW	Macrina et al. (2019)
Costs of recharge, route, pollution emission limit					✓	✓		✓		✓	✓	✓	GMFVRP PRTW	Macrina et al. (2019)
Count of vehicles and total traveled distance								✓				✓	CGVRP	Normasari et al. (2019)
Minimizing total times (fuel						✓							EVRP	Zhang, G.

and travel)														(2020).
Minimizing total times (fuel and travel)						✓						✓	EVRP	kancharla (2020)
Minimizing total costs										✓			EVRPTW	Ramin raeesi (2020)
Minimize total time and costs	✓					✓					✓	✓	DVRP	Zhenzhong Wang (2021)
Cost of distance and recharge		✓		✓	✓						✓		***PEVRP***	***Current research***

* full-words of objective functions are given at Appendix A
**full-words of problem types are given at Appendix A

Contribution of this work:

Based on the literature review and Table 1, we found that PVRP considering delivery due and mixed charging rates is yet to be comprehensively studied. The novelty of the this work comes in the face of presenting a mathematical model where customers' orders are due for delivery at a programing horizon, given that some customers must receive their orders prior to the delivery due date while some others can wait until the end of the programing period. On the other hand, utilization of EV fleet in logistics has been increasingly regarded in recent research works, motivating us toward opting for an EV fleet for distributing the orders in multiple periods in this work. We herein attempt to fill in this research gap by presenting a mathematical model and solving it by a metaheuristic algorithm. The developed optimization model is then solved by a robust SA algorithm, with the results compared to those of a nonlinear optimization algorithm implemented in GAMS software.

2. Materials and methods

2.1. Research workflow

Given that the present research is aimed at minimizing the sum of traveled distance and recharging costs during each period, the framework shown in Figure 1 was followed to solve the relevant optimization problem. This was started by identifying the type of optimization problem and its assumptions. The assumptions and constraints of the problem, including the number of stations, maximum number of vehicles, continuity of the path, demand, and vehicle capacity, along with the mathematical model of the problem were designed appropriately. Some 30 sample problems with different numbers customers, vehicles, and charging stations were considered for different periods to evaluate the developed model. Each sample problem was launched independently of the others. The mathematical model was first implemented and run in GAMS, before having it solved by the SA algorithm. Results of the two algorithms on the 30 sample problems were compared to introduce the best algorithm for this problem. In the following, we begin by introducing the mathematical model and applied constraints and then continue to explain the SA algorithm and its structure. Details of the framework are illustrated in Figure 1.

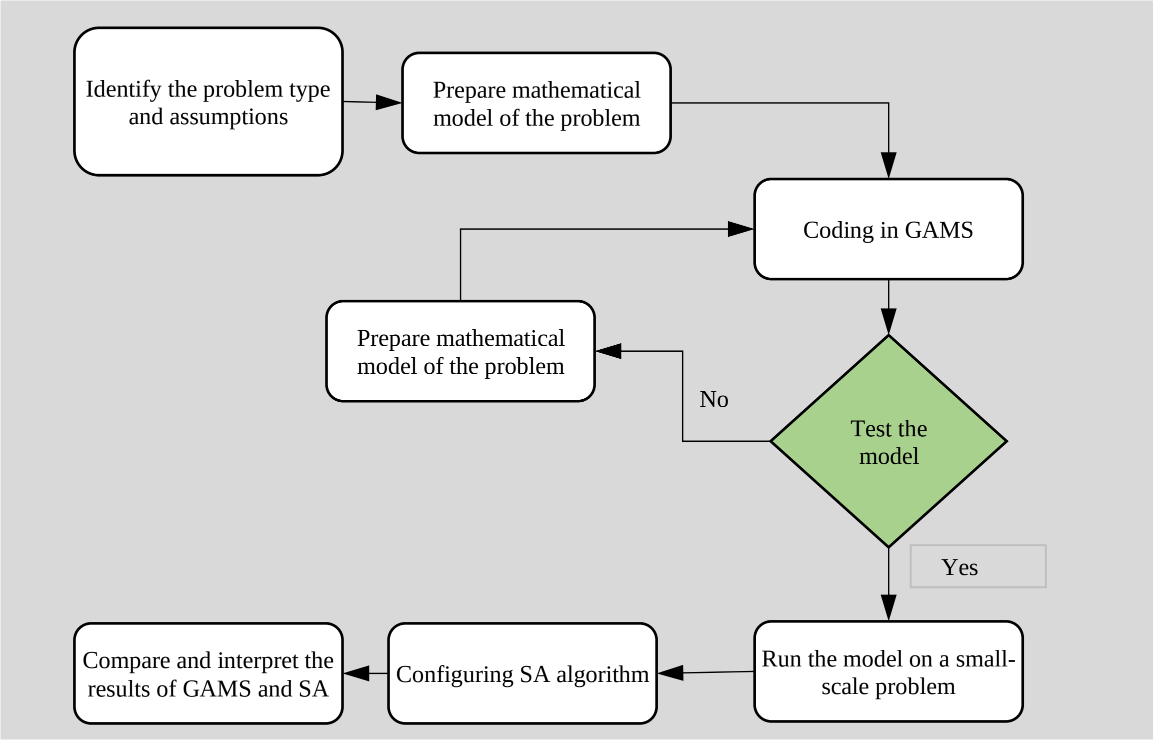

Identify the problem type and assumptions
Prepare mathematical model of the problem
Coding in GAMS
Prepare mathematical model of the problem
No
Test the model
Yes
Run the model on a small-scale problem
Configuring SA algorithm
Compare and interpret the results of GAMS and SA

Figure 1. Research framework

2.2 Mathematical modelling

To more understanding of the proposed model for this research, we have developed an illustration model in which describes the components of the problem and shown in Figure 2. In this illustration, there are two vehicles. The first vehicle starts its movement from the warehouse and then customer visits customer C1 in addition to visiting the customer C2 in this place performs recharge operations and then goes to the customer C5 and finally returns to the warehouse. However, the second vehicle performs recharge operations at F1 charging station.

Our main goal is to provide a mathematical model of periodic routing of electric vehicles with due date, in which customers are divided into two categories of customers with due date and without due date (optional)

In this section, the assumptions of the model of this research are explained.

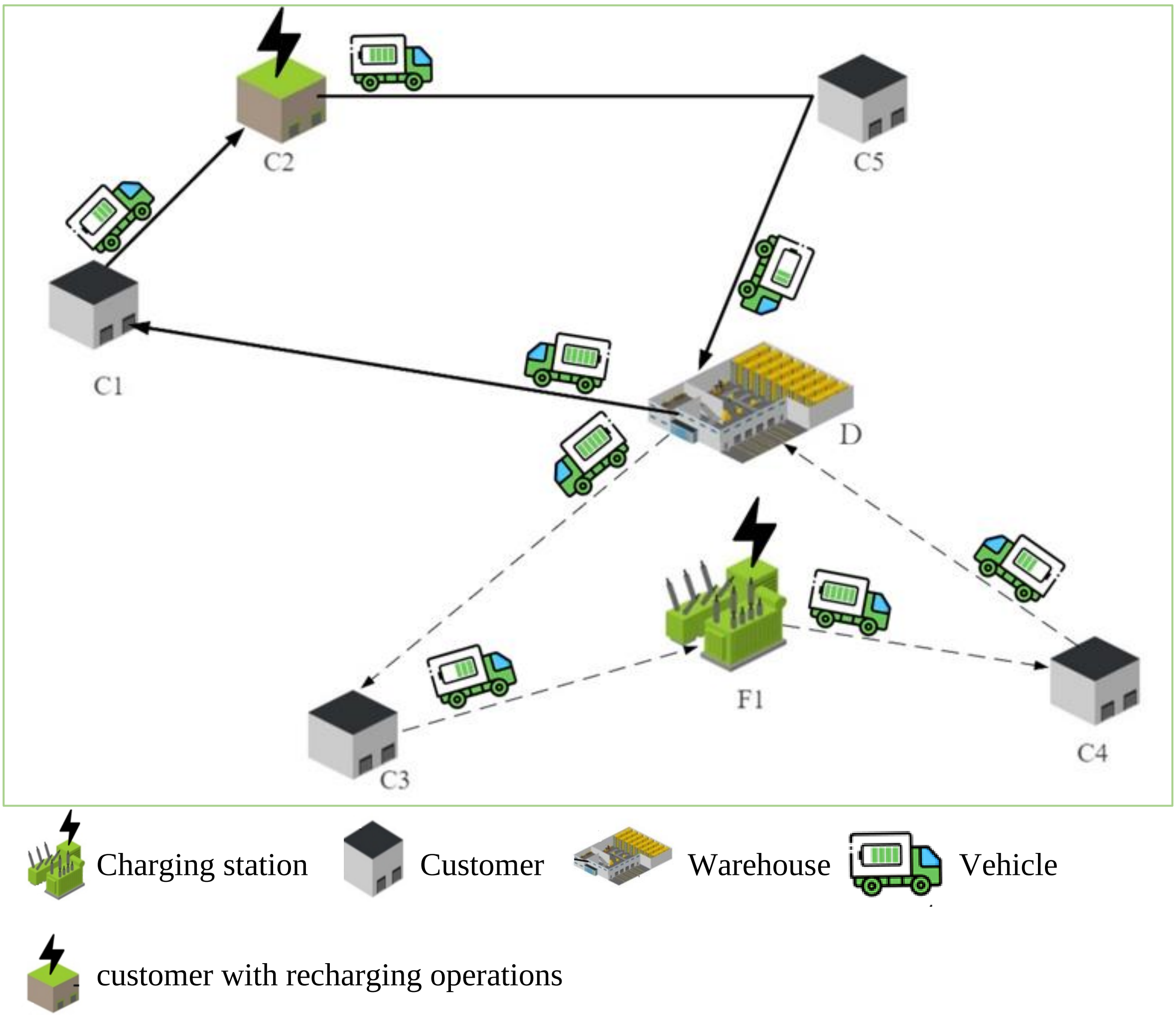


Figure 2. Illustration of the proposed electric vehicle routing Problem

Assumptions:

1. The inputs of the problem are definitive and fixed.
2. Customers with a due date should be visited prematurely, but customers without due date could be visited in any period of planning horizon.
3. The amount of used charge is dependent on mileage.
4. The amount of recharge at the charging station is complete and in the customers location could be partial.

Notations:

V: set of nodes

v': set of customers $v^{'} \subseteq V$

C: set of customers without due date $C \subseteq v^{'}$

V_f: set of charge stations $V_f \subseteq V$

H: set of time periods

d_{ij}: traveling distance from node i to j

$CAP:$ vehicle capacity

Q: battery capacity of the electric vehicle

h: energy consumption rate of the electric vehicle per unit of distance

r_i: time of ready to deliver for customer i's order

dl_i: due date delivery of customer i's order

q_i: demand of customer i

p_i: cost of each recharge unit at node i

a : cost coefficient per unit distance

m: number of available vehicles

Decision variables:

x_{ij}^t: binary variable and equal to 1 if the electric vehicle travels from node i to node j at period t, 0 otherwise

r_i^t: binary variable and equal to 1 if the electric vehicle is charged from node i to node j at period t, 0 otherwise

l_{ji}^t: total shipped demand on path ij at period t

y_i^t: level of vehicle charge while reaching to node i at period t

w_i^t: amount of charge taken by vehicle at node i at period t

u_i^t: Empty battery capacity of vehicle at node i at period t

2.2.1. Objective function

In this work, the objective function is to minimize the sum of traveled distance and cost of recharging the EVs during a particular period of time. In this regard, the first term in Eq. 1 refers to the traveled distance while the second term denotes total recharging cost.

In fact, Eq. 1 is the objective function of the model. The objective function is included two parts. The first part seeks to minimize total travelled distance. In addition, the second part aims to minimize total recharging cost, during each period.

$$Min\ z = \sum_{i,j\in V} \sum_{t\in H} a.\, d_{ij}.\, x_{ij}^t + \sum_{i\in V} \sum_{t\in H} p_i.\, w_i^t \qquad (1)$$

2.2.2. Constraints

Various constraints were applied to solve the optimization model, as expressed in the following. Eq. 2 ensures that customers with a due date are visited in the period from the end of lead time to the delivery due date. Eq. 3 guarantees that the customers with no due must be visited during any of the periods. Eq. 4 sets an upper bound of 1 to the times a charging station can be visited. In other words, this constraint ensures that a charging station is visited only if the vehicle needs a recharge. The fifth constraint (Eq. 5) refers to the path continuity, according to which there exists at least one output path from each node for each input path to that node. Eq. 6 guarantees that the number of utilized EVs per period will not exceed the available number of vehicles. Eqs. 7 and 8 evaluate the demand carried along each path during each period. These constraints eliminate the chances of sub-tours to develop. The 9th constraint (Eq. 9) ensures that the demand carried along each path does not exceed the vehicle capacity. Eq. 10 expresses that the vehicle's battery is fully charged when the EV leaves the warehouse, while Eq. 11 evaluates the battery level at the destination based on the charge level at previous check point. According to Eq. 12, if a recharging process is performed at a charging station in a particular period, the battery level reaches 100%. Eq. 13 sets that recharging at customer location increases the battery level to at most the battery capacity (allowing partial recharging at customer location). Based on Eq. 14, the charging the battery at customer location cannot increase the battery level beyond the battery capacity. Finally, Eq. 15 ensures that the battery level is at some feasible level (> 0) when a customer is visited. Eqs. 16 through 18 express the types and ranges of the decision variables.

$$\sum_{j\in v}\sum_{t\in[r_i,dl_i]} x_{ij}^t = 1 \qquad i \in v'\backslash C \tag{2}$$

$$\sum_{j\in v}\sum_{t\in H} x_{ij}^t = 1 \qquad i \in C \tag{3}$$

$$\sum_{j\in v}\sum_{t\in H} x_{ij}^t \leq 1 \qquad i \in V_f \tag{4}$$

$$\sum_{j\in v} x_{ij}^t = \sum_{j\in v} x_{ji}^t \qquad i \in v, t \in H \tag{5}$$

$$\sum_{j\in v} x_{0j}^t \leq m \qquad t \in H \tag{6}$$

$$\sum_{j\in v} l_{ji}^t - \sum_{j\in v} l_{ij}^t = q_i \sum_{j\in v} x_{ij}^t \qquad i \in v \cup V_f{}', t \in H \tag{7}$$

$$\sum_{j\in v} l_{0j}^t - \sum_{j\in v} l_{j0}^t = \sum_{j\in v} q_i x_{ij}^t \qquad i \in v \cup V_f{}', t \in H \tag{8}$$

$$l_{ij}^t \leq CAP.x_{ij}^t \qquad i,j \in A, t \in H \tag{9}$$

$$y_0^t = Q \qquad t \in H \tag{10}$$

$$y_j^t + h.d_{ij}.x_{ij}^t \leq y_i^t + w_i^t + Q\left(1 - x_{ij}^t\right) \qquad i,j \in V, t \in H \tag{11}$$

$$w_i^t + y_i^t = Q.r_i^t \qquad i \in V_f\,, t \in H \tag{12}$$

$$w_i^t + u_i^t + y_i^t = Q \qquad i \in v'\,, t \in H \tag{13}$$

$$w_i^t \leq Q.r_i^t \qquad i \in v'\,, t \in H \tag{14}$$

$$u_i^t \leq Q.(1 - r_i^t) \qquad i \in v'\,, t \in H \tag{15}$$

$$x_{ij}^{t} \in \{0,1\} \qquad i,j \in V, t \in H \tag{16}$$

$$r_{i}^{t} \in \{0,1\} \qquad i \in V, t \in H \tag{17}$$

$$l_{ji}^{t}, w_{i}^{t}, u_{i}^{t}, y_{i}^{t} \geq 0 \qquad i,j \in V, t \in H \tag{18}$$

where x_{ij}^{t} is 1 if the vehicle travels from the point i to the point j in the time period t, and 0 otherwise; l_{ji}^{t} is total demand traveled along the path ij in the period t; y_{i}^{t} is the battery level of the EV when it reaches the node i in the period t; w_{i}^{t} is the charge loaded into the EV battery at the node i in the period t; u_{i}^{t} is the discharged capacity of the EV battery when it reaches the node i in the period t; and r_{i}^{t} is a binary variable which takes a value of 1 if the EV is charged at the node i in the period t and 0 otherwise.

2.3. SA algorithm

Simulated annealing (SA) algorithm has been inspired by the annealing process in the metallurgy. It was first introduced by Kirkpatrick *et al.* in 1983. The SA algorithm emulates the process of material cool cooling gradually down to a temperature where the material takes a stable solid form. This algorithm converges toward optimal solution in a stepwise fashion by iteratively generating new solutions and having them evaluated. For this purpose, a new neighborhood is randomly generated and evaluated at each iteration. In this method, the points near a considered point in the search space are investigated. Accordingly, if a point was found to be superior (*i.e.*, returns a lower value of the cost function) to the considered point, it replaces it and serves as the new selected point in the search space, and if it is rather worse than the considered point (*i.e.*, returns a higher value of the cost function), it has still chances of being selected according to a probability function. To put it simply, in order to optimize the cost function, search is monotonously led toward lower values of the cost function although there are still chances that an increase in the cost function occurs. The acceptance of the next point is usually based on a criterion called metropolis criterion (Eq. 19).

$$P\{accept\} = \begin{cases} 1 & \Delta f < 0 \\ e^{-\frac{\Delta f}{c}} & \Delta f \geq 0 \end{cases} \tag{19}$$

where P is the probability of acceptance for the proceeding point, c is a control parameter, and Δf denotes the change in cost.

The control parameter in SA resembles the temperature in physical phenomena. Initially, the particle (*i.e.*, the current point in the search space) is fixed at its place by a huge level of energy (*i.e.*, a high value of control parameter, c). This high level of energy allows the particle to avoid from a local minimum, and as the search process continues, the particle has its energy level reduced (*i.e.*, c decreases) to an eventual point where the search leads to universal minimum. It must be noted that the chances of avoiding the local minimum decreases at lower temperatures, implying that the higher the initial level of energy, the higher the probability of ending up at a local minimum.

The SA optimization starts with a randomly generated initial solution for the decision variables and produces, again on random basis, a new solution near the initial solution using an appropriate neighborhood structure. Accordingly, an important issue in SA is the procedure for generating the neighborhood. The following three elements are required for implementing the SA algorithm (Figure 3):

1. Starting point: a point in the search space from where the search process is initiated. This point is usually set on random basis.
2. Motion generator: this generator is responsible for generating the next state. Accordingly, by calculating the costs at the current point and the next point, it determines the way algorithm moves toward the final solution.
3. A plan for annealing the parameters that determine the annealing scheme of the algorithm. This plan defines the initial and final temperatures and the rate at which the temperature decreases.

Different stages of this algorithm are shown in Figure 3.

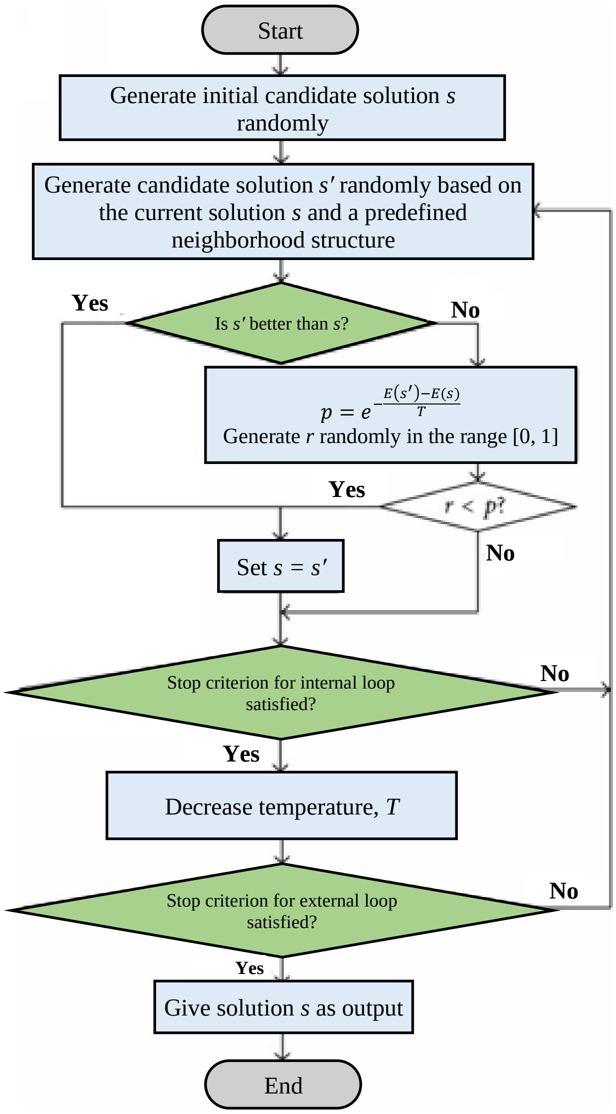

Start
Generate initial candidate solution s randomly
Generate candidate solution s' randomly based on the current solution s and a predefined neighborhood structure
Yes
Is s' better than s?
No
$p = e^{-\frac{E(s')-E(s)}{T}}$
Generate r randomly in the range [0, 1]
Yes
$r < p$?
No
Set $s = s'$
Stop criterion for internal loop satisfied?
No
Yes
Decrease temperature, T
Stop criterion for external loop satisfied?
No
Yes
Give solution s as output
End

Figure 3. Flowchart of SA algorithm (Rajan, 2010).

Steps demonstrated in Figure 3 can be explained as follows:

1. Generate an initial candidate solution on a random basis and evaluate the cost function, *Z*, for the initial candidate solution.
2. Randomly generate a new solution in the neighborhood of the current candidate solution and evaluate the cost function for this new solution.
3. Accept the new solution if it returns a better value of the cost function.
4. Conditional acceptance (if the new solution is not any better than).
5. Update the best solution found.
6. Decrease the temperature and go to step 2, if needed (*i.e.*, the stop criteria are not satisfied).

Before a problem can be solved by the SA algorithm, one should design a solution vector for the problem. For this purpose, a *T* × *I* matrix was designed where *T* is the number of programing horizons and *I* is the number of customers. All elements of this matrix were 0s and 1s, with the 1s indicating the period in which the customer *i* is visited (Table 1). For instance, consider 5 customers that must be visited in 3 periods. Accordingly, a random candidate solution for visiting the customers in different periods can be of the following form:

The matrix presented in Table 2 implies that the customers i_1 through i_5 are visited in periods 1, 3, 2, 2, and 3, respectively. When establishing this matrix, one should take care to assign 1 to the customers with delivery dues for the lead and delivery periods.

Table 2. Vector of customer visiting in different periods.

Periods/vector	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}	i_{12}
t_1	1	0	0	0	0	1	0	0	0	1	0	1
t_2	0	0	1	1	0	0	1	1	0	0	0	0
t_3	0	1	0	0	1	0	0	0	1	0	1	0

Following with the algorithm, one should identify a route for each vehicle and the customers at which place the recharging shall be done (Table 3). In order to assign the vehicles to different nodes and prioritize the visits and recharging cycles, a three-row matrix was used, where the first row defines the vehicle number to visit the considered node, the second row gives the visiting priority of the node, and the third row indicates whether the vehicle is to be recharged (1) or not (0) at that node. The visiting priority is a number in the range of [0, 1] whose lower value implies that the corresponding customer will be visited sooner. For example, considering 2 vehicles and customers (obtained from the previous matrix) that must be visited in the second period, Table 2 gives a random vector of customer assignment and visiting priority.

Table 3. Customer assignment and prioritization in period t_2.

i_3	i_4	i_7	i_8	fs_1	fs_2
2	1	2	1	1	2
0.245	0.746	0.409	0.162	0.510	0.014
0	1	0	0	0	0

3. Solution method and results

3.1. Taguchi’s method

Taguchi’s method is one of the most popular statistical methods in the field of design of experiments (DoE) for performing sensitivity analysis on the outputs of a process. It helps optimize different parameters of a process to achieve the best possible output by performing only a fraction of all experiments that were otherwise required to identify the optimal levels of all parameters. In this method, once finished with defining different levels for each of the parameters affecting the experiment outputs, several DoEs are proposed to the researcher, of which the researcher can opt for one depending on the selected levels and type of experiment. After performing the selected set of experiments, the results are returned back to the Taguchi’s method for analysis. Results of this analysis then led us toward understanding the effect of each factor on the dependent variable. This method compiles repeated data from the experiments into an indicator called signal-to-noise ratio (SNR). An objective of the Taguchi’s method is to maximize this SNR. Table 4 presents the levels considered for different parameters of the SA algorithm, where low, fair, and high levels are defined for each parameter. Table 5 gives a list of the experiments performed to find optimum values of different parameters. These include a total of 9 experiments. According to the table, the cost function returned the minimum value (6583.99) at the third iteration with T_0 = 3 and α = 3.

Table 4. Levels of different parameters of the SA algorithm.

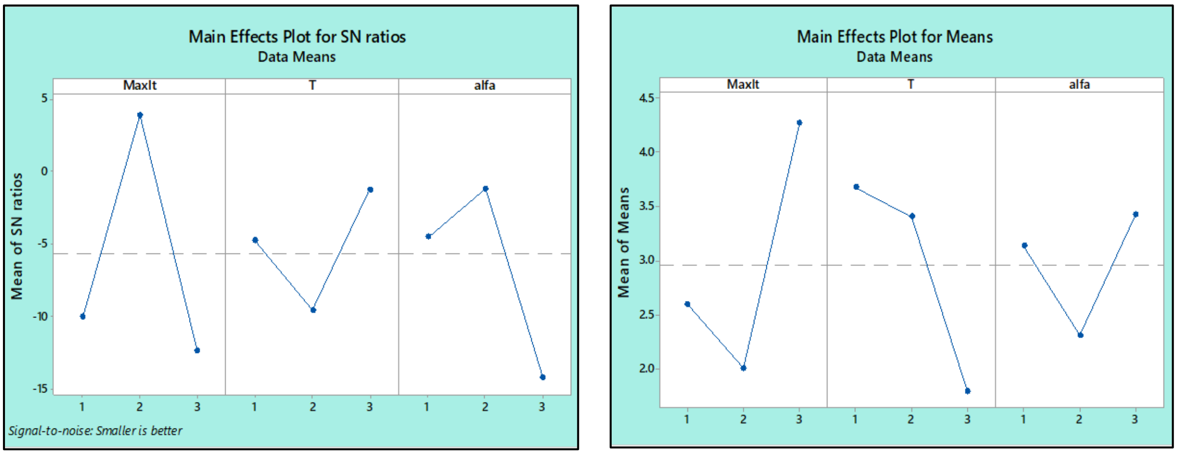
Parameters	Low(1)	Medium(2)	High(3)
MaxIt	1000	2000	3000
T0	50	75	100
alpha	0.8	0.85	0.9

Table 5. Designed experiments for SA algorithm.

Run	MaxIt	T0	alpha	Objective	RPD
1	1	1	1	6988.66	6.15
2	1	2	2	6691.86	1.64
3	1	3	3	6583.99	0
4	2	1	2	6595.23	0.18

5	2	2	3	6951.18	5.58
6	2	3	1	6600.51	0.26
7	3	1	3	6893.04	4.7
8	3	2	1	6780.84	2.99
9	3	3	2	6920.35	5.11

Parameter tuning was performed in Minitab software. Figure 4A shows that data repeating frequency was maximal in the second stage, while Figure 4B demonstrates that the frequency is minimal in the same stage. This indicates that, among the three ranges of number of iterations, the best results were anticipated with 2000 iterations. In a similar way, optimal values of T_0 and α can be found considering Figures 4A and 4B, as reported in Table 5.



(A) (B)

Figure 4. Results of Taguchi's method for optimizing the parameters of SA algorithm: (A) SNR plot of SA algorithm, and (B) means of means diagram.

Table 6 reports the values considered for the parameters of SA algorithm in this study. According to this table, minimum and maximum numbers of iteration have been set to 1000 and 3000, respectively, with the values of T_0 and α set to range in 50 – 100 and 0.8 – 0.9, respectively. Finally, optimum values of the three parameters are indicated in the same table.

Table 6. Levels of parameters of SA algorithm.

Parameters	Low(1)	Medium(2)	High (3)	optimum values
MaxIt	1000	2000	3000	2000
T0	50	75	100	100
alpha	0.8	0.85	0.9	0.85

First of all, the presented mathematical model was validated. For this purpose, a sample problem was generated and solved in GAMS. This sample is presented in Table 7. In this stage, maximum processing time was limited to 2 h, meaning that the GAMS software was set to process the problem for no longer than 2 h. The SA algorithm was implemented and run in MATLAB R2017b. Table 8 indicates dimensions of the problem in this research. As is evident from Table 8, 30 sample problems with different counts of customers, periods, charging stations, and vehicles were considered. It is worth noting that the number of customers ranged from 5 (for problems 1 and 2) to 120 (for problems 29 and 30). That is, as the problem number increased, the problem complexity increased in terms of the number of customers and vehicles. The developed problems were solved by GAMS and the SA optimization algorithm, with their results analyzed and evaluated properly.

Table 7. Dimensions of the produced problems.

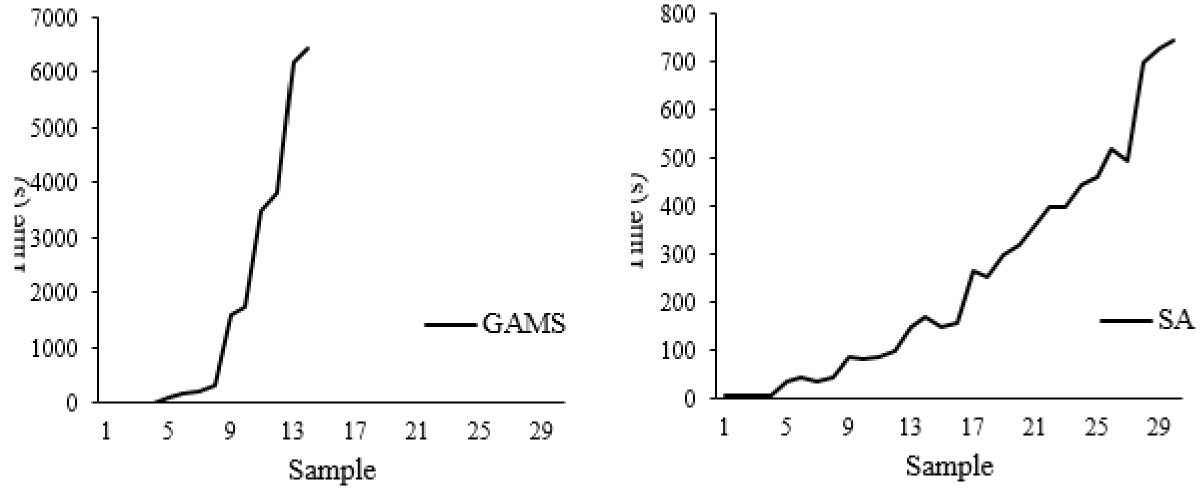
Vehicle	Charging station	Period	Customer	Sample problem	Vehicle	Charging station	Period	Customer	Sample problem
1	5	2	2	2	16	45	5	4	5
2	5	3	2	2	17	50	3	4	5
3	7	2	2	3	18	50	5	5	6
4	7	3	2	3	19	60	4	5	6
5	10	2	2	3	20	60	7	5	6
6	10	3	2	3	21	70	4	5	6
7	12	3	2	3	22	70	7	5	6
8	12	4	2	3	23	80	4	5	6
9	15	3	2	3	24	80	7	8	8
10	15	4	3	4	25	90	4	8	8
11	20	3	3	4	26	90	7	8	8
12	20	5	3	4	27	100	4	8	8
13	30	3	3	4	28	100	7	10	10
14	30	5	4	5	29	120	4	10	10
15	45	3	4	5	30	120	7	10	10

Table 3 reports the results of solving the 30 sample problems by the two models. Focusing on this table, it is evident that the GAMS could return the solution for problems 1 up to problem 14. That is, any further increase in the complexity of the problem made the GAMS fail to produce the solution within the defined maximum processing time. In initial sample problems, both models returned the same or closely similar values of cost function. However, with increasing the numbers of customers, vehicles and charging stations, superior power of SA algorithm over GAMS became clearly evident. For instance, processing time of the GAMS solver for problem 13 was as long as 6181 s while the same problem could be solved by SA algorithm within no longer than 151 s. This was while both methods returned very close cost function values. For small-scale problems, the processing time of the GAMS code was shorter than SA, but it took much longer as the problem complexity increased, so that the processing time went beyond our measurement window from the problem 14 on. In general, it was observed that the SA algorithm gives exact solution for small-scale problems, indicating the suitability of this algorithm for such problems, so that it can be used to solve the model presented in this research. On the other hand, it was evident that the processing time for the exact solution method increased exponentially with increasing the problem size, making it infeasible to apply for large-scale problems. This was while the processing time of the metaheuristic algorithm increased so slowly with increasing the problem size, indicating its suitability for solving large-scale problems.

Table 8. Values of cost function and processing time for running the sample problems.

Sample problem	Objective function		Processing time (second)		Optimality gap (%)
	GAMS	SA	GAMS	SA	
1	1158.594	1158.594	2.24	7.48	0
2	1175.352	1175.352	3.41	9.13	0
3	1263.331	1263.331	10.47	9.4	0
4	1301.31	1301.31	9.37	9.79	0
5	1571.671	1587.85	112.41	36.69	2.22
6	1602.315	1610.128	163.85	45.07	1.03
7	1592.205	1592.205	235.19	38.09	0
8	1658.427	1686.749	311.76	45.45	3.47
9	1817.051	1862.55	1589.78	87.16	4.67
10	1772.314	1809.73	1737.84	83.35	4.02
11	1863.484	1890.61	3491.64	86.94	2.66
12	1909.552	1948.96	3807.29	101.62	3.69
13	2315.105	2384.22	6181.33	151.05	4.69
14	2413.154	2510.84	6440.21	169.95	6.22
15	-	1744.68	-	151.02	-
16	-	1660.27	-	158.11	-
17	-	2688.45	-	267.04	-
18	-	3032.41	-	254.77	-
19	-	2955.78	-	299.5	-
20	-	3385.63	-	319.42	-
21	-	3648.59	-	360.19	-
22	-	3892.55	-	400.99	-
23	-	4144.22	-	397.78	-
24	-	3758.84	-	443.22	-
25	-	7681.19	-	459.32	-
26	-	7913.85	-	520.63	-
27	-	8147.57	-	493.36	-
28	-	10444.81	-	696.32	-
29	-	10854.37	-	725.29	-
30	-	11774.13	-	742.01	-

Figure 5 shows the evolution of processing time with GAMS and SA. As mentioned earlier, with increasing the sample problem size, the processing time increased exponentially when GAMS software was employed, while it increased at a rather slow rate with SA. As is evident from the figure, focusing on problem 14, the two models returned processing times that differed by about 6270 s, highlighting the efficiency of the SA algorithm for large-scale problems. In addition, GAMS needed some 6500 second to solve problem 14 while SA algorithm could solve the same problem within about 200 s. Ultimately, SA algorithm needed only 750 s to solve problem 30, far better than the result of GAMS. All of these results indicated high efficiency of the SA algorithm for large-scale problems.



(a)

(b)

Figure 5. Processing time for running the optimization model on different sample problems: (A) GAMS, and (B) SA.

In Figure 6, it is evident that the processing time grows exponentially with increasing the problem size, indicating NP-Hard nature of the problem considered in this research. Our results show that the SA algorithm can produce similar solutions to those of GAMS for small-scale problems. With increasing the number of customers and periods, however, the processing time of GAMS software increased so much that it failed to converge to solution following several hours. This was while the SA algorithm returned much better solutions for the same problem. Therefore, SA algorithm is recommended for cases with larger numbers of customers. Figure 5 demonstrates the changes in the value of cost function with the problem size. According to this figure, it is trivial that the cost function value increases gradually with the problem size when SA algorithm is the solution method of choice. This was while the GAMS model could provide solutions for problems 1 through problem 14, failing to address further problems due to extended processing time.

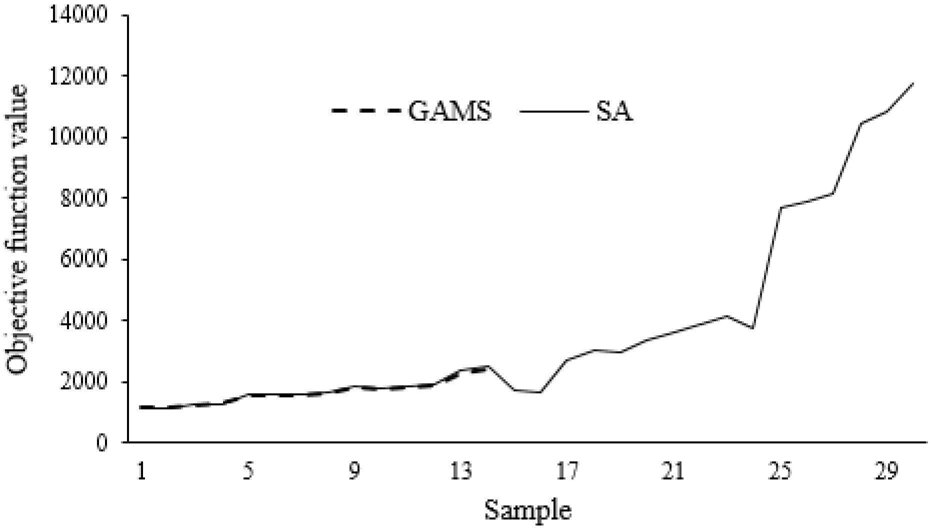


Figure 6. Values of cost function for SA and GAMS.

Lack of real data on the parameters of this research represents the main limitation suffered by this study. As of present, utilization of EVs in the logistics systems and distribution networks is very limited. Indeed, today’s EVs are mostly developed for personal transportation applications, with the EV manufacturers largely ignoring the development of EVs for product distribution applications. In general, it can be stipulated that most of the research on EV routing has been merely academic. Due to lack of required infrastructures for EV transportation in Iran, it is almost impossible to find an Iranian company working on EVs for distribution and logistics purposes.

4. Conclusion and further research

In this research, a mathematical model and a metaheuristic algorithm based on simulated annealing (SA) were presented for a multi-period electric vehicle (EV) routing problem. In order to evaluate the performance of the proposed SA algorithm, its results were further compared to those of GAMS optimization model on 30 different sample problems with the same cost function. Focusing on the electrical nature of the vehicles, we assumed that the EVs can be recharged at not only the charging stations but also customer nodes where charging facilities were available. Considering the periodic nature of the problem, we assumed two groups of customers, namely those with a delivery due date and those without one. Operationally speaking, the present research possesses different aspects for executives at distribution companies. Indeed, this model can be applied if customers’ demands and the corresponding delivery dues are known beforehand. Our results indicated the superior performance of SA algorithm compared to GAMS for complex problems. The results further showed that the SA can well solve the optimization problem within reasonably short processing times, as compared to GAMS, as the problem complexity increases. The results also showed that a requirement for achieving optimal solution is proper setting of the SA algorithm parameters. All in all, this model can serve as an appropriate decision-making model for managers in the distribution and logistics industry. There are some opportunity of future research based on the findings of this work for researchers. Further researches may be conducted by focusing on considering a time window for customers in the corresponding period, application of a charging function for forecasting the recharging time, considering the parameters with associated uncertainties, and utilization of other optimization algorithms, other metaheuristic methods and comparing & benchmarking the results.

Declarations:

Conflict of interest statement

Not applicable.

Ethical Approval:

The authors declare no conflict of interest.

Consent to Participate:

The authors declare that they agree with the participation of the journal.

Consent to Publish:

The authors declare that they agree with the publication of this paper in this journal.

Authors Contributions:

Maryam Elahi (ME) has contributed to the Methodology, Investigation, Project administration, Validation, Data collection, Software. Soroush Avakh Darestani (SAD) contributed to the Conceptualization, Writing - review & editing, Visualization and Supervision.

Availability of data and materials:

The authors declare that the data are not available and can be presented upon the requested of the readers.

Funding

There is no funding reported in this paper.

References:

Baldacci, R., Boschetti, M.A., Christofides, N. and Christofides, S., 2009. Exact methods for large-scale multi-period financial planning problems. Computational Management Science, 6(3), pp.281-306.

Baldacci, R., Bartolini, E., Mingozzi, A. and Valletta, A., 2011. An exact algorithm for the period routing problem. Operations research, 59(1), pp.228-241.

Basso, R., Kulcsár, B., Egardt, B., Lindroth, P., & Sanchez-Diaz, I. (2019). Energy consumption estimation integrated into the electric vehicle routing problem. Transportation Research Part D: Transport and Environment, 69, 141-167.

Beltrami, E.J. and Bodin, L.D., 1974. Networks and vehicle routing for municipal waste collection. Networks, 4(1), pp.65-94.s, 65, 111-127.

Breunig, U., Baldacci, R., Hartl, R. F., & Vidal, T. (2019). The electric two-echelon vehicle routing problem. Computers & Operations Research, 103, 198-210.

Francis, P. and Smilowitz, K., 2006. Modeling techniques for periodic vehicle routing problems. Transportation Research Part B: Methodological, 40(10), pp.872-884.

Froger, A., Mendoza, J. E., Jabali, O., & Laporte, G. (2019). Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions. Computers & Operations Research, 104, 256-294.

Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. Future generation computer systems, 97, 849-872.

Hiermann, G., Hartl, R. F., Puchinger, J., & Vidal, T. (2019). Routing a mix of conventional, plug-in hybrid, and electric vehicles. European Journal of Operational Research, 272(1), 235-248.

Jie, W., Yang, J., Zhang, M., & Huang, Y. (2019). The two-echelon capacitated electric vehicle routing problem with battery swapping stations: Formulation and efficient methodology. European Journal of Operational Research, 272(3), 879-904.

Kancharla, S. R., & Ramadurai, G. (2020). Electric vehicle routing problem with non-linear charging and load-dependent discharging. *Expert Systems with Applications*, *160*, 113714

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

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. Science, 220(4598), 671-680.

Koyuncu, I., & Yavuz, M. (2019). Duplicating nodes or arcs in green vehicle routing: A computational comparison of two formulations. Transportation Research Part E: Logistics and Transportation Review, 122, 605-623.

Lu, C. C., Yan, S., & Huang, Y. W. (2018). Optimal scheduling of a taxi fleet with mixed electric and gasoline vehicles to service advance reservations. Transportation Research Part C: Emerging Technologies, 93, 479-500.

Macrina, G., Laporte, G., Guerriero, F., & Pugliese, L. D. P. (2019). An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. European Journal of Operational Research, 276(3), 971-982.

Macrina, G., Pugliese, L. D. P., Guerriero, F., & Laporte, G. (2019). The green mixed fleet vehicle routing problem with partial battery recharging and time windows. Computers & Operations Research, 101, 183-199.

Mao, H., Shi, J., Zhou, Y., & Zhang, G. (2020). The electric vehicle routing problem with time windows and multiple recharging options. *IEEE Access*, *8*, 114864-114875.

Masmoudi, M. A., Hosny, M., Demir, E., Genikomsakis, K. N., & Cheikhrouhou, N. (2018). The dial-a-ride problem with electric vehicles and battery swapping stations. Transportation research part E: logistics and transportation review, 118, 392-420.

Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, *95*, 51-67.

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

Normasari, N. M. E., Yu, V. F., & Bachtiyar, C. (2019). A simulated annealing heuristic for the capacitated green vehicle routing problem. Mathematical Problems in Engineering, 2019.

Paz, J., Granada-Echeverri, M., & Escobar, J. (2018). The multi-depot electric vehicle location routing problem with time windows. International Journal of Industrial Engineering Computations, 9(1), 123-136.

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

Poonthalir, G., & Nadarajan, R. (2018). A fuel efficient green vehicle routing problem with varying speed constraint (F-GVRP). Expert Systems with Applications, 100, 131-144.

Raeesi, R., & Zografos, K. G. (2020). The electric vehicle routing problem with time windows and synchronised mobile battery swapping. *Transportation Research Part B: Methodological*, *140*, 101-129.

Rajan, C. C. A. (2010). Hybridizing evolutionary programming, simulated annealing and tabu search method to solve the unit commitment problem. *Journal of Electrical Engineering*, *10*(1), 8-8.

Revesz, R.L., Howard, P.H., Arrow, K., Goulder, L.H., Kopp, R.E., Livermore, M.A .Sterner, T., 2014. Global warming: improve economic models of climate change Nature 508 (7495), 173–175.

Schneider M., Stenger A., Goek D., (2014), The electric vehicle-routing problem with time windows and recharging stations, Transportation Science 48(4), 500-520.

Schiffer, M., & Walther, G. (2018). Strategic planning of electric logistics fleet networks: A robust location-routing approach. Omega, 80, 31-42.

Vincent, F. Y., Redi, A. P., Yang, C. L., Ruskartina, E., & Santosa, B. (2017). Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem. Applied Soft Computing, 52, 657-672.

Vincent, F. Y., Redi, A. A. N. P., Hidayat, Y. A., & Wibowo, O. J. (2017). A simulated annealing heuristic for the hybrid vehicle routing problem. *Applied Soft Computing*, *53*, 119-132.

Wang, Y., Bi, J., Guan, W., & Zhao, X. (2018). Optimizing route choices for the travelling and charging of battery electric vehicles by considering multiple objectives. Transportation Research Part D: Transport and Environment, 64, 246-261.

Xiao, Y., Zuo, X., Kaku, I., Zhou, S., & Pan, X. (2019). Development of energy consumption optimization model for the electric vehicle routing problem with time windows. *Journal of Cleaner Production*, *225*, 647-663.

Zhang, S., Gajpal, Y., Appadoo, S. S., & Abdulkader, M. M. S. (2018). Electric vehicle routing problem with recharging stations for minimizing energy consumption. International Journal of Production Economics, 203, 404-413.

Zhenzhong Wang, Kai Ye, Min Jiang, Junfeng Yao, Neal N. Xiong Gary G. Yen, Solving Hybrid Charging Strategy Electric Vehicle based Dynamic Routing Problem via Evolutionary Multi-objective Optimization, Swarm and Evolutionary Computation (2021), doi: https://doi.org/10.1016/j.swevo.2021.100975

Appendix A: Full-words list:

EVRPTW: Electric Vehicle Routing Problem with Time Window
GVRP: Green Vehicle Routing Problem
G-VRPPD: Green-Vehicle routing Problem with Pickup and Delivery
HVRP: Hybrid Vehicle Routing Problem
EVRP: Electric Vehicle Routing Problem
DARP-EV : Dial-a-Ride Problem with Electric Vehicle
MDVLRPTW: Multi-Depot Electric Vehicle Location Routing Problem with Time Windows
EFV-CSP: electric freight vehicle Charge Scheduling Problem
F-GVRP: Fuel efficient Green Vehicle Routing Problem
LRPIF: location-routing problem with intra-route facilities
2sEVRP: Two-stage Electric Vehicle Routing Problem
E-VRP-NL: electric vehicle routing problem with nonlinear charging
H2E-FTW: hybrid heterogeneous electric fleet routing problem with Time Window
BEVRP: battery electric vehicle routing
2E-EVRP-BSS: two-echelon capacitated electric vehicle routing problem with battery swapping stations
MFGVRP: mixed-fleet green vehicle routing problem

GMFVRPPRTW: green mixed fleet vehicle routing problem with partial battery recharging and time windows
CGVRP: Capacitated Green Vehicle Routing Problem
PEVRP: Periodic Electric Vehicle Routing Problem
DFC: design for changeability
BS: Battery substitute
TD: Time dependent
H: hybrid