Paper ID 2005200

Optimized Route Planning for Electric Vehicles with Charging Stops

Jean-Sébastien Gonsette[1*], Arnaud Jacques[2], Noémie Meunier[3]

1. AISIN AW – AWTC Europe SA, Belgium – jean-sebastien.gonsette@aisin-europe.com
2. Euranova SA, Belgium – arnaud.jacques@euranova.eu
3. AISIN AW – AWTC Europe SA, Belgium – noemie.meunier@aisin-europe.com

Abstract

The rapid adoption of electric vehicles (EVs) has introduced unique challenges in route planning, particularly for long-distance travel, where constraints such as limited battery capacity, non-linear charging times, and sparse charging infrastructure must be considered. This paper presents an EV routing solution designed to optimize total travel time by dynamically integrating charging stops into long-distance navigation. To address computational complexity, the solution employs techniques such as clustering of charging stations and incorporates real-time data for adaptive decision-making. The algorithm's effectiveness is further enhanced by a scoring system that evaluates charging stations based on availability, speed, and fallback options. Experimental results on a North American map demonstrate significant reductions in total and expected travel times, showcasing the solution's potential for scalable, real-time deployment in EV navigation systems.

KEYWORDS

Electric Vehicles (EVs), Route Planning, Charging Station Optimization, Travel Time Optimization.

Introduction

The rapid growth of electric vehicles (EVs) brings new challenges in route planning, especially for long-distance journeys. Unlike traditional vehicles that can rely on a dense network of fuel stations with short refueling times, EVs are constrained by limited battery capacity and a sparse, variable-speed charging infrastructure. Planning routes for EVs therefore demands careful consideration of battery levels, charging station locations, charging speeds, and waiting times, which can vary based on demand and time of day. The goal is to not only determine the shortest route from origin to destination but also to optimize charging stops to reduce total travel time and mitigate risks associated with inoperable charging stations.

At the core of this challenge lies a combinatorial explosion in the number of possible routes and charging stop combinations. For a single long-distance trip, there can be numerous potential charging stations, each with distinct properties such as location, power output, availability, and expected waiting times. An effective EV route planner must select the best combination of charging stops and adjust the route dynamically to account for these variables. However, considering every possible path that includes all potential charging stations is computationally infeasible. This complexity increases further when considering the state of charge (SoC) at which to recharge at each station, as fully recharging the battery is often not the best option.

Time dependency adds another layer of complexity. Charging stations may be unavailable at certain hours or have fluctuating wait times based on real-time demand. As a result, routing algorithms that traditionally assume fixed costs and straightforward paths struggle to efficiently integrate dynamic data into their calculations. Moreover, real-time data availability, such as live station availability or traffic conditions, is often limited, requiring the route planner to make predictions that balance accuracy with computational efficiency.

To address these issues, an EV routing solution must intelligently reduce the search space while maintaining high accuracy in estimating the trip time. It needs to prioritize efficiency by pre-computing certain paths and optimizing charging station selection based on real-time data wherever possible. This paper explores a planner design that manages the inherent combinatorial explosion in EV routing, minimizes the total trip time, and that effectively incorporates real-time and dynamic information to make electric vehicle travel more feasible over long distances.

Background

Optimal pathfinding for electric vehicles, including necessary recharge stops, can be integrated directly into the non-EV routing graph, as demonstrated in (1). However, the approach faces scalability issues when new constraints are introduced. For example, while it limits the number of charging stops, it doesn't minimize total trip time. Reference (2) refines the battery charging model and introduces partial recharges. Like in (1), they use a side graph of charging stations with a modified Contraction Hierarchy (CH) algorithm to accelerate the queries. Personalized road planning is explored in (3), using a side graph construction similar to Customized Road Planning (CRP). However, the results are limited to German highways.

The impact of neglecting nonlinear charging times is underlined in (4), as it can significantly increase the operational costs or result in infeasible routes. It brings the conclusion that piecewise linear approximation provides a practical balance between complexity and accuracy. In (5), the authors propose a planner that minimizes the total expected trip time, which includes waiting, charging and driving time. They also emphasize the importance of considering contingency plans at each charging station to deal with unexpected unavailability.

Paper (6) introduces the Charging Function Propagating (CFP) algorithm that uses labels representing all possible tradeoffs between charging time and resulting SoC on top of a CH algorithm. It gives very good results at minimizing the total trip time, even on big map instance, but does not consider waiting times and alternative paths. The same authors focus on energy optimal routes in (7). While part of the approach is similar, partial recharge is simplified by only considering a small discrete set of targets SoC level. This is done by expanding the charging station in multiples nodes. All those works only consider the minimization of one objective at a time (e.g. time, or energy) with fixed driving speeds. In (8), the authors tackle the problem of minimizing the total trip time with adaptive speeds. Adjusting this speed along the road segments enables us to adjust the consumption and to lower the number of recharge operations, and thus to potentially reach the target quicker.

Even by using a pre-computed charging stations graph, the computational effort to find good solution may be high considering the number of charging stations across a continental-wide map and given the high density of graph connections. The problem will only get worse with the increase in charging stations number that will follow EV adoption. This issue can be mitigated by clustering nearby stations, as proposed in (9).

Contribution

This paper introduces a library designed to solve the complex problem of electric vehicle (EV) route planning by integrating optimal charging stops into long-distance navigation. It addresses the combinatorial explosion of possible routes and charging station choices by employing a layered approach that combines pre-computed data structures with dynamic, real-time calculations. This approach facilitates efficient selection of charging stations that minimize total travel time while considering both battery constraints and real-time charging station availability.

The library's architecture is built around two main graph structures: the **Driving Station Graph** and the **Planning Station Graph**. The Driving Station Graph connects charging stations that can be reached within a single battery charge, ensuring the planner does not attempt unrealistic paths. The Planning Station Graph, in turn, represents each charging station with nodes that reflect different states of charge (SoC), allowing the planner to optimize charging levels dynamically at each stop. Together, these graphs allow to narrow the set of feasible charging stops to those that align with battery levels and driving range constraints. To further improve efficiency, our library uses **clustered station groups** in areas with dense charging networks to lower the computation charge and incorporates real-time data regarding station availability and expected wait times.

Finally, an evaluation methodology is introduced to measure robustness of the computed plans, as uncertainties of the real world can always put them down.

The planner library

Our library is structured to operate efficiently in a cloud-based environment, integrating pre-computed data with real-time updates to deliver optimized EV routes. It leverages an external router algorithm, based on Contraction Hierarchies (CH), to compute driving times between stations rapidly (10). By reducing the graph size through hierarchical compression, this router allows our planning library to perform quick and efficient lookups of travel times between charging spots, even in large geographic areas.

The architecture of the library is designed to support concurrent requests, ensuring scalability for high-demand applications. It pre-processes a **Driving Station Graph** that connects reachable stations within a single battery charge and computes a **Planning Station Graph** that models each station with discrete state-of-charge (SoC) levels. These pre-computed structures enable efficient query-time pathfinding, minimizing the computational load.

Functionally, diverse requirements must be handled: support for multiple vehicle models with different battery capacities, integration of real-time charging station availability, and adaptability to user-defined waypoints. The system is configured to optimize not only for the fastest route but also to minimize waiting and charging times by selecting the best charging stations along the route. The library is built to be flexible in threading, ensuring that multiple instances can operate independently while sharing pre-computed graph data, maximizing efficiency and responsiveness in multi-user scenarios.

Driving and planning graphs

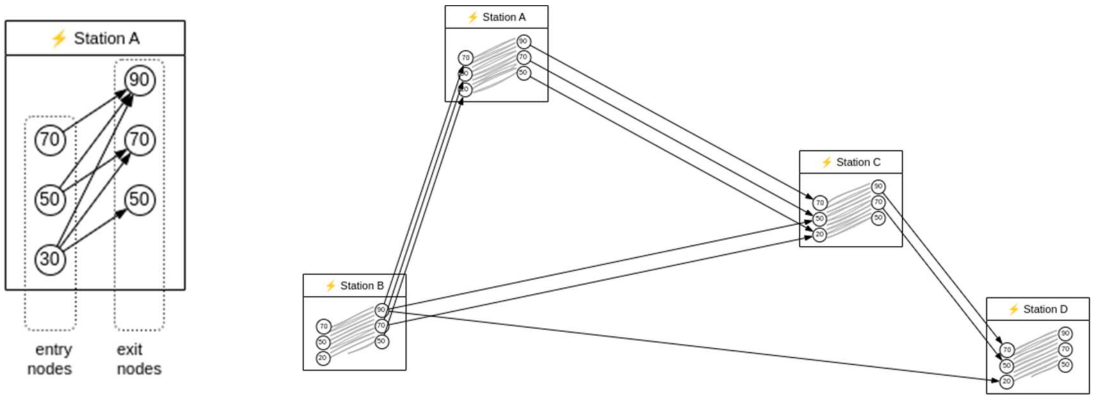
Our planning algorithm relies on two foundational graph structures that manage the complexities of EV route planning: the **Driving Station** and the **Planning Station Graphs**. These graphs streamline route calculations by limiting search spaces and incorporating key EV-specific considerations, such as battery range and charging times.

The **Driving Station Graph** connects charging stations together, provided they are reachable within the full battery capacity of the vehicle. This graph is pre-computed and reflects the fastest paths between stations that can be traversed without recharging with a full battery. Each node represents a charging station, and edges carry weights based on driving time and energy consumption. By limiting connections to reachable stations within a full charge, the Driving Station Graph reduces unnecessary computation and ensures that planned routes remain feasible based on the vehicle's battery constraints.

This driving station graph is not enough on its own to solve the planning problem. Indeed, it does not capture the different possible strategies that can be followed at each charging station, given that fully charging a battery is not always the best option. For instance, if the charging station power is low, it may be beneficial to charge less and to benefit from a fast-charging station later.

To deal with this issue, the **Planning Station Graph** builds upon the Driving Station Graph by modeling each station with multiple nodes that represent discrete states of charge (SoC). This finer granularity allows the algorithm to determine the optimal charging level at each stop along a route. Each node in the Planning Station Graph represents a specific SoC at a station (e.g., 30%, 50%, 90%), and edges in between represent transitions due to charging. Additionally, the graph incorporates station-specific properties such as charging power and estimated waiting times, enabling the planner to minimize overall trip time by selecting ideal charging stops and SOC levels.

An example of SoC transitions at some charging station node is given in Figure 1.a, where the transitions are weighed by their corresponding charging time. A small planning graph example is illustrated in Figure 1.b, where the decreases of SoC level from one station to the next are based on the weights of the driving station graph.



a) b)

Figure 1: a) Illustration of the transitions inside a single charging station for some given SoC discretization level **(11)**. **b)** A small planning graph where the different possible SoC levels at the exit of a charging station are linked to the inputs of the other stations in range.

Charging stations clustering

To further optimize computation time and manage the high density of charging stations in certain areas, we implement a **clustering** strategy (12). This allows the planner to treat groups of nearby stations as single entities during route calculations, reducing the complexity of the search space while preserving accuracy in route planning. This is especially critical in urban areas or along popular routes where multiple stations are within proximity.

The clustering process begins by grouping charging stations based on proximity and reachability within a specified energy range. We adopted a modified density-based clustering algorithm (modified DBSCAN), which ensures that all stations within a cluster are accessible to one another with less than a configurable energy threshold. This approach forms clusters within which each station is reachable from any other station.

For each cluster, a **representative station** is selected based on specific criteria, such as the station's charging power, availability, and waiting times. By only considering this representative node when building the planning graph, it is possible to greatly reduce the search space to explore when resolving a query. Indeed, this heuristic enables a query to be solved in two steps. First, the general planning solution is sketched by selecting which sequence of clusters must be visited to reach the destination. Then, a second step enables us to determine more precisely which charging station to select in each visited cluster. This clustering approach enables the system to retain high computational efficiency without sacrificing too much the route quality, making it feasible to handle real-time, large-scale route requests in cloud-based environments.

Scoring and charging stations selection

To optimize the total trip time, the planner integrates a scoring system that evaluates charging stations based on multiple factors, including waiting time, charging speed, and fallback options. This scoring system guides the algorithm in selecting the most efficient charging stops along the route, ensuring that each stop minimizes delays while meeting the EV's energy needs.

Each charging station is assigned a **score** that combines several key elements:

- **Waiting Time**: Estimated time the EV may need to wait before starting to charge, influenced by the number of charging points and real-time occupancy data (if available).
- **Charging Time**: Calculated based on the station's charging power and the vehicle's battery characteristics, indicating the time required to reach a target state of charge (SoC).

- **Fallback Time**: To account for station unavailability due to unexpected issues (e.g., station out of service), fallback time represents the additional time required to reach a nearby alternative station.

The **station score** is computed as a weighted sum of these elements, with fallback time weighted by the probability that the station may be unavailable. This score penalizes stations with high waiting times or low charging speeds, directing the algorithm toward stations that offer quicker, more reliable experience.

Once stations are scored, the algorithm integrates the scores into the **Planning Station Graph**, where each station cluster is considered based on its calculated score. During the pathfinding step, the planner selects the sequence of clusters that minimizes the total route score, effectively balancing driving time, charging time, and potential delays. If a cluster has multiple viable stations, the algorithm selects the representative with the best score, ensuring both accuracy and computational efficiency.

Pathfinding algorithms

The pathfinding algorithm works in *two steps*. First, it computes the optimal sequence of charging station clusters enabling to join the start and end positions of the planning request. Then, in the second phase, it selects one charging station in each of the selected clusters. Since clusters may have many potential charging stations, this design choice allows handling large graph sizes, although it may not always find the optimal solution.

To efficiently compute the optimal sequence of clusters during the first phase, the planning library utilizes the A* algorithm, an enhancement of Dijkstra's algorithm that speeds up pathfinding by incorporating a heuristic. A* is particularly well-suited for the EV routing problem, as it allows the planner to estimate the remaining cost to the destination, guiding the search toward more promising paths and minimizing computation time.

In this case, the A* heuristic combines a **straight-line distance** to the destination with an estimate of the **ideal travel time**, which assumes maximum driving speeds and optimal charging sessions. This heuristic underestimates the true cost, ensuring that A* maintains optimality while providing a clear sense of directionality. For each node (charging cluster and SoC), the heuristic accounts for the estimated driving time to reach the destination as well as some minimal but necessary charging sessions. This approach is particularly effective in scenarios involving significant distance, as it directs the search toward stations that are most likely to reduce the overall trip duration.

Once the sequence of cluster is determined, *the second step* focuses on the selection of one charging station in each of them, to compute the final trip. To do so, the search space is reduced by only considering the charging stations belonging to the selected clusters, and by visiting them in the same order. This problem is much simpler than the first and can be solved quickly with a simple vanilla Dijkstra algorithm.

Time-dependent constraints

One of the unique challenges in EV route planning is managing time-dependent constraints, especially about charging station availability and potential wait times. Unlike traditional routing, where refueling is typically quick and available at all hours, EV charging requires careful consideration of station opening hours and fluctuating demand that may impact wait times. To provide realistic and efficient routes, the planning algorithm tracks the current time at each explored node. This enables the integration of these dynamic factors into the planning process.

Charging station demand often varies by time of day, leading to fluctuating wait times that can significantly impact travel efficiency. Real-time or historical data are incorporated, when available, to estimate waiting times at each station. This is integrated into the Planning Station Graph, where each node's score dynamically adjusts based on expected wait times, as explained in the scoring section. For stations that frequently experience high demand, the planner may prioritize alternative stops with shorter queues, even if these require a slight detour.

Some charging stations may be available only during specific hours, which can affect route feasibility and total trip time. The planning algorithm integrates these time constraints by tagging each station with its opening hours and by adding a potential waiting time at each charging node. This additional waiting time is used when arriving at a closed station during the graph exploration, in order to wait for its next opening slot. As a result, this penalty is likely to favor other scenarios that prevent the EV to arrive at a closed station, avoiding unnecessary delays.

Route quality evaluation

Evaluating and comparing EV route planning algorithms presents unique challenges. Many route planners function as black boxes, each potentially using different data sources, energy models, or minimization objectives. For instance, two planners might rely on different maps, use varied lists of charging stations, or access distinct traffic data. They may also differ in their optimization goals, with some prioritizing travel time, while others might focus on energy efficiency or minimizing charging stops. These variations can lead to vastly different outcomes, even for the same origin-destination pair, complicating any attempt to evaluate different planners side by side.

We do not address this challenge in this paper. Rather, we establish our own quality criteria and use them to evaluate our own implementation. Our criteria focus on both the planned route's resilience to unexpected events and its efficiency under ideal conditions. Specifically, we assess **total trip time** as well as **expected trip time**, which accounts for potential issues such as station closures, malfunctions, or any unexpected event preventing the driver from following the plan and charging where he should have to.

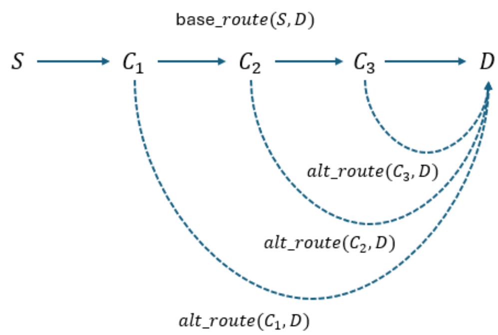


Figure 2: Planned route joining S to D through charging stations C_1, C_2 and C_3, along with the branching cases in case of failure. Those alternatives routes contain their own (and potentially different) charging stations, but we do not consider they could fail.

Total trip time

A fundamental criterion for evaluating the quality of any route planner is the **total trip time**. This metric represents the sum of all driving, charging, and waiting times along the planned route, providing a straightforward measure of efficiency. Minimizing total trip time is crucial for EV drivers, as it directly impacts the practicality and convenience of their journey. An effective planner must optimize the sequence of charging stops and their durations to ensure that the total travel time is as short as possible while adhering to battery and route constraints.

Expected trip time

While simple to compute, the total trip time alone may not fully capture the robustness of the plan under real-world conditions, necessitating complementary evaluations. This is because some real-world events are simply impossible to forecast, resulting in potential failures of even the most carefully crafted plans. Many things can go wrong when reaching a charging station. For example, all the spots may be occupied by drivers that are nowhere to be seen. Or, the waiting time may be unexpectedly high, which may occur for stations that only have a few low-power chargers. Sometimes, the chargers can malfunction or refuse the driver payment method. Whatever the reason, there is always a small but non-marginal probability that the driver must deviate from the plan and find another charging station. Of course, this is only possible if another station is in range considering what remains in its battery. This perspective is stressful for an EV driver and a good planner must therefore keep enough battery margin and backup solutions to cope with this seldom but nonetheless possible scenario.

That is why our second criterion is the **expected trip time**, a metric that considers the possibility that any scheduled charging stop fails. When this happens, an alternative plan must be re-computed from the corresponding location and remaining SoC. Figure 2 illustrates the concept with three scheduled charging stations, C_1 to C_3, along a route planned from starting location S to destination D.

The expected time score is computed *recursively* and *backward* from the destination to the source. For instance, if we call $p(C_3)$ the probability of success at C_3, then the expected time from C_3 to D is given by: $\varepsilon(C_3 D) = p(C_3).\tau(C_3 D) + \big(1 - p(C_3)\big).\tau(alt\_route(C_3 D))$, where $\varepsilon(.)$ stands for the expected time, $\tau(.)$ designates the total trip time (without failure), and $alt\_route(.)$ is the backup plan starting at a disabled charging station. In this equation, we do not envision that the backup alternative may fail at its turn. This is a reasonable assumption since the probability of double failures is expected to be very low.

For a general plan $P = P_0 \ldots P_n$, starting from the source P_0 and ending to the destination P_n, we can generalize in the following *recursive* way:

$$\forall 0 \leq i < n: \quad \varepsilon(P_i P_n) = p(P_i).\big(\tau(P_i P_{i+1}) + \varepsilon(P_{i+1} P_n)\big) + \big(1 - p(P_i)\big).\tau(alt\_route(P_i P_n)).$$

It may happen that the planner could not find any alternative route in case of failure at some charging station P_i. In that case, the total trip time is increased by some *penalty* and the corresponding station is marked as a *mandatory* station. The total number of such mandatory stations is a *third indicator* that we use to assess the quality of a plan computed by the planner algorithm.

Table 1 : Summary of different results computed over the map of North America, for three different trip length categories and averaged on 1000 requests. The total trip time indicates the driving, waiting, and charging time for a route without incident. The expected trip time includes the likelihood that rerouting through another charging spot is required.

	Route length category		
	Short (330)	**Medium (616)**	**Long (54)**
∞ battery trip time	2h 50m	6h 27m	15h 36m
Comp. time	0.67 s	0.85 s	1.47 s
Tot. trip time	4h 27m	9h 23m	21h 50m
Exp. trip time	4h 31m	9h 44m	22h 44m
#stops / #mandatory stops	1.3 / 0.1	3.5 / 0.2	10.3 / 0.5

Experiments

The planner algorithm has been implemented in *C++* and all the experiments ran on a map of *North America* provided by *Here*. This map has been used to precompute the driving and planning graphs interconnecting *~34000 charging stations* coming from the same data provider. As explained above, those graphs contain the driving time, and the energy required to travel between the station pairs. Travel times come from the *Here historical traffic data*, while the energies come from a *private EV consumption model* using the speeds and slopes of the road network. The average consumption of this vehicle and its battery capacity are around 17k Wh/100km and 71k Wh respectively. Charging from 20% to 80% takes roughly 25 minutes.

The measurements in Table 1 have been conducted *single thread* on an AMD Ryzen 9 5900X Processor@3.70 GHz on 1000 randomly generated source, destination pairs. The *initial battery SoC* is chosen uniformly in the interval [20, 80%] and each request requires at least one recharge. The crow fly distance between the source and destination of each pair is not uniform. Instead, we used a *lognormal distribution* to better fit what is encountered by a server coping with real user requests. By using the parameters $\mu = 5.6, \sigma = 1.1$, we got 48% of *short routes* in the interval [0, 250 km], 45% of *medium routes* in the interval [250, 1250 km] and 7% of *long routes* in the interval [1250, 3500 km]. The planner always attempts to reach a charging spot with at least 15% of left battery.

The expected time is computed by assuming that the plan can fail at any station with an arbitrary probability of *5%*. In this case, the planner tries to compute an alternative path without reusing the departure station and without limiting the minimum battery level. If not possible, an arbitrary penalty of *one hour* is added to the initial trip time.

Conclusion

We presented in this paper the development of our solution to the complex problem of electric vehicle (EV) route planning, emphasizing the integration of optimized charging stops into long-distance navigation. By leveraging dual-layered graph structures, dynamic real-time data at charging stations, and an innovative clustering mechanism, our approach effectively addresses the combinatorial challenges inherent in EV routing. Experimental results on a North American map validated the efficiency and scalability of the proposed system, highlighting good performances in both total and expected travel times. The inclusion of a robust scoring system, accounting for waiting times, charging speeds, and fallback options, enhances the reliability of the planner under real-world conditions. Finally, the computation times, even for very long routes, remain around a bunch of seconds.

Future research could explore further enhancements in real-time data integration and adaptive algorithms to optimize for evolving EV infrastructure and user needs. Indeed, the present algorithm requires to pre-compute all the inter-stations travel times and energies. The heavy computational resources it requires limits the scalability of the solution for more subtle kinds of routing: eco-routes, personalized planning, etc. A future axe of research could explore if the introduction of deep learning techniques could provide good alternatives to those precomputations.

References

1. *Quick and Energy-Efficient Routes – Computing Constrained Shortest Paths for Electric Vehicles.* **Storandt, Sabine.** 2012, IWCTS '12: Proceedings of the 5th ACM SIGSPATIAL International Workshop on Computational Transportation Science.

2. *Electric Vehicle Routing with Realistic Recharging Models.* **Zündorf, Tobias.** s.l. : Master Thesis, 2014.

3. *Personalized Route Planning with Recharging for Electric Vehicles.* **Højsleth, Mathias.** 2016, Aalborg University.

4. *The electric vehicle routing problem with nonlinear charging function.* **Montoya, Alejandro, et al.** s.l. : Transportation Research Part B: Methodological, vol 103, 2017, Transportation Research Part B: Methodological.

5. *An Efficient Electric Vehicle Path-Planner That Considers the Waiting Time.* **Gareau, Jaël Champagne, Beaudry, Éric et Makarenkov, Vladimir.** s.l. : SIGSPATIAL '19: Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2019.

6. *Shortest Feasible Paths with Charging Stops for Battery Electric Vehicles.* **Baum, Moritz, et al.** 2019, Transportation Science, vol 53.

7. *Energy-Optimal Routes for Battery Electric Vehicles.* **Baum, Moritz, et al.** 2019, Algorithmica, vol 82.

8. *Modeling and Engineering Constrained Shortest Path Algorithms for Battery Electric Vehicles.* **Baum, Moritz, et al.** 2020, Transportation Science, vol 54.

9. *Cluster - Based Approach for Route Planning Optimization of Electric Vehicles.* **Argiolas, Luca.** s.l. : Politecnico Di Torino, 2023, Master's Degree Thesis.

10. *Fast Matrix Queries and Application to Routing Optimization Problems.* **Gonsette, Jean-Sébastien et Meunier, Noémie.** s.l. : Proceedings of the 29th ITS WORLD CONGRESS Suzhou, 2023.

11. **Kostas Kollias, Sreenivas Gollapudi.** Addressing Range Anxiety with Smart Electric Vehicle Routing. *https://research.google/blog/addressing-range-anxiety-with-smart-electric-vehicle-routing/.* [En ligne] 2021.

12. *Efficient Electric Vehicle Route Planning Using Station Clustering in Graphs.* **Meunier Noémie, Jacques Arnaud, Gonsette Jean-Sébastien.** s.l. : Proceedings of the 31st ITS WORLD CONGRESS Atlanta, 2025.