Electric Vehicles Routing Problem With Variable Speed And Time Windows

Khaled S. Abdallah[1]
Yasmin Adel[2]

Arab Academy for Science, Technology and Maritime Transport (AASTMT), Egypt
The French University in Egypt, Egypt
Khaled.seif@aast.edu; Yasmeen.adel@ufe.edu.eg

Abstract: Vehicle routing is a major concern for a distribution channel of any supply chain. It plays a crucial role in attaining a competitive advantage for a company by being cost efficient or responsive. Transportation as a key logistics activity represents a relevant component (generally from 10% to 20%) of the final cost of goods, and one third to two thirds of the cost of logistics. The literature recently shifted towards the use of more energy efficient vehicles. Electric vehicles are characterized by being energy efficient, and do not produce polluting gas emissions such as carbon dioxide. However, the electric vehicles suffer from the limited capacity of the battery and the large charging time. In this paper, the dispatching and routing of battery-operated electric vehicles is considered. The vehicles can move at variable speeds when moving from a customer to another. When the speed is fast, the charge is depleted fast and small number of customers are served in a route. While when the speed is slow, the charge is depleted slowly, and more customers can be accommodated in a route. A genetic algorithm is developed to solve the problem. A piece linear range function based on finite speeds is proposed, as the average speed is used in planning for a given route in real life. A proposed genetic algorithm is proposed and applied on many cases from the literature. The results show that the model is able to optimize the performance and that the model behavior is consistent.

1. INTRODUCTION

Distribution of goods is a main activity in any supply chain. It can help the organization to boost its competitive strategy when it is aligned with the company's strategic objective. Distribution not only has a significant contribution on the product cost, but also results in gas emission and loss of energy. The use of electric vehicles is one of the proposed methods to reduce environmental emission from the distribution network. The large number of real-world applications has widely shown that optimizing the distribution process planning produces substantial savings (Toth and Vigo, 2002). Transportation as a key logistics activity represents a relevant component (generally from 10% to 20%) of the final cost of goods and one third to two thirds the cost of logistics (Swenseth and Godfrey, 2002; Tseng el al. 2005).

Vehicles as main transportation means have recently focused on improving the performance of electric vehicles as they are energy efficient and do not produce gas emissions. However, electric vehicles face several challenges. Storing the electric energy for later use and the drainage of the battery are major research areas with significant improvements to the battery performance introduced. Another concern is the charging time which takes a long period. There is a relation between the limited capacity of the battery which needs a long time to recharge, and the possible range that the vehicle can reach with the change in speed.

This research focuses on studying the routing and dispatching of electric vehicle with time and other resource windows requirements when the range of the vehicle is affected by the travel speed. Due to the trade-off between the speed and the range, when the time window of a customer is not in the immediate future, the EV can move slowly conserving power. In this case, the ability to accommodate more customers increases. On the other hand, if a vehicle moves faster, it may serve a customer whose time window is very close and cannot be served if the vehicle moves slowly. But, moving fast decreases the vehicle range and hence the ability to accommodate more customers in a route.

The rest of the paper is as follows. In section 2, the relevant literature is reviewed. In section 3, the problem is described along with its assumptions and limitations. A mathematical formulation is also introduced. In section 4, a solution approach is proposed and then tested in section 5. Finally, in section 6, the conclusions and findings are discussed with suggestions of future work.

2. LITERATURE REVIEW

The Capacitated constrained VRP (CVRP) is one of the first model introduced to solve the vehicle routing problem (Dantzig and Ramser, 1959). In CVRP, all customers correspond to requested deliveries, and demands are deterministic, known in advance and may not be split. The objective is to minimize the total cost while observing the capacity consumption of the vehicles. When the distance that can be travelled is also a constraint, as in the case of using an electric vehicle, the problem is then called a distance constrained VRP (DCVRP).

CVRP and DCVRP are known to be *NP* hard in the strong sense as they are a generalization of the travelling salesman problem (TSP) where there are capacity and distance constraints imposed by the problem.

According to the type of service offered by the distributer, the VRP can be classified into several categories, including Capacitated VRP (CVRP), Distance Constrained VRP (DVRP), VRP with backhauling (VRPB), VRP with Time Windows (VRPTW), VRP with Pickup and Delivery (VRPPD), VRP with Backhaul and Time Windows (VRPBTW), and VRP with Pickup, Delivery and Time Windows (VRPPDTW).

Wang and Shen (2007) consider electric bus scheduling problem that can be defined as a vehicle routing problem with fueling time and range constraints. They consider the objective function of minimizing the number of tours (or vehicles) and minimizing the total deadhead time. They present multiple ant colony algorithms to solve the Traveling Salesman Problem (TSP). They adopt several improvements on the route construction rule, and the pheromone updating rule is adopted.

Several solution approaches are used to solve the VRPTW. Exact solution methods include column generation to the VRPTW is tracked back to Desrosiers et al. (1984). Dumas et al. (1991) extend the problem to pick up-and-delivery with time window. Desrochers (1988) formulates the master problems of a few variations of the VPRTW as set covering problems and the subproblems as non-elementary shortest path problems with resource constraints.

Several variations are applied to the VRPTW such as soft windows as in (Qureshi *et al.*, 2009; and Taş et al., 2014), semi hard resource windows as in (Abdallah and Jang, 2014) stochastic travel times as in (Gutierrez *et al.*, 2018).

When it comes to the electric vehicles, Schneider et. Al. (2014) consider the inclusion of charging stations which consume time. Partial charging is considered in (Keskin and Çatay,2016). Lin et al. (2016) suggest that the battery consumption is affected by the weight a vehicle carries. An exact solution is developed by Desaulniers et. al. (2016). Gutierrez et al. (2018) consider a log-normal approximation to model the stochastic arrival time and propose a multi-population memetic algorithm to solve the problem. Xu at al. (2019) consider the case of green vehicle routing problem with soft time window. They develop an algorithm based on the non-dominated sorting genetic algorithm (NSGA-II), however, the soft time window is a relaxed model over the model considered in this paper.

3. PROBLEM DESCRIPTION

Consider a set V with N customers and a depot. Its network is $G=(V,A)$, where A is the set of arcs connecting customers with each other and with the depot, and $A=\{(i,j): i,j \leq N\}$. The coordinates of all nodes are known, and the lengths of the arcs are known as the Cartesian distances between their coordinates. Each customer requires a delivery of a certain quantity d_i and has an earliest time ,a_i, after which the delivery can take place and a latest time ,b_i, after which the delivery can not be carried out. The service at each customer takes a given time s_i. The departure time does not necessarily have to be within the time window, but the service must start within the customer's specified time window. An example of the considered network is shown in figure 1.

There are *K* electric vehicles available at the depot, each with capacity *D*. The vehicle can start after a given earliest start time and can only return by the latest return time. Without the loss of generality, all the vehicles are assumed to have the same capacity, earliest start times, and latest return times. When a vehicle arrives at a node before the earliest time, the vehicle can wait till the earliest start time to start the service and cannot start the service before this time.

The vehicles start from the depot with a full battery charge. The discharge function $\varphi^{-1}(i,j,v)$ represents the energy consumed when a vehicle moves from node *i* to node *j* at speed *v*. Each vehicle can use only the attached battery which has a total charge capacity of *Q*. The following assumptions are used in the model:

- Each vehicle starts fully charged and return to the depot after completing a route.
- A vehicle performs only one route.
- No intermediate charging occurs.
- The discharging function is known.
- The discharge during the idle time is assumed to be minor and it is not considered in the model.

The following symbols are used in the rest of this paper:

k index of the available *K* vehicles

i,j index of customers, each customer is represented by a node in the network graph

x_{ijk} binary decision variable for whether arc *i-j* will be served by vehicle *k*

c_{ij} transportation cost from *i* to *j*

D_{ik} cumulative demand just before serving customer *i* by vehicle *k*. This value is 0 when customer *i* is not served by vehicle *k*.

P_{ik} Cumulative power required to visit *i* using vehicle *k*

T_{ik} cumulative time consumption just before serving customer *i* by vehicle *k*. This value is 0 when customer *i* is not served by vehicle *k*.

d_{ij} distance between *i* and *j*

$\Delta^+(i)$, $\Delta^-(i)$ set of nodes that can be served directly after (before) *i* in *G(V,A)*

a_i, b_i the lower and upper limit of time window for customer *i*

S_i service time at node *i*.

E_i, L_i earliest start time and latest return time at the depot

$\varphi(i,j,v)$ energy consumed when a vehicle moves from node *i* to node *j* at a speed of *v*.

$\tau(i,j,v)$ time consumed when a vehicle moves from node *i* to node *j* at a speed of *v*.

Q_k battery k initial charge

v travel speed

Then the problem can be formulated as follows:

Min $\sum_{k\in K}\sum_{(i,j)\in A} c_{ij}x_{ijk}\,\varphi(i,j,v)$ (1)

Subject to

$\sum_{k\in K}\sum_{j\in\Delta^+(i)} x_{ijk}=1$ $\quad\forall i\in \mathrm{V}/\{0,\mathrm{n}+1\}$ (2)

$\sum_{j\in\Delta^+(0)} x_{0jk}=1$ $\quad\forall k\in \mathrm{K}$ (3)

$\sum_{i\in\Delta^-(j)} x_{ijk}-\sum_{n\in\Delta^+(j)} x_{jnk}=0$ $\quad\forall k\in K, j\in V/n+1$ (4)

$\sum_{i\in\Delta^-(n+1)} x_{i,n+1,k}=1$ $\quad\forall k\in K$ (5)

$T_{ik}+s_i+\tau(i,j,v)-T_{jk}+Mx_{ijk}\le M$ $\quad\forall(i,j)\epsilon A,\forall k\in K$ (6)

$(\sum_{j\in V} x_{jik})(a_i)\le T_{ik}\le(\sum_{j\in V} x_{jik})(b_i)$ $\quad(i,j)\in A,\forall k\in K$ (7)

$D_{jk}=\sum_{i\in v'}\left(x_{ijk}*\left(D_{ik}+d_j\right)\right)$ $\quad\forall i,j\in V,\forall k\in K$ (8)

$D_{ik}\le D$ $\quad i\in V,\forall k\in K$ (9)

$P_{jk}=\sum_{i\in v'}\left(x_{ijk}*\left(P_{ik}+\varphi(i,j,v)\right)\right)$ $\quad\forall i,j\in V,\forall k\in K$ (10)

$P_{ik}\le Q_k$ $\quad i\in V,\forall k\in$ (11)

$E_i\le T_{ik}\le L_i$ $\quad i\in\{0,n+1\},\forall k\in K$ (12)

$P_{0k}=0, D_{0k}=0$ $\quad\forall k\epsilon K$ (13)

$x_{ijk}\in\{0,1\}$ $\quad\forall k\in K,\forall(i,j)\in A$ (14)

The objective function (1) minimizes the total cost. Constraint (2) assigns every customer to just one vehicle. Constraint (3) allows a vehicle to originate from the depot once. Constraint (4) ensures that if a vehicle visits a customer, this vehicle has to leave that customer. Constraint (5) allows a vehicle to reach the depot once. Constraints (6) and (7) ensure that the cumulative time consumption through customer i when visited by vehicle k is within the its time window$[a_i,b_i]$. Constraints (8) and (9) ensure the total demand of a partial route up to reaching customer i when visited by vehicle k does not exceed the vehicle capacity. Constraints (10) and (11) ensure the total power consumption of a partial route up to reaching customer i when visited by vehicle k does not exceed the battery capacity. Constraint (12) imposes the boundary conditions at the depot. Constraints (13) and (14) impose the binary conditions.

4. SOLUTION APPROACH

A piecewise linear function is used in the proposed model instead of continuous discharging function to represent the power consumption. If a power discharge function is as shown in figure 1, then the range function is as shown in figure 2. As the moving speed between 2 given nodes will vary in the real life, so, a mean speed is used instead. As a result, the range function $R(v)$ can be approximated by a piece linear function $R'(v)$ that results from a finite set of speeds V={v_1, v_2, ...} as shown in figure 3. The modified discharge function can be represented as $\varphi^*(i,j,v)=\frac{d_{ij}}{R'(v)}*Q$ where d_{ij} is the travel distance, $R'(v)$ is the piece linear range function, and Q is the power capacity of the battery.

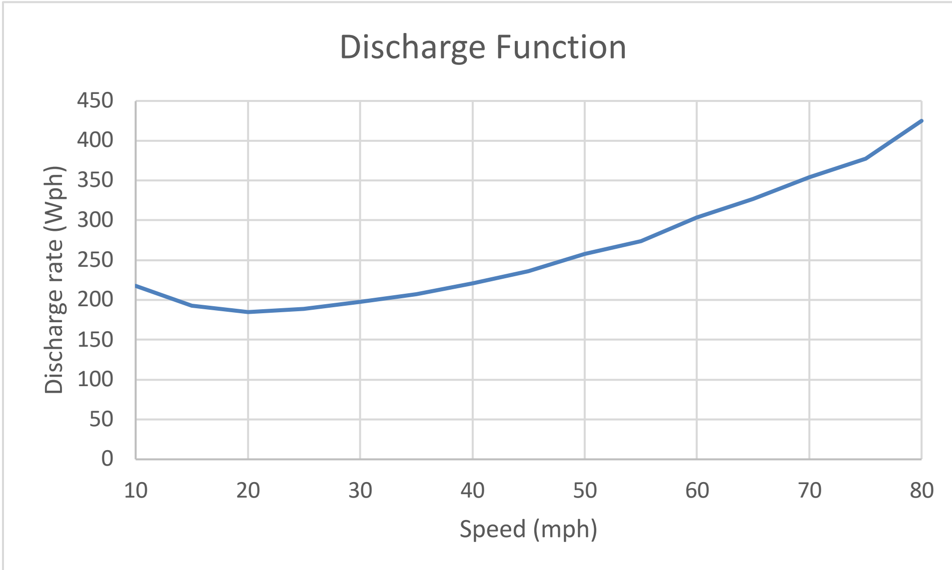

Figure 1. Discharge function

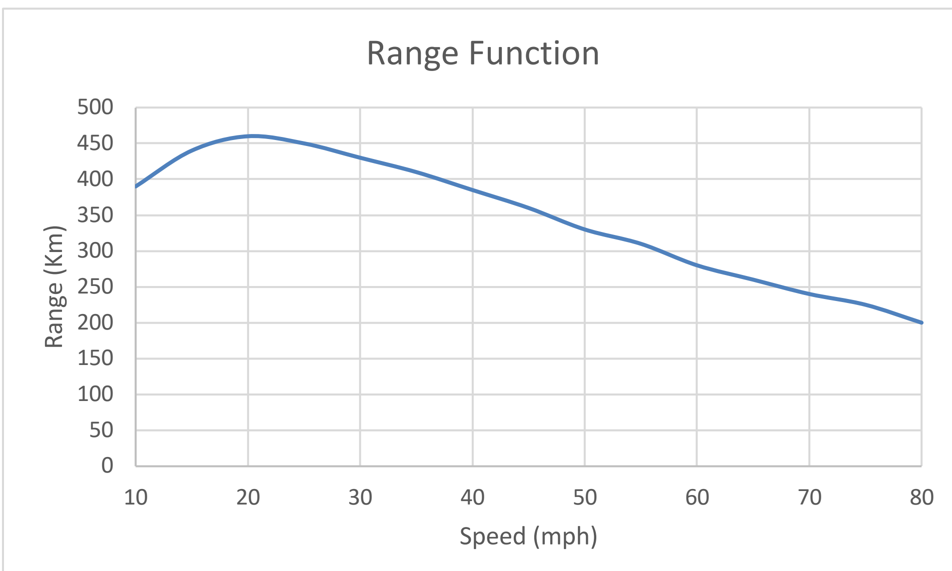

Figure 2 Range function

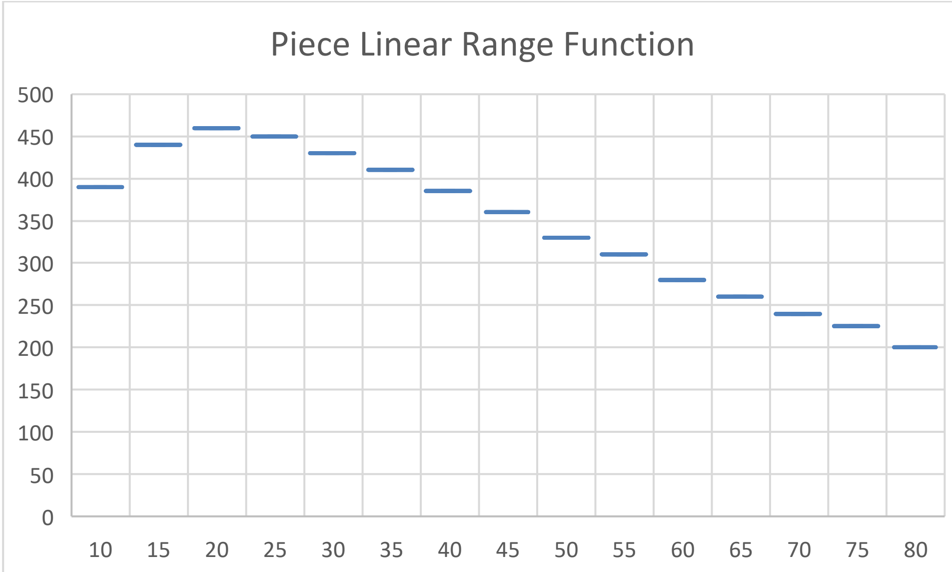

Figure 3 modified function

A genetic algorithm is proposed to solve the formulated problem. The chromosome in the proposed Genetic Algorithm is formed of two parts. The first part consists *n* cells that represent the order of visits of the nodes. The second consists of *n+1* cells representing the speeds of visiting different nodes. A given chromosome for 4 customers is shown in figure 4.

3	2	4	1	1	2	2	4	3

Figure 4. An example of the used chromosome for 4 customers.

The first part of the chromosome is used to interpret the visit order. Each vehicle is considered to follow the sequence in the chromosome until there is a customer that cannot be served in the current route under investigation due to various resources constraints. If the vehicle can feasibly return to the depot (i.e., satisfying the depot constraints), the route of the current vehicle is considered finished, and a new vehicle is considered. Otherwise, the last customer is dropped. This is repeated until the vehicle can feasibly return to the depot, and a new route using a new vehicle is started by the last dropped customer.

The second part of the chromosome is used in conjunction with the first part as the moving speed. When the first vehicle moves from the depot to the node of the first cell of the first part, it moves at the speed in the corresponding cell in the second part. The vehicle then moves to the node in the second cell in the first part using the speed in the second cell in the second part. The vehicle keeps moving at the corresponding speed until deciding from the first part that the vehicle needs to return to depot. The return speed is then determined as the slowest feasible speed to reach the depot. This is achieved by selecting the slowest speed to return, and compute the feasibility of the resource requirement. If not feasible, the next speed is selected and tested. This is repeated until reaching a feasible return speed. If there is no feasible speed to satisfy the resource requirements, the current node is dropped from the current vehicle route and the possibility to move from the previous node in the chromosome to the depot is examined. A new vehicle is considered to be moving from the depot to the first unassigned customer in the first part of the chromosome using the first unassigned speed in the second part of the chromosome. The algorithm is then repeated until all the nodes are visited.

It should be noted that that procedures require the first part of the chromosome as a permutation of the *n* nodes (customers). While the second part needs to be of *n+1* cell length.

When computing the fit of a chromosome, the fitting function computes the time, cumulative demand, cumulative power consumption, and the number of vehicles required to reach the next node from the current node. If the chromosome can serve all the customers feasibly, then the fit function returns the cost of the route (i.e., the sum of the cost of the power consumed, the cost of labor time and the fixed cost).

The initial population is formed by random permutation of the *n* customers available for the first part of the chromosomes, and then, random generation of one of the available speeds for a vector of size $n+1$ for the second part.

Several types of crossovers are used in the proposed algorithm. Two parents are selected and the first part of the chromosomes (i.e., the routing part) are exchanged keeping the speed part as shown in figure 5.

Another type of crossover is used by creating a binary vector *r* with a size that is equal to the chromosome's size. The first part of the vector is initialized with zero values. Then two numbers are generated randomly. These are used as start and end points of a portion of the generated vector that are replaced by ones rather than zeros. The second part of the generated vector *r* is formed of random binary values.

The generated child is formed by taking two parents and follow the sequence of the generated vector in different ways for the first and second parts. For the first part, if the current cell in vector *r* is 0, we take the content of the current cell from the first parent (e.g. 3), delete it from the first parent, and delete the cell with the same content of that cell (e.g. 3) from the second parent. If the current cell contains 1, we do the same but we take it from the second parent and delete it from the first parent. This is illustrated in figure 6. For the second part of the chromosomes, if the corresponding cell in vector *r* is zero, then the child inherits the current cell from the first parent; and, inherits it from the second parent if it is one.

Parent 1	3	2	4	1	1	2	2	4	3
Parent 2	2	4	1	3	3	1	1	2	4

Child 1	2	4	1	3	1	2	2	4	3
Child 2	3	2	4	1	3	1	1	2	4

Figure 5. Crossover of the first part of the chromosomes.

Parent 1	3	2	4	1	1	2	2	4	3
Parent 2	1	3	4	2	3	1	1	2	4
r	0	1	1	0	0	0	1	0	1
Child	3	1	4	2	1	2	1	4	4

Figure 6. Crossover for visiting order and speed

For the mutation, two ways of mutation are used. For the first part of the chromosome, a random value is generated corresponding to a cell. Then the content this cell is swapped with the cell whose order is equivalent to the content of the generated cell number. In figure 7, if the generated number is one, then the 1st cell is swapped with the 3rd cell where 3 is the content of the first cell.

The second mutation is for the second part of the chromosome (i.e., the speed part of the chromosome). A random number is generated, then the content of the cell with the generated order is randomly generated from the available mean speeds in the stepwise linear function proposed above. An example is illustrated in figure 8 where the fifth cell speed is changed from 3 to 2.

Parent	3	2	4	1
Mutated chromosome	4	2	3	1

Figure 7. Mutation visiting order when the first cell is chosen. Content of the first cell is replaced with the cell order as in the first cell (i.e., content of cell 3 which is 4).

Parent	1	3	4	2	3	1	1	2	4
Child	1	3	4	2	2	1	1	2	4

Figure 8. Mutation of the speed for the first cell in the second part of the chromosome.

5. NUMERICAL RESULTS

The numerical experiments were developed via MatLab 2016 on an intel i5-3470 at 3.2 GHz and 4 GB ram. The test data used is Solomon's data set (Solomon, 1987). The data sets used are for 100 customers for random (r), clustered (c), and mixed (rc) customers. We solve for 25, 50 and 100 customers. When solving for 25 (50) customers, we consider the first 25 (50) customers of the 100 customers given in any data set. There are 3 speed steps used in the numerical testing 15, 30, and 45 mph.

The initial population is formed of 50 parents. The routing parts are formed of a random permutation of the *n* customers chosen. The second parts of the chromosomes are formed as vectors of size *n+1* containing random values representing one of the speeds available.

The numerical testing is carried out to answer two questions. The first question is whether the proposed model improves the performance over using a fixed speed or not, and the second question is whether the proposed model is reliable when a test is replicated.

To answer the first question, the developed model using the same code is applied to all the test cases and replicated 3 times for the case of a fixed speed 15 mph, then, another time for the case when variable speeds are allowed. The best answer for each case is recorded and compared as shown in tables 1-3 for *r* data sets, *c* data sets, and *rc* data sets; respectively. The proposed model is able to find a better solution in all the cases. This shows the practical importance of the model. The model is able to overcome the tight time and distance constraints by moving the vehicles at a faster speed. This takes place despite the rapid depletion of the battery and the higher cost of the moving faster.

Table 1. Comparison of the best solution without speed variation vs the best solution when speed variation is allowed for the r data sets.

	R								
	25			**50**			**100**		
	One Speed	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**
101	3323.09	2773.81	16.5%	6999.34	5869.08	16.1%	13305.1	11184.6	15.9%
102	2645	2124.27	19.7%	5624.46	4548.14	19.1%	10674.9	8664.67	18.8%
103	1898.22	1476.2	22.2%	3992.7	3164.72	20.7%	7382.78	5858.46	20.6%
104	1510.63	1134.56	24.9%	2275.1	1696.06	25.5%	4394.96	3519.22	19.9%
105	1331.16	955.208	28.2%	2340.56	2005.75	14.3%	4426.34	3720.17	16.0%
106	1212.17	876.566	27.7%	2099.79	1702.35	18.9%	3748.71	3284.06	12.4%
107	1154.43	785.355	32.0%	1902.61	1561.64	17.9%	3502.96	2858.83	18.4%
108	1061.97	746.326	29.7%	1862.65	1344.47	27.8%	3131.22	2936.81	6.2%
109	1200.9	788.559	34.3%	2053.27	1546.11	24.7%	3629.42	3090.24	14.9%
110	1027.99	734.463	28.6%	1993.03	1437.4	27.9%	3159.59	2612.43	17.3%
111	1100.46	810.965	26.3%	2030.22	1425.57	29.8%	3557.25	2668.19	25.0%
112	993.064	703.946	29.1%	1807.15	1361.09	24.7%	3261.4	2497.53	23.4%

To answer the second question about the reliability of the model when it is run several time using the same parameters, the proposed model is run three times, then the best value is compared to the mean value of each data set as shown in table 4-6. The value chosen for the comparison is the decrease in the solution relative to the mean solution (i.e., (mean-best)/mean)). From the table, it could be concluded that the proposed model is neither sensitive to the increase of the number of customers nor to the distribution of the customers (i.e., random, clustered, or mixed). The decrease in the solution relative to the mean does not exceed 6.2%.

Table 2. Comparison of the best solution without speed variation vs the best solution when speed variation is allowed for the c data sets.

	C								
	25			**50**			**100**		
	One Speed	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**
101	3019.19	2513.85	16.7%	6430.57	5391.43	16.2%	15389.2	12935.9	15.9%
102	2323.91	1868.29	19.6%	4859.51	3973.54	18.2%	11920.9	9790.66	17.9%
103	1433.78	1132.8	21.0%	3307.85	2628.36	20.5%	8016.6	6463.5	19.4%
104	1011.17	788.456	22.0%	1649.67	1268.53	23.1%	4755.96	3782.3	20.5%
105	691.887	509.76	26.3%	1543.83	1274.69	17.4%	4458.91	4284.85	3.9%
106	2494.13	2010.08	19.4%	4068.75	3296.08	19.0%	6091.58	5534.44	9.1%
107	511.503	453.931	11.3%	1267.66	1034.31	18.4%	3964.46	3468.8	12.5%
108	511.503	428.41	16.2%	1114.55	1013.99	9.0%	3581.06	3524.98	1.6%
109	511.503	452.526	11.5%	1130.35	908.899	19.6%	3498.87	3100.59	11.4%

Table 3. Comparison of the best solution without speed variation vs the best solution when speed variation is allowed for the rc data sets.

	RC								
	25			**50**			**100**		
	One Speed	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**	**One Speed**	**Variable speed**	**Cost reduction**
101	1108.04	862.146	22.2%	2544.6	1992.74	21.7%	4803.13	4469.11	7.0%
102	924.398	809.706	12.4%	2073.44	1849.63	10.8%	4774.27	3738.58	21.7%
103	885.447	665.754	24.8%	2055.13	1548.22	24.7%	4326.22	3629.58	16.1%
104	800.625	605.503	24.4%	1815	1615.96	11.0%	4242.89	3386.71	20.2%
105	1796.69	1436.51	20.0%	3313.37	2708.53	18.3%	6579.76	5414.97	17.7%
106	911.253	691.693	24.1%	2144.48	1765.07	17.7%	4154.52	3647.13	12.2%
107	791.549	600.763	24.1%	1972.41	1608.68	18.4%	4239.66	3568.85	15.8%
108	790.699	600.148	24.1%	2008.06	1521.07	24.3%	4076.38	3290.31	19.3%

Table 4. Comparison of the best solution and the mean solution among replications for the r data sets.

	r								
	25			50			100		
	Best	Mean		Best	Mean		Best	Mean	
101	2773.8	2773.8	0.0%	5869.1	5869.1	0.0%	11184.6	11184.6	0.0%
102	2124.3	2124.3	0.0%	4548.1	4552.3	0.1%	8664.7	8680.2	0.2%
103	1476.2	1489.7	0.9%	3164.7	3179.2	0.5%	5858.5	5932.5	1.2%
104	1134.6	1168.4	2.9%	1696.1	1715.0	1.1%	3519.2	3563.7	1.2%
105	955.2	979.1	2.4%	2005.8	2040.2	1.7%	3720.2	3780.8	1.6%
106	876.6	905.9	3.2%	1702.4	1770.8	3.9%	3284.1	3375.8	2.7%
107	785.4	809.3	3.0%	1561.6	1618.1	3.5%	2858.8	3049.2	6.2%
108	746.3	747.1	0.1%	1344.5	1378.7	2.5%	2936.8	2951.2	0.5%
109	788.6	821.4	4.0%	1546.1	1556.5	0.7%	3090.2	3172.1	2.6%
110	734.5	758.6	3.2%	1437.4	1516.3	5.2%	2612.4	2702.0	3.3%
111	811.0	815.5	0.6%	1425.6	1469.7	3.0%	2668.2	2813.2	5.2%
112	703.9	718.2	2.0%	1361.1	1377.2	1.2%	2497.5	2527.3	1.2%

Table 5. Comparison of the best solution and the mean solution among replications for the c data sets.

	c								
	25			50			100		
	Best	Mean		Best	Mean		Best	Mean	
101	2513.9	2513.9	0.0%	5391.4	5391.4	0.0%	12935.9	12935.9	0.0%
102	1868.3	1869.1	0.0%	3973.5	3981.0	0.2%	9790.7	9798.1	0.1%
103	1132.8	1136.2	0.3%	2628.4	2645.9	0.7%	6463.5	6521.9	0.9%
104	788.5	799.4	1.4%	1268.5	1309.3	3.1%	3782.3	3927.8	3.7%
105	509.8	511.8	0.4%	1274.7	1290.7	1.2%	4284.9	4365.6	1.8%
106	2010.1	2011.7	0.1%	3296.1	3322.2	0.8%	5534.4	5586.1	0.9%
107	453.9	464.1	2.2%	1034.3	1093.0	5.4%	3468.8	3661.7	5.3%
108	428.4	442.2	3.1%	1014.0	1045.5	3.0%	3525.0	3707.2	4.9%
109	452.5	456.8	0.9%	908.9	947.2	4.0%	3100.6	3276.5	5.4%

6. CONCLUSIONS AND FUTURE WORK

Recently, there has been an increasing interest in reducing gas emissions from different transportation means. Hybrid and battery-operated vehicles serve this purpose as they do not consume power while idle, and, the efficiency of the motor is relatively high. However, storing electricity, the capacity of the batteries, and the charging time are major challenges. This research focuses on routing a fleet of electric battery-operated vehicles that can move at different speeds consuming different powers. The batteries are assumed to be fully charged at the beginning and the different customer requirements are satisfied.

The paper introduces a proposed genetics algorithm that is based on the idea that, for practical reasons, the cost of operating a vehicle on a route is, most of the time, evaluated using the average vehicle speed.

Moreover, in actual life, there may be a deviation from a fixed speed, so again the average speed could be used. So, the model proposed in this paper utilizes a piece-linear range function resulting from the average travel speed of a vehicle. It should be noted that increasing the number of speed steps in the proposed model makes the model close to the continuous case.

The proposed model was tested using Solomon's datasets. The model was proved to be effective in improving the solution over the case when a single speed is used and reliable in finding consistent solutions. The model shows that increasing the speed limits the range for the car and hence limits the number of customers that could be served along a route. On the other side, moving slowly extends the range and the ability to accommodate more customers; however, the slow serving time limits the number of customers. The proposed helps improving the performance under these trade-offs. Variable speeds can overcome the obstacles of tight windows or distance constrains at the expense of the battery capacity. The model is capable of routing the vehicles to satisfy the customer needs (demand, time…etc.) in addition to determining the average speed a vehicle should run at when moving between two successive customers.

This model can be extended in different ways for future works. The model can be extended to include intermediate charging stations, allow multi route for the vehicle, and include multiple routes.

Table 6. Comparison of the best solution and the mean solution among replications for the rc data sets.

	rc								
	25			**50**			**100**		
	Best	**Mean**		**Best**	**Mean**		**Best**	**Mean**	
101	862.1	903.0	4.5%	1992.7	2066.2	3.6%	4469.1	4646.6	3.8%
102	809.7	831.8	2.7%	1849.6	1930.8	4.2%	3738.6	3850.4	2.9%
103	665.8	672.1	0.9%	1548.2	1612.9	4.0%	3629.6	3651.6	0.6%
104	605.5	612.0	1.1%	1616.0	1683.6	4.0%	3386.7	3574.9	5.3%
105	1436.5	1471.8	2.4%	2708.5	2830.2	4.3%	5415.0	5490.4	1.4%
106	691.7	755.2	8.4%	1765.1	1833.7	3.7%	3647.1	3789.1	3.7%
107	600.8	616.1	2.5%	1608.7	1671.7	3.8%	3568.9	3679.1	3.0%
108	600.1	603.0	0.5%	1521.1	1573.5	3.3%	3290.3	3422.2	3.9%

7. REFERENCES

[1] Abdallah, K.S. & Jang, J. (2014) An exact solution for vehicle routing problems with semi-hard resource constraints. Computers & Industrial Engineering, 76, 366–377.

[2] Dantzig, G.B. and Ramser, J.H. (1959) The Truck Dispatching Problem. Management Science, 6, 80-91.

[3] Desaulniers G., Errico F., Irnich S. & Schneider M. (2016), Exact Algorithms for Electric Vehicle-Routing Problems with Time Windows, Operations Research,Vol. 64, No. 6

[4] Desrochers, M. (1988). An algorithm for the shortest path problem with resource constraints. Technical Report G-88-27, GERAD.

[5] Desrosiers, J., Soumis, F., & Desrochers, M. (1984). Routing with time windows by column generation. Networks, 14, 545–565.

[6] Dumas, Y., Desrosiers, J., & Soumis, F. (1991). The pickup and delivery problem with time windows. European Journal of Operational Research, 54(1), 7–22.

[7] Gutierrez, A., Dieulle, L., Labadie, N. and Velasco, N., 2018. A multi-population algorithm to solve the VRP with stochastic service and travel times. Computers & Industrial Engineering, 125, pp.144-156.

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

[9] Lin J., Zhou W.& Wolfson O. (2016) Electric Vehicle Routing Problem, Transportation Research Procedia, Volume 12, 508-521.

[10] Qureshi, A., Taniguchi, E. and Yamada, T., 2009. An exact solution approach for vehicle routing and scheduling problems with soft time windows. Transportation Research Part E: Logistics and Transportation Review, 45(6), pp.960-977.

[11] Schneider M., Stenger A. & Goeke D. (2014) The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations. Transportation Science, Vol. 48, No. 4

[12] Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problem with time window constraints. Operations Research, 35, 254–265.

[13] Swenseth, S. R. & Godfrey M. (2002) Incorporating transportation costs into inventory replenishment decisions. International Journal of Production Economics, 77(2):113-130.

[14] Taş, D., Gendreau, M., Dellaert, N., van Woensel, T. and de Kok, A., 2014. Vehicle routing with soft time windows and stochastic travel times: A column generation and branch-and-price solution approach. European Journal of Operational Research, 236(3), pp.789-799.

[15] Toth, P. and Vigo, D. (2002) The Vehicle Routing Problem. Siam, Philadelphia.

[16] Tseng, Y. Y., Yue, W. L., & Taylor, M. A. P. (2005) The role of transportation in logistics chain. Proc. East. Asia Soc. Transpn Stud, 5, 1657-1672

[17] Wang H.& Shen J, (2007) Heuristic approaches for solving transit vehicle scheduling problem with route and fueling time constraints. Applied Mathematics and Computation, v 190, n 2, Jul 15.

[18] Xu, Z., Elomri, A., Pokharel, S. and Mutlu, F., 2019. A model for capacitated green vehicle routing problem with the time-varying vehicle speed and soft time windows. Computers & Industrial Engineering, 137, p.106011.