An Efficient Electric Vehicle Path-Planner That Considers the Waiting Time

Jaël Champagne Gareau
Éric Beaudry
Vladimir Makarenkov
champagne_gareau.jael@courrier.uqam.ca
beaudry.eric@uqam.ca
makarenkov.vladimir@uqam.ca
Université du Québec à Montréal
Montréal, Québec

Figure 1: An electric vehicle charging at a public level 2 station in Montreal

ABSTRACT

In the last few years, several studies have considered different variants of the Electric Vehicle Journey Planning (EVJP) problem that consists in finding the shortest path (according to time) between two given points, passing by several charging stations and respecting the range of the vehicle. The total time taken by the vehicle is the sum of the driving time, the charging time and the waiting time. Unfortunately, the consideration of the waiting time has been neglected by previous studies. This study aims to fill this gap by introducing: (1) a graph relabeling technique using a probabilistic model of charging station occupancy generated using real EV stations data; (2) an alternative paths generation technique which accounts for worse than expected waiting time at various charging stations. Our empirical results indicate that the *a priori* consideration of charging station occupancy by graph relabeling can reduce the waiting time by more than 75%, while having a negligible impact on the driving time, and that the generation of alternative paths helps reduce the waiting (and total) time even more. For our public station network dataset and the current station occupancy (for now quite low), the mean total journey time (computed over 1000 requests) decreased by 17.3 minutes when our new technique was used.

SIGSPATIAL '19, November 5–8, 2019, Chicago, IL, USA

ACM ISBN 978-1-4503-6909-1/19/11…$15.00
https://doi.org/10.1145/3347146.3359064

CCS CONCEPTS

• **Theory of computation → Shortest paths**; *Dynamic graph algorithms*; • **Computing methodologies → Planning under uncertainty**; **Discrete space search**; *Search with partial observations*; *Probabilistic reasoning*; • **Mathematics of computing →** *Queueing theory*.

KEYWORDS

Electric Vehicles, Planning, Waiting Time, Minimization, Occupation, Contingency, Alternative Path, Charging Stations

ACM Reference Format:
Jaël Champagne Gareau, Éric Beaudry, and Vladimir Makarenkov. 2019. An Efficient Electric Vehicle Path-Planner That Considers the Waiting Time. In *27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '19), November 5–8, 2019, Chicago, IL, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3347146.3359064

1 INTRODUCTION

Electric vehicles (EVs) are an attractive alternative to fossil-fuel vehicles to reduce greenhouse gas emissions. However, their limited range and their high charging time represent a major obstacle to their massive adoption. Moreover, long journeys require careful planning to determine the charging stations to be used in order to avoid running out of energy. For example, today (year 2019), affordable EVs have an average range of around 250 km. This limited range implies the need to make recharging stops many times on long journeys.

EV journey planning (EVJP) is a complex problem which cannot be effectively solved by conventional approaches. Indeed, EV planners need to take into account not only various driving factors applicable to conventional vehicles (the wind, the energy needed to fight the air resistance relative to the speed, the traffic, eventual detours, etc.), but also factors specific to EVs, such as the level of charging stations (which influences the charging speed), the non-linearity of the charging curve of the battery, the topography of the map (EVs can recover some energy when moving downhill), the probability of the stations' occupancy as well as the expected waiting time at the charging stations.

While many of these factors have been addressed by previous studies, very few of them have considered the waiting time in the objective function to minimize. This factor is increasingly important because in many countries, the number of EVs on the road increases faster than the number of charging stations [6, 15, 17]. The waiting time will thus probably continue to increase at the majority of the stations for some time.

This paper proposes some improvements to EV planning techniques by considering expected waiting time at charging stations. Our first contribution is the construction of a probabilistic model to predict waiting time at each charging station and its integration in the planning algorithm using dynamic edge relabeling. This improvement allows the planner to optimize the total journey time by considering a trade-off between more driving time and less waiting time. Our second contribution is to take into account alternative paths which are dynamically chosen during the journey. Indeed, while executing the plan, the real-time occupancy data might suggest a waiting time which is worse than what was initially expected. If this happens, it is advantageous to have alternative paths at our disposition. We therefore propose an alternative paths generation mechanism to account for worse than expected charging station occupancy. To the best of our knowledge, this is also the first study that quantifies the time saving of considering the waiting time when using real data on charging station placement and occupancy (the existing approaches considering the waiting time use artificial grid networks and artificial probabilities of station occupancy).

The remainder of the paper is structured as follows. In Section 2, existing studies related to the various EV planning problems are presented. In Section 3, a mathematical formulation of the EVJP and a basic algorithm to solve the problem (on which we base our techniques) are presented. Sections 4 and 5 then describes the proposed methods to account for the waiting time in the planner. Finally, Sections 6 and 7 present the evaluation of our methods (testing methodology and results) and the conclusion, respectively.

2 RELATED WORK

An EV planning framework taking into account the possibility that the battery recharges itself during a journey (through braking or potential energy loss) already exists [16]. This framework also considers other factors which may influence energy expenditure, including the speed limit on each road segment, the mass of the vehicle and occupants, the air resistance coefficient, etc. However, it does not take into account the possibility of using charging stations to charge at an intermediary node along the path. This problem is usually called EV path planning (EVPP). The algorithm used is a variant of A* [10], called *Energy-A**. It uses a consistent heuristic $h = c + \tilde{c}$, where c is a combination of the potential energy difference between the current node and the goal, including the energy loss inherent to the displacement, and $\tilde{c}$ represents the battery constraints. No pre-calculation is possible in this model since the energy cost of each edge may depend on data specific to each request. This method makes it possible to find a solution in $O(n \log n)$ (where n is the number of nodes). Similar results have been found [7].

Many studies have extended the aforementioned EVPP technique to solve some generalized instances of the original problem. Many of them have extended the problem by considering the possibility of using charging stations to charge the battery if the range is not sufficient to directly go from the departure to the arrival point [1]; some of them using techniques based on dynamic programming [19]. One of their weaknesses is that they do not consider the inherent uncertainty of the problem and, in particular, the probability of occupancy and the expected waiting time at every charging station.

The possibility of recharging partially at charging stations has been studied by using the formalism called *State of charge* [2]. This approach considers different possible states of charge and optimizes them for the best possible choice (e.g., it might be advantageous to recharge partially if the next part of the journey goes downhill).

Different factors affecting the EVs energy consumption have also been studied [21]. For example, the effect of the road gradient [14] (which impacts linearly the energy consumption) and the effect of the ambient temperature [4, 13] (which affects linearly between $-15°$C and $20°$C the capacity of an electric vehicle battery). As an example of the temperature impact on the battery, the Nissan Leaf has a range that varied from about 50 km to about 165 km, depending on the temperature.

Recently, some studies have started to take into account the waiting time [12, 18]. The former work focuses on EV routing to deliver multiple packages in an optimal order, starting and ending at a depot (similar to the traveling salesman problem (TSP), but considering the EV range and charging stations). This problem is called EVRP-MRUA (EV routing problem with mid-route recharging and uncertain availability). This research models the problem using a Markov Decision Process (MDP) that aims to minimize the cost for the operator of an EV fleet and the global time to deliver all the packages.

The latter work focuses on a problem which is more closely related to our problem (i.e., path-planning for a single EV going from a departure node to an arrival node while respecting the range of the EV, and considering the waiting time and the uncertainty related to the availability of EV charging stations). The authors first described some efficient algorithms to find an optimal recharging policy (how long to charge at every station) for a given path, then extended these algorithms to find an *a priori* optimal routing and recharging strategy (also using MDP). The graph considered had a charging station at each of its nodes. The optimal policy (including the path and the charging policy on that path) was computed in $O(n^4)$, where n is the number of nodes (i.e., charging stations) in the network. Heuristics were used to reduce the size of the problem. In that model, the availability of charging stations was known only when the EV was arriving at a given station. The evaluation of this technique was done on a grid of 500×500 nodes, with a common distance of 5 miles between every pair of adjacent nodes. The probability of occupancy and waiting time at each station were generated randomly using uniform distributions.

3 ELECTRIC VEHICLES JOURNEY PLANNING

This section defines formally the EVJP problem and presents a baseline planner that solves the problem optimally when considering the driving and charging time, but without considering the waiting time (i.e., assuming waiting time is inexistent). This baseline will be used as a base to which we will add the two proposed techniques. It will also be used in the evaluation of our novel techniques to measure the average time reduction when considering the waiting time.

3.1 Problem Formulation

Definition 1. A *road network* M is modeled by a tuple (V, E, λ, μ, S), where (V, E) is a digraph and λ, μ are two labelings of the edges. More specifically:

- V is the set of nodes (latitude, longitude) considered on the map;
- E is the set of road segments (edges);
- $\lambda\colon E \longrightarrow \mathbb{R}^+$ gives the length (in m) of every edge;
- $\mu\colon E \longrightarrow \mathbb{R}^+$ gives the expected speed (in m/s) at every edge;
- S is the set of all charging stations.

We associate every charging station $s \in S$ with the nearest vertex $v_s \in V$.

Remark. The λ and μ labeling can be used to define a labeling giving the expected time to cross an edge (i.e., using edges representing time instead of distances in the graph). Both formulations are equivalent.

The μ labeling can be based on empirical data on the average speed of the vehicles on every edge, or can simply be set to the maximum allowed speed of each road segment. We now formally define an EV journey planning problem and define its solution formulation:

Definition 2. An *EVJP* problem is defined by the tuple $(M, \rho, \alpha, \omega)$, where

- M is the road network;
- $\rho \in \mathbb{R}^+$ is the range of the EV;
- $\alpha, \omega \in V$ are the departure and arrival nodes.

Remark. It is possible to generalize it to the case where $\alpha, \omega \notin V$ by using a KD-Tree to find the nearest corresponding node in the graph.

Definition 3. A *solution* of the EVJP problem $(M, \rho, \alpha, \omega)$ is a tuple (P, Q), where

- P is a finite sequence of $k + 1$ nodes $\langle P_0, P_1, \dots, P_k\rangle$ (where $P_i \in V$);
- Q is a subsequence $\langle P_{i_0}, \dots, P_{i_{b+1}}\rangle \subseteq P$ containing the b used charging station in the solution, and containing α and ω;
- $P_{i_0} = P_0 = \alpha$;
- $P_{i_{b+1}} = P_k = \omega$;
- $\forall j \in \{0, 1, \dots, b\}, \mathrm{dg}(Q_j, Q_{j+1})^1 \leq \rho$ (i.e., every consecutive node in Q are at a distance which is smaller than or equal to the range ρ).

In other words, P is the sequence of nodes by which the EV needs to travel according to the solution, and Q is a subsequence of P containing the charging stations which need to be used in the journey (as well as α and ω).

Our objective is to find a solution to the EVJP problem that minimizes the total time of the journey from α to ω, including the driving time, the charging time and the waiting time. This is formalized in the next definition.

Definition 4. An *optimal solution* to EVJP is a solution (P, Q), as stated in Definition 3, minimizing the following objective function:

$$\mathrm{Z}(P, Q) = \mathrm{DT}(P) + \mathrm{CT}(Q) + \mathrm{WT}(Q),$$

where DT, CT and WT are respectively the expected driving time, charging time and waiting time. They are given by:

$$\mathrm{DT}(P) = \sum_{i=0}^{k-1} \frac{\lambda(P_i, P_{i+1})}{\mu(P_i, P_{i+1})},$$

$$\mathrm{CT}(Q) = \sum_{i=1}^{b} \mathrm{ECT}(Q_i),$$

where $\mathrm{ECT}(Q_i)$ is the expected charging time at the station Q_i when considering states of charge [2] and the station level (e.g., a 220VAC or a 400VCC station).

3.2 Base Planner

We now present the base planner we will use as a baseline in the tests. First, the distance between each pair of charging stations is pre-computed and stored in a matrix $D = (D_{ij})$, where $D_{ij} = \mathrm{dg}(s_i, s_j)$. We also store the optimal path between each of these stations for further use.

Next, for every request, we build a simplified graph (s-graph) (V', E') containing only the nodes associated with the charging stations as well as the nodes of departure and arrival ($V' = S \cup \{\alpha, \omega\}$). Edges with weights corresponding to the pre-calculated distances are added to this new graph between every pair of charging stations. Dijkstra's algorithm [5] is then used twice (the first time from the starting point, and the second time from the arrival point, on the reversed graph) to add edges from the departure to every charging stations and from every charging stations to the arrival. The computations for this step can be accelerated by using contraction hierarchies [9]. The two executed Dijkstra passes can also be run in

[1] $\mathrm{dg}(A, B)$ is the distance in the graph between A and B

parallel. The new graph we obtain is a complete graph, from which we remove every edge whose length is larger than the EV range.

With this simplified graph, we execute the A* algorithm (using the Great-circle distance as heuristic) from α to ω, which is enough to get a sequence Q as specified in Definition 3. This sequence satisfies the last part of the definition (the EV range constraint) insofar as every intermediary node is a charging station. Consequently, there is no need to consider the range of the vehicle at this point. The sequence P can then be found using the data from the previously computed paths from the two Dijkstra passes on the original graph and from the pre-calculation of the path between every pair of charging stations.

The pseudo-code of this base planner is shown in Algorithm 1.

Algorithm 1 Base planner algorithm

```
Compute the matrix D and the optimal path between every pair of stations
Construct the s-graph containing every charging station
for all request (α, ω, ρ) do
    Run Dijkstra from α on the original graph
    Run Dijkstra from ω on the reversed original graph
    Add α and ω to the s-graph and add edges with length ≤ ρ
    Run the A* algorithm on the s-graph from α to ω to find the sequence Q
    Find the sequence P from Q using all computed paths
end for
```

Remark. We make some assumptions in this work to keep the presentation clutter-free and stay focused strictly on the consideration of expected waiting time and real-time occupancy. However, many of these assumptions can be easily removed by slightly modifying the base planner. For example, we assume the initial EV charge at α to be full (i.e., $= \rho$), but this assumption can be removed by simply modifying the weight of the out-edges of α. The charging model of the battery and a remaining charge variable could also be easily incorporated in our model allowing one to consider partial recharge and the non-linearity of the battery's charging curve.

Let $n = |V|$ and $m = |E|$. The execution of a request using this base planner has a time complexity of $O(n \log n + m)$ time, i.e., the same complexity as Dijkstra's algorithm (when implemented with a Fibonacci heap [8]). In a real world road network, the maximum degree of a node is bounded by a small number (based on the lowest number of road intersections possible at any given node). This implies that we can reduce the time complexity of the algorithm to $O(n \log n)$.

Remark. Many other studies consider that $\alpha, \omega \in S$ and suppose the simplified graph is given in input. If it is the case, note that the two Dijkstra passes are unnecessary, and the time complexity becomes simply $O(|S| \log |S|)$.

4 GRAPH RELABELING

Since minimizing the waiting time is done by going to a charging station that might require a detour from the optimal path (thus increasing the driving time), we need to find the optimal trade-off between the waiting time and the driving time.

To consider the waiting time and find the optimal trade-off, the planner needs some *a priori* information about the probability of occupancy and the expected waiting time at every charging station (which depends on the day of the week, the time of the day, etc.). To get this information, we need historic data at every station. We assume these data are available and can be used to generate a family of functions that give the probability of occupancy and the expected waiting time at every hour and day. The following definition makes this more precise:

Definition 5. for every station $s \in S$, the *a priori probability of occupancy* at station s is given by

$$\begin{aligned} f_s \colon \{Monday, \ldots, Sunday\} \times \{0..23\} &\to [0, 1] \\ (d, h) &\mapsto \mathbb{P}(s \text{ is occupied} \mid \text{Day} = d \wedge \text{Hour} = h). \end{aligned}$$

and the *expected waiting time* is given by

$$g_s \colon \{Monday, \ldots, Sunday\} \times \{0..23\} \to \mathbb{R}^+$$

The numbers $g_s(d, h)$ can be seen as being the mean service time in a waiting queue (normally denoted $\frac{1}{\mu}$ in queueing theory). In other words, the time of service corresponds to a time-dependent (i.e., non-homogeneous) exponential distribution. In our model, we consider the *a priori* mean waiting time instead of a complete waiting queue formalism. We did so because the information about the real-time presence of a waiting line at an EV station is not known in the majority of EV stations network (including the one we used in our tests). That being said, the g_s functions could be easily modified to consider such waiting queues.

Remark. Of course, the model can easily be refined by considering smaller time intervals. We chose one hour as the interval because we didn't have enough historic data to run our tests on smaller intervals while keeping a reasonable variance.

We now adapt the base planner presented in Section 3.2 to consider the $\{f_s\}$ and $\{g_s\}$ function families in order to minimize the mean waiting time.

In addition to the edge labeling given by the mapping λ, we define another edge labeling ξ that considers the length of the path (like λ), while also considering the expected waiting time.

Definition 6. Let $e = (u, v) \in E$. The labeling

$$\xi \colon E \times \{Monday, \ldots, Sunday\} \times \{0..23\} \to \mathbb{R}^+$$

is given by

$$\xi(e, d, h) = \begin{cases} \lambda(e) & \text{if } u \notin S \\ \lambda(e) + f_u(d, h) \cdot g_u(d, h) \cdot \mu(e) & \text{if } u \in S \end{cases}$$

i.e., ξ corresponds to the labeling λ to which a *virtual* distance equivalent to the expected waiting time at s, if arriving at time (d, h), is added.

Remark. Equivalently, instead of adding a value proportional to the waiting time on every out-edges, we could have added a weight on the nodes and modified A* consequently. We chose to modify edges instead of adding node weight because it seems more frequent in the literature. Moreover, time-dependent edges formalism can be used for many other path planning related consideration, like traffic or weather.

Since we now use a time-dependent labeling, and the time (d, h) when arriving at a station s is unknown before starting the search algorithm, the base planner (Algorithm 1) must be modified by adapting the call to the A* algorithm on line 7. A survey of possible modifications to well-known path planning techniques allowing the consideration of time-dependent edges is presented in [3]. We use A* in the same way that Dijkstra's algorithm was adapted in this survey. In summary, when a node $s \in S$ of the s-graph is extracted out of the priority queue, the shortest path to s has been found, so the time when arriving at s is known (the initial time of departure plus the time of travel from α to s, including, when applicable, the waiting and the charging time). Hence, the labeling $\xi(e, d, h)$ of every out-edge $e = (s, v)$ of s can be determined and A* can add every out-neighbor of s to the priority queue and continue as it normally does.

Remark. The aforementioned modification to the planner only adds a constant number of operations to every iteration of the A* algorithm. Consequently, the running time of our planner remains $O(n \log n)$.

5 ALTERNATIVE PATHS GENERATION

As we will see in Section 6, the modification to the planner made in the previous section decreases a lot the average waiting time. However, it is only based on *a priori* probability of occupancy and *a priori* waiting time. If, by misfortune, the real-time probability of occupancy is worse than what was assumed, it would be advantageous to have an online planner which takes advantage of this information. In this section, we propose to consider this by precomputing, for every station in the Q sequence, an alternative path passing by alternative charging station(s) which could be chosen by the planner at runtime if our first charging station choice is unavailable when we arrive at a branching node. The branching node is chosen to be the last common node between the base path and the new alternative path.

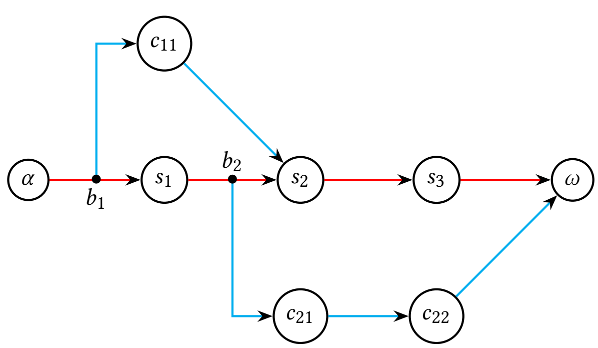

Figure 2: Alternative path generation

Graphically, we can illustrate the situation we are looking for as in Figure 2. In this figure, the middle path $Q = (\alpha, s_1, s_2, s_3, \omega)$ is the path obtained by the method in Section 4, while the upper path (respectively the lower path) is an alternative path that could be taken if s_1 (respectively s_2) is occupied at runtime, even though the *a priori* probability of occupancy would be low. The node c_{ij} is the j^{th} charging station on the i^{th} alternative path (the path that will be used if the station s_i had worse than expected occupancy) and the node b_i is the branching node for the i^{th} alternative path.

Algorithm 2 Alternative path generation for station s_i

Assume $f_{s_i} \equiv 1$
Run the relabeled time-dependent planner
if new path is same as base path **then**
  return
end if
$b_i \leftarrow$ last common node in prefix of new path and base path
Set the new path as an alternative path on node b_i

To find such paths, we run Algorithm 2 on every station $s \in Q$. As a result, we obtain a mapping $\pi : V \rightarrow V^2$ such that

$$\pi(x) = \begin{cases} (s_{i+1}, -) & \text{if } x = s_i \wedge \nexists b_{i+1} \\ (b_i, -) & \text{if } x = s_i \wedge \exists b_{i+1} \\ (s_i, c_{i1}) & \text{if } x = b_i \\ (c_{i,j+1}, -) & \text{if } x = c_{ij}, \end{cases}$$

assuming that $\forall i, \exists k \exists p$ such that $c_{ik} = s_p$ (i.e., the last node of an alternative path is a node on the base path, possibly the arrival node ω).

Remark. Not all stations s will necessarily lead to the generation of an alternative path because sometimes waiting is more efficient than going elsewhere.

During runtime, a query to a server containing the real-time data of occupancy (rto) for every station could be sent when arriving at a branching node between the base path and an alternative path. If the station on the base path is occupied, the online planner could send the EV through the alternative path. This online plan execution strategy is presented in Algorithm 3.

Algorithm 3 Online plan execution

procedure EXECUTEPLAN(π)
  $n \leftarrow \alpha$
  while $n \neq \omega$ **do**
    $(x, y) \leftarrow \pi(n)$
    if $y = - \vee \neg \text{rto}(x)$ **then**
      $n \leftarrow x$
    else
      $n \leftarrow y$
    end if
    Move EV to node n
  end while
end procedure

Remark. We assume that the occupancy of every station (occupied or not) can be known at runtime, but not the actual waiting time (i.e., if a waiting queue of 3 EVs is at a station, this information cannot be known by the planner). Such an assumption is realistic based on the current lack of available public data of the EV charging infrastructure.

This technique is simple, has a negligible pre-calculation overhead (it increases the time of computation by less than 1%) and reduces significantly the mean total journey time when compared to relabeling without alternative paths (Section 4). While it may seem at first that we could have simply recomputed the path at every node during the trip using the real-time occupancy, while having similar results, this would cause serious disadvantages. In fact, the number of calls to the server containing the data of occupancy of the charging stations network would be $O(|P|)$, while our approach needs to call the server only $O(|Q|)$ times, thus giving similar results to systematic replanning (because of the placement of the branching node), while inducing a much lower resource usage on the server side.

Remark. We could also use the proposed techniques to consider the actual travel time (e.g., by considering real-time traffic information) in addition to (or instead of) the actual waiting time. Both of our techniques would work almost the same way with dynamic travel time (including expected time based on historic data, like in our first method, and real-time travel time information, like in our second method). However, many papers already considers dynamic traveling time, and it was not the main focus of our paper (we wanted to focus strictly on waiting time, since it is specific to EV).

6 EVALUATION

Like it was mentionned in Section 2, the most related work is the one of Sweda et al. [18]. However, this study does not provide an evaluation of the proposed method using real-world data. It also has a different formulation of the problem, with different assumptions, thus making it hard to make an honest comparison against our technique (e.g., it considers that every node of the graph is a charging station, including the departure and arrival nodes). Also, this method generates a policy that indicates the best thing to do *when arriving* at a charging station (charge there, go elsewhere, etc.). Our alternative path technique has the advantage of considering real-time charging stations occupancy *prior to arriving* at a station, thus allowing additional time saving in many situations. For all these reasons, we don't directly compare our techniques to that of Sweda et al. [18].

For the evaluation of the proposed method, the baseline (Algorithm 1, without the ξ labeling) was compared to the two proposed techniques (graph relabeling and alternative path generation). The map data (i.e., the nodes and the road segments) were taken from the OpenStreetMap project [20]. The territory of the Québec Province (Canada) was chosen to carry out our tests because it is vast, because the journeys between certain pairs of cities can be very long (thus requiring a large number of recharges) and because the network of charging stations is relatively well developed. The graph generated from these data had 2 923 013 vertices and 5 907 672 edges.

The charging stations considered in the tests were real stations from the Québec Province's public network of EV charging stations (called *Circuit Électrique*). The dataset contained 1318 charging stations (level 2 and 3). All stations in the dataset were considered as if they were level 3 stations because the EV planners are primarily used for long itineraries, where fast charging is a must. Furthermore, there was not enough level 3 charging stations currently available in our data to test the proposed algorithm adequately using only them (only 140 level 3 charging stations were present in our dataset).

To compare different placements and densities of stations on the map, artificial charging stations were also generated. They were uniformly distributed among the nodes of the map (i.e., since big cities have more nodes than rural village, more stations were generated there). Overall, the tests were run with, respectively, the 140 true level 3 stations, the 1318 true stations in the network, and the artificial stations (250, 500, 1000, 2000 stations).

The f_s and g_s functions were constructed for every real station (the data for the occupancy and waiting time at every day and hour) by retrieving data on the public network's website every 5 minutes from December 2017 to June 2019. Since many stations have been for now underused, a multiplicative factor on the f_s was used in the tests (1 for the true value or $2, 3$ and 5 to simulate a bigger probability of occupancy). For the artificial stations, the values $f_s(d, h)$ were generated uniformly between 0 and 1 and the values $g_s(d, h)$ were generated using a Kumaraswamy distribution [11] (with the parameters 2 and 100, scaled between 1 minute and 6 hours) which gives a mean of about 31.6 minutes. This distribution was chosen because in real life, an average EV driver stays around 30 min at a charging station but some people leave their car plugged in for way longer, so a positive skewed distribution makes more sense than a symmetric one.

The 1000 generated requests were composed of the departure point α, the terminal point ω (both chosen at random from all the nodes in the graph), the EV range ρ (generated uniformly between 90 km and 550 km, based on the minimal and maximal EV range currently available on the market) and the day and initial hour of departure. The requests' optimal solution length had a high variability, ranging from around 200 to 1500 km. All of the generated requests necessitated at least one stop at a charging station for the EV to be able to reach the destination (some of them necessitating even 4 stops).

For the sake of simplicity, it was assumed in the tests that $\mu(e) =$ 90 km/h $\forall e \in E$. It was also assumed that $\mathrm{ECT}(s) = 30$ min $\forall s \in S$. In a consumer-ready planner, the level of the charging station and the state of charge of the vehicle need to be taken into consideration, but it was not relevant for this study (it can be easily added to our model if needed).

Table 1 reports the results obtained by running the tests described above. The first two columns present our main parameters: the station network data SN (where R stands for real, and A stands for artificial stations), and the probability modifier PM (where Rand means that the probabilities were generated uniformly). The next two columns, respectively, present the average waiting time (Wait) and the total time (W+D) of the baseline. Finally, the two four-column blocs present, respectively, the results for the relabeling and the alternative path generation techniques. The Wait and W+R columns have the same meaning than previously, and the WR and TTR columns compare the technique with the baseline by showing the difference in the waiting time (in percent) and the difference in the average total (wait + drive) time (in minutes). Every value in the table was rounded to one decimal.

When considering the column SN together with the two WR and TTR columns, we observe that the increase of stations on the map (R140 to R1318, or A250 to A500 to A1000 to A2000) leads

Table 1: Results for 1000 journeys: considering occupation

Parameters		Baseline		Relabeling				Alternative Paths			
SN	PM	Wait min	W+D min	Wait min	W+D min	WR %	TTR min	Wait min	W+D min	WR %	TTR min
R140	× 1	10.2	350.1	3.3	343.9	-67.5	-6.2	2.2	342.8	-78.8	-7.2
R140	× 2	22.7	362.5	6.0	347.0	-73.6	-15.5	4.1	345.3	-82.0	-17.2
R140	× 3	37.5	377.3	7.7	349.0	-79.3	-28.3	7.2	348.5	-80.9	-28.7
R140	Rand	51.3	391.2	10.9	352.6	-78.7	-38.5	9.8	351.6	-80.9	-39.5
R1318	× 1	19.4	356.0	2.2	339.2	-88.6	-16.7	1.7	338.7	-91.5	-17.3
R1318	× 2	37.5	374.0	4.0	341.2	-89.3	-32.8	3.0	340.3	-92.0	-33.7
R1318	× 3	50.5	387.1	6.5	343.9	-87.1	-43.1	4.7	342.3	-90.6	-44.8
R1318	Rand	62.0	398.6	6.9	345.3	-88.8	-53.3	6.0	344.4	-90.4	-54.2
A250	Rand	29.3	368.2	12.0	354.2	-59.0	-13.9	10.3	353.1	-64.8	-15.1
A500	Rand	28.9	363.4	9.4	346.9	-67.6	-16.5	8.3	346.1	-71.2	-17.3
A1000	Rand	28.7	362.5	7.4	343.8	-74.2	-18.7	6.5	343.1	-77.3	-19.4
A2000	Rand	27.1	359.9	4.9	340.0	-81.9	-19.9	3.8	339.0	-86.0	-20.9

SN: Station network (where R = real and A = artificial); **PM**: Probability modifier; **Wait**: Average waiting time **W+D**: Average total time (waiting + driving time) **WR**: Waiting time reduction (vs Baseline) **TTR**: Total time reduction (vs Baseline)

to a bigger decrease in both the waiting time and the total time (for the two proposed techniques). For example, when the number of charging stations increased from 140 to 1318, the waiting time decreased from 67.5% to 88.6% for the relabeling technique, and from 78.8% to 91.5% for the replanning technique. The same trend can be observed when comparing the four sizes of artificial charging stations network (last four rows in the table).

The augmentation of the probability of charging stations occupancy (by increasing the probability modifier PM) obviously increased the average charging time, hence allowing a bigger reduction of the total journey time (as can be seen by looking at the values of the two TTR columns when comparing rows with different PM values).

Figures 3 and 4 shows box and whisker plots of the three techniques (baseline, relabeling, alternative paths generation) respectively for the R140 × 1, R1318 × 1 and A250 datasets on the former, and the R140 × R, R1318 × R and A2000 datasets on the latter. They show the minimum, first quartile, median, third quartile and maximum of the total journey time (in seconds). While the alternative paths technique doesn't allow a significant median time reduction, it decreased significantly the maximum time on all datasets.

Overall, the waiting time reduction, together with the negligible increase of the driving time, allowed a significant decrease of the total EV trip time. These results apply not only to our specific real charging stations data, but also to artificially generated data, including stations, probabilities of occupancy and waiting times, suggesting that the observed trends are characteristic for many EV stations disposition, density and occupancy scenarios.

7 CONCLUSION

In this paper, we proposed to extend current EV path-planning techniques to consider the waiting time at charging stations in the total cost function of an EV trip. We did so in two different ways. The first technique is a dynamic time-dependent graph relabeling that adds an artificial cost to every out-edge of every charging station to let the planner account for the average waiting and charging time at the station at a specific moment (hour and day). The second technique is a contingency generation technique that precomputes alternative paths in case a station where we planned to recharge is occupied prior to arriving to it, despite the historic data. The complete planner that includes the two described techniques is an efficient $O(n \log n)$ planner.

Our results show that the first technique gave impressive results on average for the 1000 EV trips used in our tests, dropping by more than 3/4 the waiting time while having a negligible increase in the driving time. Overall, the mean total trip time decreased by 17.3 minutes when using the real data of stations placement and occupancy. The second technique was tested on top of the first, and further amplified the results (though to a lesser extent). It helped to reduce the waiting time of some requests which were not effectively processed by the first technique.

The waiting time reduction provided by our techniques was generally proportional to the density of charging stations on the map (the more stations were on the map, the more the waiting time was reduced) and to the probability of the station occupancy. This indicates that the effectiveness of our method will increase with the increase in the number of charging stations on the map (currently, new charging stations are installed faster and faster [17]) and the

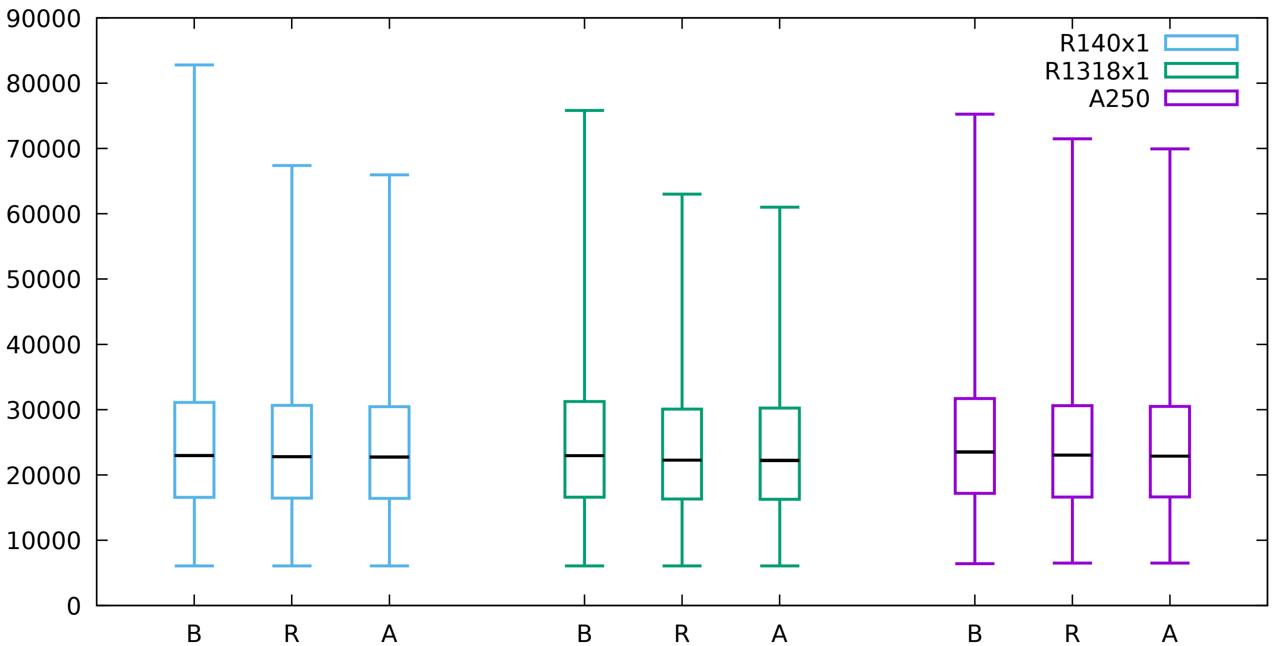

Figure 3: Box plots showing the five number summary of the total time
B: Baseline; **R**: Relabeling; **A**: Alternative paths.

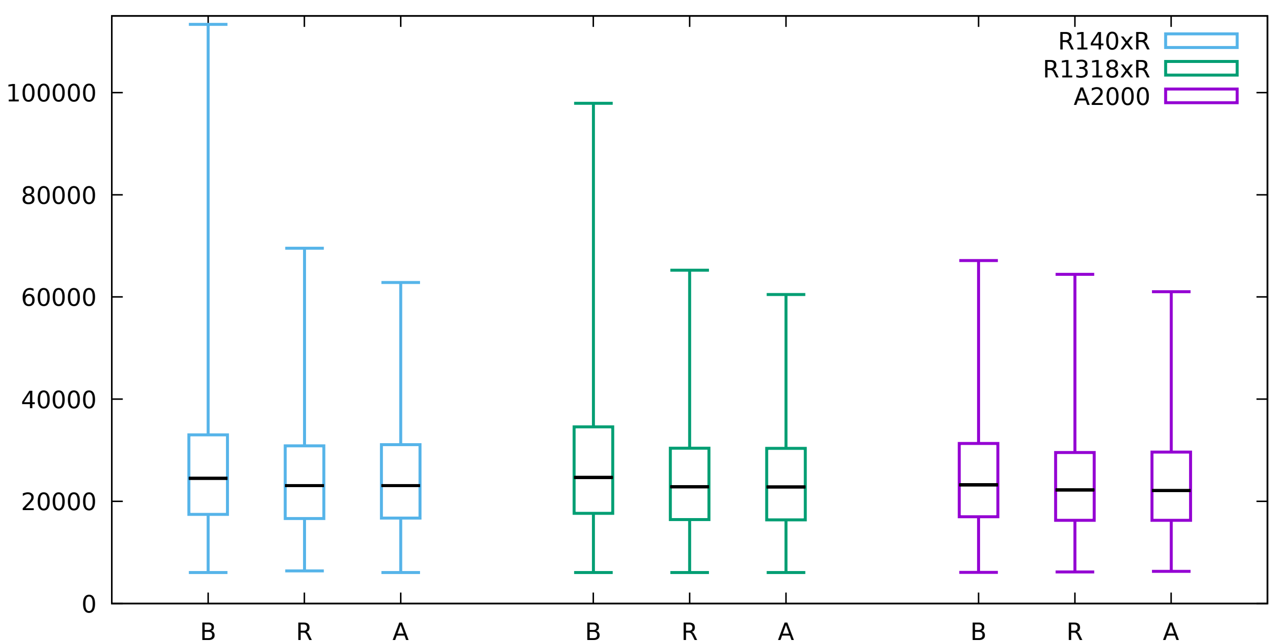

Figure 4: Box plots showing the five number summary of the total time
B: Baseline; **R**: Relabeling; **A**: Alternative paths.

increase of the EV on the road (the number of EV also increases worldwide [15]).

While the graph relabeling and alternative path generation models are important, a good probabilistic model to estimate the global demand is also needed. Currently, we have access to real-time occupancy data for charging stations which are part of our station network. The presence/absence of waiting queues, allowing an estimation of the waiting time at these stations, is not available so far. However, when it becomes available, this information can be easily incorporated into our techniques.

As future work, we plan to develop an EV charging stations reservation system together with an online EV planner that will consider the EV stations reservation by the other EV drivers when determining the optimal path.

ACKNOWLEDGMENTS

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC). We also acknowledge the support of the Fonds de Recherche du Québec — Nature et Technologies (FRQNT)

REFERENCES

[1] Fouad Baouche, Romain Billot, Rochdi Trigui, and Nour Eddin El Faouzi. 2014. Electric Vehicle green routing with possible en-route recharging. In *International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2787–2792. https://doi.org/10.1109/ITSC.2014.6958136

[2] Moritz Baum, Jonas Sauer, Dorothea Wagner, and Tobias Zündorf. 2017. Consumption Profiles in Route Planning for Electric Vehicles: Theory and Applications *. In *International Symposium on Experimental Algorithms (SEA)*, Vol. 75. Dagstuhl Publishing, 19:1–19:18. https://doi.org/10.4230/LIPIcs.SEA.2017.19

[3] Daniel Delling and Dorothea Wagner. 2009. Time-dependent route planning. In *Lecture Notes in Computer Science*, Vol. 5868 LNCS. 207–230. https://doi.org/10.1007/978-3-642-05465-5_8

[4] Jose Rizalino M. Delos Reyes, Robert V. Parsons, and Ray Hoemsen. 2016. Winter Happens: The Effect of Ambient Temperature on the Travel Range of Electric Vehicles. *IEEE Transactions on Vehicular Technology* 65, 6 (2016), 4016–4022. https://doi.org/10.1109/TVT.2016.2544178

[5] E. W. Dijkstra. 1959. A note on two problems in connexion with graphs. *Numer. Math.* 1, 1 (1959), 269–271. https://doi.org/10.1007/BF01386390

[6] Brian Eckhouse, David Stringer, and Jeremy Hodges. 2019. The World Still Doesn't Have Enough Places to Plug In Cars - Bloomberg. https://www.bloomberg.com/news/features/2019-02-14/the-world-still-doesn-t-have-enough-places-to-plug-in-cars

[7] Jochen Eisner, Stefan Funke, and Sabine Storandt. 2011. Optimal Route Planning for Electric Vehicles in Large Networks.. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*. AAAI Press, 1108–1113. https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/viewFile/3637/4005

[8] Michael L Fredman and Robert Endre Tarjan. 1987. Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM* 34, 3 (1987), 596–615. https://doi.org/10.1145/28869.28874

[9] Robert Geisberger, Peter Sanders, Dominik Schultes, and Christian Vetter. 2012. Exact Routing in Large Road Networks Using Contraction Hierarchies. *Transportation Science* 46, 3 (2012), 388–404. https://doi.org/10.1287/trsc.1110.0401

[10] Peter Hart, Nils Nilsson, and Raphael Bertram. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans - TSMCA* , 4, 2 (1968), 100–107. https://doi.org/10.1109/tssc.1968.300136

[11] M. C. Jones. 2009. Kumaraswamy's distribution: A beta-type distribution with some tractability advantages. *Statistical Methodology* 6, 1 (2009), 70–81. https://doi.org/10.1016/j.stamet.2008.04.001

[12] Nicholas Kullman, Justin Goodson, Jorge E Mendoza, Nicholas Kullman, Justin Goodson, Jorge E Mendoza, Electric Vehicle, Justin C Goodson, and Jorge E Mendoza. 2017. *Electric vehicle routing with uncertain charging station availability dynamic decision making.* Technical Report. https://hal.archives-ouvertes.fr/hal-01497793

[13] Juuso Lindgren and Peter D Lund. 2016. Effect of extreme temperatures on battery charging and performance of electric vehicles. *Journal of Power Sources* 328 (2016), 37–45. https://doi.org/10.1016/j.jpowsour.2016.07.038

[14] Kai Liu, Toshiyuki Yamamoto, and Takayuki Morikawa. 2017. Impact of road gradient on energy consumption of electric vehicles. *Transportation Research Part D: Transport and Environment* 54 (2017), 74–81. https://doi.org/10.1016/j.trd.2017.05.005

[15] Brad Plumer. 2016. *The rapid growth of electric cars worldwide.* Retrieved May 1, 2019 from https://www.vox.com/2016/6/6/11867894/electric-cars-global-sales

[16] Martin Sachenbacher, Martin Leucker, Andreas Artmeier, and Julian Haselmayr. 2011. Efficient Energy-Optimal Routing for Electric Vehicles. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*. AAAI Press, 1402–1407. https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/viewFile/3735/4092

[17] Richard Stubbe. 2018. *Charging Stations Keep Pace With Electric-Vehicle Growth - Bloomberg.* Retrieved May 1, 2019 from https://www.bloomberg.com/news/articles/2018-05-30/charging-stations-keep-pace-with-electric-vehicle-growth

[18] Timothy M Sweda, Irina S Dolinskaya, and Diego Klabjan. 2017. Adaptive Routing and Recharging Policies for Electric Vehicles. *Transportation Science* 51, 4 (2017), 1326–1348. https://doi.org/10.1287/trsc.2016.0724

[19] Timothy M Sweda and Diego Klabjan. 2012. Finding minimum-cost paths for electric vehicles. In *IEEE International Electric Vehicle Conference (IEVC)*. IEEE, 1–4. https://doi.org/10.1109/IEVC.2012.6183286

[20] Patrick Weber and Muki Haklay. 2008. OpenStreetMap: user-generated street maps. *IEEE Pervasive Computing* 7, 4 (2008), 12–18. www.openstreetmap.

[21] Xinkai Wu, Assistant Professor, Alfredo Cabrera, David Freese, William A Kitch, and Xudong Jia. 2013. Electric Vehicles' Energy Consumption Measurement and Estimation. *Transportation Research* (2013), 1–17. https://pdfs.semanticscholar.org/7ca7/ea1189c40879614e8914074d4b3035aaf46a.pdf