JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Real-Time Algorithmic Approach to Reduce EV Charging Delays

Yash Sharma[1] and Aashish Khosla[2]

[1]Mathematics and Computing Department, Indian Institute of Technology, Bhilai, India
[2]St. Gregorios High School, Mumbai, Maharashtra, India.

Abstract

The growing adoption of electric vehicles (EVs) presents unique challenges, par- ticularly in terms of efficient charging during long trips. This paper introduces an algorithm designed to optimize the selection of charging stations along a pre-planned route. The algorithm models the road network as a graph, where certain nodes rep- resent charging stations, and the edges represent roads with weights based on real traffic times. By analyzing the current location of the vehicle and the planned route, the algorithm identifies charging stations along the way and calculates real-time in- formation such as the number of cars waiting at each station. It then ranks and displays the top three charging stations with the least total time for both travel and charging. This approach demonstrates the potential for minimizing overall journey and charging delays, enhancing the user experience for EV drivers.

Keywords - Graph-based algorithms, Charging station optimization, Electric vehicles, Route planning, Real-time traffic

1 Introduction

1.1 Background and Related Work

The growing adoption of electric vehicles (EVs) presents unique challenges, particularly in terms of efficient charging during long trips. Ensuring the vehicle does not run out of battery while identifying the most time-efficient charging station nearby is a critical aspect of EV route planning. Sebai et al. [10] addressed EV route planning by deriving optimal routes based on energy consumption, utilizing historical trajectory data for analy- sis. Schoenberg and Dressler [9] proposed an adaptive charging and routing strategy that incorporates driving time, waiting time, and charging time to reduce overall travel delays.

Their approach contrasts with traditional routing systems, which typically prioritize the shortest or fastest routes; instead, their methodology focuses specifically on energy-optimal routing while also minimizing battery stress [7]. Additionally, Mariyasagayam et al. [6] in- troduced a forecasting method for predicting the availability of charging spots at stations. By leveraging the location and reachability information of other EVs along the route, their method estimates the occupancy ratio of charging spots at the respective stations.

The rapid proliferation of electric vehicles (EVs) worldwide has significantly transformed the transportation landscape, emphasizing the urgent need for optimized charging infras- tructure. This need arises from the challenges of balancing charging demand, minimizing driver wait times, and managing grid stability during peak loads. Addressing these issues has been the focus of extensive research in the domains of charging station placement, routing strategies, and grid-aware charging optimization.

To achieve an efficient charging infrastructure, Sayarshad [8] proposed an integrated model for optimizing EV routing and charging station coordination, considering power- aware operations to ensure grid stability. Similarly, Cheng et al. [3] focused on the place- ment of charging stations based on voltage stability in distribution systems, highlighting the critical relationship between station placement and grid reliability.

Routing optimization has also been explored extensively. Zhu et al. [11] developed a real-time traffic-informed path-planning algorithm for EVs, demonstrating the benefits of integrating traffic data to minimize overall travel time and charging delays. Li et al. [4] extended this approach by incorporating traffic impedance guidance to further refine EV routing strategies, underscoring the importance of dynamic traffic conditions in route plan- ning.

The density and localization of charging piles are critical components of infrastructure optimization. Li et al. [5] presented a strategic framework to determine the optimal density and location of charging stations, balancing user convenience with operational costs. Their work emphasizes how localized station planning can mitigate congestion and improve ac- cess for EV users.

Another significant challenge lies in managing charging demand during peak periods. Aoun et al. [1] proposed a dynamic optimization algorithm to mitigate grid power peaks, offering a robust solution to balance user satisfaction with grid performance. This is com- plemented by Chen et al. [2], who introduced a high-efficiency routing and charging strategy for EV fleets operating in dynamic transit systems, demonstrating the advantages of fleet- wide coordination in reducing operational delays.

Building on these insights, this paper introduces an adaptive, graph-based algorithm designed to dynamically identify and rank EV charging stations. By integrating real-time traffic data, station availability, and wait times, the proposed model ensures that EV drivers

receive optimal routing recommendations. This approach advances the field by addressing the dual challenges of minimizing driver wait times and enhancing the overall efficiency of the EV charging network.

1.2 Problem Statement

As more people switch to electric vehicles (EVs), it’s crucial to make sure they can charge their cars without wasting time. However, the way we guide EVs to nearby charging stations is inefficient. This study addresses what helps EV drivers identify the most time- effective charging stations along their planned routes by proposing and evaluating inno- vative algorithms and models that can dynamically route EVs to the most appropriate charging stations based on real-time data.

1.3 Research Gap and Motivation

Although various methods have been proposed for optimal EV routing and charging, most fail to dynamically incorporate real-time traffic conditions and waiting times at charging stations. Additionally, existing methods often lack effective algorithms for ranking stations based on the total journey and charging time.

1.4 Contributions

This paper introduces an adaptive algorithm that:

- Dynamically identifies and ranks charging stations based on total travel and charging time.
- Integrates real-time traffic data and waiting times at charging stations.
- Provides drivers with the top three charging stations, enabling informed decision- making.

Our approach demonstrates a significant reduction in both travel and charging delays, improving the overall EV driving experience.

2 Model

The model proposed in this research paper is designed to optimize the allocation of charging stations for electric vehicles (EVs) to minimize total charging time during long-distance trips. It employs a graph-based approach, where:

2.1 Graph Representation

The road network is modeled as a connected graph $G = (V, E)$, where V is the set of nodes. Where some nodes represent charging stations, and E is the set of edges representing roads. The weight of each edge corresponds to the estimated travel time, including real-time traffic conditions obtained via API integration. Figure 1 represents a graph with weighted edges that depict real traffic times

- **Edges** denote roads, weighted by real traffic time.

The model is dynamic, analyzing real-time data such as the current vehicle position, the planned route, and the number of cars waiting at each charging station.

2.2 Steps involved in the Model

1. **Graph Representation:** The entire road network is represented as a connected graph $G = (V, E)$, where V is the set of nodes, and E represents the roads between them with associated travel times as

weights.

2. **Charging Station Identification (CS List Algorithm):** Based on the current vehicle location, Dijkstra's algorithm is used to determine all reachable charging stations within a certain radius. The model adds these stations to a dynamic list based on their distance from the vehicle's current position.

3. **Computation of Total Time:** For each charging station in the list, the model cal- culates the total time(calculated using Dijkstra's algorithm), combining travel time, queue length, and charging time, to estimate the overall time required to reach and charge at that station.

4. **Output:** The top three charging stations with the minimum total time are displayed, helping EV drivers make informed decisions about where to charge their vehicles while constantly moving and heading towards the destination.

This model provides a comprehensive and systematic way of handling charging station selection, allowing EV drivers to save both travel and charging time while contributing to an overall efficient charging network, thereby enhancing the EV driving experience. It offers a practical, real-time solution for EV route planning, effectively dispersing vehicles across charging stations to avoid congestion and delays.

3 Method

3.1 Algorithm: Displaying Optimized Charging Stations

Initially, the status variable of all cars requesting charging is set to *uncharged*. After charg- ing is completed, the status is updated to *charged*. Based on the Battery Duration Before

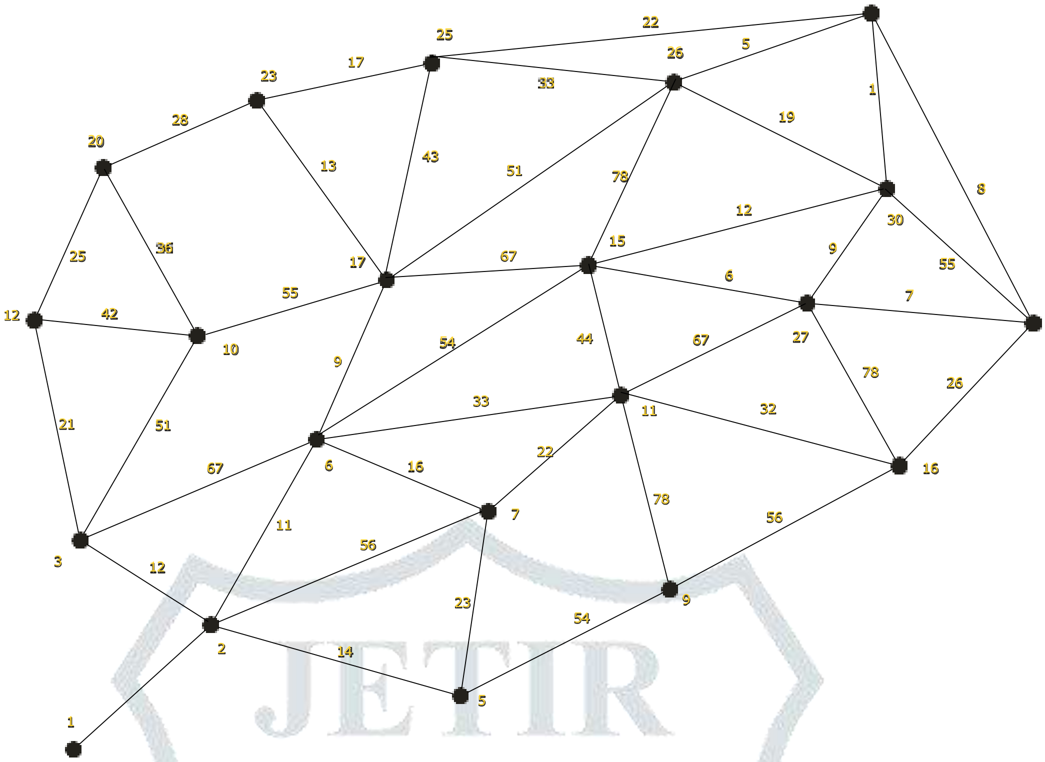

Figure 1: Graph with weighted edges representing real traffic times

Dead (BDBD), we have estimated how far a car can travel before requiring a charge. The BDBD indicates the maximum time the car can run without recharging. Using this BDBD as an input parameter for the *CS_List* function, we retrieve a list of charging stations that are reachable from the car's current location, given its BDBD status. We store this list under the variable *check*. The algorithm is then executed only on the stations in this list for a car, identified by its CN (Unique Identification Number). This selective execution improves the algorithm's efficiency in terms of time complexity, as it avoids running calcu- lations on charging stations that the car cannot reach. Hence, we focus the algorithm only on the *check* list, based on the car's current location and BDBD status.

Our algorithm then displays the top three charging stations with the minimal charging time for the vehicle, considering factors such as the car's current location, the planned route (from the current location to the destination), real-time traffic along the route, the number of cars waiting at each charging station, and the number of cars heading towards charging station. All these factors are computed within the *computation* algorithm. This algorithm returns a list of charging stations along with the estimated time to reach the final destination (factoring in charging time). This list is displayed to the driver, enabling

Symbol	Description
check, Enable	List of charging stations that are reachable and in line with the route of the current location of the vehicle
CL	Current location of vehicle
CS	Charging station
optimal list	A list of charging stations with the total time required to reach the destination, including charging time
z	Input variable to accept or decline to charge for given admissible charging station.
CN	Unique car identification number
T	Termination point
t_1	Estimated time taken from starting point to charging station
t_2	Estimated time taken from charging station to termination point
k	Cars waiting at E_i
p	Cars ahead me in this round in admissible list
c	Time required to charge a particular car
AL	Admissible List
WL	Waiting List
d_i	Admissible range distance for i^{th} charging station
$d(A, B)$	minimum distance between point A and B

Table 1: Symbols and descriptions used in algorithms below

them to make an informed decision about where to charge. The list includes the name of each charging station and the estimated time to reach the destination, including the charging time on the route.

After each execution, our algorithm refreshes at a set rate, continuously recalculating and displaying the optimal charging station based on the car's current location, planned route, real-time traffic, and the number of cars waiting or heading towards charging station.

Each charging station is surrounded by an imaginary boundary, which may not be per- fectly circular. We can visualize this boundary as the area within which the time required to reach the charging station is equal from every point of boundary. Since different routes take varying amounts of time due to traffic conditions, the boundary shape adapts accord- ingly. Routes with less traffic might have a broader boundary compared to routes with heavy congestion, as illustrated in the accompanying diagram(Figure 2). This boundary is determined solely by the time (affected by traffic) required to reach the charging station. Once a car enters this boundary, it has the option to accept or decline charging at that particular station. The car's input is numeric: 0 indicates a decision to decline charging atthe station, while any value other than 0 indicates acceptance. Once accepted, the car is added to the "Admissible List," which contains all cars that have agreed to charge at that particular station. Note that each charging station maintains its own Admissible List.

3.2 Admissible Range Concept

The concept of an admissible range is introduced to address potential issues arising when vehicles can select charging stations without proximity restrictions. In other words, why is it necessary to have an admissible range? To understand this, consider the limitations that occur in the algorithm without an admissibility constraint.

Suppose a vehicle on a long journey evaluates charging stations and finds that nearby stations have significant waiting times, while a more distant station—let's call it E_p—has a shorter waiting time. According to the algorithm, E_p would be recommended as it cur- rently has the least overall wait time. However, this recommendation is problematic: by the time the vehicle arrives at E_p, other cars, which were closer and activated later, might converge there, leading to unexpectedly long wait times. Consequently, the initial decision to charge at E_p becomes suboptimal as the increased demand results in prolonged waiting times upon arrival.

To prevent this scenario, the concept of admissibility is applied. We define an admissi- ble range as a boundary around each charging station within which vehicles are permitted to confirm their decision to charge. This boundary is based on a parameter we term the *Average Time to Reach After Admitting (ATRAA)*—the average travel time for vehicles to reach the station once they have chosen to charge there. An imaginary boundary is created around each station, ensuring that all cars within this range from boundary will take approximately the same time to arrive at the station.

Only vehicles within this admissible range are allowed to *admit* or confirm their in- tent to charge at that station. This approach provides two main benefits: (1) it ensures that waiting times calculated for each station are more accurate, as only vehicles within a reasonable distance can commit to the station, and (2) it minimizes computational errors associated with estimating wait times, as only nearby vehicles impact the queue length in the subsequent decision phase. Notably, an EV driver's admission decision directly affects the wait times calculated for other vehicles; hence, the admissibility requirement helps stabilize queue lengths and reduce discrepancies in time estimations.

Additionally, admissibility mitigates the impact of long-distance reservations. For in- stance, if many distant vehicles preemptively select E_p due to temporarily low waiting times, the admissibility constraint prevents their selection from misleading nearby vehi- cles. Vehicles close to the station can make informed decisions based on more accurate,

real-time availability. In effect, the admissibility range improves the precision of total time computations and enhances the algorithm's robustness in managing station demand.

When a car (CN) accepts to charge, it is added to both the Admissible List and the Record List (along with the time of acceptance). Once the car reaches the charging station, it is removed from the Admissible List. The car remains on the Admissible List from the moment of acceptance until it arrives at the charging station. Upon arrival, two scenarios are possible: either the station is available, in this case, the car will be charged immedi- ately as per the algorithm, or the station is occupied, in this case the car is placed on the station's Waiting List. Each charging station also maintains its own Waiting List. The car stays in the Waiting List from the moment it arrives until it is ready to charge.

Prioritization within the Waiting List is based on the time of acceptance recorded in the system. Cars that accepted charging earlier will be given priority, following a First- In-First-Out (FIFO) approach. After the car (CN) has been allocated a charging spot, it is removed from the Waiting List and the record dictionary for that charging station. The Admissible List tracks cars that have accepted charging (CN), while the Waiting List tracks cars waiting to charge (CN), and the record dictionary holds information about the car (CN) along with its acceptance time. The car's presence in the record dictionary spans from the time of acceptance until it is ready to charge.

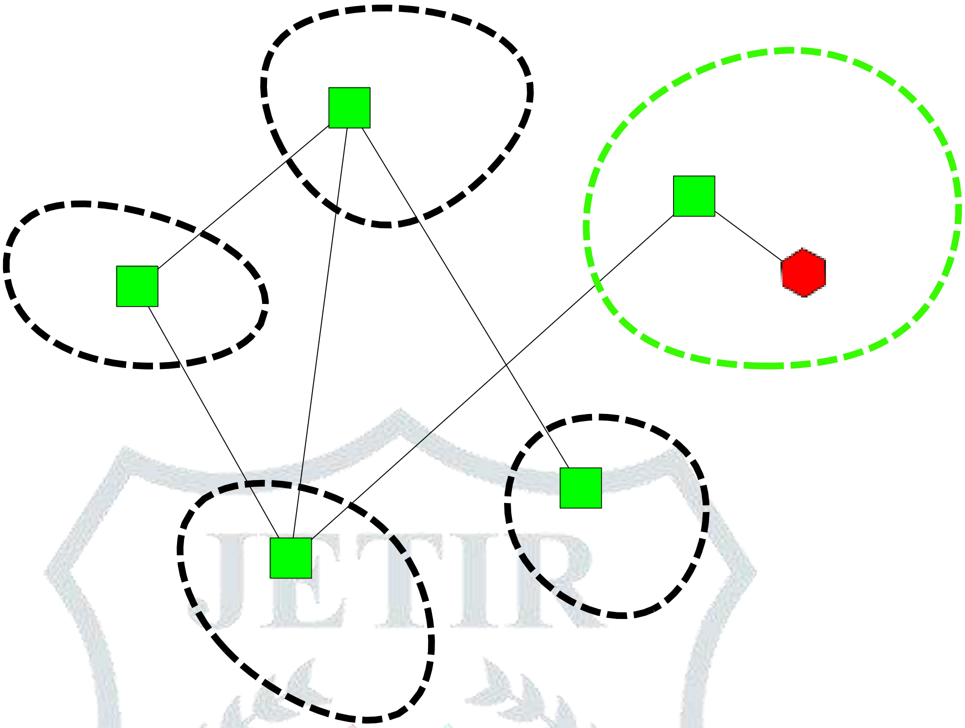

Figure 2: Admissible Range

Car entering the admissable range of a charging station

Admissable range of the given charging station

Reachable charging station EV vehicle

Algorithm 1: Optimized Charging Station Selection Algorithm

```
1: check ← CS_List()
2: while status! = charged do
3:   optimal list ← Computation(CN, CL, T, check)
4:   Display(optimal list)
5:   for each i in check do
6:   if min d(CL, CS_i) < d_i then
7:   z ← Input("You are in admissible range")
8:   if z = 0 then
9:   continue
10:  else
11:  AL_CS_i.append(CN)
12:  Record _CS_i.append(CN, t)
13:  while True do
14:  if min d(CL, CS_i) == 0 then
15:  AL_CS_i.remove(CN)
16:  if CS_i is empty then
17:  Charge CN
18:  status = charged
19:  break
20:  else
21:  WL_CS_i.add(CN)
22:  Extract CN_k from WL_CS_i with min t in Record{}
23:  if CN_k is CN then
24:  Charge CN
25:  status = charged
26:  WL_CS_i.remove(CN )
27:  Record_CS_i.remove(CN , t)
28:  break
29:  else
30:  Wait until current CN_k charged completely and charging node becomes free for next
31:  Goto STEP 21
32:  end if
33:  end if
34:  end if
35:  end while
36:  end if
37:  end if
38:  end while
39: end for
```

3.3 Algorithm: CS_List

Based on the car's Battery Duration Before Dead (BDBD) status and its current location, we add all reachable charging stations to the Enable list, considering the constraints im- posed by the BDBD. The primary objective of this step is to optimize the algorithm's time complexity by restricting computations to a relevant subset of charging stations. For a given car, identified by its CN (Unique Identification Number), the algorithm is executed only on the charging stations in the Enable list. Charging stations that are beyond the vehicle's BDBD range are excluded, as the vehicle would deplete its battery before reaching them. Consequently, these stations are not taken into consideration, and the final output is a list of reachable charging stations store in *Enable* list.

Algorithm 2: CS_List(CN, CL, BDBD)

1: Set BDBD as the boundary from CL
2: Enable[] ← [] {Initialize an empty list to store charging stations that are reachable from the current location of a car}
3: Add all Charging Stations (CS) in Enable[] that fall within the BDBD boundary
4: **return** Enable[]

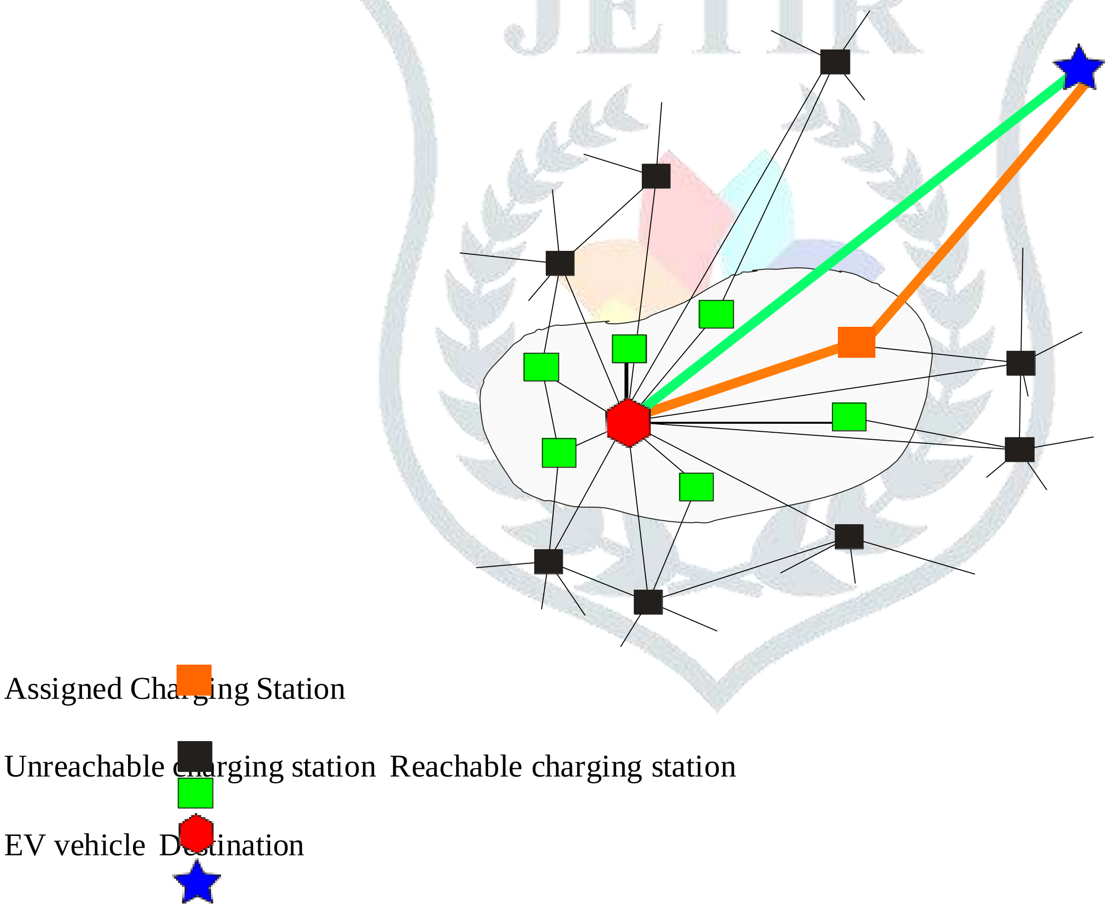

Figure 3: Charging Station Allocation

Flowchart

3.4 Algorithm: Computation

The most critical component of our approach is the **computation algorithm**, which takes the following input parameters:

- **CN**: The unique identification number of the car.
- **CL**: The car's current location.
- **T**: Destination or Termination point
- **check**: A set of charging stations selected for the algorithm based on reachability.

The goal of this algorithm is to estimate the total time required from the moment the charging request is made until the vehicle reaches its final destination. To calculate this, the total time is broken down into several components:

1. **Time from the car's current location (CL) to the charging station** E_i:
This is computed using Dijkstra's algorithm, where each edge weight corresponds to the real-time traffic conditions, integrated via an API. Dijkstra's algorithm provides the optimal path with the least time to reach the selected charging station.

2. **Waiting time at the charging station**:
The cars who have admitted earlier and waiting for their dien to get charged.

3. **Charging time for cars admitted earlier but not yet arrived**:
These are vehicles that accepted charging earlier and are en route but have not reached the charging station. Their total charging time is factored in.

4. **Time from the charging station to the destination**:
This is the time required to travel from the charging station E_i to the final destina- tion(Calculated based on Dijkstra's algorithm)

The total time is calculated as:

$$\text{Total Time} = \text{Time from CL to reach Charging Station} + \text{Time for Waiting cars to charged} + \text{Time for Admitted cars to charged} + \text{Time to reach Destination}$$

For all charging stations in the *check* list, the computation algorithm determines the total time and stores it, along with the corresponding station names, in a list *optimal list*. This list helps the driver make an informed decision by selecting the charging station with the minimal overall time.

Algorithm 3: Computation(CN, CL, T, check)

Input: CN, CL, T, $check$
optimal list ← [] {Initialize an empty list to store time and charging station pairs}
for each CS_i in check **do**
  $t_1 \leftarrow Dijkstra(G, CL, CS_i)$ {Time from current location CL to charging station E_i}
  $t_2 \leftarrow Dijkstra(G, CS_i, T)$ {Time from charging station E_i to destination T}
  $k, p \leftarrow \text{Server}(CS_i)$ {k is the number of cars waiting, p is position in the admissible list}
  $t \leftarrow t_1 + t_2 + k \cdot c + (p - 1) \cdot c$ {Total time with waiting time factored in}
  optimal list.append(t, CS_i) {Store the time and corresponding charging station}
end for
Sort *optimal list* in ascending order of t {Sort by total time}
return *optimal list* {Return the sorted list of charging stations and times}

4 Conclusion and Future Work

This paper presents an algorithm designed to assist electric vehicles (EVs) in efficiently reaching their destinations by identifying the optimal path and charging station with mini- mal total time. The algorithm considers real-time factors such as traffic conditions, waiting times at charging stations, and the reachability of stations based on the vehicle's current battery status.

In its current scope, the study assumes full charging at a single station for simplicity and practicality. However, future research can extend this work by incorporating partial charging strategies at multiple stations along the route. This extension could enable a more flexible and time-efficient approach, allowing vehicles to optimize their total charging time further while adapting to dynamic conditions. Such advancements could significantly enhance the practicality of the algorithm and improve the overall EV driving experience.

5 References References

[1] Alain Aoun, Mehdi Adda, Adrian Ilinca, Mazen Ghandour, and Hussein Ibrahim. Dynamic charging optimization algorithm for electric vehicles to mitigate grid power peaks. *World Electric*

Vehicle Journal, 15(7), 2024.

[2] Tao Chen, Bowen Zhang, Hajir Pourbabak, Abdollah Kavousi-Fard, and Wencong Su. Optimal routing and charging of an electric vehicle fleet for high-efficiency dynamic transit systems. *IEEE Transactions on Smart Grid*, 9(4):3563–3572, 2016.

[3] Shan CHENG, Linfeng XU, Weibin SUN, Gaocan SU, and Yuzhang FENG. Opti- mal planning of charging stations for electric vehicles based on voltage stability of distribution system. *Journal of Electric Power Science and Technology*, 35(4):3–12, 2020.

[4] Jingyu Li, Shiyuan Tian, Na Zhang, Guangchen Liu, Zhaoyuan Wu, and Wenyi Li. Optimization strategy for electric vehicle routing under traffic impedance guidance. *Applied Sciences*, 13(20):11474, 2023.

[5] Wenzao Li, Lingling Yang, Zhan Wen, Jiali Chen, and Xi Wu. On the optimiza- tion strategy of ev charging station localization and charging piles density. *Wireless Communications and Mobile Computing*, 2021(1):6675841, 2021.

[6] Marie Nestor Mariyasagayam and Yuichi Kobayashi. Electric vehicle route assistance using forecast on charging station. *ENERGY*, pages 134–142, 2013.

[7] Theresia Perger and Hans Auer. Energy efficient route planning for electric vehicles with special consideration of the topography and battery lifetime. *Energy Efficiency*, 13(8):1705–1726, 2020.

[8] Hamid R Sayarshad. Optimization of electric charging infrastructure: integrated model for routing and charging coordination with power-aware operations. *npj Sus- tainable Mobility and Transport*, 1(1):4, 2024.

[9] Sven Schoenberg and Falko Dressler. Reducing waiting times at charging stations with adaptive electric vehicle route planning. *IEEE Transactions on Intelligent Vehicles*, 8(1):95–107, 2022.

[10] Meriem Sebai, Lilia Rejeb, Mohamed Ali Denden, Yasmine Amor, Lasaad Baati, and Lamjed Ben Said. Optimal electric vehicles route planning with traffic flow predic- tion and real-time traffic incidents. *International Journal of Electrical and Computer Engineering Research*, 2(1):1–12, 2022.

[11] Wan Qing Zhu, Zhang Yu, Yuan Xingfu, and Li Zhixuan. Electric vehicle charging path planning based on real-time traffic information. In *2022 IEEE 6th Conference on Energy Internet and Energy System Integration (EI2)*, pages 768–774. IEEE, 2022.