Electric Vehicles En-route Charging Navigation Systems: Joint Charging and Routing Optimization

Liu, C.; Zhou, M.; Wu, J.; Long, C.; Wang, Y.

TR2017-126 November 2017

Abstract

Widely recognized as an excellent solution of global warming and oil crisis, electric vehicles (EVs) however suffer remarkable weakness such as the limited cruise range, which can be partly addressed by introducing en-route charging navigation systems. Different from traditional navigation, which solves a shortest path problem, the en-route charging navigation resorts to a joint charging and routing optimization. In this paper, we formulate the en-route charging navigation in a dynamic programming (DP) setting in both a deterministic and a stochastic traffic network. Specifically, to relieve computational complexity in navigation systems, a simplified charge-control (SCC) algorithm is presented in the deterministic case, which can simplify the charging control decisions within an SCC set. In the stochastic case, an online state recursion (OSR) algorithm is designed, which can provide an accurate navigation utilizing online information. Numerical simulation verifies the computing burden and accuracy of the proposed algorithms in a deterministic and a stochastic networks.

IEEE Transactions on Control Systems Technology

Electric Vehicles En-route Charging Navigation Systems: Joint Charging and Routing Optimization

Chensheng Liu, *Student Member, IEEE,* Min Zhou, *Student Member, IEEE* Jing Wu, *Member, IEEE,* Chengnian Long, *Member, IEEE* Yebin Wang, *Member, IEEE,*

***Abstract*—Widely recognized as an excellent solution of global warming and oil crisis, electric vehicles (EVs) however suffer remarkable weakness such as the limited cruise range, which can be partly addressed by introducing en-route charging navigation systems. Different from traditional navigation, which solves a shortest path problem, the en-route charging navigation resorts to a joint charging and routing optimization. In this paper, we formulate the en-route charging navigation in a dynamic programming (DP) setting in both a deterministic and a stochastic traffic network. Specifically, to relieve computational complexity in navigation systems, a simplified charge-control (SCC) algorithm is presented in the deterministic case, which can simplify the charging control decisions within an SCC set. In the stochastic case, an online state recursion (OSR) algorithm is designed, which can provide an accurate navigation utilizing online information. Numerical simulation verifies the computing burden and accuracy of the proposed algorithms in a deterministic and a stochastic networks.**

***Index Terms*—Electric Vehicle Navigation; Joint Charging and Routing Optimization; Deterministic Traffic Network; Stochastic Time-Dependent Network; Online Navigation.**

I. Introduction

ELECTRIC vehicles (EVs) have been drawing much attention in recent years due to their efficiency and environmental friendliness. However, the limited cruise range, caused by the battery capacity, makes drivers in "range anxiety" especially on a long distance trip, and it makes EVs en-route charging navigation systems crucial. Since real-time pricing is widely used in electricity market to shift load and stabilize power systems, en-route charging navigation system under real-time pricing is different from the traditional navigation system, where the path is both weighted by the time-dependent electricity prices and limited by the time-dependent traffic properties. This means that the routing cost is determined by not only the routing decisions but also the charging control decisions. As a result, the charging navigation becomes a joint charging and routing optimization rather than a common shortest path routing.

In literature, EV charging and routing problems have been studied, separately. For example, demanded load model formulation [1] and impact analysis [2] were discussed in uncoordinated charging field. In order to solve the problems caused by EVs' penetration, coordinated charging control was presented in both deregulated and regulated electricity markets to minimize power losses, generation cost, voltage deviations, load variance, and charging cost [3], [4], [5], [6], [7]. However, the charging control strategies presented above were designed in a centralized form, which are not suitable for power systems with vast EVs. Decentralized algorithms and protocols were designed with the goal of shifting load, regulating frequency, optimizing EV charging set, and minimizing generation cost [8], [9], [10], [11]. Even though many excellent work has been done in charging control field, they mainly focus on coordinating EVs charging at home or parking plots, and ignore the coupling between routing and en-route charging.

In EVs routing field, authors in [12] studied an energy-optimal routing problem with constraints of recuperation and battery capacity. However, the routing problem is defined within one-charge distance rather than en-route charging. For long distance trips, optimal routing problems were considered, where optimal objective might be charging delays [13], energy-efficient route [14], [15], or a combination objective of travel time, charging time, and energy consumption [16]. The aforementioned work on en-route charging navigation however does not consider the charging control process, which implicity assumes constant electricity price.

The joint charging and routing problem in deterministic networks, is formulated and solved by dynamic programming (DP) in our previous work [17], where the time-dependent travel cost under real-time pricing is a much more appropriate formulation compared with the shortest path [18] and eco-routing [19] formulation. Even though similar problems in deterministic networks are solved in [20] and [21] using mixed integer non-linear programming (MINLP) and DP, respectively, we optimize the travel cost in time-dependent networks, e.g., real-time electricity prices. Moreover, we propose a simplified algorithm in [17] to relieve the computing burden in navigation under real-time pricing. As there are randomness and uncertainty in traffic networks, the deviation between the simplified deterministic model and the actual traffic network may lead to over-discharging of batteries or even driving out of power halfway. In order to improve the accuracy of navigation, we extend our previous deterministic charging navigation to an online navigation system based on stochastic traffic network models and online information.

In this paper, we present an en-route charging navigation problem under real-time pricing in a deterministic and a stochastic traffic network. Specifically,

- The en-route charging navigation problem under real-time

The material in this paper was partially presented at the 19th IFAC World Congress, August 24∼29, 2014, Cape Town, South Africa.

Chensheng Liu, Jing Wu, Chengnian Long (corresponding author) are with the Department of Automation, Shanghai Jiao Tong University, Shanghai, 200240 China. Email:longcn@sjtu.edu.cn

Yebin Wang is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA. E-mail: yebinwang@merl.com

pricing is first formulated in a dynamic programming setting, where the charging and routing decisions are affected by the real-time electricity prices.
- A simplified charge-control (SCC) algorithm is designed based on the characteristic in the SCC set, which can relieve the computing burden in the deterministic case.
- An online state recursion (OSC) algorithm is designed based on the stochastic time-dependent traffic model and online traffic information, which can improve the accuracy of the navigation.

The following sections are organized as follows: detailed system models are presented in Section II. The joint charging and routing optimization in a deterministic traffic network is presented in Section III. The joint charging and routing optimization in a stochastic traffic network is discussed in Section IV. Simulation and conclusion are given in Section V and VI, respectively.

II. System Model and Notation

A. Notation

In the rest of the paper, t denotes time, and τ denotes a period of time. Variables with a subscript i or j refer to variables of EVs at the ith or the jth station, while variables with a subscript ij refer to variables of the link (i, j). For example, e_i is the EV's arrival energy at the ith station, while e_{ij} is the energy consumption of EVs on link (i, j). Moreover, the variables at a station without a superscript "$'$" denote arrival variables, while the variables at a station with a superscript "$'$" denote departure variables, e.g., t_i is the arrival time, t'_i is the departure time. In the deterministic case, we denote τ_{ij} and e_{ij} as the link travel time and link energy consumption, while we use $\tau_{ij}(t)$ and $e_{ij}(t)$ in the stochastic case. Note that in the following sections, we denote $\pi_{ij}(t)$ and $\xi_{ij}(t)$ as a special value of variables $\tau_{ij}(t)$ and $e_{ij}(t)$, respectively. For ease of reading, a brief summarization of notation and definition is given in Table I.

B. System Model

In this part, a charging and routing cost model is presented in a deterministic traffic network.

1) Traffic Networks: As EVs can not arrive at the destination without recharging at long distance trips, we assume that a network is defined as a directed graph $G = (\mathcal{N}, \mathcal{A})$ with $\mathcal{N} = \{1, \ldots, n\}$, where node $i \in \mathcal{N} \backslash \{1, n\}$ represents a fast charging station. Node 1 and n are the origin and destination of a long distance trip. $(i, j) \in \mathcal{A}$ is the road connecting node i and j. Node 1 may be a fast charging station or a place equipped with slow chargers. As it takes a much longer charging time to recharge an EV using a slow charger, usually 6 $\sim$ 8h, we don't consider the charging process at node 1 if it is not a fast charging station. Denote d_{ij} and τ_{ij} as the distance and travel time of link (i, j). Let $O(i) = \{j \in \mathcal{N} | (i, j) \in \mathcal{A}\}$ be the successor stations of the ith station. Denote $T = \{t_1, \cdots, t_{end}\}$ as the travel time limit, where t_{end} is preset, based on the traffic network, to avoid unpractical solution in navigation.

TABLE I
The Summarization of Notation

Notation	Definition
$\mathcal{N}$	Node set in traffic network.
$\mathcal{A}$	Arc set in traffic network.
T	Travel time limit.
E	Possible energy in a battery.
$Ex[\cdot]$	Expected value operator.
d_{ij}	Distance of link (i, j).
$\tau_{ij}(t)$	Link travel time (variable) at time t.
$\pi_{ij}(t)$	A special value of $\tau_{ij}(t)$.
$e_{ij}(t)$	Link energy consumption (variable) at time t.
$\xi_{ij}(t)$	A special value of $e_{ij}(t)$.
t_i/t'_i	Arrival/departure time at the ith station.
e_i/e'_i	Arrival/departure energy at the ith station.
Δe_i	Charged energy within a unit time at the ith station.
τ_i^w	Waiting time at the ith station.
τ_i^c	Time spent for charging at the ith station.
τ_i	Elapsed time at the ith station, including waiting time.
s_i	State when a EV arrives at the ith station.
u_i	Routing and charging decisions at the ith station.
m_r	A link travel time mode.
$M(t)$	The identified (link travel time) mode set at time t.
$\mathcal{M}(t)$	The possible mode set at time t.
$p(t)$	Real-time electricity prices.
$P(\cdot)$	Probability of a traffic mode in stochastic traffic networks.
$D(\cdot)$	Charging control decision set in the ICS algorithm.
$\mathcal{D}(\cdot)$	Charging control decision set in the OSR algorithm.
$C(s_i)$	Minimal cost from the state s_1 to the current state s_i.
$\mathcal{C}(\cdot)$	Minimal cost from the current state $(\cdot)$ to the destination.

2) Link Energy Consumption: Link energy consumption expresses the energy consumed at links, which is a constraint of the charging control decisions. Based on the power-speed relationship of EVs, the link energy consumption e_{ij} can be expressed as follows:

$$e_{ij} = \tau_{ij} \cdot f(d_{ij}/\tau_{ij}), \tag{1}$$

where d_{ij}/τ_{ij} is the average speed of link (i, j), f is the EV's power-speed function.

3) Rapid Charging: Let $E = [e_{min}, \cdots, e_{stop}, \cdots, e_{rate}]$ be the possible energy in the battery, where e_{min} denotes the stop discharging energy to protect the battery cycle life, e_{rate} denotes the rated energy of battery. In order not to damage the battery, the rapid charging process stops when the energy in battery rises to e_{stop}. For example, CHAdeMo chargers are usually configured to stop automatically once the battery is 80% full [22]. Before reaching e_{stop}, the rapid charger keeps the charging current constant [23], i.e., the charging curve is approximately linear. We denote Δe_i as the energy charged in unit time at the ith station. Let the waiting time at the ith station be τ_i^w, which is obtained by a waiting-time predictor[1] [24]. For a EV starting to charge with the arrival energy e_i at time t_i, and departing with the energy e'_i, the elapsed time at the ith station can be expressed as

$$\tau_i = \tau_i^w + \tau_i^c = \tau_i^w + (e'_i - e_i)/\Delta e_i, \tag{2}$$

where τ_i^c is the time spent for charging at the ith station.

4) Routing and Charging Cost at Stations: Let the (discrete) real-time prices, obtained by a price predictor, be $p(t)$, $t \in T$. Suppose a EV arrives at the ith station with a energy

[1] Waiting time at charging stations can be predicted using electric vehicle's online information, e.g., online intension information [24].

e_i at time t_i, and departs to the jth station with energy e_i'. Denote $s_i = (i, e_i, t_i)$ and $u_i = (e_i', j)$ as the arrival state and decisions, respectively, then the routing and charging cost at the ith station is

$$c(s_i, u_i) = \sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t) \cdot \Delta e_i + \alpha \cdot (\tau_i + \tau_{ij}), \tag{3}$$

where the first part is the charging cost, and the second part is the elapsed time. $\alpha \geq 0$, is used to coordinate the charging cost and the elapsed time. α should be large (small) enough when minimize the elapsed time (charging cost) in priority. α can be preset for different navigation modes. Because the successor state s_j can be deduced by s_i and u_i as shown in (5), the charging cost $c(s_i, u_i)$ can be expressed as $c(s_i, s_j)$, equivalently.

III. Deterministic Navigation Optimization

A. Dynamic Programming Formulation

1) State Variables: We define the state variable at the ith station as $s_i = (i, e_i, t_i)$, where i is the current station, e_i is arrival energy, i.e., the energy left in battery when arriving at the ith station, t_i is the arrival time. Assume a EV departs form node 1 with a energy e_1 at time t_1, then the starting state is $s_1 = (1, e_1, t_1)$. Similarly, the state when a EV arrives at the destination, node n, the state can be expressed as $s_n = (n, e_n, t_n)$, where $t_n \leq t_{end}$. As we minimize the charging cost and elapsed time in the navigation, it is intuitively obvious that $e_n = e_{min}$.

2) Decision Variables: Assume that charging control and routing decisions are made when a EV arrives at a station. Based on the current state at the ith station, $s_i = (i, e_i, t_i)$, we define the decision variable as $u_i = (e_i', j)$, where $e_i' \in E$ is the departure energy, $j \in O(i)$ is the successor charging station, i.e., the routing decision is j. Given the current state $s_i = (i, e_i, t_i)$ and a routing decision $j \in O(i)$, the charging control decision set can be expressed as

$$D(s_i, j) = \{e_i' \in E | \max\{e_i, e_{min} + e_{ij}\} \leq e_i' \leq \max\{e_i, e_{stop}\}\}. \tag{4}$$

If node 1 is not a charging station, the decision variable based on the state $s_1 = (1, e_1, t_1)$ is $u_1 = (e_1, j)$ with $j \in O(1)$. As $O(n) = \emptyset$, the decision set is empty at node n.

Based on the current state $s_i = (i, e_i, t_i)$ and decision $u_i = (e_i', j)$, the state at the jth station, $s_j = (j, e_j, t_j)$, can be calculated

$$\begin{aligned} e_j &= e_i' - e_{ij} \\ t_j &= t_i + \tau_i + \tau_{ij}, \end{aligned} \tag{5}$$

where τ_i is the elapsed time at the ith station.

3) Recursive Value Equation: Define the value function of the current state, $C(s_i)$, as the minimal summation of the charging cost and the elapsed time from the starting state s_1 to the current state s_i, and $C(s_1) = 0$ (boundary condition). Then the value function of the successor state s_j, determined by the decision $u_i = (e_i', j)$, can be expressed as below

$$C(s_j) = \min_{j \in O(i), e_i' \in D(s_i, j)} \{C(s_i) + c(s_i, u_i)\}, \tag{6}$$

where $c(s_i, u_i)$ is given in (3). In this paper, our objective is to find an optimal solution sequence of $C(s_n)$.

B. Deterministic Navigation Algorithms

In this part, an improved Chrono-SPT[2] (ICS) algorithm is designed in a deterministic traffic network. In order to relieve the computational complexity, an SCC algorithm is designed further.

1) Improved Chrono-SPT (ICS) Algorithm: Borrowing the conception of chronological searching (a variant of Dijkstra's algorithm) in time-dependent networks [25], we design an ICS algorithm, which searches the optimal charging and routing decision sequences, chronologically.

In the ICS algorithm, we denote $s_i = (i, e_i, t_i)$, $u_i = (e_i', j)$, and $s_j = (j, e_j, t_j)$ as the current state, current decision , and successor state, respectively. Let $C(s_i)$ be the minimal cost from the starting state $s_1 = (1, e_1, t_1)$ to the state s_i, and denote $\psi(s_i) = u_i$ as the optimal decisions based on the current state s_i, which will be used in constructing optimal decision sequences. We use a bucket-list $B = \{B_t | t \in T\}$ to efficiently perform the selection operations, where B_t is the state bucket at time t. The definition is given below.

***Definition** 1:* Given a traffic network $G = (\mathcal{N}, \mathcal{A})$ and possible arrival states set $\mathcal{S} = \{(1, e_1, t_1), \cdots, (n, e_n, t_n)\}$, the state bucket at time t is $B_t = \{(i, e_i, t_i) \in \mathcal{S} | t_i = t\}$.

In initialization steps, we initialize $B_{t_1} = \{s_1\}$, $s_1 = (1, e_1, t_1)$, while the other buckets are all empty. What's more, the cost label of starting state is initialized as zero, i.e., $C(s_1) = 0$, while the others are initialized as positive infinite. The termination condition is verified when all the buckets are all empty, i.e., $B = \phi$. Note that an extended state network can be obtained at the end of the algorithm, where the optimal routing and charging decision sequences can be found. The main iteration of ICS algorithm is given in Algorithm 1.

Algorithm 1: ICS Algorithm

$*$ main iteration $*$

1: **select** $s_i = (i, e_i, t_i)$ from B_{t_i};
2: $B_{t_i} = B_{t_i} \backslash \{s_i\}$;
3: **for each** $j \in O(i)$
4: **for each** $e_i' \in D(s_i, j)$ % calculate successor state
5: $u_i = (e_i', j)$;
6: $\tau_i = \tau_i^w + (e_i' - e_i)/\Delta e_i$;
7: $t_j = t_i + \tau_i + \tau_{ij}$;
8: $e_j = e_i' - e_{ij}$;
9: $s_j = (j, e_j, t_j)$;
10: $c(s_i, u_i) = \sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t) \cdot \Delta e_i + \alpha \cdot (\tau_i + \tau_{ij})$;
11: $B_{t_j} = B_{t_j} \cup \{s_j\}$;
12: **if** $C(s_i) + c(s_i, s_j) < C(s_j)$ **then**
13: $C(s_j) = C(s_i) + c(s_i, s_j)$; % update cost
14: $\psi(s_i) = u_i$;
15: **end if**
16: **end for**
17: **end for**

In Algorithm 1, lines $3 \sim 9$ are the steps of calculating the successor states, lines $12 \sim 15$ are corresponding to the relaxation steps in Dijkstra's algorithm. Lines $3 \sim 5$ are used to search all the possible routing and charging control decisions. Once state s_i is selected, it is removed from state bucket B_{t_i} as shown in line 2, and the successor state s_j is added to the corresponding state bucket B_{t_j}.

[2]Chrono-SPT is short for chronological shortest path tree.

Compared with the Chrono-SPT algorithm [25], the ICS algorithm can search the optimal decision sequences in both time- and energy-dependent network, not just a time-dependent network. Similar to the Chrono-SPT algorithm, the convergence of the ICS algorithm follows the theorem below.

***Theorem** 1:* Given a directed acyclic graph $G = (\mathcal{N}, \mathcal{A})$ and the travel time limit T, the ICS algorithm terminates after a finite number of iterations, which is no more than $|\mathcal{A}|\cdot|E|^2\cdot|T|$.

Proof: Since the given graph $G = (\mathcal{N}, \mathcal{A})$ is a directed acyclic graph, and the feasible charging control decision set $D(s_i, j) \subseteq E$, the ICS algorithm terminates after a finite number of iterations. Suppose there is a link (i, j) in the given graph $G = (\mathcal{N}, \mathcal{A})$, then the possible arrival state at the ith station is $x = (i, e_i, t_i)$, $e_i \in E$, $t_i \in T$, i.e., the number of possible arrival state is no more than $|E|\cdot|T|$. For each possible arrival state, the charging control decisions $e'_i \in D(s_i, j) \subseteq E$, then the iterations in Algorithm 1 is no more than $|\mathcal{A}|\cdot|E|^2\cdot|T|$. ■

As the desensitization level of E and T is related to both the computational complexity of the ICS algorithm and the navigation accuracy, the desensitization level of E and T is a tradeoff between the accuracy and the computational complexity. As the time granularity of 1 min is usually acceptable in navigation, we define the desensitization level of T is 1 min, and the desensitization level of E is 1 kW·h, according to the charging rate.

The ICS algorithm enumerates all the possible charging control decisions, it has a high computational complexity. Since all the information, related to the routing and charging decisions, can be collected locally in deterministic cases, the navigation system can be deployed on traditional navigation devices by simply upgrading the software. Taking the low computing power of the embedded devices and the rapid response requirement of the navigation system, the computational complexity of the navigation algorithm must be low enough, which brings new requirements in improving the ICS algorithm above.

2) Simplified Charging Control (SCC) Algorithm: Since real-time electricity prices usually fluctuate over time, hourly [29], the charging control decisions can be simplified in a constant price interval for some cases.

Suppose that charging rates at the ith and the jth station are the same[3], and the waiting time at the jth station is 0[4], i.e., $\Delta e_i = \Delta e_j$, and $\tau_j^w = 0$. Denote $s_i = (i, e_i, t_i)$ and j as the current state and routing decision, respectively. Suppose the electricity price is constant for t, $t_1^c \le t_2^c$, i.e., the constant price time set can be expressed as $T_c = \{t \in T | t_1^c \le t \le t_2^c\}$. Then the SCC set can be defined as follows.

***Definition** 2:* Given the current state $s_i = (i, e_i, t_i)$, routing decision j, and constant price time set T_c, the SCC set is $S(s_i, j, T_c) = \{e'_i \in D(s_i, j) | t_i + \tau_i \in T_c \ \&\ t_i + \tau_i + \tau_{ij} + \tau_j^w \in T_c, \tau_i = \tau_i^w + (e'_i - e_i)/\Delta e_i, \Delta e_i = \Delta e_j, \tau_j^w = 0\}$.

Simply put, the SCC set is a charging control decisions set, which can be simplified. The SCC set can be calculated offline.

[3]As nearly all of Japan's and 3/4 of American fast chargers are CHAdeMO chargers [28], there are many stations with the same charging rate.

[4]As the penetration rate of EVs is low, waiting times may be zero especially at off-peak hours.

Note that if the SCC set exists, for different charging control decisions in the SCC set, we have the following theorem.

***Theorem** 2:* Suppose there are charging control decisions $e'_i > \tilde{e}'_i$, $e'_i, \tilde{e}'_i \in S(s_i, j, T_c)$, and the successor states, determined by e'_i and $\tilde{e}'_i$, are $s_j = (j, e_j, t_j)$ and $\tilde{s}_j = (j, \tilde{e}_j, \tilde{t}_j)$, the equation $c(s_i, s_j) = c(s_i, \tilde{s}_j) + c(s_j, \tilde{s}_j)$ holds.

Proof: Given the current state $s_i = (i, e_i, t_i)$, routing decision j, two different charging control decision $e'_i > \tilde{e}'_i$, $e'_i, \tilde{e}'_i \in S(s_i, j, T_c)$ and the corresponding arrival states at the jth station $s_j = (j, e_j, t_j)$ and $\tilde{s}_j = (j, \tilde{e}_j, \tilde{t}_j)$, we have $t_j = t_i + \tau_i + \tau_{ij}$ and $\tilde{t}_j = t_i + \tilde{\tau}_i + \tau_{ij}$, where $\tau_i = \tau_i^w + (e'_i - e_i)/\Delta e_i$, $\tilde{\tau}_i = \tau_i^w + (\tilde{e}'_i - e_i)/\Delta e_i$. Since $e'_i > \tilde{e}'_i$, we have $\tau_i > \tilde{\tau}_i$ and $t_j > \tilde{t}_j$. As described in Definition 2, the electricity price is the same for t, $t_i + \tilde{\tau}_i \le t \le t_i + \tau_i + \tau_{ij} + \tau_j^w$. As $\Delta e_i = \Delta e_j$ and $\tau_j^w = 0$, the cost from current state s_i to the successor state s_j satisfies

$$
\begin{aligned}
&c(s_i, s_j) = \textstyle\sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t)\cdot\Delta e_i + \alpha\cdot(\tau_i + \tau_{ij}) \\
&= \textstyle\sum_{t=t_i+\tau_i^w}^{t_i+\tilde{\tau}_i} p(t)\cdot\Delta e_i + \alpha\cdot(\tau_i + \tau_{ij}) + \sum_{t=\tilde{t}_j+\tau_j^w}^{t_j+\tau_j^w} p(t)\cdot\Delta e_j \\
&= c(s_i, \tilde{s}_j) + c(\tilde{s}_j, s_j)
\end{aligned}
$$

It means that the charging control decisions in the SCC set can be simplified. ■

Note that, if the SCC set exists, the charging control decisions in the set can be simplified to depart with any element in the set and recharge to the corresponding energy at the successor station. Based on theorem 2, the SCC algorithm is designed below. For simplicity, only changes are presented in Algorithm 2, corresponding to lines 4 ∼ 15 in Algorithm 1. In line 2, the charging control decisions in set $S(s_i, j, T_c)$ is simplified when the SCC set exists.

Algorithm 2: SCC Algorithm

$*$ main iteration $*$

1: **if** $S(s_i, j, T_c) \ne \phi$ **then** % simplify set $D(s_i, j)$
2: $\quad D(s_i, j) = D(s_i, j) \backslash \{S(s_i, j, T_c)\} \cup min\{S(s_i, j, T_c)\}$;
3: **end if**
4: **for each** $e'_i \in D(s_i, j)$ % calculate successor state
5: $\quad u_i = (e'_i, j)$;
6: $\quad \tau_i = (e'_i - e_i)/\Delta e_i$;
7: $\quad t_j = t_i + \tau_i + \tau_{ij}$;
8: $\quad e_j = e'_i - e_{ij}$;
9: $\quad s_j = (j, e_j, t_j)$;
10: $\quad c(s_i, u_i) = \sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t)\cdot\Delta e_i + \alpha\cdot(\tau_i + \tau_{ij})$;
11: $\quad B_{t_j} = B_{t_j} \cup \{s_j\}$;
12: $\quad$ **if** $C(s_i) + c(s_i, u_i) < C(s_j)$ **then**
13: $\quad\quad C(s_j) = C(s_i) + c(s_i, u_i)$; % update cost
14: $\quad\quad \psi(s_i) = u_i$;
15: $\quad$ **end if**
16: **end for**

***Remark** 1:* Given a directed acyclic graph $G = (\mathcal{N}, \mathcal{A})$ and the real-time prices $p(t)$, the derived optimal cost in the SCC algorithm equals to that in the ICS algorithm.

As described in Theorem 2 that the charging control decisions in the simplified charging control set can be simplified, and recharged to the same state at the next station, the optimal cost in SCC algorithm equals to that in ICS algorithm, obviously.

In order to further explain the ICS and SCC algorithms, a simple example is given below.

***Example** 1:* Suppose there is a 5-node traffic network, as shown in Fig. 1, where each links are marked with the link travel time (min) and corresponding link energy consumption (kW·h). Assume the charging rate at node 2, 3 and 4 are the same, and the waiting time at node 3 and 4 are 0. The parameters about the battery of the electric vehicle are e_{min} =2 kW·h, e_{stop} = 22 kW·h, e_{rate} = 24 kW·h , and the state at node 1 is s_1 =(1,24,7:00). Let the granularity of charging control decisions be 2 kW·h, for simplicity, and $\Delta e_i = 40$ kW, $i = 2, 3, 4$. Based on the instance of Illinois Power Company real-time pricing [29], the electricity prices are 4.0 Cent/kW·h, 4.2 Cent/kW·h, and 3.5 Cent/kW·h, for $7:00 \leq t \leq 7:59$, $8:00 \leq t \leq 9:59$, and $10:00 \leq t \leq 10:59$, respectively.

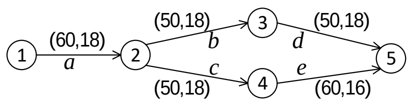
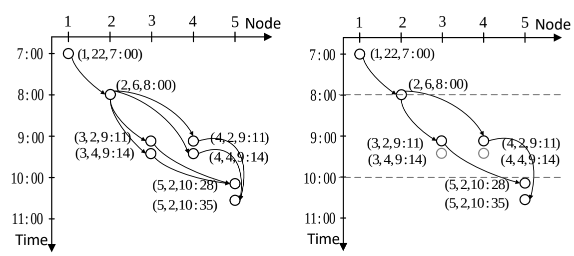


Fig. 1. A 5-node traffic network.

Based on the state at node 2, s_2 =(2,6,8:00), the feasible charging control decisions in the ICS algorithm are $D(s_2, 3) = \{20, 22\}$ and $D(s_2, 4) = \{20, 22\}$. Since the electricity prices are constant for $t \in T_c = \{t \in T | 8:00 \leq t \leq 9:59\}$, the SCC sets are $S(s_2, 3, T_c) = \{20, 22\}$ and $S(s_2, 4, T_c) = \{20, 22\}$. As shown in Fig. 2(b), the charging control decisions at station 2 can be simplified in the SCC algorithm, i.e., the SCC algorithm has a lower computational complexity. Since electricity prices are the same when the EV charge at stations 2, 3, and 4, the derived optimal costs in the ICS and SCC algorithms are equivalent. The extended state networks corresponding to the ICS and SCC algorithms are presented in Fig. 2.



(a) ICS extended network. (b) SCC extended network.

Fig. 2. Extended network obtained by the ICS and SCC algorithms.

IV. Stochastic Navigation Optimization

As an outstanding feature of traffic networks, the traffic profiles, such as link travel time, are stochastic. In order to improve the accuracy of EV navigation systems, a stochastic navigation optimization is presented below, based on a stochastic link travel time model.

A. Stochastic Link Travel Times

There are multiple random events in traffic network, such as incident, vehicle breakdown, and bad weather, which affect the link travel time. Additionally, the link travel time is usually correlated link-wisely and time-wisely. For example, if the randomness comes from the weather, link travel times of the whole network over a certain time period are related. If the randomness comes from incidents, link travel times around the incident location are related to the incident duration. In this part, we use the stochastic link travel time model in [26], with the assumption that link travel times are temporal and spatial dependent, and travellers have perfect online information, i.e., they know the realizations of all the link travel times up to the current time.

TABLE II
Temporal and Spatial Probability Distribution

Time	Link Travel Time (min)					Probability
	a	b	c	d	e	
7:00$\leq t \leq$ 8:59	60	50	50	50	60	1
9:00$\leq t \leq$ 9:59	60	50	50	50	50	0.5
	60	50	50	50	70	0.5
10:00$\leq t \leq$10:59	50	50	50	50	60	0.4
	70	50	50	50	60	0.6

Denote $\Gamma(t)$ as the possible travel time set of all links at time t (an illustration is given in Example 2). We define the link travel time mode in the stochastic time-dependent networks as follows.

***Definition** 3:* Given a traffic network $G = (\mathcal{N}, \mathcal{A})$ and the travel time limitation set T, a link travel time mode, m_r, is a possible combination of all the link travel time over T, determined by the temporal and spatial dependence, i.e., $m_r \in R^{|T| \times |\mathcal{A}|}$, and $m_r = (\tau_{12}(t_1), \cdots, \tau_{ij}(t_1), \cdots, \tau_{12}(t), \cdots, \tau_{ij}(t), \cdots, \tau_{12}(t_{end}), \cdots, \tau_{ij}(t_{end}))$.

Denote $m_1, \cdots, m_r, \cdots, m_R$ as the possible link travel time modes, the joint link travel time (discrete) probability distribution is the distribution of such modes. Denote $P(m_r)$ as the possibility of mode m_r, we have $\sum_{r=1}^{R} P(m_r) = 1$.

***Example** 2:* For a given traffic network as shown in Fig. 1, suppose there is a temporal and spatial probability distribution over $T = \{7:00, \cdots, 10:59\}$ as shown in Table II, where the rows reflect the spatial dependence and the columns reflect the temporal dependence. The joint link travel time probability distribution can be obtained from Table II, as shown in Table III. Note that the rows in Table II correspond to $\Gamma(t)$, e.g., for 7:00 $\leq t \leq$ 8:59, $\Gamma(t) = \{(60, 50, 50, 50, 60)\}$, and for 9:00 $\leq t \leq$ 9:59, $\Gamma(t) = \{(60, 50, 50, 50, 50), (60, 50, 50, 50, 70)\}$. Moreover, the rows in Table III are the possible link travel time modes m_1, $\cdots$, m_4.

Given the joint link travel time distribution, the link travel time modes can be identified utilizing online link travel time information. Denote $\pi_{ij}(\tilde{t})$ as the known link travel times before current time t, where $(i, j) \in \mathcal{A}$, $\tilde{t} \leq t$. The mode set, identified by $\pi_{ij}(\tilde{t})$, can be defined as follows.

***Definition** 4:* For a given joint link travel time distribution, the identified mode set at time t is $M(t) = \{m_r | \forall (i, j) \in \mathcal{A}, \forall \tilde{t} \leq t, \tau_{ij}(\tilde{t}) = \pi_{ij}(\tilde{t})\}$, where $\pi_{ij}(\tilde{t})$ is a special value of $\tau_{ij}(\tilde{t})$.

For example, suppose the current time is $t = 9:30$, and the known link travel time is $\pi_{ij}(\tilde{t})$, $\tilde{t} \leq t$, $(i, j) \in \mathcal{A}$. In Table III, if the link travel time sequence is (60, $\cdots$, 50), then the identified mode set at time t is $M(t) = \{m_1, m_2\}$. If the link travel time sequence is (60, $\cdots$, 70), then the identified mode

TABLE III

AN EXAMPLE OF JOINT LINK TRAVEL TIME DISTRIBUTION.

Time & Link		7:00 $\leq t \leq$ 8:59					9:00 $\leq t \leq$ 9:59					10:00 $\leq t \leq$ 10:59					Probability
		a	b	c	d	e	a	b	c	d	e	a	b	c	d	e	
Mode	m_1	60	50	50	50	60	60	50	50	50	50	50	50	50	50	60	0.2
	m_2	60	50	50	50	60	60	50	50	50	50	70	50	50	50	60	0.3
	m_3	60	50	50	50	60	60	50	50	50	70	50	50	50	50	60	0.2
	m_4	60	50	50	50	60	60	50	50	50	70	70	50	50	50	60	0.3

set at time t is $M(t) = \{m_3, m_4\}$.

Suppose the current time is t, and denote the possible identified mode set at time $t^* > t$ as $M_1(t^*), \cdots, M_k(t^*)$, we define the possible mode set at time $t^* > t$ as $\mathcal{M}(t^*) = \{M_1(t^*), \cdots, M_k(t^*)\}$, $t^* > t$. For example, suppose current time is $t = 8:00$, then the possible mode set at time $t^* = 9:00$ is $\mathcal{M}(t^*) = \{\{m_1, m_2\}, \{m_3, m_4\}\}$.

Suppose there are mode sets $M(t)$ and $M(t^*)$, $t^* > t$, then the conditional probability of such mode sets can be expressed as

$$P(M(t^*)|M(t)) = \frac{\sum\limits_{m_r \in M(t) \cap M(t^*)} P(m_r)}{\sum\limits_{m_r \in M(t)} P(m_r)}, \tag{7}$$

where $M(t) \cap M(t^*) = \phi$ or $M(t^*)$.

B. Dynamic Programming Formulation

Compared with the deterministic case, the time-dependent link travel times in stochastic traffic network are related to the charging and routing cost. In this part, we describe state variables with the EV arrival state and the identified mode set.

Suppose that the current state is $s_i = (i, e_i, t_i)$, and the identified mode set is $M(t_i)$. Denote $\pi_{ij}(t)$ and $\xi_{ij}(t)$, $(i,j) \in \mathcal{A}$, $t \leq t_i$, as a special value of $\tau_{ij}(t)$ and $e_{ij}(t)$ in the identified mode set $M(t_i)$. Let $u_i = (e'_i, j)$ be the decision at the ith station, where the decisions satisfy $j \in O(i)$, and $e'_i \in \mathcal{D}(s_i, M(t_i), j)$. $\mathcal{D}(s_i, M(t_i), j)$ is defined as

$$\mathcal{D}(\cdot) = \{e'_i \in E | \max\{e_i, e_{min} + \xi_{ij}(t_i)\} \leq e'_i \leq \max\{e_i, e_{stop}\}\} \tag{8}$$

Similarly, the charging cost at the ith station can be expressed as

$$c(s_i, u_i) = \sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t) \cdot \Delta e_i + \alpha \cdot (\tau_i + \pi_{ij}(t_i)) \tag{9}$$

According to the current state $(s_i, M(t_i))$ and decision u_i, the successor state is $s_j = (j, e_j, t_j)$, where $e_j = e'_i - \xi_{ij}(t_i)$, $t_j = t_i + \tau_i + \pi_{ij}(t_i)$, and τ_i can be calculated by (2). Suppose the possible mode set at time t_j is $\mathcal{M}(t_j)$. Define $\mathcal{C}(s_i, M(t_i))$ as the minimal cost from the current state $(s_i, M(t_i))$ to the destination node, then it can be formulated in a recursion form

$$\mathcal{C}(s_i, M(t_i)) = \min\left\{c(s_i, u_i) + \underset{M(t_j) \in \mathcal{M}(t_j)}{Ex}[\mathcal{C}(s_j, M(t_j))]\right\}, \tag{10}$$

C. Online State Recursion (OSR) Algorithm

Similar to the ICS algorithm, we search the optimal charging and routing decisions in a stochastic traffic network, chronologically.

In the OSR algorithm, we denote $\mathcal{C}(s_i, M(t_i))$ as the minimal cost from the currents state $(s_i, M(t_i))$ to the destination node. Let $\tilde{\mathcal{C}}(s_i, M(t_i))$ be the expectation of $\mathcal{C}(s_i, M(t_i))$, $M(t_i) \in \mathcal{M}(t_i)$. Denote $\psi(s_i, M(t_i))$ as the optimal decisions based on the current state $(s_i, M(t_i))$. We assume that EVs make decision as soon as they arrive at a station. The detailed steps are presented in Algorithm 3.

Algorithm 3: OSR Algorithm

$*$Main iteration$*$

1: **select** $(s_i, M(t_i))$ from B_{t_i};
2: $B_{t_i} = B_{t_i} \backslash \{(x, M(t_i))\}$;
3: **if** $i == n$
4: $\quad \mathcal{C}(s_i, M(t_i)) = 0$; % boundary condition
5: **else**
6: $\quad$ **for each** $j \in O(i)$ % calculate successor state
7: $\quad\quad$ **for each** $e'_i \in \mathcal{D}(s_i, M(t_i), j)$
8: $\quad\quad\quad u_i = (e'_i, j)$;
9: $\quad\quad\quad \tau_i = \tau_i^w + (e'_i - e_i)/\Delta e_i$;
10: $\quad\quad\quad t_j = t_i + \tau_i + \pi_{ij}(t_i)$;
11: $\quad\quad\quad e_j = e'_i - \xi_{ij}(t_i)$;
12: $\quad\quad\quad s_j = (j, e_j, t_j)$;
13: $\quad\quad\quad c(s_i, u_i) = \sum_{t=t_i+\tau_i^w}^{t_i+\tau_i} p(t) \cdot \Delta e_i + \alpha \cdot (\tau_i + \pi_{ij}(t_i))$;
14: $\quad\quad\quad$ **for each** $M(t_j) \in \mathcal{M}(t_j) \cap M(t_i)$
15: $\quad\quad\quad\quad B_{t_j} = B_{t_j} \cup \{(s_j, M(t_j))\}$;
16: $\quad\quad\quad\quad P = P(M(t_j)|M(t_i))$; % conditional probability
17: $\quad\quad\quad\quad \tilde{\mathcal{C}}(s_j, M(t_j)) = \tilde{\mathcal{C}}(s_j, M(t_j)) + P \cdot \mathcal{C}(s_j, M(t_j))$;
18: $\quad\quad\quad$ **end for**
19: $\quad\quad\quad$ **if** $\mathcal{C}(s_i, M(t_i)) > c(s_i, u_i) + \tilde{\mathcal{C}}(s_j, M(t_j))$
20: $\quad\quad\quad\quad \mathcal{C}(s_i, M(t_i)) = c(s_i, u_i) + \tilde{\mathcal{C}}(s_j, M(t_j))$;
21: $\quad\quad\quad\quad \psi(s_i, M(t_i)) = u_i$; % update cost
22: $\quad\quad\quad$ **end if**
23: $\quad\quad$ **end for**
24: $\quad$ **end for**
25: **end if**

Corresponding to the recursion formulation in (10), the cost of the successor state $\mathcal{C}(s_j, M(t_j))$ is called when calculating the cost of the current state $(s_i, M(t_i))$. Line 3 is the boundary condition of iterations. Lines 14 $\sim$ 18 are used to calculate the expected minimal cost of the possible successor state $(s_j, M(t_j))$. Lines 19 $\sim$ 22 are used to update the minimal cost of the current state $(s_i, M(t_i))$. In line 24, $\psi(s_i, M(t_i))$ is used to record the optimal charging and routing decisions. The intermediate variable $\tilde{\mathcal{C}}(s_j, M(t_j))$ (in line 17) is initialized as zero.

***Theorem** 3:* Given a traffic network $G = (\mathcal{N}, \mathcal{A})$, the travel time limit T, and the possible link travel times mode $m_1, \cdots, m_R$, the OSR algorithm terminates after a finite number of iterations, which is no more than $|\mathcal{A}| \cdot |E|^2 \cdot |T| \cdot |R|$.

We omit the proof for brevity, which is similar to the proof of Theorem 1.

In OSR algorithm, on-line traffic information must be obtained to identify the current mode set $M(t)$, which brings high-density communication between EVs and traffic infrastructures. Moreover, detailed history data are needed in constructing the joint link travel time distribution, which introduces heavy computing burden. In order to balance the navigation accuracy, computing burden, and communication overload, a new navigation infrastructure, such as an on-line server-client infrastructure, is necessary, where the servers, with powerful computational capacities, is used to construct the joint link travel time distribution and search the optimal decisions, and on-vehicle clients are used to upload EV's states and receive the optimal routing and charging decisions.

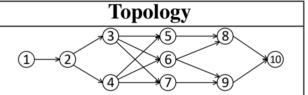
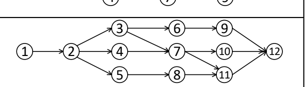
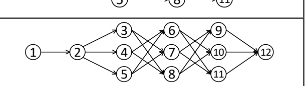
V. Numerical Simulation

In this section, we simulate the joint charging and routing optimization in both deterministic and stochastic networks. In the deterministic case, we compare the CPU time and cost in the ICS and SCC algorithms. In the stochastic case, we compare the cost in the OSR algorithm with an assumed omniscient driver, who knows the real traffic properties in advance. The computer used in simulation has a 2.93GHz CPU and 2 GB RAM.

In simulation, we use the parameters of Renault ZOE, where $e_{min} = 2$ kW · h, $e_{stop} = 22$ kW· h, and $e_{rate} = 24$kW· h. The ZOE power-speed curve [27] is used in calculating the link energy consumption. As CHAdeMO chargers are in the majority of the existing fast chargers [28], we set all the charging rate to 40kW. Assume the electric vehicle departs from node 1 at time $7:00$ with the energy 24 kW·h, and the travel time limit is $T = \{7:00, \cdots, 11:59\}$. The charging control decision granularity is assumed to be 1 kW·h. We use the real-time price of the Illinois Power Company [29] in simulation.

As shown in Table IV, three traffic networks with different topology are used, where case 1 and case 2 have the same link numbers, case 2 and case 3 have the same node numbers. We assume that the distance between two charging stations is subject to uniform distribution within [60 km, 80 km], which is suitable for EVs with a cruise range of about 100 km. The distance between the origin and the destination nodes is about 300 km $\sim$ 400 km, corresponding to a near 5 hours trip for the EV drivers.

TABLE IV
Simulated traffic networks and topologies.

Case	Node	Link	Topology
1	10	15	
2	12	15	
3	12	20	

A. Simulation in Deterministic Networks

The ICS and SCC algorithms are simulated in deterministic traffic networks, where the travel speed is subject to uniform distribution within in [60 km/h, 90km/h]. To evaluate the performance of the SCC algorithm, we set the waiting times as 0 here. We simulate 100 times for each cases, the average costs, iteration numbers, and CPU times are presented in Table V.

TABLE V
Simulation results of the ICS and SCC algorithms

Item / Case	ICS algorithm			SCC algorithm		
	Iteration	Times (s)	Cost	Iteration	Times (s)	Cost
Case 1	3311	5.74	46.81	2503	3.94	46.81
Case 2	2460	3.56	46.35	1967	2.66	46.35
Case 3	4637	9.35	46.10	3736	7.04	46.10

As shown in Table V, the number of iterations and the CPU times in the SCC algorithm are much smaller than those in the ICS algorithm. In detail, the SCC algorithm saves about 1/4 $\sim$ 1/3 CPU times compared with the ICS algorithm. Since the algorithms are deployed on on-vehicle embedded devices with low computational capacities, the SCC algorithm with a shorter CPU time is more suitable than the ICS algorithm.

B. Simulation in Stochastic Networks

According to the analysis of spatial and temporal correlation in [30], a joint link travel time distribution is presented in Table VI. The columns in Table VI is travel time of all the links over 7:00 $\sim$ 11:59. Assume the hour between 8:00 and 9:00 is peak hour, the hour between 10:00 and 11:00 is off-peak hour, and the other hours are intermediate hours. The link travel times can be obtained by the link distance and the uniformly distributed link speed, which have an average speed 60 km/h at peak hours, 90 km/h at off-peak hours, and 70 km/h at intermediate hours. The detailed link travel times are omitted, and the table is rotated (compared with Table III) for space saving. The waiting time at each node is generated randomly within 0$\sim$5 min.

TABLE VI
The joint link travel time distribution in simulation.

Time & Link		Link travel time (min) mode							
		m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
$7{:}00 \leq t \leq 7{:}59$	a	60	60	60	60	60	60	60	60
	b	50	50	50	50	50	50	50	50
	...	...	...	...	...	...	...	...	...
$8{:}00 \leq t \leq 8{:}59$	...	...	...	...	...	...	...	...	...
$9{:}00 \leq t \leq 9{:}59$	...	...	...	...	...	...	...	...	...
$10{:}00 \leq t \leq 10{:}59$	...	...	...	...	...	...	...	...	...
$11{:}00 \leq t \leq 11{:}59$	...	...	...	...	...	...	...	...	...
$P(m_r)$		0.28	0.12	0.07	0.03	0.28	0.12	0.07	0.03

Assume there is an omniscient EV driver who knows the real link travel times of each simulation scenarios in advance. The ideal cost of the omniscient driver is calculated utilizing the time-dependent deterministic traffic properties. In order to evaluate the accuracy of the OSR algorithm in stochastic networks, we define the normalized residual as

$$\delta = \frac{1}{N} \cdot \sum\nolimits_{k=1}^{N} (\mathcal{C}_k(s_1, M(t_1)) - \mathcal{C}_k(s_1, m_r))/\mathcal{C}_k(s_1, m_r),$$

where δ is the normalized residual, N is the number of simulations, $\mathcal{C}_k(s_1, M(t_1))$ is the cost of the OSR algorithm at the kth simulation with a starting state s_1 and the identified mode set $M(t_1)$, $\mathcal{C}_k(s_1, m_r)$ is the ideal cost at the kth simulation with the starting state s_1 and the known ahead link

TABLE VII
SIMULATION RESULTS OF THE OSR ALGORITHMS

Item / Case	Omniscient driver			OSR algorithm			δ (%)
	Iteration	Times (s)	Cost	Iteration	Times (s)	Cost	
Case 1	4432	34.92	44.61	30710	273.10	48.50	8.72
Case 2	3896	32.17	45.01	27468	276.21	46.46	3.22
Case 3	4734	43.58	40.79	30951	366.50	44.18	8.31

travel time mode m_r. The average costs, iteration numbers, and CPU times are given in Table VII.

As shown in Table VII, the cost in the OSR algorithm is quite close to the ideal cost of the omniscient EV driver, within a normalized residual of about 10%. The CPU times (iteration numbers) in the OSR algorithm are much larger than those in the omniscient cases, because all the possible link travel time modes must be discussed in stochastic cases. Moreover, additional CPU times are introduced by the states searching in the minimal cost update of each iteration, which can be reduced by storing the states in the corresponding place of the matrix, if there are enough caches or RAMs. Since the OSR algorithm will be deployed on servers with powerful computational capacities and large caches, EV drivers may receive the optimal routing and charging decisions within a much smaller response time.

VI. CONCLUSION

In this work, we study the electric vehicle navigation problem, which can provide the minimal routing and charging cost during a long distance trip. Because of the advantages such as none/less communication requirement, low computational complexity, the SCC algorithm based navigation system is compatible to the traditional navigation systems. Utilizing the stochastic traffic model, the OSR algorithm based navigation system guarantees the accuracy of the navigation systems, when there is enough on-line traffic information. In this paper, we focus on the navigation of single electric vehicle, the interactions between electric vehicles and navigation of multiple electric vehicles will be studied in the future.

REFERENCES

[1] K. Qian, C. Zhou, M. Allan, and Y. Yuan, "Modeling of load demand due to EV battery charging in distribution systems," *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 802-810, Mar. 2010.

[2] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 371-380, Jan. 2010.

[3] N. Rotering, and M. Ilic, "Optimal charge control of plug-in hybrid electric vehicles in deregulated electricity markets,". *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1021-1029, Jul. 2011.

[4] S. Deilami, A. S. Masoum, P. S. Moses, and M. A. Masoum, "Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile,". *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 456-467, Aug. 2011.

[5] Y. Cao, S. Tang, C. Li, P. Zhang, Y. Tan, Z. Zhang, and J. Li, "An optimized EV charging model considering TOU price and SOC curve," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 388-393, Feb. 2012.

[6] R. J. Bessa, M. A. Matos, F. J. Soares, and J. A. P. Lopes, "Optimized bidding of a EV aggregation agent in the electricity market," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 443-452, Feb. 2012.

[7] P. Richardson, D. Flynn, and A. Keane, "Optimal charging of electric vehicles in low-voltage distribution systems,". *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 268-279, Jan. 2012.

[8] C. Wen, J. Chen, J. Teng, and P. Ting, "Decentralized plug-in electric vehicle charging selection algorithm in power systems,". *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1779-1789, Dec. 2012.

[9] L. Gan, U. Topcu, and S. H. Low, "Optimal decentralized protocol for electric vehicle charging," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 940-95, May 2013.

[10] Z. Ma, D. S. Callaway, and I. A. Hiskens, "Decentralized charging control of large populations of plug-in electric vehicles,". *IEEE Transactions on Control Systems Technology*, vol. 21, no. 1, pp. 67-78, Dec. 2012.

[11] C. Ahn, C. Li, and H. Peng, "Optimal decentralized charging control algorithm for electrified vehicles connected to smart grid," *Journal of Power Sources*, vol. 196, no. 23, pp. 10369-10379, Jul. 2011.

[12] M. Sachenbacher, M. Leucker, A. Artmeier, and J. Haselmayr, "Efficient energy optimal routing for electric vehicles," in *Proceedings of the 25th Conference on Artificial Intelligence*, San Francisco, US, Aug. 2011, pp. 1402-1407.

[13] M. M. DeWeerdt, E. H. Gerding, S. Stein, V. Robu, and N. R. Jennings, "Intention-aware routing to minimise delays at electric vehicle charging stations," in *Proceedings of the 23rd International Joint conference on Artificial Intelligence*, Beijing, China, Aug. 2013, pp. 83-89.

[14] F. Baouche, R. Billot, R. Trigui, E. Faouzi, and N. Eddin, "Electric vehicle green routing with possible en-route recharging," in *Proceedings of the 17th International Conference on Intelligent Transportation Systems*, Qingdao, China, Oct. 2014, pp. 2787-2792.

[15] M. Schneider, A. Stenger, and D. Goeke, "The electric vehicle routing problem with time windows and recharging stations,". *Transportation Science*, vol. 48, no. 4, pp. 500-520, 2014.

[16] F. Alesiani, and N. Maslekar, "Optimization of charging stops for fleet of electric vehicles: a genetic approach," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 3, pp. 10-21, 2014.

[17] C. Liu, J. Wu, C. Long, "Joint charging and routing optimization for electric vehicle navigation systems," in *Proceedings of the 19th World Congress of the International Federation of Automatic Control*, Cape Town, South Africa, Aug. 2014, pp. 2106-2111.

[18] R. W. Floyd, "Algorithm 97: shortest path," *Communications of the ACM*, vol. 5, no. 6, pp. 345, Jun. 1962.

[19] K. Boriboonsomsin, M. J. Barth, W. Zhu, and A. Vu, "Eco-routing navigation system based on multisource historical and real-time traffic information," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1694-1704, 2012.

[20] T. Wang, C. G. Cassandras, and S. Pourazarm, "Energy-aware vehicle routing in networks with charging nodes," in *Proceedings of the 19th World Congress of the International Federation of Automatic Control*, Cape Town, South Africa, 2014, pp. 9611-9616.

[21] S. Pourazarm, C. G. Cassandras, and A. Malikopoulos, "Optimal routing of electric vehicles in networks with charging nodes: a dynamic programming approach," *IEEE International Electric Vehicle Conference*, Florence, Italy, Dec. 2014, pp. 1-7.

[22] CHAdeMO User Guide [Online]. Available: http://teslapedia.org/model-s/tesla-driver/chademo-user-guide/.

[23] H. Hoimoja, A. Rufer, G. Dziechciaruk, and A. Vezzini, "An ulstrfast EV charging station demonstrator," *International Symposium on Power Electronics, Electrical Drives, Automatiion and Motion*, 2012, pp. 1390-1395.

[24] M. M. Weerdt, S. Stein, E. H. Gerding, V. Robu, and N. R. Jennings, "Intention-aware routing of electric vehicles," *IEEE Transactions on Intelligent Transportation Systmes*, vol. 17, no. 5, pp. 1472-1482, 2016.

[25] R. B. Dial, "Algorithm 360: Shortest-path forest with topological ordering," *Communications of the ACM*, vol. 12, no. 11, pp. 632-633, 1969.

[26] S. Gao,and I. Chabini, "Optimal routing policy problems in stochastic time-dependent networks," *Transportation Research Part B: Methodological*, vol. 40, no. 2, pp. 93-122, 2006.

[27] Maximising range in a ZOE 2, 2013. Retrieved from http://myrenaultzoe.com/index.php/author/umberto/

[28] A simple guide to DC fast charging, 2016. Retrieved from http://www.fleetcarma.com/dc-fast-charging-guide/

[29] A. H. Mohsenian-Rad, and A. Leon-Garcia, "Optimal residential load control with price prediction in real-time electricity pricing environments," *IEEE Transactions on Smart Grid*, vol. 1, no.2, pp. 120-133, 2010.

[30] P. Rachtan, "Spatial and temporal correlations of freeway link speeds: an empirical study," M.S. thesis, University of Massachusetts Amherst, Massachusetts, US, 2012. Retrieved from http://scholarworks.umass.edu/theses/940/