Charging scheduling with minimal waiting in a network of electric vehicles and charging stations

Conference Paper · September 2011

DOI: 10.1145/2030698.2030706 · Source: DBLP

2 authors:

Hua Qin
Hunan University

Wensheng Zhang
Iowa State University

Charging Scheduling with Minimal Waiting in A Network of Electric Vehicles and Charging Stations

Hua Qin and Wensheng Zhang
Department of Computer Science
Iowa State University, IA, USA
{hqin,wzhang}@iastate.edu

ABSTRACT

Environment preservation has become a prominent issue around the world. As traditional internal combustion engine (ICE) vehicles have been major contributors of air pollution, electric vehicles (EVs) are gaining popularity. However, due to the limited electricity supply of battery pack, EVs need to be charged frequently and each charge takes long time. This may degrade travel efficiency and driver comfort. To address this issue, this paper aims to minimize charging waiting time through intelligently scheduling charging activities spatially and temporally. A theoretical study has been conducted to formulate the waiting time minimized charging scheduling problem and derive a performance upper bound (i.e., the theoretical lower bound of charging waiting time). Based on the insights discovered from the theoretical analysis, a practical distributed scheme has been proposed. Extensive simulation results verify that the proposed design can achieve a waiting time near the theoretical lower bound.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Distributed networks*

General Terms

Theory, Design, Algorithms

Keywords

Electric vehicle, charging scheduling, delay minimization

1. INTRODUCTION

As environment preservation becomes a prominent issue around the world, electric vehicles (EVs) are poised to gain mass acceptance from the general public. Electric vehicles offer many benefits over traditional ones, such as high energy efficiency, low greenhouse gas emissions, potential to run on locally produced renewable energy (e.g., wind and solar power) and so on. EVs are gaining popularity in market, for not only short-distance commuting but also long-distance journeys. It is forecasted that about 1.5 million EVs will be in the US by 2015, and over 10 million by 2020 [7].

Many models of EVs (e.g., [8, 2, 3, 4, 6, 5]) have been or will soon be released to the market. To cite a few, the Hyundai BlueOn [4] is equipped with a battery pack that offers a range of 140 km per charge and can be charged to 80% of the capacity in 25 minutes; the Nissan Leaf electric car [5] runs up to 160 km per charge and takes 30 minutes to charge 80% of the capacity of the battery pack. Even though, EVs still need frequent charging and long charging time, compared to traditional internal combustion engine (ICE) vehicles powered by gasoline. The long charging time may further lead to long waiting at charging stations. For travel efficiency and driver comfort, it is necessary to intelligently schedule the electricity charging of EVs so as to minimize waiting without disrupting the travel plans or habits of the drivers.

Coexistence of EVs with the emerging smart grids has been recently an attractive research topic. A considerable amount of efforts have been made to develop standards, such as [10, 9], for EV to smart grid communication (also known as vehicle-to-grid (V2G)). In connection with these standards, a wide range of organizations are involved in developing wireless software and hardware solutions (e.g., Zigbee, WiFi, WiMax, cellular networks, etc [10]) to support V2G. With the support of these technologies, we propose a charging scheduling scheme with minimal waiting in this paper to intelligently schedule the EV charging in a *large-scale* road network by taking advantage of the interoperability between EVs and charging stations.

Achieving minimal waiting time for EV charging does not come in without challenges. Due to the mobile nature, EVs could be very randomly distributed spatially and temporally. The exact knowledge about the EVs may not be available, and it may change over time and space. Hence, effectively scheduling charging would be difficult, especially in a large-scale range. Because of the potentially huge number of EVs on road simultaneously and their wide distribution, collecting the information from them to a central server which directs the electricity charging of every EV is not practical.

To overcome the above difficulties, we first conduct a theoretical study by modelling and analyzing the electricity charging behaviors in a large-scale network of EVs and charging stations. Particularly, a performance upper bound (i.e., the lower bound of charging waiting time) has been found by assuming the presence of a central server which knows all information about the road system, the electricity charging stations, and the EVs. Then, based on the insights discovered from the theoretical study, an efficient, practical and distributed scheme is designed.

To evaluate our proposed design, extensive simulations have been conducted. The results show that the proposed scheme can achieve an average waiting time that approximates its theoretical lower bound.

The rest of the paper is organized as follows. Section 2 presents the system model and theoretical analysis. Section 3 elaborates the proposed distributed charging scheduling scheme, which is followed by the report of simulation evaluation in Section 4. Section 5 summarizes related work. Finally, Section 6 concludes the paper and discusses the future work.

2. SYSTEM MODEL AND THEORETICAL ANALYSIS

This section presents the system model, formulates the problem to be studied as an optimization problem, and derives a performance upper bound for solutions to the problem.

2.1 System Model

We consider a highway road system, where charging stations are deployed at some entrances/exits. At each entrance/exit, some EVs may enter or exit the road system. If there is a charging station at the entrance/exit, some EVs may be charged and then continue moving to their destinations. All charging stations are connected via the Internet and therefore can communicate with each other. They are also equipped with wireless communication devices (e.g., WI-FI, 3G), which allow them to communicate with nearby EVs that are also equipped with wireless devices. Hereafter, we use *charging station* and *station*, *EV* and *vehicle* interchangeably.

Formally, the connected system of charging stations can be represented as a graph $G = \langle V, E\rangle$, where V is the set of stations, E is the set of communication paths connecting stations. Each station v has c_v charging outlets, each has a charging capacity of r_v^+ (Watts); that is, the outlet can charge electricity to a vehicle at the rate of r_v^+ (Watts). The battery of each vehicle i has an electricity capacity of $e_{i,max}$ (J) and it consumes electricity at a rate of r_i^- (J/meter).

We assume that the rate of charging demands of all vehicles is lower than the overall rate of electricity charging such that there exists a charging schedule with which no vehicle needs to wait infinitely. In addition, the following assumptions are made to simplify theoretical analysis. Note that, *our proposed practical scheme will not be restricted by these additional assumptions.*

- For any pair of stations $(s, d) \in V^2$, the flow of vehicles originating from s and destinating at d follows the Poisson process with arrival rate $\lambda_{s,d}$. Each vehicle always takes the shortest route between s and d, denoted as $R_{s,d}$.
- All stations have the same charging capacity; i.e., $c_v = c$ and $r_v^+ = r^+$ for any $v \in V$, where c and r^+ are constants.
- All vehicles have the same electricity capacity e_{max} and consumption rate r^-. They all enter the road system with full electricity e_{max} , and are fully charged whenever they are charged. Hence, the maximum distance that it can travel after one charge (called *maximum one-charge travel distance* hereafter) is denoted as $d_{max} = e_{max}/r^-$.
- We only consider vehicles whose routes are longer the maximum one-charge travel distance. Vehicles whose routes are shorter than the maximum one-charge travel distance need no charging and thereby will not affect the system.

2.2 Problem Formulation

The problem we aim to solve is to find a charging scheduling scheme that can minimize the average time period that each vehicle has to wait for charge during its trip. Here, the *waiting time* includes the time period that the vehicle waits at a queue for a charging outlet to become available (called *queuing time*) and the time period during which the vehicle is being charged (called *charging time*).

To facilitate problem formulation, we introduce the following notations:

- $F_{s,d}$: the set of stations on the shortest route between stations s and d such that each of the stations is not away from s for longer than the maximum one-charge travel distance. Here, we assume the shortest route is unique.
- $\Omega = \{p^v_{s,d} \,|\, \forall (s, d) \in V^2, v \in F_{s,d}\}$: the formal representation of a charging schedule. Specifically, $p^v_{s,d}$ denotes the probability that a vehicle originating from s (or charged at) station s and destinating at station d should be charged at station v.
- λ_v, μ_v for any station v: the charging process at a station v can be modelled as a multi-server queueing system with expected arrival rate λ_v (i.e., the expected number of vehicles arriving at station v for charging per time unit) and expected service rate μ_v (i.e., the expected number of vehicles that station v can charge per time unit).
- $\Lambda_{s,d}$: flow of vehicles that originate from or are charged at station s, and are destinated at station d.
- $\Gamma_{s,v}$, $\gamma_{s,v}$: $\Gamma_{s,v}$ denotes the flow of vehicles that originate from or are charged at station s and pass station v, where v is on the shortest routes between s and their destinations. $\gamma_{s,v}$ denotes the arrival rate of flow $\Gamma_{s,v}$.

The problem is formulated as follows.

Objective:

$$\text{Find } \Omega \text{ to minimize } \quad W = \sum_{v \in V} \lambda_v w_v, \tag{1}$$

where w_v be the expected waiting time that a randomly selected vehicle spends for charging at station $v \in V$.

s.t.,

$$p^v_{s,d} = 0, \qquad \forall (s, d) : d \in F_{s,d} \tag{2}$$

$$\sum_{v \in F_{s,d}} p^v_{s,d} = 1, \qquad \forall (s, d) : d \notin F_{s,d} \tag{3}$$

$$\lambda'_{s,d} = \lambda_{s,d} + \sum_{\forall (s',d): s \in R_{s',d}} \lambda'_{s',d} \cdot p^s_{s',d}, \quad \forall (s, d) \in V^2 \tag{4}$$

$$\gamma_{s,v} = \sum_{\forall (s,d'): v \in R_{s,d'}} \lambda'_{s,d'} \cdot p^v_{s,d'}, \qquad \forall (s, v) \in V^2 \tag{5}$$

$$\sum_{v \in V} \sum_{s \in V} (\gamma_{s,v} \cdot E_{s,v}) = \sum_{s \in V} \sum_{d \in V} \Theta(\lambda_{s,d}, \Omega_{s,d}), \tag{6}$$

$$\lambda_v = \sum_{s \in V} \gamma_{s,v}, \qquad \forall v \in V \tag{7}$$

$$\mu_v = \left[\sum_{s \in V} \frac{\gamma_{s,v}}{\lambda_v} \cdot \frac{E_{s,v}}{r^+} \right]^{-1}, \qquad \forall v \in V \tag{8}$$

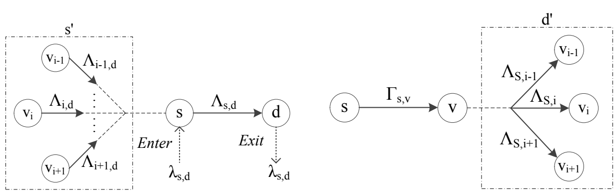
$$w_v = \Psi(\lambda_v, \mu_v, c), \quad \forall v \in V \tag{9}$$

$$0 \le \rho_v < 1, \qquad \forall v \in V \tag{10}$$

$$0 \le p^v_{s,d} \le 1, \qquad \forall (s,d) \in V^2, v \in F_{s,d} \tag{11}$$

Here, Eq. (2) and (3) express the constraints for each vehicle. Eq. (2) indicates that a vehicle needs no more charging if the farthest station that it can reach without additional charging is farther than its destination station. Eq. (3) specifies that a vehicle has to choose a station in $F_{s,d}$ to get charged if it cannot reach its destination without additional charging.

Since the given arrival rate $\lambda_{s,d}$ only counts the vehicles originating at s, we introduce an auxiliary arrival rate $\lambda'_{s,d}$ in Eq. (4), which is the arrival rate of all vehicles on flow $\Lambda_{s,d}$ (i.e., vehicles originating from or charged at s). The second part of Eq. (4) computes the arrival rate of vehicles charged at but not originating from s recursively, by summing up the expected contribution of every other vehicle flow $\Lambda_{s',d}$ that passes s. Particularly, the vehicles belonging to flow $\Lambda_{s',d}$ have the probability $p^s_{s',d}$ to merge into flow $\Lambda_{s,d}$. For example, as shown in Fig. 1-(a), vehicle flows $\Lambda_{v_{i-1},d}, \Lambda_{v_i,d}$ and $\Lambda_{v_{i+1},d}$ all destinate at d and pass station s. Regarding vehicle arrival rate, the contributions of these flows to $\Lambda_{s,d}$ are $p^s_{v_{i-1},d}\lambda'_{v_{i-1},d}$, $p^s_{v_i,d}\lambda'_{v_i,d}$ and $p^s_{v_{i+1},d}\lambda'_{v_{i+1},d}$, respectively. Besides the vehicle flows from other stations, new vehicles entering the road system (with arrival rate $\lambda_{s,d}$) immediately become part of vehicle flow $\Lambda_{s,d}$.



(a) (b)

Figure 1: Examples. (a) Formation of vehicle flow $\Lambda_{s,d}$. (b) Charging flow $\Gamma_{s,v}$ originating at station s and feeding into station v

In Eq. (5), λ_v is computed by summing up the arrival rates for all possible $\Gamma_{s,v}$. An example is shown in Fig. 1-(b), where three vehicle flows, $\Lambda_{s,v_{i-1}}, \Lambda_{s,v_i}$ and $\Lambda_{s,v_{i+1}}$, having the same source s and also passing station v contribute to charging flow $\Gamma_{s,v}$ with vehicle rates $p^v_{s,v_{i-1}}\lambda'_{s,v_{i-1}}$, $p^v_{s,v_i}\lambda'_{s,v_i}$ and $p^v_{s,v_{i+1}}\lambda'_{s,v_{i+1}}$, respectively.

Besides the vehicle flow constraint (i.e., Eq. (4) and (5)), *electricity flow* constraint in Eq. (6) also needs to be considered when the system is in steady state. Specifically, the sum of the expected electricity charging requests serviced at all stations should be equal to the sum of the expected electricity charging requests input to the network at all stations (called *network charging input*). The expected electricity charging requests at a station v is given by $\gamma_{s,v} \cdot E_{s,v}$, where the expected electricity demand $E_{s,v}$ made by a vehicle on flow $\Gamma_{s,v}$ to station s can be computed by

$$E_{s,v} = e_{max} - r^- \cdot D_{s,v}, \tag{12}$$

where $D_{s,v}$ is the (shortest) road distance between s and v. The expected electricity charging requests input at a station s is determined by the flows of vehicles that originates at s. For a flow of vehicles originating at s and destinating at d, the corresponding charging input can be computed by a function $\Theta(\lambda_{s,d}, \Omega_{s,d})$, where $\Omega_{s,d}$ is the charging schedules for the flow of vehicles (with arrival rate $\lambda_{s,d}$), namely, $\Omega_{s,d} = \{p^v_{s',d'} \mid \forall \{s', d'\} \subseteq R_{s,d}, v \in F_{s',d'}\}$. The reason that the charging input is determined by charging schedules (i.e., Ω) is that different schedules may result in different residual electricity when vehicles leave the road system, which cause different charging requests. Thus, given $\lambda_{s,d}$ and $\Omega_{s,d}$, the function Θ computes the expected electricity charging requests resulted from the corresponding flow of vehicles on $R_{s,d}$.

Eq. (7) and (8) compute the expected arrival rate λ_v and service rate μ_v, respectively. In Eq. (8), the expected service time is computed by $E_{s,v}/r^+$. In addition, the probability that a randomly selected vehicle at v belongs to $\Gamma_{s,v}$ is $\gamma_{s,v}/\lambda_v$, so the overall expected service time is $\sum_{s \in V}[(\gamma_{s,v}/\lambda_v) \times (E_{s,v}/r^+)]$, whose reciprocal is the expected service rate.

Since each flow $\Gamma_{s,v}$ is independent of each other and has a constant service time ($E_{s,v}/r^+$), the overall service time has a Poisson characteristic [13]. Thus, the charging station can be modelled as M/M/c queuing system, and the waiting time w for M/M/c queuing is a function of arrival rate λ, service rate of each server μ and number of servers c, given by

$$w = \Psi(\lambda, \mu, c) = \frac{r^c \cdot \rho}{\lambda c!(1-\rho)^2 \cdot (\frac{cr^c}{c!(c-r)} + \sum_{n=0}^{c-1} \frac{r^n}{n!})} + \frac{1}{\mu}, \tag{13}$$

where $r = \lambda/\mu$ and $\rho = r/c$ [13]. Note that ρ is *server utilization*. By applying Eq. (13) to our system, we can get Eq. (9) with $r_v = \lambda_v/\mu_v$ and $\rho_v = r_v/c$, respectively. Finally, Eq. (10) ensures the system is steady (i.e., the queue length will not increase infinitely).

2.3 Performance Upper Bound (Lower Bound of Waiting Time)

The optimization problem defined above (called *original problem* hereafter) is hard to solve efficiently. However, to analyze the performance of our practical design, it is necessary to find a performance upper bound for solutions to the problem. In the following, we first relax the original problem to a *relaxed problem* by eliminating the vehicle flow constraints. Then, we analyze the relaxed problem and compute a performance upper for solutions to the relaxed problem, which is consequently a performance upper bound for solutions to the original problem.

2.3.1 Relaxed Problem

By eliminating the vehicle flow constraints (i.e., Eq. (4) and (5)) of the original problem, the original problem can be relaxed.

LEMMA 1. *The solution to the relaxed problem is an upper bound of the solutions to the corresponding original problem.*

This is obviously true as the relaxed problem has less constraints than the original problem. Note that since Eq. (5) is eliminated in the relaxed problem, the set $\Upsilon = \{\gamma_{s,v} \,|\, \forall(s,v) \in V^2\}$ becomes unknown, which is *independent* of the set of unknowns (i.e., $\Omega = \{p^v_{s,d} \,|\, \forall(s,d) \in V^2, v \in F_{s,d}\}$) in the original problem. Moreover, as only the electricity flow constraint (Eq. (6)) depends on Ω, we can rewrite it as follows.

$$\sum_{v \in V} \sum_{s \in V} (\gamma_{s,v} \cdot E_{s,v}) = \phi, \tag{14}$$

where $\phi \in \Phi$. Here, Φ is the set of all possible values given by $\sum_{s \in V} \sum_{d \in V} \Theta(\lambda_{s,d}, \Omega_{s,d})$ subjected to Eq. (2), (3), and (11).

2.3.2 Upper Bound Analysis

By Eq. (9) and (13), we can know that $\lambda_v \cdot w_v$ is a function of r_v (i.e., λ_v/μ_v). Let $\Psi'(r_v) = \lambda_v \cdot w_v$. Then, for any given $\phi \in \Phi$, the relaxed problem can be rewritten as follows.

Objective:

$$\text{Find } \Upsilon \text{ to minimize } \quad W = \sum_{v \in V} \Psi'(r_v), \tag{15}$$

s.t.,

$$\sum_{v \in V} \sum_{s \in V} (\gamma_{s,v} \cdot E_{s,v}) = \phi, \tag{16}$$

$$r_v = \frac{\lambda_v}{\mu_v} = \frac{\sum_{s \in V} (\gamma_{s,v} E_{s,v})}{r^+}, \tag{17}$$

$$\Psi'(r_v) = \frac{c r_v^{c+1}}{c!(c - r_v)^2 \cdot (\frac{c r_v^c}{c!(c - r_v)} + \sum_{n=0}^{c-1} \frac{r_v^n}{n!})} + r_v, \tag{18}$$

$$0 \le r_v/c < 1, \qquad \forall v \in V \tag{19}$$

Assuming ϕ is given, we can apply Lagrange multipliers to the above problem and get the following lemma.

LEMMA 2. *For a given network charging input, the objective function of the relaxed problem is minimized if the utilizations of all stations are equal. Namely, given a $\phi \in \Phi$, W is minimized, if $r_v = \phi/(|V| \cdot r^+)$ and $0 \le r_v/c < 1$, $\forall v \in V$.*

Furthermore, if all vehicles carry full electricity when they enter the road system and drain all electricity when exit, we can derive the minimum value ϕ_{min} in Φ by

$$\phi_{min} = \sum_{(s,d) \in V^2} \lambda_{s,d} \left[r^- \cdot D_{s,d} - e_{max} \right]. \tag{20}$$

Since $\Psi'(r_v)$ is monotonically increasing with respect to r_v and $\phi \ge \phi_{min}$, $\Psi'(\frac{\phi}{|V| \cdot r^+}) \ge \Psi'(\frac{\phi_{min}}{|V| \cdot r^+})$. Then, by *Lemma 1* and *Lemma 2*, we can get a performance upper bound of the original problem as follows.

THEOREM 1. *A performance upper bound of the solutions to the original problem (lower bound of charging waiting time) is given by*

$$W \ge |V| \cdot \Psi'(\frac{\phi_{min}}{|V| \cdot r^+}), \tag{21}$$

if

$$0 \le \frac{\phi_{min}}{|V| \cdot c \cdot r^+} < 1, \tag{22}$$

which is the expected minimal network utilization.

3. THE PROPOSED SCHEME

Due to the potentially huge number of highly mobile EVs, it is not practical to schedule charging in a centralized manner. Hence, we propose a distributed scheme based on local optimization.

3.1 Overview

Our design is motivated by the following observation obtained from the above theoretical analysis: *the overall waiting time is minimized if charging demands of all charging stations are balanced* (*Lemma 2*). In practice, however, it is costly or even impossible to achieve the uniform distribution in the whole system. Hence, we take a practical, distributed approach of balancing the distribution of charging demands locally in every small portions of the system so as to approximate the system-wide balanced distribution.

The basic idea is as follows. Periodically, each EV makes a *reservation* at a certain station for its next charging, or adjusts its prior reservation (note: the reservations are dynamically changeable), such that the charging schedule can result in the minimum waiting time for the EV based on the current knowledge. Meanwhile, based on the reservation information of EVs and the status information from other stations, each station periodically refines its estimation of the current and future charging waiting time at its own station and other stations, such that it can help EVs to select the best stations for their next charging to minimize their waiting time.

3.2 Protocol for Charging Scheduling

Consider an EV, which destinates at station d, was most recently charged at station s, and currently runs near station u. The vehicle interacts with station u directly and stations along the shortest route from u to d (denoted as $R_{u,d}$) indirectly, to determine the station where it should be charged next such that its overall waiting time can be minimized based on the currently information of the stations. More specifically, the protocol runs as follows.

- Requested by the EV, station u sends a reservation request packet (RREQ) to the next station on the route. The RREQ contains the following fields: (1) *id* of the vehicle, (2) current timestamp (t), (3) average speed (ν) of the vehicle, (4) maximum capacity of the battery pack (e_{max}), (5) electricity consumption rate (r^-) of the vehicle, (6) the station (s) where the vehicle was most recently charged, (7) the route ($R_{u,d}$) towards station d, (8) the minimum waiting that has been found so far denoted as w_{min}, and (9) the station v_{min} that should be selected for charging next in order to achieve w_{min}. Initially, $w_{min} = \infty$ and $v_{min} = NULL$.
- Upon receiving a RREQ, station $v \in F_{u,d} \setminus \{u\}$ computes the minimal waiting time w_{min} that the vehicle can achieve if v is chosen as the next station to charge. This is done by calling a dynamic programming-based, recursive algorithm MINWAITINGTIME($R_{v,d}$, $t + D_{u,v}/\nu$, s), where $D_{u,v}$ represents the distance between u and v. In the algorithm (as presented in Alg. 1), the waiting time at station v is firstly computed by calling algorithm GETWAITINGTIME with given arrival time and electricity request, which will be presented in detail in Alg. 4 and explained later. Secondly, each possible option for the next charging station, which is in $F_{v,d}$, is considered. For each option, the consequent waiting time is computed by call

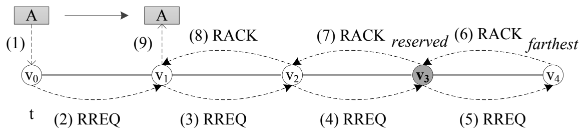


Figure 2: Charging reservation. Station v_0 launches a reservation request on behalf of vehicle A at time t. The RREQ goes through stations v_1, v_2, v_3 until the farthest station v_4. Each of these stations estimates the minimum waiting time of vehicle A if it is chosen as the next charging station. In this example, v_3 has the smallest minimum waiting time and thereby v_3 is reserved. Then, to notify v_3 and vehicle A of this new reservation as well as cancel previous reservation, v_4 sends a RACK along the reverse route. Later, when vehicle A runs near v_1, the reservation result will be informed.

the MinWaitingTime algorithm, and the minimum waiting time is returned. The returned minimum waiting time is compared to w_{min}. If it is smaller than w_{min}, w_{min} is replaced by this new waiting time and v_{min} is replaced by v, and the corresponding fields in RREQ are updated. Finally, if v is not the farthest station in $F_{u,d}$, the RREQ is forwarded to next station on the path towards station d.

Algorithm 1 MinWaitingTime($R_{v,d}$, a_v, s)

if destination $d \in F_{s,d}$ **then**
  return 0 * *return if d is reachable without charging* *\
end if
* *compute the waiting time at station v* *\
$w_v = \text{GetWaitingTime}(v, a_v, D_{s,v} \cdot r^-)$
* w^v_{min} *will record the minimum waiting time if v is selected for next charging* *\
$w^v_{min} = \infty$
for all stations $v' \in F_{v,d} \setminus \{v\}$ **do**
  * $a_{v'}$: *time to arrive* v' *\
  $a_{v'} = a_v + w_v + D_{v,v'}/\nu$
  $w'_{min} = w_v + \text{MinWaitingTime}(R_{v',d}, a_{v'}, v)$
  if $w'_{min} < w^v_{min}$ **then**
    $w^v_{min} = w'_{min}$
  end if
end for
return w^v_{min}

- The above process continues until the RREQ reaches the farthest station in $F_{u,d}$, where station v_{min} is chosen as the charging station to reserve for that vehicle. Particularly, this farthest station generates a reservation acknowledgement packet (RACK), which is forwarded along the reverse route back to v_{min}, and then to the station next to u. An example is shown in Fig. 2.
- During the forwarding of RACK, a station that was reserved previously by the vehicle removes the reservation. Station v_{min} makes a record of the reservation as follows:

$$\langle vid, sid = u, a = a_{v_{min}}, e = e_{v_{min}} \rangle,$$

where vid and sid are the ids of the vehicle and station u (called *source station* of the reservation), respectively. a is the estimated time that the vehicle arrives at v_{min}, and e is the amount of electricity that should be charged when it arrives.

- RACK is forwarded back along the reverse route until it reaches the station next to u, where RACK is buffered. Later, when the vehicle runs near that station, the reservation result will be informed.
- When a vehicle arrives at a station which has its reservation, the vehicle is charged there. After charging, the corresponding reservation record is removed.

As we can see from the above description, the success of finding a minimal waiting schedule relies on the accurate estimation of the waiting time at each individual charging station (i.e., GetWaitingTime in Alg. 4). In the following subsections, we elaborate how each station estimates the waiting time in the presence of dynamic changes of charging reservations, and how stations exchange their waiting time information with each other.

3.3 Waiting Time Estimation at a Station

For a station to estimate the waiting time for a *future* moment, it needs to know the following information regarding that future moment: (1) how many EVs are being charged, (2) how long each of the charged EVs still need to be charged, (3) how many other EVs are waiting to be charged, and (4) how much electricity is needed to charge each of these waiting EVs. To find out these information, the station can only rely on the charging reservations that have been made with it. Particularly, if all current reservations are assumed to be carried out without cancellation or modification, the QueuingTimeSim algorithm (presented at Alg. 2), which simulate the charging behaviors of the station, can be used to estimate the *queuing time* of a vehicle which arrives at some future time a.

Algorithm 2 QueuingTimeSim(a)

Data Structure:
A FIFO queue of reservations $\{(vid_i, *, a_i, e_i) | i = 0, \cdots, n-1; a_i < a\}$, where reservations are sorted according to the ascending order of EV arrival time a_i
A. At time a_i ($i = 0, \cdots, n-1$):
if reservation $(vid_i, *, a_i, e_i)$ is at the head of the queue **and** there is an idle charging outlet **then**
  dequeue the reservation
  use an idle outlet to charge vehicle vid_i for e_i/r^+, where r^+ is the charging rate of the outlet
end if
B. When finishing charging an EV:
if the queue is empty (i.e., all charging requests submitted before a have been serviced) **then**
  if $a \geq$ "current time" **then**
    output "0"; end * *There is an available outlet* *\
  else
    output "current time - a"; end * *Need to wait* *\
  end if
else
  dequeue $(vid_i, *, a_i, e_i)$ from the head of the queue and charge EV vid_i for e_i/r^+
end if

Unfortunately, reservations may be changed (i.e., cancelled or modified) and thus cannot be used directly as in the QueuingTimeSim algorithm. To address this problem, we propose a method to estimate the probability in which each reservation will be really carried out (called *success probability* of the reservation hereafter), and use the reservation information according to its success probability. The success probability is estimated based on the following ideas:

- When a reservation is made, the stability (i.e., the chance that it will not be changed or cancelled) is firstly estimated through comparing the waiting time caused by this reservation with the waiting time caused by other optional reservations. The larger is the difference, the higher is the stability.
- Furthermore, the historical data is used to model the mapping between the stability of reservations and the success probabilities. With the mapping, the success probability of a reservation can then be quantified based on both stability and the mapping between stability and success probability.

Detailed design based on the above basic ideas is as follows.

3.3.1 Reservation Stability

To quantify the stability of a reservation, we define a stability weight (denoted as δ_i) to associate with each reservation (vid_i, sid_i, a_i, e_i) made at station v_{min}. Specifically, assuming the RREQ associated with reservation $(vid_i, sid_i{=}u, a_i, e_i)$ is launched by station u on behalf of vehicle vid which was charged most recently at station s, when the vehicle reaches station u and destinates at station d, the stability weight of the reservation is calculated by the farthest station on route $F_{s,d}$ as follows:

$$\delta_i = \prod_{\forall v \in F_{s,d} \setminus (\{u \text{ and its precursors in } F_{s,d}\} \cup \{v_{min}\})} \delta_v, \quad (23)$$

where δ_v is defined as

$$\delta_v = \begin{cases} 1 - \exp\left(-\frac{w^v_{min} - w_{min}}{a_v - t}\right), & \text{if } v \text{ a precursor of } v_{min} \\ 1 - \exp\left(-\frac{w^v_{min} - w_{min}}{a_{v_{min}} - t}\right), & \text{otherwise.} \end{cases} \quad (24)$$

Here, w^v_{min} is the minimal waiting time if v is reserved for next charging, a_v is the arrival time of the vehicle at v, w_{min} is the waiting time if v_{min} is reserved for next charging, $a_{v_{min}}$ is the arrival time of the vehicle at v_{min}, and t is the time when the reservation request is issued. Intuitively, δ_v can be viewed as the probability that station v *cannot* achieve a smaller waiting time than w_{min} before the vehicle arrives at v, i.e., the chance that the current reservation at v_{min} will not be changed due to v.

The rationales behind the above definition of stability weight are as follows. On one hand, if the waiting time difference (i.e., $w^v_{min} - w_{min} > 0$) is large, then δ_v is large, indicating a small chance that station v can change the reservation within a given time duration (i.e., $a_v - t$ or $a_{v_{min}} - t$). On the other hand, if the time duration allowed for station v to make changes (i.e., $a_v - t$ or $a_{v_{min}} - t$) is long, then δ_v is small, indicating a large chance that station v can change the reservation. Since $0 < \delta_v < 1$, we treat δ_v as a probability and compute the overall weight δ_i by taking the product of δ_v for all possible v. Correspondingly, δ_i is an overall estimation of the potential impact of other stations on the reservation.

Fig. 2 shows an example, where station v_0 (i.e., $u = v_0$) launch reservation request on behalf of vehicle A at time t. Let a_i and w_i be the arrival time and minimal waiting time at station v_i, respectively, for $i = 1, \cdots, 4$. The minimal waiting time of the reserved station v_3 is the smallest (i.e., $w_{min} = w_3$ and $a_{v_{min}} = a_3$) and thereby v_3 is reserved. Hence, the stability of this reservation (recorded at v_3) is computed by $(1 - \exp(-\frac{w_1 - w_3}{a_1 - t}))(1 - \exp(-\frac{w_2 - w_3}{a_2 - t}))(1 - \exp(-\frac{w_4 - w_3}{a_3 - t}))$.

3.3.2 Historical Records of Past Reservations

Each station v keeps historical records of successful and failed reservations that have been made with it in the past. Particularly, it maintains a FIFO queue of past reservations $H^u_v = \{(t_i, \delta_i, x_i) | i = 0, \cdots, n-1\}$ for each reservation source station u. Here, each 3-tuple (t_i, δ_i, x_i) is a record of past reservation, where t_i is the time when the reservation is made, δ_i is the stability of the reservation, and $x_i = 1$ if it is a successful reservation while $x_i = 0$ if it is a failed one. Reservations with the same source station (u) are considered together due to the reason that vehicles corresponding to these reservations are expected to travel the same path leading to the reserved station and therefore potentially have similar characteristics.

To save storage space, the size of each H^u_v is regulated by two parameters: time window π and size window ω. When a new entry (t, δ, x) is added, H^u_v is purged until no entry is earlier than $t - \pi$ or the size of H^u_v is smaller than ω. In this paper, $\pi = 1$ hour and $\omega = 100$ by default.

3.3.3 Quantifying Reservation Success Probability

With the reservation history H^u_v, we define a function $P_u(\delta)$ to quantify the reservation success probability for any given pending reservation R, whose source station and stability are u and δ, respectively. Briefly, we first quantify reservation success probability of R (denoted by p_i) based on each reservation i in H^u_v. Then, to summarize probability estimations based on individual reservations, $P_u(\delta)$ is computed by moving averaging p_i of all reservations in H^u_v according to their temporal order.

Regrading an individual reservation $(t_i, \delta_i, x_i) \in H^u_v$, the success probability p_i for R is computed as follows.

- when $x_i = 1$, there exist two cases: if $\delta \geq \delta_i$, $p_i = 1$ because it is very likely that reservation R will be successful based on the observation that a past reservation succeeded with a lower stability; otherwise, $p_i = \frac{\delta - 0}{\delta_i - 0} = \frac{\delta}{\delta_i}$ because it is not certain whether reservation R will fail or succeed based on the observation that a past reservation i succeeded with a higher stability.
- Similarly, when $x_i = 0$, there also exist two cases: if $\delta \leq \delta_i$, $p_i = 0$; otherwise, $p_i = \frac{\delta - \delta_i}{1 - \delta_i}$.

The table below shows an example, where $\delta = 0.5$ for the given reservation R.

Reservation i in H^u_v	$(\delta_i, x_i, *)$	p_i
1	$(0.7, 0, *)$	0
2	$(0.4, 0, *)$	$\frac{0.5-0.4}{1-0.4} = \frac{1}{6}$
3	$(0.6, 1, *)$	$\frac{0.5-0}{0.6-0} = \frac{5}{6}$
4	$(0.3, 1, *)$	1

Then, the success probabilities computed based on individual reservations (i.e., p_i for $i = 0, \cdots, n-1$, where n is the number of reservations in H^u_v) are synthesized through *exponential moving average*. Particularly, let P_i be the reservation success probability by synthesizing the first i reservations. P_i is computed by

$$P_i = \alpha_i \cdot p_i + (1 - \alpha_i) \cdot P_{i-1}, \quad (25)$$

where $P_0 = p_0$ and α_i is the smoothing factor between 0 and 1, defined as

$$\alpha_i = 1 - \exp\left(-\frac{t_i - t_{i-1}}{t_{n-1} - t_0}\right). \quad (26)$$

Here, $t_{n-1} - t_0$ is the total time span covered by H_v^u while $t_i - t_{i-1}$ is the time difference between two sequential reservations. With moving averaging, the impact of time elapsing is considered by giving a bigger weight to recent reservations while a smaller weight to older ones. Finally, the reservation success probability of R is quantified as $P_u(\delta) = P_{n-1}$.

3.3.4 Summary

To estimate the waiting time for a future moment, we extend the QueuingTimeSim algorithm (i.e., Alg 2) by considering reservation success probability when constructing the data structure (i.e., a FIFO queue of reservations). The extended algorithm, called QueuingTimeSim*, is presented in Alg. 3.

Algorithm 3 QueuingTimeSim*(a)

Data Structure:
A FIFO queue of reservations $\{(vid_i, *, a_i, e_i) | i = 0, \cdots, n-1; a_i < a\}$, where reservations are sorted according to the ascending order of EV arrival time a_i

* *Initialization (construction of the queue)* *\

the queue is initially empty
for all vehicle vid_i that is being charged or waiting to be charged **do**
  insert $(vid_i, *, a_i, e_i)$ into the queue
end for
for all vehicle vid_i that has a pending reservation with source station u_i and stability δ_i **do**
  insert $(vid_i, *, a_i, P_{u_i}(\delta_i) \cdot e_i)$ into the queue
end for

A. the same as the **A** of QueuingTimeSim
B. the same as the **B** of QueuingTimeSim

3.4 Exchanging Waiting Time Information

To exchange the charging waiting time information with other stations, each station periodically floods a waiting time status packet (WSTA) via the Internet, containing its estimated queuing time at a certain number of future time points. That is, WSTA= $\{w_{a_i} | a_i = a_0, \cdots, a_{n-1}\}$, where w_{a_i} = QueuingTimeSim*(a_i).

As WSTA is propagated farther away from its source station, more and more waiting time information contained in the packet becomes obsolete. Consider station u receives a WSTA generated by station v at time t. Suppose the maximum speed of a vehicle is ν_{max}. Then, the waiting time information $\{w_{a_i} | a_i < t + D_{u,v}/\nu_{max}\}$ becomes obsolete for station u and it successive receivers, because no vehicle originating at u in the future can reach v before these time points. To save bandwidth, the size of WSTA can be shortened by removing the obsolete information. Consequently, the flooding process of WSTA can eventually terminate.

3.5 Getting Waiting Time at Any Station

Having the reservation information and also the waiting time information of other stations, a station u can estimate the waiting time of any vehicle at any station v by running the GetWaitingTime algorithm (as presented in Alg. 4). Here, a and e are vehicle arrival time and electricity to charge at station v, respectively.

In Alg. 4, two cases are considered. One is that station u, which is running GetWaitingTime, wants to compute its own waiting time (line 1). In this case, the QueuingTimeSim* algorithm is used. The other is that station u wants to know the waiting time of some other station $v \neq u$. In this case, if the arrival time a is later than all estimation time points provided by v, then latest estimation is used (line 6); otherwise, two bounding time points can be found and their mean is used (line 8). In addition, it is possible that station u does not have the waiting time status of v as they are too far away from each other. In some case, the information of the station that is closest to v is used instead, as shown in line 11-12.

Algorithm 4 GetWaitingTime(v, a, e) **called by station u*\\

1: **if** $v = u$ **then**
2:   **return** QueuingTimeSim*(a)+e/r_u^+
3: **else**
4:   **if** u has the waiting time information of v **then**
5:     **if** a is later than the latest estimation time point a_{n-1} of v **then**
6:       **return** $w_{a_{n-1}} + e/r_v^+$
7:     **else**
8:       **return** $\frac{w_{a_i} + w_{a_{i+1}}}{2} + e/r_v^+$, where $a_i < a \leq a_{i+1}$
9:     **end if**
10:   **else**
11:     find a station v' closest to v such that u has the waiting time information of v'
12:     GetWaitingTime(v', a, e)
13:   **end if**
14: **end if**

4. SIMULATION

To evaluate our proposed system in a large-scale network, we develop a discrete-event simulator, based on which extensive simulations have been conducted in a $300km \times 300km$ grid road network with 100 stations randomly deployed at road entrances/exits. At each entrance, vehicles are randomly generated following the Poisson process. Each generated vehicle randomly selects a destination exit and moves along the shortest path between them. The traffic pattern of each route randomly falls into one of three categories: low-traffic, moderate-traffic and high-traffic, where the average vehicle arrival rates are $\lambda/2$, λ and 2λ, respectively. In addition, vehicle speed is modelled as Normal distribution with the mean 30 m/s and the variance 2 m/s. To focus on our proposed design, the vehicles that need no additional charge to reach their destinations are ignored.

Furthermore, each vehicle follows the charging specification (electricity capacity e_{max}, range d_{max}) of one of the following five types of EVs: Wheego Whip (30 kWh, 161 km) [8], Coda Automotive (33.8 kWh, 193 km) [2], Ford Focus BEV (23 kWh, 160 km) [3], Hyundai BlueOn (16.4 kWh, 140 km) [4] and Renault Fluence Z.E. (22 kWh, 160 km) [6]. Since the capability of an EV is typically specified by its electricity capacity and range, we model the electricity consumption in our simulation as $e_{max} \cdot d_{travel}/d_{max}$, where d_{travel} is the distance travelled from the last station where the vehicle was charged, and vehicle electricity status is updated every second. For each charging station, four charging outlets are installed and each charging outlet uses the rate of 44 kW (i.e., IEC 62196 [11], an international standard for electrical connectors and charging modes) by default.

As the major purpose of our simulation is to study the impact of the size of charging input on the charging waiting time, we show the simulation results as the relations between the charging waiting time and the normalized *charging input* (not λ). Specifically, the normalized charging input is the expected minimal network utilization given by Eq (22). Namely,

$$\frac{\phi_{min}}{|V| \cdot c \cdot r^{+}}.$$

To ensure a steady system for data collection, the normalized charging input should be smaller than 1.

Unless otherwise specified, our simulation uses the settings shown in the table below.

Charging rate	44 kW	Charging input	0.5
Arrival distribution	Poisson	Average speed	30 m/s

To better evaluate our distributed charging scheduling (D-ECS), we implement two other schemes: the farthest charging scheduling (F-ECS) scheme and the greedy charging scheduling (G-ECS) scheme. In F-ECS, vehicles are always charged at the farthest station that they can reach before the next charging. In G-ECS, each station maintains the charging status information of other station by using the protocol proposed in Section 3.4 (but no reservation is made). Whenever a vehicle reaches a station, the normalized waiting time for charging at following stations (i.e., $w_v/D_{s,v}$ for all $v \in F_{s,d}$ where s is the most recent station that the vehicle was charged) is computed, and the station with the minimal normalized waiting time is chosen. A vehicle is charged when its currently chosen station has been reached. Besides, the performance of D-ECS is also compared with the performance upper bound (i.e., the lower bound of the waiting time) by using Eq. (21), which is called O-ECS in our simulation.

In the simulation, two major metrics are studied:

- *Average waiting time* (s/km): the waiting time (including both queuing and charging time) of all vehicles divided by their total travel distances.
- *Charging frequency* (#/100km): the total times that all vehicles are charged divided by their total travel distances.

Data collection starts after the simulated system becomes steady. In each run of simulation, 100,000 vehicles are sampled.

In the following, to compute and compare with the performance of O-ECS, we first let the charging rate and number of outlets of all stations be the same. Then, we randomize the charging rate and number of outlets to give more detailed comparison among the other three schemes (i.e., F-ECS, G-ECS and D-ECS).

4.1 Impact of Charging Rate

Firstly, we measure the performance of different schemes by varying charging rate, as illustrated in Fig. 3. The lowest rate (i.e., 26 kW) corresponds to the IEC 62196 standard [11] while the highest rate (i.e., 62 kW) is based on a quick charging method developed by CHAdeMO association [1]. In general, the waiting time decreases as the charging rate increases. Particularly, the waiting time of our proposed D-ECS is very close to that of O-ECS. The reason is as follows. When the charging rate is low, almost all stations are occupied. In this case, the D-ECS can balance the charging input over different stations, yielding the same effect as O-ECS. When the charging rate is high, vehicles can be quickly charged, resulting in very short waiting time. Thus, the difference between O-ECS and D-ECS is not obvious. Besides, since the vehicles that use F-ECS always choose the farthest station to charge, the farthest stations on some high-traffic routes may be very crowded, leading to very long waiting time.

To study the actual queuing time of different schemes, we also plot the average base waiting time (called *base*) that vehicles have to spend in charging (i.e., the time spent in charging only, excluding queuing time, which is estimated as ϕ_{min}/r^{+}). Take the charging rate 44 kW (i.e., IEC 62196 [11]) for example. The base waiting time is 5.02 s. The difference between the base and other schemes is as follows: 33.26 s for F-ECS, 6.84 s for G-ECS, 2.64 s for D-ECS, which correspond their respective average queuing time. As we can see that the waiting time reduction of D-ECS against F-ECS and G-ECS are about 92% and 56%, respectively, which is significant.

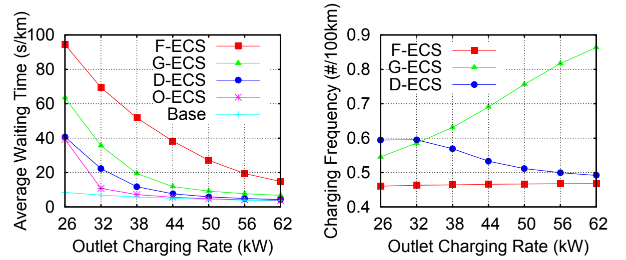


Figure 3: Impact of charging rate

As for charging frequency, F-ECS does not show any change since on a specific route the vehicles of the same type always choose the same set of stations to charge. Although F-ECS can achieve a slightly lower charging frequency, its waiting time is much longer than the other schemes. For G-ECS, the charging frequency increases as the charging rate goes up. This is because G-ECS does not take the whole route into consideration. It is likely that a vehicle conducts many times of charging in order to minimize the waiting time on each segment of the route. This could occur more often when charging rate is large, since in such case the probability that a station has zero waiting time is high. For D-ECS, the charging frequency is high at low charging rate. This is because when the charging rate is low, some charging stations may become heavily loaded. In this case, D-ECS can balance the charging workload over more stations to shorten the overall waiting time. As a side-effect, vehicles need to be charged at more stations.

4.2 Impact of Network Charging Input

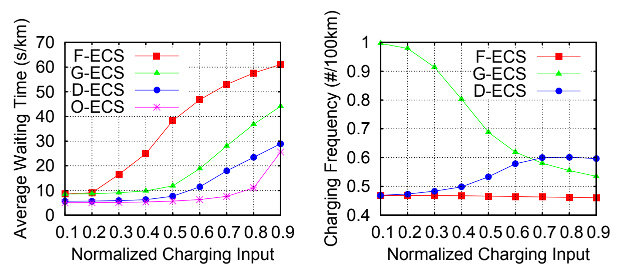


Figure 4: Impact of charging input

Fig. 4 shows the performance of different schemes as the charging input varies. In general, D-ECS outperforms F-

ECS and G-ECS, and has the performance closest to O-ECS. Note that the trend illustrated in Fig. 4 is exactly the reserve of that in Fig. 3. This is because decreasing charging rate while fixing charging input is equivalent to increasing charging input while fixing charging rate, in terms of network utilization defined by Eq. (22). And the network utilization solely determines the overall waiting time, as shown in Eq. (21). Therefore, the same arguments hold for the results in Fig. 4. Moreover, we can see that the delay is relatively large when the charging input is high. To address this issue, more charging outlets can be deployed to achieve desired performance.

4.3 Heterogeneity of Charging Stations

In order to compute and thereby compare with the performance of O-ECS, the charging rate and the number of outlets of all stations are the same in our simulation so far. In the following, we randomize the setting of each station by randomly choosing a charging rate between 26 kW and 62 kW and the number of outlets between 2 and 6. As illustrated in Fig. 5, D-ECS outperforms F-ECS and G-ECS. For example, when the network input is 0.5, the waiting time reduction of D-ECS against F-ECS and G-ECS are about 87% and 58%, respectively.

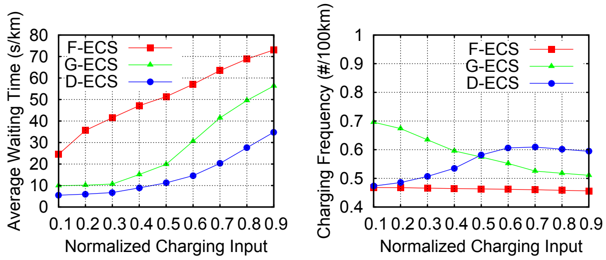


Figure 5: Heterogeneity of charging stations

4.4 Vehicle Queue Length

Fig. 6 shows the histogram of vehicle queue length (i.e., the number of vehicles being charged or waiting to be charged at a station) of all stations in the network, which is a snapshot of the status of all stations at the end of the simulation. In the figure, we can see that when D-ECS is used, about 48% of stations have zero vehicle and no station has a queue length greater than 9. In contrast, when F-ECS or G-ECS is used, there are some stations with a queue length longer than 10, which are the major contributors to long waiting time.

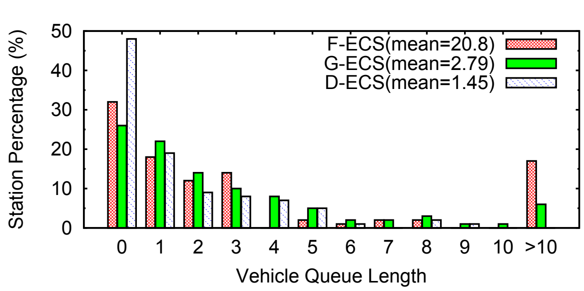


Figure 6: Vehicle queue length at different stations

4.5 Standard Deviation of Waiting Time on Different Routes

To evaluate the waiting time variance among different individual vehicles, the standard deviation of the average waiting time of the vehicles on different routes has been measured and the corresponding histogram is plotted in Fig. 7. From the results, we can see that when D-ECS is used, about 85% of all routes (i.e., about 2734 routes are counted in the simulation) have the standard deviation of average waiting time between 0 and 2 seconds (represented by the corresponding midpoint 1 second in the figure), and no route has a standard deviation greater than 6 seconds. In addition, G-ECS outperforms F-ECS, which results in the standard deviation greater than 22 seconds on as many as 28.3% of all routes.

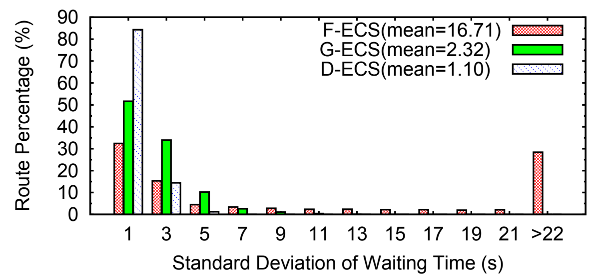


Figure 7: Standard deviation of average waiting time on different routes

4.6 Communication and Computation Overhead of D-ECS

Finally, we evaluate the communication and computation overheads of our proposed D-ECS by varying charging inputs. The communication overhead is measured as the average bandwidth consumption (Kb/s) of all stations by considering all control packets (i.e., RREQ, RACK and WSTA). The computation overhead is measured as the average number of executions of a statement in the most inner loops of Alg. 1 and Alg. 3 in one time unit (#/s) for all stations. Note that, these statements are the most basic operations in the scheme and thereby best reflect the computational cost. The results, including the average and the maximal per-station overheads, are shown in Fig. 8. As we can see, both the average and the maximum cost are very small.

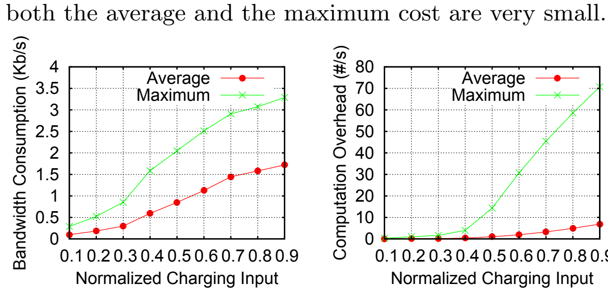


Figure 8: Communication and computation overhead

5. RELATED WORK

Recently, a number of studies have been undertaken regarding EV charging. Most of them target at saving charging cost by avoiding charging during the peak hours and making full use of the overnight demand valley. Two decentralized control strategies [17, 15] are proposed for EV charging by establishing a charging schedule that fills the overnight demand valley. In [12], a prediction-based charging scheme is presented to achieve low charging cost by dynamically predicting the market prices during the charging period and determining the appropriate time to charge. For Home Energy control, smart energy control strategies are proposed in [18] based on quadratic programming for EV charging, aiming to minimize the peak load and flatten the overall load profile. Taking electricity grid constraints into account, a novel method of planning the EV charging is presented in [20] to reduce the overloading in the electricity grid. In [19], an EV charging scheduling scheme is designed to optimize power network performance by minimizing power generation cost and power loss. In [14], three EV charging strategies, known as Dumb Charge, Smart Charge and Vehicle To Grid (V2G), are discussed, and it is found that Smart Charge is necessary to prevent large peak demand on the grid by moving EV charging to more favorable hours of energy availability and transmission grid availability.

Different from those work, our work focuses on improving the driving comfort, rather than cost, by reducing the waiting time for charging in a large-scale road network. To the best of our knowledge, this is the first work targeting on this issue. Moreover, different approaches are taken in our work. Particularly, our proposed charging scheduling attempts to optimize charging performance temporally and spatially; stations are networked and collaborated for charging scheduling; and the route information has been taken into account for performance optimization as well.

In addition, a reservation-based scheduling scheme [16] has been proposed for charging station to decide the service order of multiple charging requests, aiming to improve the satisfiability of electric vehicles. Upon receiving a charging request, the charging station decides the charging order by a rank function and then replies to all requesters with the waiting time and charge cost it can guarantee. Each requester can decide whether to charge at that station or try another station. Unlike this work, our work considers the large-scale road network, where all charging stations collaborate to minimize the charging waiting time. Moreover, our proposed protocol does not need the intervention of drivers and therefore can provide better driving comfort.

6. CONCLUSION AND FUTURE WORK

This paper aims to minimize the waiting time for EV charging in a large-scale road network. A theoretical study has been conducted to formulate and analyze the problem. Inspired by the insight discovered from theoretical analysis, a distributed scheduling protocol has been proposed for minimizing waiting time in practice. The simulation results show that the proposed design can achieve a performance close to the performance upper bound in various scenarios.

In the future, we plan to extend our work in several directions through studying the problems such as (1) how to deploy charging stations and allocate charging outlets to save electricity and lower charging cost, and (2) how to navigate EVs for minimal waiting time and travel time.

7. ACKNOWLEDGEMENT

This work is supported partly by the NSF under Grants CNS-0834593 and CNS-0831874, and by the ONR under Grant N00014-09-1-0748.

8. REFERENCES

[1] CHAdeMO Quick Charging. *www.chademo.com*.

[2] Coda Automotive. *www.codaautomotive.com*.

[3] Ford Focus BEV. *www.ford.com/electric/focuselectric/2012/*.

[4] Hyundai BlueOn. *en.wikipedia.org/wiki/Hyundai_BlueOn*.

[5] Nissan LEAF Electric Car. *www.nissanusa.com/leaf-electric-car*.

[6] Renault Fluence Z.E. *www.renault.com/en/vehicules/renault /pages/fluence-ze.aspx*.

[7] Top 10 Electric Car Makers for 2010 and 2011. *www.cleanfleetreport.com/clean-fleet-articles/top-electric-cars-2010*.

[8] Wheego electric cars. *wheego.net*.

[9] IEEE Standard 1547.3, Guide for Monitoring, Information Exchange, and Control of Distributed Resources Interconnected with Electric Power Systems. 2007.

[10] NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0. 2010.

[11] I. E. Commission. IEC 62196. *www.iec.ch*.

[12] M. Erol-Kantarci and H. T. Mouftah. Prediction-Based Charging of PHEVs from the Smart Grid with Dynamic Pricing. *1st IEEE Workshop on Smart Grid Networking Infrastructure*, 2010.

[13] R. M. Feldman and C. Valdez-Flores. *Applied Probability and Stochastic Processes*, chapter 7-8, pages 201–250. Springer, second edition, 2010.

[14] R. Freire and et al. Integration of Renewable Energy Generation with EV Charging Strategies to Optimize Grid Load Balancing. *Conference on Intelligent Transportation Systems*, 2010.

[15] L. Gan, U. Topcu, and S. Low. Optimal Decentralized Protocol for Electric Vehicle Charging. *IEEE Conference on Decision and Control*, 2011.

[16] H. J. Kim and et al. An efficient scheduling scheme on charging station for smart transportation. *International Conference on Security-Enriched Urban Computing and Smart Grid*, 2010.

[17] Z. Ma, D. Callaway, and I. Hiskens. Decentralized Charging Control for Large Populations of Plug-in Electric Vehicles. *IEEE Conference on Decision and Control*, 2010.

[18] K. Mets and et al. Optimizing Smart Energy Control Strategies for Plug-In Hybrid Electric Vehicle Charging. *Network Operations & Management Symposium Workshops*, 2010.

[19] S. Sojoudi and S. H. Low. Optimal Charging of Plug-in Hybrid Electric Vehicles in Smart Grids. *IEEE Power & Energy Society (PES) General Meeting*, 2011.

[20] O. Sundstrom and C. Binding. Planning Electric-Drive Vehicle Charging under Constrained Grid Conditions. *International Conference on Power System Technology*, 2010.