Fastest Route and Charging Optimization of an Electric Vehicle With Battery's Life Consideration

Dariush Ebrahimi
Wilfrid Laurier University
Waterloo, Canada
debrahimi@wlu.ca

Seyedmohammad Kashefi
Lakehead University
Thunder Bay, Canada
skashefi@lakeheadu.ca

Maryam Helae
Trent University
Peterborough, Canada
maryamhelae@gmail.com

Sabah Mohammed
Lakehead University
Thunder Bay, Canada
sabah.mohammed@lakeheadu.ca

Fadi Alzhouri
Concordia University
Montreal, Canada
fadi.alzhouri@concordia.ca

ABSTRACT

The tremendous development in battery technology has made the use of electric vehicles (EVs) a reality and the growing usage of autonomous electric vehicles (AEVs) over the past couple of years has posed many serious challenges. To date, much IoT research exists on EV transportation systems in general, particularly in the routing, energy, and grid system balance. In this context, throughout this paper, we revisit the task of finding the fastest path for AEVs. In contrast to the state-of-the-art, we explore the capabilities of minimizing the traveling time and maximizing the battery life for the effective utilization of electric vehicles. We first model the problem mathematically to obtain optimal solutions for small instances, and then owing to the complexity of the optimization model, we propose several heuristic algorithms to solve the problem on large instances. Our analysis and evaluations, which are based on small and large networks of nodes, demonstrate the effectiveness of our proposed approaches and which one is superior in practice. We believe that this study provides a very strong step towards finding the optimal usage of AEVs in terms of time and battery life alike.

CCS CONCEPTS

• **Computing methodologies** → *Evolutionary robotics.*

KEYWORDS

Autonomous electric vehicle, battery life, charging, and routing.

ACM Reference Format:
Dariush Ebrahimi, Seyedmohammad Kashefi, Maryam Helae, Sabah Mohammed, and Fadi Alzhouri. 2023. Fastest Route and Charging Optimization of an Electric Vehicle With Battery's Life Consideration. In *Proceedings of the Int'l ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet '23), October 30-November 3 2023, Montreal, QC, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3616392.3623411

DIVANet '23, October 30-November 3 2023, Montreal, QC, Canada

ACM ISBN 979-8-4007-0369-0/23/10…$15.00
https://doi.org/10.1145/3616392.3623411

1 INTRODUCTION

Most people believe that the future is for electric vehicles, but all indicators show that this era is the era of electric vehicles (EVs) and the future is for autonomous electric vehicles (AEVs). In recent years, the demand for EVs has dramatically increased due to their low operational cost, Government policies and laws, emission-free, new battery technologies, and improving vehicle range [10]. In this context, establishing and finding smart urban environments and systems play a crucial role in obtaining the maximum benefits of AEVs. This imposes cooperation between engineers from different fields on one hand, and developers of computer systems, networks, and applications on the other hand. While researchers have worked to provide solutions to many challenges in AEV, such as safety, security, batteries, autonomous driving, routing, energy, and grid systems among others, approaches based on both autonomous charging and driving are modest and do not take optimal battery usage into account.

Motivated by the aforementioned information, this paper proposes a practical approach that finds the fastest route from the source to the destination of an EV, and leverages the optimum battery energy levels, considering the distribution of charging stations. Currently, not only making the decision on the right time and place for charging is an important process but also the amount of charging required to reach the destination is essential, especially since electric charging is still time-consuming compared to refueling with gasoline [2]. Therefore the aim is to automate and systematize the battery charging level, routing, and setting the right times and charging stations, to reach the destination in the shortest time and the optimal level of utilizing the battery as well. In particular, we outline the paper's contributions as follows:

- We mathematically formulate the problem of EV routing, maintaining charging levels for efficient battery life, finding appropriate charging stations on the route, and charging the battery based on demand, to minimize the traveling time from the source to the destination, and obtain optimal solutions for different graphs.
- Due to the complexity of the optimization model, we propose a time-efficient routing and charging (TERC) heuristic that decomposes the problem into two sub-problems: (i) selection of charging stations on the way to the destination, and (ii) determination of the optimal paths between the source, charging stations, and the destination. The TERC method

applies a dynamic programming approach to solve the second sub-problem and uses a greedy algorithm to solve the first one.

- To improve the performance of our heuristic algorithm, we propose a newer version of the TERC method, called TERC2, which incorporates a more sophisticated charging station selection procedure based on an ant colony optimization (ACO) algorithm. The ACO algorithm assigns a pheromone value to each charging station based on its proximity to the vehicle's current location and the predicted energy consumption until the next charging stop. The vehicle then selects the charging station with the highest pheromone value, and the procedure is repeated until the vehicle reaches its destination.
- For comparison purposes, we update the well-known state-of-the-art K-shortest path (KSP) algorithm to fit our problem by considering all required constraints and propose a new method called the K-fastest path (KFP) method. This method selects a path (from K paths) with available charging stations on the route that allows the vehicle to reach its destination with the minimum number of recharging stops and time, thus minimizing the total travel time.
- Finally, we evaluate and validate the proposed methods in small and large environments by varying the number of road intersections and the distribution of charging stations and using traveling time, battery usage, distance, and computation time as the performance metrics.

The remainder of this paper is organized as follows. In the following section, we review the literature on related work and characterize the uniqueness of the proposed approaches. In Section 3, we present the system model and describe the problem. Then, in Section 4, we formulate the problem mathematically and present the objective and constraints. The heuristic approaches are introduced in Section 5, and the performance of different methods is evaluated in Section 6. Finally, Section 7 concludes the paper and highlights future work.

2 RELATED WORK

The popularity of electric vehicles has grown over the past few years as a result of their affordability and environmental friendliness. Unfortunately, route planning is significantly hampered by the short driving range and lengthy charging times [9]. For this reason, various academics have proposed different strategies to determine the fastest route for EVs considering the location of charging stations along the route. In [13], the authors proposed a nonlinear programming model that takes into account EV speed and loads to reduce energy consumption. In their studies speed and loads have been considered as key factors in reducing energy consumption rather than distance as suggested in [7]. The latter assumed the scarcity of charging stations and the habits of full charging in their studies, which contradict the increase of charging stations nowadays, and that time will be wasted in charging a vehicle to its fullest. Similarly in [3], the authors modeled the problem as a Mixed Integer Linear Programming (MILP) by adding more factors to serve a fleet of EVs in a certain area. Whereas, in [11] the authors adopted the tabu search heuristic method and used the full charge strategy to solve the routing problem using time windows for a fleet of EVs. A machine learning technique was used by [8] to predict energy consumption along road segments for energy-efficient routing. In this context, the authors in [6] extended the energy consumption prediction model introduced in [5] to include the temperature and internal resistance of the battery. In their strategy, after predicting the EV energy consumption on all roads, by using one of the shortest-path algorithms, they tried to calculate and find the best energy-efficient route. It should be noted that the optimal energy-efficient route is different from the fastest route although the two objectives may be met in special cases. In addition, no battery recharging has been considered on the route of the EV from the source to the destination. There are very few research studies that consider recharging EVs. For example, the research study introduced in [1] coordinates the allocation of charging stations to a flow of EVs. However, this solution does not serve an individual vehicle in all situations. To extend battery life, the authors in [4] focused on the state of battery charging along the EV routing problem, and proposed four models to charge an EV battery, such as partial or full charging. Their goal is to reduce the delivery time while taking into account the battery life. They also mentioned that their suggested model did not work with large instances due to the increased complexity of the proposed model.

In this paper, we present a solution to the problem of routing and charging an electric vehicle to reach its destination as fast as possible, considering the extension of battery life. In contrast to [4], our proposed heuristic solutions can be applied to large instances.

3 SYSTEM MODEL AND PROBLEM DESCRIPTION

The system model, as shown in Figure 1, is presented with a weighted graph $G = (N, E)$, where N is a set of nodes (intersections and charging stations) in the scope of the system, and E is a set of edges (road segments) connecting any two nodes, and weighted based on the distance, traveling time, and battery level. The source and destination nodes are shown in the figure with black nodes, respectively with notations S and D. Whereas, the intersections and charging stations are respectively shown with blue and red nodes. The dotted arrow illustrates the EV traveling direction on the road visiting a set of nodes on its way from the source to the destination. We let the energy depletion and charging of the EV per time unit be $E_{Depletion}$ and $E_{Charging}$, respectively. For longevity of battery life, the maximum and minimum battery level percentages are kept respectively between B_{min} and B_{max}. We also let $I_{Station}$ represent the set of charging stations. For simplicity, we assume the EV takes a fixed time $T^{Edge}_{(i,j)}$ to traverse any edge. Consequently, the arrival time and battery level percentage of the vehicle on each visited node can be calculated, and they are highlighted in the graph respectively with T_i and B_i. The red arrow illustrates the direction to the nearest charging station, which is estimated to consume the energy of $E_{toStation}$.

Hence, the problem is to find the optimum route on the road, traversing a set of nodes from the source to the destination, such that the battery level maintains within B_{min} and B_{max}, choosing and charging EV at minimum charging station points as needed, which eventually minimizes the traveling time. In addition, the battery level of the EV at the destination should be at least enough

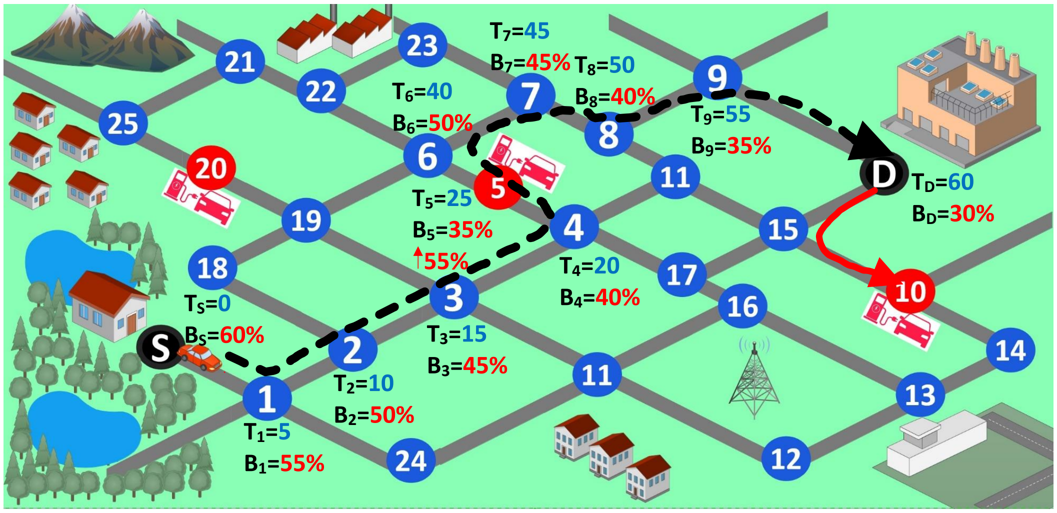


Figure 1: Illustration of the system model; the intersections are shown with blue nodes, charging stations with red nodes, the source and destination nodes respectively with black S and D nodes, and roads connecting nodes with links. The dotted black arrow shows the EV traveling direction on the road with arriving time (T_i) and battery level percentage (B_i) at each node.

to reach the nearest charging station, and still, its battery level is above B_{min}. In other words, the battery level of the EV at the destination should be $B_D = B_{min} + E_{toStation}$. The EV first collects all necessary information about the environment with the help of a roadside unit or base station and then solves the routing and charging problem. The figure shows an example where the battery level of the EV at the source is $B_S = 60\%$, and obviously, since the EV has not yet started, the traveling time is $T_S = 0$. Next, when the EV reaches node 1, if we assume $T^{Edge}_{(1,2)} = 5$ and $B_{Depletion} = 5$, then $T_1 = 5$ and $B_1 = 55\%$. Notice that the EV did not take the shortest path to the destination because the battery level could be below $B_{min} + E_{toStation}$. Therefore, the EV had to change its course at node 4, to reach a charging station at node 5, to charge with enough energy to let it get to the destination satisfying the battery level constraint. The EV arrives at node 5 at $T_5 = 25$ and with a battery level of $B_5 = 35\%$. It charged with 20% only within 5 unit time to get to 55% battery level and then continued its course to node 6. When arrived at that node, as shown in the figure, the battery as expected was $B_6 = 50\%$ (i.e., $B_6 = B_5 - E_{Depletion}$). Finally, the EV arrives at the destination in $T_D = 60$ with a battery level of $B_D = 30\%$. Note that the EV has enough battery energy to get to the nearest charging station (node 10) and still its battery level will be equal or above B_{min} if we assume $B_{min} = 20\%$.

4 PROBLEM FORMULATION

In this section, we mathematically formulate the problem as a mixed integer linear programming (MILP). The used notations are listed in Table 1. Let T_D indicate the time required for the EV to reach its destination. The objective of the optimization model is to minimize the arrival time of the EV. It can be mathematically formulated as follows:

$$Min \qquad T_D \tag{1}$$

subject to: (2) - (9), where these constraints are derived in detail in subsections 4.1 to 4.4.

4.1 Routing Constraints

In this subsection, the route for the EV from its source (starting point, S) to its destination (D) is constructed. Let $R_{(i,j)} \in \{0, 1\}$ indicate whether link (i, j) is set on the path of the EV from the source to the destination (i.e., $R_{(i,j)} = 1$, if link (i, j) is on the route of the EV and zero otherwise). The route construction is mathematically formulated as shown in eq. (2).

$$\sum_{j:(i,j)\in E} R_{(i,j)} - \sum_{j:(j,i)\in E} R_{(j,i)} = \begin{cases} 1, & i = S; \\ -1, & i = D; \\ 0, & otherwise. \end{cases} \tag{2}$$

It is obvious that the above constraint obtains the differences between the number of outgoing links to incoming links on node i for route construction. If node i is the source of the EV, obviously the difference between the number of outgoing links to incoming links is equal to 1 and equal to -1 when node i is the destination node since the total number of assigned incoming links to node i is one and the total number of assigned outgoing links is zero, hence the subtraction result is -1. Consequently, when node i is a relay node or neutral (none of the above), the difference is zero; that is, if there is an incoming link to a relay node (for route construction), there should be an outgoing link, and if there is no incoming link (not on a route), there must be no outgoing link as well; so the subtraction in both cases is equal to zero. To avoid a loop, the following constraint restricts the edge to be active in one direction but not in both directions at the same time.

$$R_{(i,j)} + R_{(j,i)} \leq 1; \qquad \forall (i, j) \in E \tag{3}$$

4.2 Travel Time Constraints

The EV's presence at any node except the source means reaching that node has taken a while and the time increases as the EV travels to another node. To evaluate the arrival time of the EV to any visited node, let T_i be the arrival time of the EV to node i. Constraint (4) obtains the arrival time of the EV at node i (i.e., T_i) after traversing link (h, i). If node h is a charging station and the EV has charged its

Table 1: Notations used in the problem formulation

Parameters		
N	Set of nodes.	
E	Set of edges (links).	
S	Starting point (source) of the EV.	
D	Destination of the EV.	
L	Large constant (larger than any possible system time).	
$I_{Station}$	Set of charging stations.	
$T^{Edge}_{(i,j)}$	Time to traverse edge (i, j).	
B_{Max}	Maximum battery level.	
B_{Min}	Minimum battery level.	
$E_{toStation}$	Required energy to get to the nearest station.	
$E_{Depletion}$	Energy depletion per time unit.	
$E_{Charging}$	Energy charging per time unit.	
Decision Variables		
T_D	≥ 0	Arrival time at destination.
2*$R_{(i,j)}$	2*$\in \{0,1\}$	Indicate whether link (i, j) is set on the route of the EV.
B_i	≥ 0	EV's battery level at node i.
T_i	≥ 0	EV's arrival time at node i.
2*C_i	2*$\in \{0,1\}$	Indicate whether the EV has been charged at node i.
G_i	≥ 0	Required time to charge EV at station i.

battery there, then the charging time G_h is added to the traversed time of the EV on edge (h, i). It is noted that when edge (h, i) is not on the route of the EV (i.e., $R_{(h,i)} = 0$), then the constraint is always satisfied since the left-hand-side of the inequality is very large (i.e., equal to the large constant L).

$$T_i + L(1 - R_{(h,i)}) \geq T_h + T^{Edge}_{(h,i)} + G_h, \quad \forall (h,i), \forall i \in N : i \neq S \tag{4}$$

4.3 Charging Constraints

In this work, to restrict charging only to charging stations when needed, nodes belonging to subset $I_{Station} \subseteq N$ of set N are considered to act as charging stations. And since charging can only occur at some charging stations depending on the energy level of the battery, $C_i \in \{0, 1\}$ indicates whether node i is a charging station or not as shown in (5).

$$C_i = \begin{cases} 0, & i \notin I_{Station}; \\ 1, & i \in I_{Station}, \end{cases} \tag{5}$$

This constraint assures that when a node i is not a charging station, then no EV will be charged at that node, and therefore, $C_i = 0$, while $C_i = 1$ indicates that node i is a charging station. Therefore, it is normal to use constraint (6) to make sure that the charging time G_i on the non-charging station node, where $C_i = 0$, is zero and it is positive value $G_i > 0$ when EV is charged.

$$G_i \leq L * C_i \qquad \forall i \in N \tag{6}$$

4.4 Battery Constraints

To ensure that the EV arrives at the specified destination without any problems related to running out of energy, and to maintain the quality of battery work for longevity, the battery energy level must be maintained within the recommended minimum (B_{min}) and maximum (B_{max}) values. Let B_i be the battery level of the EV at node i. Indeed, constraint (7) ensures that the battery level is within the range at any node and time from source S to destination D. Whereas, constraint (8) makes sure that the battery level is above the minimum level plus enough energy (i.e., $E_{toStation}$) to reach the nearest charging station. Thus, in our study, we raised the advantage of maintaining enough energy not only to safely finish the current trip but also to launch the next journey. Finally, constraint (9) calculates the battery level of the EV at each node it traverses from the source to the destination. In this constraint, when the EV arrives at node i after traversing link (h, i), the battery level of the EV at node i is calculated by subtracting the energy needed to traverse link (h, i) (that is the time requires to traverse the link multiplying the depletion energy $E_{Depletion}$ per unit time) from the EV battery level at node h. Obviously, if node h is a charging station and the EV has been charged for G_h time, then the charging energy $E_{Charging}$ per unit time is added to the battery level. Also, it is to be noted that the constraint is always satisfied when the EV does not traverse edge (h, i) as explained for constraint (4).

$$B_{min} \leq B_i \geq B_{max} \qquad \forall i \in N \tag{7}$$

$$B_D \geq B_{Min} + E_{toStation} \tag{8}$$

$$B_i \leq B_h - E_{Depletion} * T^{Edge}_{(h,i)} + G_h * E_{Charging} + L(1 - R_{(h,i)}) \quad \forall (h,i), \forall i \in N : i \neq S \tag{9}$$

5 PROPOSED HEURISTIC APPROACHES

From the fact that the optimization model presented in Section 4 is very complex and cannot solve large environments (graphs). Hence, in this section, we propose three heuristic methods.

5.1 TERC (Time Efficient Routing & Charging)

The TERC method is given in Algorithm 1, where it takes as an input; graph $G(N, E)$, the initial battery level B_s, the source node S, and the destination node D, and returns the fastest path from the source to the destination by considering all the constraints necessary for the longevity of the battery's life. The algorithm works in a recursive function. First, it finds the fastest path (P_T) from S to D using the $Dijkstra$ algorithm in line 3, and finds respectively the traveling time (T_P) and battery consumption (B_{used}) using $TimeOnPath$ and $BatteryUsedOnPath$ functions in lines 4 and 5. Then, the algorithm checks whether the battery percentage after traversing the fastest path P_T and arriving at the destination node is above the required minimum battery level (i.e., B_{Min}) plus the energy needed to reach the nearest charging station (i.e., $E_{toStation}$); mathematically, we can write it as $B_d - B_{used} >= B_{min} + E_{toStation}$. It is noted that the battery level B_d in the if condition in line 6 is written as B_i because of the recursive function $FindPath$. If the condition is met, then the algorithm terminates and returns the fastest path P_T, where the travel time T_D is the result of the TERC method. Else, the algorithm, in lines 11-16, finds the closest charging station to the source node, and obtains the fastest route and battery consumption, to be added later to the path from the current charging station to the destination node, in a recursive fashion. Finally, after calculating the total traveling time in line 18 (that is

Algorithm 1: TERC

Data: Graph $G(N, E)$, B_s, S, D.
Result: P_T: The fastest path by considering all constraints from S to D.

1 $T_D = 0$;
2 **Function** `FindPath`(S, D, B_S)**:**
3    $P_T \longleftarrow Dijkstra(S, D)$;
4    $T_P \longleftarrow TimeOnPath(P_T)$;
5    $B_{used} \longleftarrow BatteryUsedOnPath(P_T)$;
6   **if** $B_i - B_{used} >= B_{min} + E_{toStation}$ **then**
7      $T_D = T_D + T_P$;
8   **return** P_T
9   **else**
10      $T_P = L$;
11   **for** *each* $i \in \mathcal{I}_{Station}$ **do**
12        $P_{temp} \longleftarrow Dijkstra(S, i)$;
13        $T_{temp} \longleftarrow TimeOnPath(P_{temp})$;
14   **if** $T_P > T_{temp}$ **then**
15          $T_P = T_{temp}$;
16          $S_{new} \longleftarrow i$;
17      $B_{new} = B_{max}$;
18      $T_D = T_D + T_P + G_i$;
19   **return** $P_{temp} + FindPath(S_{new}, D, B_{new})$

time to traverse the path from the source to the current charging station and the time to charge the battery), the algorithm returns the concatenation path resulting from the current path plus the path that will be obtained from the recursive function $FindPath$, which will find the fastest route from the current charging station to the destination node (line 19).

5.2 TERC2 (the updated TERC algorithm)

The details of the TERC2 method are given in Algorithm 2. It follows the same steps as in Algorithm 1 for the TERC method, except for the step to find the charging station in case the battery level constraint is not satisfied. As shown in lines 12-16 of Algorithm 2, the closest charging station to the source node which is also closest to the destination node is chosen. This step optimizes the process of finding the nearest charging station to the source and destination nodes. Line 16 calculates T_{temp}, the time required to reach a charging station from the source node (i.e., T_{temp1}), plus the time to get to the destination node (i.e., T_{temp2}). Lines 16 to 18 find the charging station through which the distance from the source to the destination can be traversed in the least possible time. The advantage of the TERC2 method is that it can find a charging station on a route from the source to the destination without deviating much from the shortest path. In other words, it prevents the EV from charging at a charging station which is located far from the shortest path between the source and destination although it is the closest to the source node.

5.3 KFP (K-Fastest Path)

The KFP approach presented in this subsection is the updated method of the K-shortest path that finds the fastest path between

Algorithm 2: TERC2

Data: Graph $G(N, E)$, B_s, S, D.
Result: P_T: The fastest path by considering all constraints from S to D.

1 $T_D = 0$;
2 **Function** `FindPath`(S, D, B_S)**:**
3    $P_T \longleftarrow Dijkstra(S, D)$;
4    $T_P \longleftarrow TimeOnPath(P_T)$;
5    $B_{used} \longleftarrow BatteryUsedOnPath(P_T)$;
6   **if** $B_i - B_{used} >= B_{min} + E_{toStation}$ **then**
7      $T_D \leftarrow T_D + T_P$;
8   **return** P_T
9   **else**
10      $T_P = L$;
11   **for** *each* $i \in \mathcal{I}_{Station}$ **do**
12        $P_{temp1} \leftarrow Dijkstra(S, i)$;
13        $P_{temp2} \leftarrow Dijkstra(i, D)$;
14        $T_{temp1} \longleftarrow TimeOnPath(P_{temp1})$;
15        $T_{temp2} \longleftarrow TimeOnPath(P_{temp2})$;
16        $T_{temp} = T_{temp1} + T_{temp2}$;
17   **if** $T_P > T_{temp}$ **then**
18          $T_P = T_{temp}$;
19          $S_{new} \leftarrow i$;
20      $B_{new} = B_{max}$;
21      $T_D = T_D + T_{temp1} + G_i$;
22   **return** $P_{temp1} + FindPath(S_{new}, D, B_{new})$

the source and destination nodes considering the EV battery level for the longevity of battery life. It begins by finding the K number of fastest paths and selects the best one that satisfies all battery constraints. The details of this method are given in Algorithm 3. In line 2 of the algorithm, the $KNearestPaths$ function is similar to finding the K shortest paths, it determines the K fastest paths between S and D, and stores all candidate paths in table P_{Kpaths}. The for loop in line 3 checks all the paths one by one from the fastest to the next fastest until it finds the one that satisfies all battery conditions. Line 4 measures the energy required to traverse the current path P. If the current energy level is sufficient to reach the destination and the remaining battery level is still above $B_{min} + E_{toStation}$, then the algorithm returns the current path and time and terminates. Otherwise, in line 8, the algorithm calculates the amount of energy shortage B_m required to reach the destination. The for loop in line 9 traverses all edges on the current path P, where at each node the remaining battery level is monitored (lines 10 and 11). Whenever, after traversing any node, the required energy which is $B_{min} + E_{toStation}$ is unsatisfied, then the current path is dropped, and the next shortest path from table P_{Kpaths} is retrieved (lines 12 and 13). When the next node after traversing an edge is a charging station, although the battery can be charged to the maximum allowable limit which is B_{Max}, it is not required to do that if there is enough battery to reach the destination and satisfied the battery level constraint. Since the KFP method unlike the TERC and TERC2 methods does not find the next node step by step (node-by-node) and rather finds the entire path in advance,

Algorithm 3: KFP

Data: Graph $G(N, E), S, D, B_s$
Result: P = The fastest path from S to D

1 **Function** FindPath(S, D, B_S)**:**
2 $P_{Kpaths} \longleftarrow KNearestPaths(S, D)$;
3 **for** *each* $P \in \mathcal{P}_{Kpaths}$ **do**
4 $B_{used} \leftarrow BatteryUsedOnPath(P)$;
5 **if** $B_i - B_{used} > B_{min} + E_{toStation}$ **then**
6 Return(P)
7 **else**
8 $B_m = B_{used} - B_s$;
9 **for** *each* $i \in \mathcal{P}$ **do**
10 $B_{toi+1} \leftarrow BatteryUsed(i, i+1)$;
11 $B_{i+1} = B_i - B_{toi+1}$;
12 **if** $B_{i+1} < B_{min} + E_{toStation}$ **then**
13 Jump to the next P;
14 **if** $(i+1)$ *in* $I_{station}$ **then**
15 $B_n = B_{max} - B_{i+1}$;
16 **if** $B_m > B_n$ **then**
17 $B_{i+1} = B_{max}$;
18 $B_m = B_m - B_n$;
19 **else**
20 $B_{i+1} = B_{i+1} + (B_m - B_n)$;
21 Return(P)

therefore it is not required to charge the EV to the maximum limit. Consequently, the required charging level at each charging station on path P can be measured to save time and not waste time on extra charging; lines 16-18 check if it requires more than B_{Max} to reach the destination, then charge the EV to its maximum battery level. Otherwise, line 20 calculates the amount of energy required to charge to reach the destination and satisfy the battery constraint. Eventually, the algorithm terminates in line 21 when the battery energy level is sufficient to reach the destination.

It is worth noting that the KFP method finds simple paths without revisiting any node twice. Therefore the direct implementation of this algorithm suffers a deficit because it conflicts with some scenarios where the EV has to visit an intersection more than once during its journey to the destination node. Figure 2 explains such a scenario. Assume that EV is at node A, node D is the destination node, and node C is a charging station. Let us also assume that the EV is required to charge at node C. Then, the KFP method can not find a path such that the EV goes from node A to the charging station (node C), and then from node C to the destination node (node D). That is because the EV has to traverse node B twice (once to reach node C from node A, and once when going to node D from node C). Therefore, the KFP can not solve such a scenario.

6 PERFORMANCE EVALUATION

To evaluate the performance of our heuristic approaches (TERC, TERC2, and KFP), in this section, we compare them first with the optimal solutions obtained from the optimization model (OPL) on small networks, and then we evaluate their performance on big

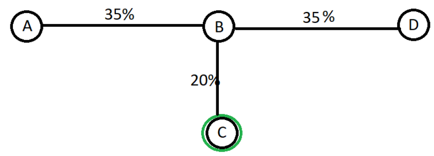


Figure 2: Example of the KFP issue.

instances with respect to the total traveling time, distance, charging time, and computation time. The inputs for our algorithm are derived from the specifications of the Tesla Model 3. For our evaluation, we consider a travel range of 545km per full battery, which reflects the distance that the Tesla Model 3 can cover under combined weather usage. Furthermore, the average traveling speed is considered 60km per hour. All available charging stations are considered AC charging stations. This choice is based on the prevailing trend of AC charging being the most commonly used method for EV charging at the present time. While other charging technologies exist, such as DC fast charging, focusing on AC charging allows us to assess the algorithm's performance under typical charging infrastructure conditions. To calculate the charging process time, we made an assumption regarding the time required to charge the battery from 0 to 100 percent. In our evaluation, as the Tesla Model 3 can be fully charged in 8 hours and 15 minutes, To calculate the time required for charging each percent of the battery, we divide the total charging time by 100. While this assumption does not account for variations in battery charging speed due to factors such as different charging speeds based on battery level and temperature, it provides a reasonable approximation for evaluating the algorithm's performance within the scope of this study. To validate our proposed approaches based on battery life, empirical evaluations have been performed with the maximum and minimum battery level of $B_{max} = 80\%$ and $B_{min} = 20\%$ respectively. The initial charging level of the EV at the starting point is set to 80%. The results are averaged over ten runs, where in each run, the source and destination have been chosen randomly but equally for different methods for fair comparison.

6.1 Graph Generation

To construct graphs for performance evaluation of our proposed methods (i.e., creating nodes and weighted edges), we used Erdős Rényi model [12] to generate random graphs with different sizes (number of nodes and edges). The number of edges is obtained by taking an appropriate probability of the existence of an edge between a pair of nodes. It is very crucial to select a good probability for edge creation so that the graph is not disconnected. Furthermore, the amount of battery energy consumption in percentage and time to traverse an edge is considered as the fixed weight of any edge. For simplicity, we consider the edge distance and vehicle speed to be fixed so that we can calculate the traveling time on each edge easily. After generating the graph and making sure that it is connected, we add a number of charging stations to the graph. First, we choose a random node in the graph as the first charging station and add it to the list $I_{Station}$. Then, for the next charging station, we select the farthest node in the radius of the first chosen one that can be reached with the maximum battery charged B_{Max}. Recursively, we repeat the same strategy from all charging stations

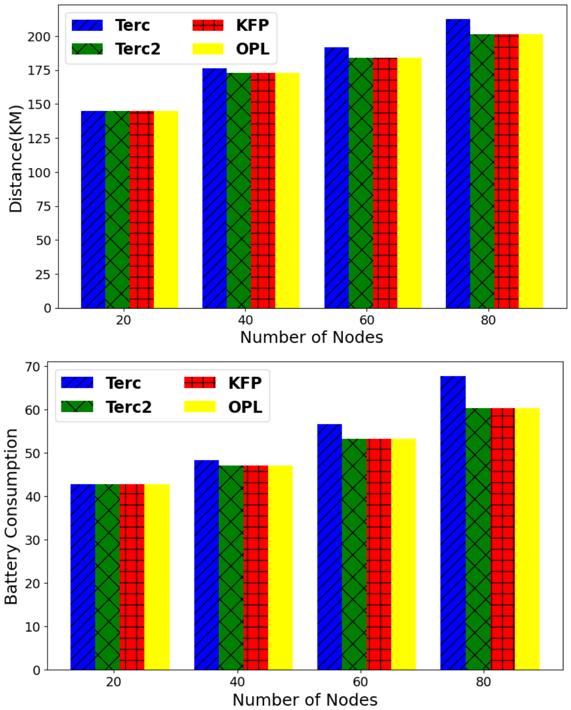


Figure 3: Comparing different methods (OPL, TERC, TERC2, and KFP) by varying the size of the network considering different performance metrics: (a) traveling Distance, and (b) battery consumption.

in the $I_{Station}$ list until the entire graph has been covered and there is no part that cannot be reached with B_{Max}. Finally, we randomly choose a source node and a destination node on the graph with a specific range distance for the EV. In addition, to avoid having a scenario where the KFP method cannot find a route from the source to the destination nodes as explained earlier, we make sure that each charging station has no dead-end path to avoid returning to the same visited node twice.

6.2 Evaluation over small Graphs

In this subsection, we evaluate the performance of our heuristic methods by comparing them to the optimization model (OPL). Figure 3 illustrates the gap between the three heuristic methods (TERC, TERC2, and KFP) to the OPL in terms of total travel distance and battery consumption. As shown in both sub-figures for travel distance and battery consumption, the performance of different methods is almost identical with the same variation as the size of the graph increases from 20 to 80 nodes. This is because as the EV takes longer routes, obviously more battery energy is consumed. It is also observable that both TERC2 and KFP reach the optimal solutions in tested small graphs. However, the TERC method performs a near-optimal solution with a maximum gap of 7.4%, and its results deviate slightly from OPL as the number of nodes increases.

These results are in the interest of proving the validity of our proposed methods, where in the case of large graphs, we expect that the heuristic methods will perform very close to optimal solutions. However, because of the complexity of the optimization method, we cannot obtain results for large instances as shown in Figure 4. The

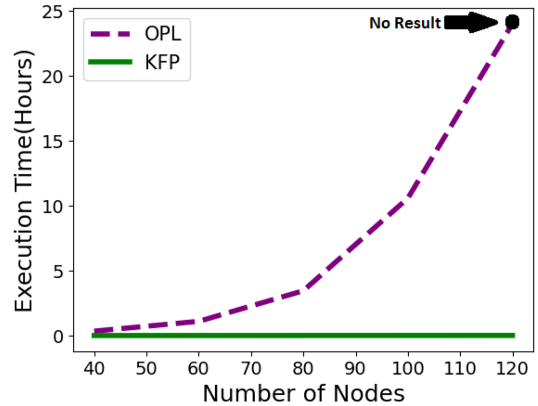


Figure 4: Comparing the execution time of the optimization model (OPL) versus the KFP method.

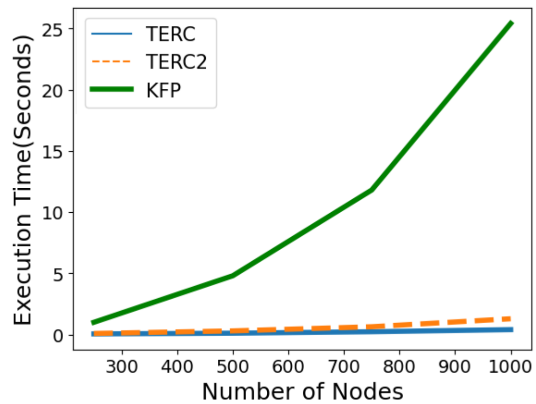
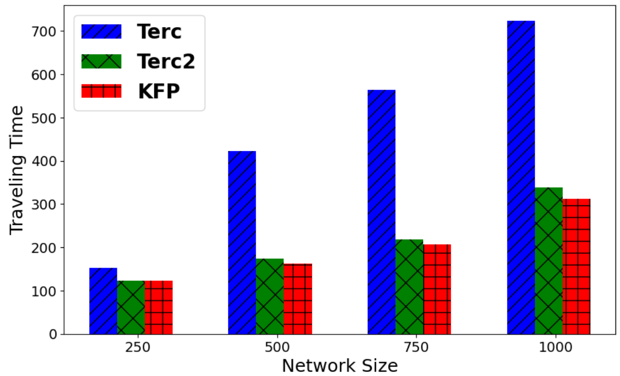


Figure 5: Comparing the execution time of the heuristic methods (TERC, TERC2 and KFP).

figure shows that the computation time of the optimization model (OPL) increases exponentially, starting with a few seconds for 40 to 60 nodes, increasing to around 10 minutes for 120 nodes, and eventually no results (failed to obtain results) for a slightly large graph of 150 nodes due to complexity of the optimization model and lack of memory. In contrast, one of the heuristic methods, like the KFP method, obtains results in less than a minute for the same size graph. Figure 5 illustrates the computation time for all proposed heuristic methods. In the figure, as the number of nodes increases, the computation time of the TERC and TERC2 methods increases slightly (from a very few milliseconds for a graph with 300 nodes to a maximum of 14 seconds for a graph of size 1000 nodes). Whereas, the KFP method increases exponentially for large graphs (7.5 seconds to 76.5 seconds respectively for graphs of size 300 to 1000 nodes). In the next subsection, we will compare the performance of the heuristic methods in terms of total EV travel time and battery consumption.

6.3 Evaluation Over Large Graphs

For performance evaluation of heuristic methods on large graphs, the size is varied from 250 nodes to 1000 nodes. Figure 6 illustrates the total traveling time and battery consumption of the three methods: TERC, TERC2, and KFP. It is to be noted that the total traveling



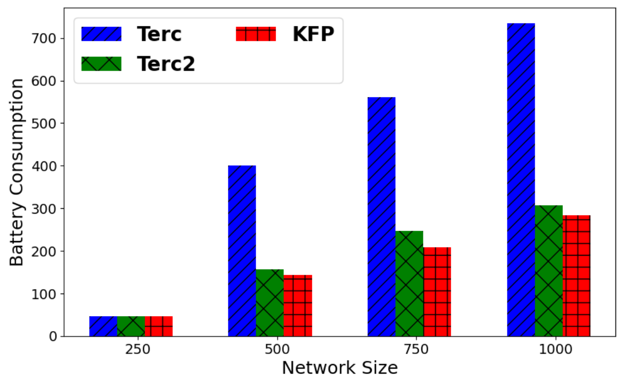


Figure 6: Comparing different heuristic methods (TERC, TERC2, and KFP) by varying the size of the network considering different performance metrics: (a) Traveling Time, and (b) battery consumption.

time covers both traversing the path and charging the EV. From both sub-figures, we can observe that as expected the total travailing time and battery consumption of the EV increase gradually with the size of the graph. The KFP outperforms the TERC2 very slightly (a maximum of 4.5% and 8% gap respectively for traveling time and battery consumption when considering 1000 nodes and 750 nodes, respectively). However, the performance of the TERC is the worst among other methods in which its performance worsens as the size of the graph is enlarged. Luckily, the battery consumption of the TERC is leveled with other methods when evaluated with a graph of size 250 nodes. Similarly, the TERC2 method in terms of total traveling time leveled with the KFP method in a graph of 250 nodes. The simulation results in this section show that although the performance of the TERC2 method is slightly behind the KFP method in some cases, it is more reliable, efficient, and effective in solving the routing and charging problem for all graphs. That is because the KFP method fails to solve the routing and charging problem for some graphs as explained in Section 5.3, has a higher computation time, and it cannot be used for a dynamic environment since it requires finding the entire path in advance.

7 CONCLUSION AND FUTURE WORKS

This paper investigated the problem of routing and charging an electric vehicle to travel from a source to a destination in the shortest time, considering the maximum and minimum battery level for the longevity of its life and the energy required to reach the nearest charging station for the next trip. The problem has been formulated mathematically to obtain optimal solutions. However, due to the complexity of the optimization model, three heuristic approaches have been proposed: TERC, TERC2, and KFP. To evaluate the performance of the proposed methods, several performance metrics such as travel time, travel distance, battery consumption, and execution time, have been considered by varying the size of the network. The numerical results showed that both TERC2 and KFP methods perform very close to the optimal solution. Furthermore, the KFP method in a few cases outperforms the TERC2 method in terms of traveling time and battery consumption. However, the former requires much more computation time and fails to find a path in some rare cases. Therefore, the latter is more reliable, effective, and efficient for solving the problem. For future work, we are interested in using AI-based methods like reinforcement learning and/or deep reinforcement learning to integrate dynamic situations like traffic and charging queues into the EV charging and route planning process. We are also interested in conducting extensive simulations and real-world experiments to provide valuable insights into the strengths and limitations of our proposed approaches.

REFERENCES

[1] Xiaowen Bi, Andrew John Chipperfield, and Wallace K. S. Tang. 2021. Coordinating Electric Vehicle Flow Distribution and Charger Allocation by Joint Optimization. *IEEE Transactions on Industrial Informatics* 17 (2021), 8112–8121.
[2] Mohamed Nadir Boukoberine, Teresa Donateo, and Mohamed Benbouzid. 2022. Optimized Energy Management Strategy for Hybrid Fuel Cell Powered Drones in Persistent Missions using Real Flight Test Data. *IEEE Transactions on Energy Conversion* 37, 3 (2022), 2080–2091.
[3] M. Bruglieri, M. Paolucci, and O. Pisacane. 2023. A matheuristic for the electric vehicle routing problem with time windows and a realistic energy consumption model. *Computers & Operations Research* 157 (2023).
[4] Cristian Cataldo-Díaz, Rodrigo Linfati, and John Willmer Escobar. 2022. Mathematical Model for the Electric Vehicle Routing Problem Considering the State of Charge of the Batteries. *Sustainability* 14 (2022).
[5] Cedric De Cauwer, Wouter Verbeke, Thierry Coosemans, Saphir Faid, and Joeri Van Mierlo. 2017. A Data-Driven Method for Energy Consumption Prediction and Energy-Efficient Routing of Electric Vehicles in Real-World Conditions. *Energies* 10 (2017).
[6] Cedric De Cauwer, Wouter Verbeke, Joeri Van Mierlo, and Thierry Coosemans. 2020. A Model for Range Estimation and Energy-Efficient Routing of Electric Vehicles in Real-World Conditions. *IEEE Transactions on Intelligent Transportation Systems* 21 (2020), 2787–2800.
[7] Gerhard Hiermann, Jakob Puchinger, Stefan Ropke, and Richard F. Hartl. 2016. The Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations. *European Journal of Operational Research* 252 (2016), 995–1018.
[8] Michail Masikos, Konstantinos Demestichas, Evgenia Adamopoulou, and Michael Theologou. 2014. Machine-learning methodology for energy efficient routing. *IET Intelligent Transport Systems* 8 (2014), 255–265.
[9] Amir Mirheli and Leila Hajibabai. 2023. Charging Network Design and Service Pricing for Electric Vehicles With User-Equilibrium Decisions. *IEEE Transactions on Intelligent Transportation Systems* 24 (2023), 2888–2902.
[10] Jennifer Morris, David Hone, Martin Haigh, Andrei Sokolov, and Sergey Paltsev. 2022. Future energy: in search of a scenario reflecting current and future pressures and trends. *Environmental Economics and Policy Studies* 25 (2022). https://doi.org/10.1007/s10018-021-00339-1
[11] Michael Schneider, Andreas Stenger, and Dominik Goeke. 2014. The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations. *Transportation Science* 48 (2014), 500–520.
[12] Comandur Seshadhri, Tamara G Kolda, and Ali Pinar. 2012. Community structure and scale-free collections of Erdős-Rényi graphs. *Physical Review E* 85, 5 (2012), 056109.
[13] Shuai Zhang, Yuvraj Gajpal, S.S. Appadoo, and M.M.S. Abdulkader. 2018. Electric vehicle routing problem with recharging stations for minimizing energy consumption. *International Journal of Production Economics* 203 (2018), 404–413.