This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

Predictive Scheduling Framework for Electric Vehicles with Uncertainties of User Behaviors

Bin Wang, *Student Member, IEEE,* Yubo Wang, *Student Member, IEEE,* Hamidreza Nazaripouya, *Student Member, IEEE,* Charlie Qiu, Chi-cheng Chu and Rajit Gadh

***Abstract*—The randomness of user behaviors plays a significant role in Electric Vehicle (EV) scheduling problems, especially when the power supply for Electric Vehicle Supply Equipment (EVSE) is limited. Existing EV scheduling methods do not consider this limitation and assume charging session parameters, such as stay duration and energy demand values, are perfectly known, which is not realistic in practice. In this paper, based on real-world implementations of networked EVSEs on UCLA campus, we developed a predictive scheduling framework, including a predictive control paradigm and a kernel-based session parameter estimator. Specifically, the scheduling service periodically computes for cost-efficient solutions, considering the predicted session parameters, by the adaptive kernel-based estimator with improved estimation accuracies. We also consider the power sharing strategy of existing EVSEs and formulate the virtual load constraint to handle the future EV arrivals with unexpected energy demand. To validate the proposed framework, 20-fold cross validation is performed on the historical dataset of charging behaviors for over one-year period. The simulation results demonstrate that average unit energy cost per kWh can be reduced by 29.42% with the proposed scheduling framework and 66.71% by further integrating solar generations with the given capacity, after the initial infrastructure investment. The effectiveness of kernel-based estimator, virtual load constraint and event-based control scheme are also discussed in detail.**

***Index Terms*—Electric Vehicle Charging, User Behavior, Predictive Control, Kernel Density Estimation.**

I. Introduction

ELECTRIC Vehicles (EVs) and Plug-in Hybrid Electric Vehicles(PHEVs) are gaining more popularity in the auto-market in recent years according to the statistics published in [1], [2]. Due to the pressure from the public to reduce air pollution, 1.5 million zero emission vehicles (ZEV) will be put on roads in California by 2025, which requires the EVSEs to support 1 million ZEV by 2020[3]. As the penetration of EVs grows larger, uncoordinated charging behaviors will create new load peaks in the aggregated load curve, leading to a myriad of issues, such as power quality degradation[4], [5] and operational cost increase[6]. Furthermore, there are uncertainties (*e.g.* start time, stay duration and energy demand, etc.) within the scheduling problem for EV charging behaviors, which cannot be completely solved by deterministic problem formulations. However, coordinating numerous EV charging behaviors in real-time is a challenging task due to the following reasons: 1) lack of sharing strategy to accommodate more EVs per EVSE; 2) lack of stochastic model to handle uncertainties of EV users' behaviors, including arriving time, leaving time and energy demand; 3) lack of predictive scheduling framework, that adaptively computes for cost optimal energy allocations, considering both current and future system states.

Previous researchers have developed numerous approaches to solve the aforementioned challenges. However, to the best of authors' knowledge, none of them provides a comprehensive solution and practical validation based on the real-world implementations. [7]–[9] have defined the load from EV charging as deferrable load, which can be shifted to a different time window without compromising user's schedule requirements. EV charging load has also been considered in demand response researches [10], [11], where the problem is formulated as a convex optimization problem with the objective to minimize the overall operational cost. In addition, valley filling and load following strategies[9] are also supported in the formulation. However, the simulation-based work assumes the battery Status of Charge (SOC) values and charging session parameters, *i.e.* the arrival and departure time, energy demand, etc., are perfectly known once vehicles are plugged, which is not realistic in practical implementations. Time-varying electricity price signals are utilized for controlling the EV energy scheduling in order to achieve cost optimal solutions[11]–[16]. The validity of using Time-of-Use (TOU) prices for EV scheduling is discussed in [11], [13], [14], [17]. Maximum revenue model is defined in [15], where both regulation price and electricity price for curtailing EV charging load are defined. A social optimal pricing scheme is developed in [12] between utility and load aggregator, which is applied to a number of fleet vehicles. Vehicle-to-Grid and Vehicle-to-Building services[17], [18] are considered in EV scheduling problem. A framework for smart energy management is proposed in [19], considering time-varying load properties and user participation, etc.

To handle uncertainties in the scheduling system, including renewable generations, base load and charging demand, scheduling algorithms based on Model Predictive Control (MPC), are proposed in [7], [8], where virtual load is modelled for future EV energy demands. Markov Decision Process (MDP) and Queue Theory (QT) are utilized to handle stochastic EV arrival rate and intermittency of renewable generation in [20], [21]. [22] models EV load, time schedules and energy

This work has been sponsored in part by grants from the LADWP/DOE fund 20699 & 20686 (Smart Grid Regional Demonstration Project).

Bin Wang, Yubo Wang, Hamidreza Nazaripouya, Charlie Qiu, Chi-cheng Chu, and Rajit Gadh are with the department of mechanical engineering, University of California, Los Angeles.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

prices with Monte Carlo method. These methods cannot be applied directly to the EVSEs with multiple power sources and outlets in our study since the constraints on different outlets are not explicitly formulated. To estimate the aggregated EV charging load, [23] evaluates multiple methods on EV charging load predictions on each EVSE, including k-Nearest Neighbors(KNN), Lazy-learning Algorithm and Pattern Sequence-based Algorithm (PSA). [16] utilizes Auto-Regressive Integrated Moving Average (ARIMA) to predict aggregated EV load on UCLA campus for the next week. Estimation methods in [13], [24] assume that there are underlying stochastic models, such as Gaussian or Poisson distribution for EV charging behaviors, which is sometimes not realistic, especially for the data collected on UCLA campus. However, the practical implementation needs real-time parameter estimations for each charging session instead of aggregated load predictions.

This paper focuses on an implementable solution that considers uncertainties of user behaviors, time-varying energy prices, renewable generation integration and other practical concerns, such as the power source limitation and power sharing strategies. We first introduce a practical system architecture for data collection in detail, based on which we show the exploratory analysis for EV charging behavioral data by associating session parameters, such as start time, stay duration and energy consumption, etc., with specific users. The EV scheduling problem is formulated as a predictive convex optimization problem, which achieves a cost-efficient solution while maintaining high level of energy delivery rate with uncertainties of user behaviors. An online predictive control paradigm is developed, which adaptively estimates charging session parameters using kernel-based methods. Specifically, Gaussian kernel is utilized to model the joint probability density distributions based on the qualified historical records, which reduces the estimation deviations for both values of stay duration (h) and energy consumption (kWh). To handle future vehicle arrivals with unknown energy demand, we also model a virtual load constraint with proper relaxation strategies to reduce the level of deferability for EV load by limiting power supply for future time intervals. The effects of virtual load constraint on energy delivery rate and average unit energy cost are also studied. Finally, to minimize the number of estimations and controls, we extend the proposed Predictive Energy Scheduling Algorithm (PESA) by developing an event-based trigger scheme in Event-based Cost-optimal Scheduling Algorithm (ECSA), where re-computation is only initiated by pre-defined events. The real-world data for EV charging behaviors, which is collected from UCLA test bed for 15 months, is randomly partitioned into training and test datasets to further evaluate the overall system performance.

Compared to the preliminary work in [25], the following new contributions are added: 1) More comprehensive description and analysis for the predictive framework are provided, including the details of scheduling services and exploratory analysis for EV charging behaviors, etc. 2) Kernel-based method is proposed to adaptively estimate parameters in charging sessions by constructing joint probability density distribution for the qualified data points with improved estimation accuracy. 3) Virtual load constraint is added to handle the unexpected EV energy demand by adjusting the deferability level of EV load. Its effects on operational cost and energy delivery rate are analyzed using experiment results; 4) 20-fold cross validation is utilized as the evaluation method for our proposed scheduling framework. Total charging records are randomly divided into 20 partitions, each of which is used as test set, and the remaining 19 partitions are used as training sets.

This paper is organized as follows: Section II introduces the system architecture. Section III discusses the problem formulation for predictive EV scheduling. Finally, discussions on experiment setup and system performance are provided in Section V. Section VI concludes this paper.

II. System Architecture

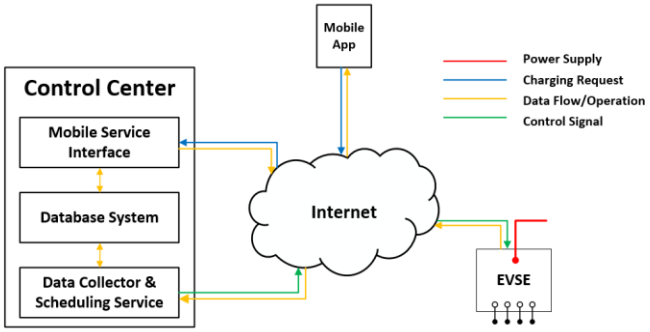

Fig. 1 System Overview

The proposed scheduling system includes three main components, *i.e.* EVSE, control center on server side and mobile application on user side, which are shown in Fig. 1. The networked EVSEs are controllable by remote commands from Internet, which can be either from the mobile applications or from scheduling services running on the server. Charging requests from EV users are transmitted from the mobile applications to EV control center, which maintains an active interface that accepts the real-time requests via HTTP secured messages. After a verification process, the requests are stored in a database system and meanwhile directed to corresponding EVSEs. In addition, the control center also maintains active scheduling services based on the real-time monitoring data retrieved by the data collector. Various scheduling algorithms with different objectives and constraints can be supported by this architecture that is built on top of the complex communication network within UCLA campus, involving multiple communication protocols, such as Zigbee, 3G, Wifi and Ethernet, etc. In addition, this architecture also supports event-based control strategies, with customized triggers from both server side and user side.

The hardware modeled in this paper is the level II EVSE developed by UCLA Smart Grid Energy Research Center (SMERC) [26], [27], which has power sharing capability, *i.e.* split the power supply from single source to multiple charging outlets within the preset range. The charging duty-cycle for each outlet, defined by SAE J1772, is linearly correlated with the charging current allocated for this outlet. In our implementation, 50% duty-cycle denotes *30A* and 10% duty-cycle denotes *6A*. The firmware in our EVSEs provides explicit interfaces to modify the duty-cycles in order to adjust the power consumption for specific outlet.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

A. Data Collection

In our previous implementation, the existing scheduling algorithm that serves the purpose to collect EV charging related data is a simple energy-sharing algorithm, *i.e.* Equal-Sharing Scheduling Algorithm (ESSA). It splits the total power supply from the power source equally by the number of connected vehicles in each scheduling loop. For simplicity, this ESSA does not require any input for user preferences but only a click on mobile application to initiate the charging session. However, significant session parameters for each user are preserved this way, such as charging start time, finish time, leave time and the energy consumption value for each session. Algorithm 1 indicates the details of ESSA when multiple vehicles are connected.

Algorithm 1: Equal Sharing Scheduling Algorithm (ESSA)

Each Loop:
- Retrieve EVSE status;
- V ← connected vehicles;
- n ←number of active vehicles V;
- Calculate average Duty-cycle: $DC \leftarrow \frac{maxDC}{n} \cdot \eta \cdot 100\%$;
- **For** $i \in [1,2,\dots,n]$
  - **If** current value drops close to 0 or unplug
    - Close charging session;
    - Record session parameters in database;
  - **Else if** different DC value detected
    - Set duty-cycle to DC for i^{th} vehicle in V;
    - Wait for current to stabilize;
  - **End**
- **End**

$maxDC$ denotes the maximum duty-cycle for each power source and η is a safety coefficient. In this algorithm, each vehicle will be assigned a percentage of circuit duty-cycle and continues charging until current drops below a pre-defined threshold. Accordingly, the start time, finish time and energy consumption are collected. Another significant parameter indicating the vehicle leave time is the plug-in status, which is also returned by the firmware in EVSEs and hereby user's stay duration can be obtained by the difference between start time and leave time.

B. Scheduling Services

The scheduling service running on server can perform schedule optimization either periodically or triggered by pre-defined events. As shown in Fig. 2, charging requests from users are submitted through the mobile application and are then stored in database as records before being directed to scheduling service, from which specific control commands are sent. Once the control action finishes, operation status is returned to users. Meanwhile, the scheduling service is able to host numerous threads, each of which can be a specific scheduling algorithm with varied optimization objective and constraints. The algorithm can be initiated periodically at the pre-set time interval, which is shown in the red box of Fig. 2. Before any optimization is made in each loop, the first action is to retrieve the real-time data and status from EVSEs, which enables the algorithm to compute the optimal schedules based on the most up-to-date system states. Another interface to initiate the scheduling service is via pre-defined events through the interface between database and scheduling service. The events are detected by monitoring the real-time data from EVSEs. Once a charging session is terminated or any status updates are detected, notifications will be sent to users through mobile applications. Note that the data and status for EVSE come through mesh networks (from EVSE to control center) and thereby communication delay exists. In addition, if the commands lead to adjustments of power consumption, it will take longer time for the circuit to stabilize. Thus, delays of several seconds based on practical experiences are expected and too frequent control (*e.g.* more than 5 times per minute) is not recommended in this system.

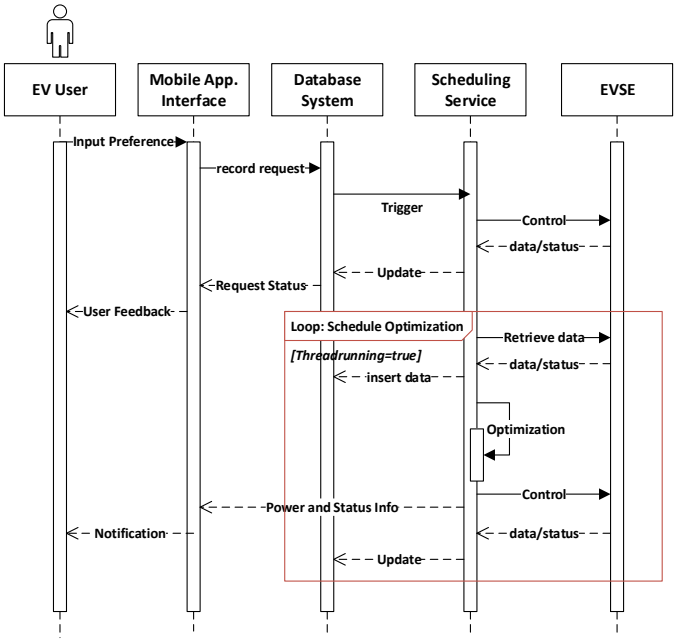

Fig. 2. Sequence chart for scheduling service

III. Predictive Scheduling Framework

In this section, we will discuss the predictive scheduling framework, which includes two main components: kernel-based parameter estimator and predictive scheduling paradigm inspired by model predictive control (MPC).

A. Kernel-based Estimation for Session Parameters

1) Tuple Construction for Session Parameters

In order to simplify the process of behavioral data modeling, a 5-tuple is created for each charging session:

$$s \triangleq (u, t_s, t_f, t_l, e)$$

where u is the unique identifier (index) for each user in our system; t_s and t_f denote start time and finish time, respectively; t_l is the leave time for each charging session; e denotes energy consumption. Note that t_l is usually later than t_f since some vehicles get fully charged before being un-plugged by users. Stay duration d can be obtained by $d = t_l - t_s$ for each session. This sequence is illustrated in Fig. 4, where t_0 is the initial time for each day.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

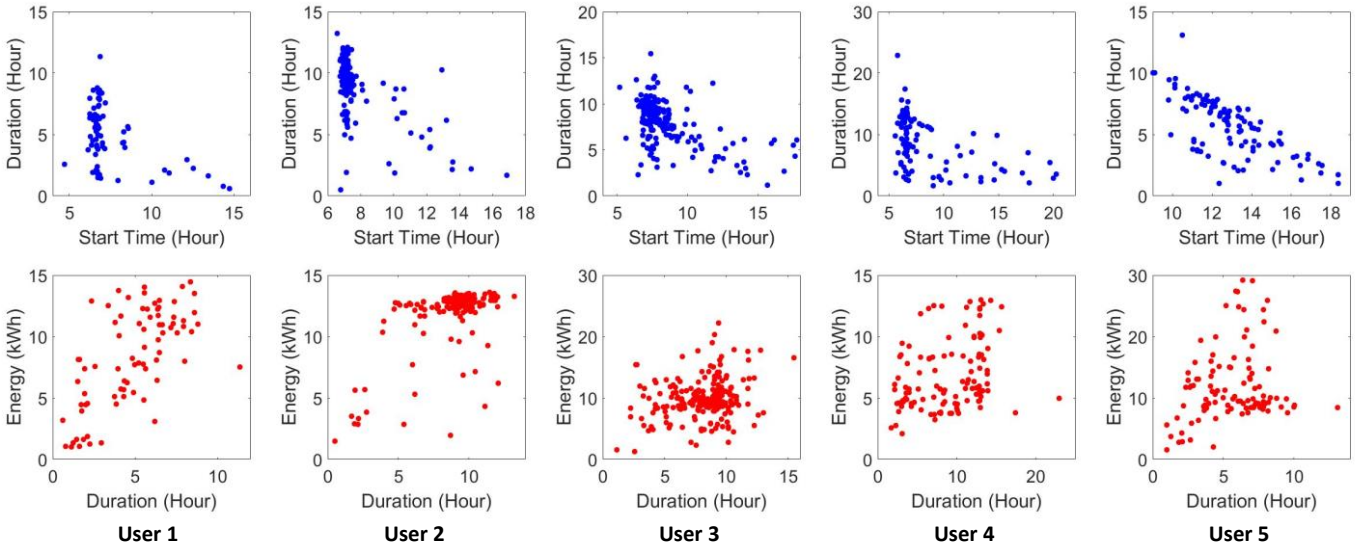

Fig. 3 Typical user behaviors

Session parameters are significant for scheduling algorithms to determine optimal solutions. Once a charging session is initiated by a specific user, the estimated stay duration and energy consumption values are needed for the scheduling service to compute for energy allocation schedules. Thus, the purpose of estimation algorithm is to obtain the estimated values of stay duration $\hat{d}$ and energy consumption $\hat{e}$, given the start time t_s, user's index u and historical records.

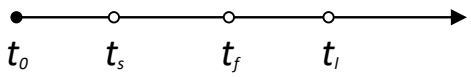

Fig. 4. Charging session parameters

2) *Exploratory Data Analysis*

Exploratory data analysis is performed for each user to exploit the distributions of session parameters and latent relations between them, *i.e.* start time vs. stay duration and stay duration vs. energy consumption. The charts in Fig. 3 show session parameters for typical users in our system. From the plots of start time vs. stay duration (upper level), one can tell that users tend to have a relatively stable start time in the morning, such as user 1, 2 and 4. However, the plots also show a few deviations of start time, *i.e.* the tails, indicating that users may plug in their vehicles later than usual. The tail effect is different among varied users. For instance, tail effect of user 3 is much heavier than that of user 1. Another observation is that in the tail part of the plots, the later users plug in their vehicles, the shorter their stay durations will be. This makes sense since most EV drivers in university campus tend to have fixed departure time. On the other hand, the duration vs. energy plots(lower level) cannot show apparent relations between users' energy consumption and their stay durations even though for user 1 and user 5, one can find that the longer user stay plugged the more energy will be consumed. However, as the duration grows larger, the variance of energy consumption values also increases. For user 2, 3 and 4, no obvious pattern can be identified from the plots. These plots are only 5 samples from more than 100 users in our system, whose parameter distributions are far more diverse. Therefore, it is difficult to develop a comprehensive parametric model to describe the behavioral data for all users, which leads us to develop nonparametric model-free method.

3) *Kernel Density Estimator*

Nonparametric estimation method, such as kernel density estimation, does not require explicit parametric model to fit the data. Discrete kernel estimator with tutorial is discussed in [28]. Given the parameters already known (*e.g.* start time t_s or estimated stay duration $\hat{d}$), the objective here is to estimate the unknown session parameters (*e.g.* stay duration d and energy consumption e) with kernel methods. As discussed above, there exists latent relationship between session parameters, shown by plots in Fig. 3, so that a bi-variate kernel density estimator is formulated. One can obtain the joint probability distribution of start time vs. stay duration or stay duration vs. energy consumption, respectively. Suppose $p(x, y)$ is joint probability for one of aforementioned chart, point estimation of random variable X, *i.e.* $\hat{x}$, can be calculated by the marginalization operation in equation (1):

$$\hat{x} = E[X] = \int_X x \cdot p(x)dx = \int_X x \cdot \int_Y p(x,y)dy \cdot dx \qquad (1)$$

As an example, to estimate stay duration d for a specific user, joint distribution of start time t_s vs. duration d is utilized and univariate distribution for d is calculated by:

$$p(d) = \int_{t_S} p(t_S, d)dt_S = \int_{\underline{t_s}}^{\overline{t_s}} p(t_S, d)dt_S \qquad (2)$$

where $\overline{t_s}$ is the upper bound of start time, denoted by $\overline{t_s} = t_s + \Delta t$. Δt is a tunable parameter, denoting the tolerance bandwidth for start time selections. Similarly, lower bound start time is $\underline{t_s} = t_s - \Delta t$. The assumption for this modeling is the latent dependence of stay duration on start time and the consistency of users' behaviors, i.e. user's future behaviors resemble her historical charging records. Thus, the similar sessions with start time falling within the tolerance range of the start time t_s for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

current session are used as the base dataset to construct the kernelized joint distribution. For instance, if start time for current charging session is 8:00 AM and the tolerance interval is set to 1 hour, then historical sessions for current user with start time between 7:00 AM and 9:00 AM will be extracted. Thus, the following constraints have to be satisfied for each qualified tuple s in qualified tuple set S:

$$\overline{t_s} \geq s.t_s \geq \underline{t_s} \tag{3}$$

$$s.d \geq t - s.t_s \tag{4}$$

$$s.e \geq e_t^c \tag{5}$$

$$s.u = u \tag{6}$$

where $s.t_s$ denotes the start time t_s of tuple s; t is the current time when the estimation function is called and the e_t^c is energy already consumed by the time t. u denotes the user index for current user. These additional constraints serve to refine the selection of historical sessions. Similarly, energy consumption value can also be estimated by the distribution of stay duration vs. energy consumption, given estimated duration $\hat{d}$ and tolerance bandwidth Δt. However, as charging session proceeds, the qualified dataset extracted from historical records by equation $(3)-(6)$ is quite different, which leads to the diversity of the joint distributions. The joint probability can be obtained as follows:

$$p_{KDE}(x) = \frac{1}{N}\sum_{i=1}^{N} K(x, B) \tag{7}$$

where B denotes the base dataset extracted for modeling and N is the total number of data points in B. Thus, $B \triangleq \{b_1, b_2, \cdots, b_N\}$ and each data point has D dimensions, i.e. $b \in \mathbb{R}^D$. For instance, $D = 2$ if we model a bi-variate distribution, such as start time vs. stay duration. $K(x, B)$ is the kernel function that is used to model the weight of each data point x. We use Gaussian kernel for a continuous probability density, *i.e.*:

$$K(x, B) = \frac{1}{\prod_{j=1}^{D} h_j} \cdot \prod_{j=1}^{D} K_j\left(\frac{x - b_i}{h_j}\right) \tag{8}$$

where h_j is the bandwidth for *j-th* dimension of the data point; K_j is the kernel function for *j-th* dimension with the following form, where g is a random variable:

$$K_j(g) = (2\pi)^{-\frac{1}{2}} \cdot e^{-\frac{1}{2}g^2} \tag{9}$$

Note that kernel modeling in equation $(7)-(9)$ does not imply that variables are independent of each other, since the multiplication operation is before the summation operation. This modeling process is performed every loop when the estimation is needed. As an illustrative example, sample probability distributions for a user are displayed in Fig. 5 and Fig. 6. The peak of joint probability distribution represents the region with highest probability, i.e. the highest density of qualified data points.

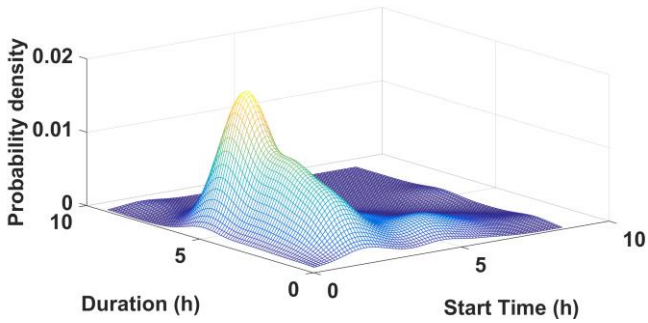

Fig. 5 Joint probability for start time and stay duration

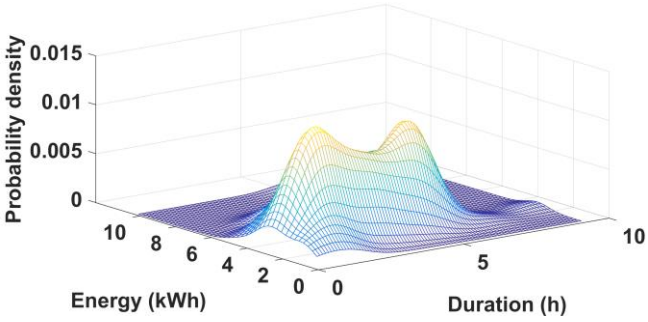

Fig. 6 Joint probability for stay duration and energy consumption

Following the above steps, session parameters can be estimated adaptively. The complete steps for parameter estimation are summarized in algorithm 2:

Algorithm 2: Kernel based Parameter Estimator

Input: current session ***s***, current time ***t***
Output: $\hat{e}$, $\hat{d}$
Extract historical tuples with the constraints $(3)-(6)$;
If number of tuples found > threshold number
  Calculate Gaussian kernel by $(7)-(9)$;
  Calculate estimated $\hat{\mathrm{d}}$ and $\hat{\mathrm{e}}$ by (1), (2);
  $\hat{e} \leftarrow max\{\hat{e}, e_t^c + \Delta e\}$;
  $\hat{d} \leftarrow max\{\hat{d}, t - s.t_s + \Delta d\}$;
Else
  $\hat{e} \leftarrow e + \Delta e$
  $\hat{d} \leftarrow d + \Delta d$
End

Note that, there will be less qualified tuples as the charging session proceeds. Specifically, a new user may also have less similar historical records. In this case, we grant priority to these users by updating the estimated energy demand and stay durations in each loop with preset values. For example, estimation algorithm may assume the energy demand is about 2 kWh for a new user within the next half an hour. In cases where the estimated stay duration $\hat{d}$ is smaller than modified current value by time t, which is denoted by $t - s.t_s + \Delta d$, the *max* operation is added to prevent the early terminations of certain charging sessions with less estimation accuracy, which may further lead to lower energy delivery rate. Accordingly, the modified estimated leave time is always later than the current time t. Similar operations are made for the estimated energy consumption $\hat{e}$, so that the charging sessions will not be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

terminated pre-maturely. Note that the estimation is computed every time if the scheduling optimization service is set to run periodically at a fixed time interval. However, under the event-based control paradigm, where computation is only triggered by pre-defined events, a considerable number of computations can be reduced, which we will discuss in later sections.

B. Problem Formulation

1) EVSE Model

Since the EVSE in this paper can be equipped with multiple power sources and certain outlets may share the same power source, one more constraint is added to ensure that total power drawn from each power source cannot exceed its upper limit. In addition, power consumption rate at each outlet cannot exceed the maximum value for that power source:

$$0 \le r_n^k(t) \le r_k^{max} \cdot \eta, \qquad \forall t \in [t_n^s, t_n^s + \hat{d}_n] \tag{10}$$

where the charging rate at time t for vehicle n connected to power source k, is defined as $r_n^k(t)$. r_k^{max} denotes the limitation of power source k and η is the safety coefficient for each power source. t_n^s is the start time for vehicle n. Let $k \triangleq \{1,2,\dots,K\}$ denote the order of power source number in one EVSE. For each power source k in the EVSE, we have:

$$0 \le \sum_{n \in N_k} r_n^k(t) \le r_k^{max} \cdot \eta, \qquad \forall t \in [t_n^s, t_n^s + \hat{d}_n] \tag{11}$$

where N_k denotes active charging sessions for power source k.

2) Battery Model

As discussed above, each charging session for user n can be described by the aforementioned parameters defined in the tuple $s_n \triangleq (u_n, t_n^s, t_n^f, t_n^l, e_n)$. Thus, the ideal scenario is that scheduling algorithm allocates more energy than expected, i.e. $e_n > \hat{e}_n$, and meanwhile below the battery capacity e_B, before user's leave time t_n^l, which is represented by $t_n^s + \hat{d}_n$. The actual energy consumption e_n increases as the charging process goes on.

$$e_n(t) = e_n(t - \Delta t) + r_n(t) \cdot \Delta t, \ \ \forall t \in [t_n^s, t_n^s + \hat{d}_n] \tag{12}$$

$$e_B \ge e_n\left(t_n^s + \hat{d}_n\right) \ge \hat{e}_n \tag{13}$$

3) Virtual Load Constraint

Since the hardware we are modeling in this paper has the power sharing circuit design, it means that the charging schedules for vehicles connected to the same power source will be interrelated with each other. Another concern with the EV scheduling problem with random user behavior is that the scheduling process is not quite robust if the power supply is limited. In other words, the pre-computed schedules may be not valid if unexpected additional energy demands are requested by new coming users for the same power source. For instance, if the scheduling algorithm arranges the energy consumptions in several hours later without considering the future new demand, it is highly possible that the limited power source fails to deliver enough energy to satisfy the unexpected charging demand because the total power consumption violates the power capacity constraints.

We propose a method based on virtual load constraint to solve this issue, by adding constraints on power supply for a future time window. Intuitively, if the future power supply is further limited and the scheduling algorithm has to arrange earlier time intervals for vehicle charging. Thus, the deferability level of EV load is reduced and more energy consumption will be shifted forward to avoid infeasible solutions. The detailed mathematical formulation is in equation (14):

$$\sum_{\tau=t+\Delta H}^{\tau=T} \sum_{n \in N_k} r_n^k(\tau) < \lambda \cdot \sum_{\tau=t+\Delta H}^{\tau=T} r_k^{max}, \forall \tau \in [t + \Delta H, T] \tag{14}$$

This constraint is to limit the total power consumption for a specific EVSE by virtual load constraint coefficient λ. Note this limit is only valid for the time range from $t + \Delta H$ to the end time T. $\lambda = 1$ is equivalent to remove this constraint and $\lambda = 0$ is actually only allowing power consumption from time t to $t + \Delta H$. Thus, for different scenarios, λ value and ΔH should be tuned to achieve better overall scheduling performance. The effects of λ parameter are discussed in later section.

4) Receding Horizon Control

We formulate EV charging scheduling problem as a predictive control problem, which can be applied to a variety of objectives, as long as the problem can be formulated as a convex optimization problem. At each time interval, the algorithm hosted by scheduling service will call optimization program to compute an optimal EV charging schedule for the remaining time intervals, considering the estimated session parameters and energy consumption values for all active charging sessions. With the solar generation integration, EV power consumption as deferrable load, will be shifted to the time interval with abundant solar generations. On the other hand, when solar generations cannot support the total EV charging load, algorithm will choose the time range with lower energy prices for EV charging. The optimization objective is formulated as follows:

$$\min_{r_n^k(\tau), \tau \in [t,T]} \sum_{\tau=t}^{\tau=T} P(\tau) \cdot max\left(\sum_{k \in K} \sum_{n \in N_k} r_n^k(\tau) - PV(\tau)\right), 0) \tag{15}$$

$$s.t.\ (10) - (14)$$

τ denotes to the current time when the scheduling algorithm is called and T is the maximum time step for the scheduling horizon. $PV(\tau)$ denotes the forecast solar generation at time τ from the installed panels. $P(\tau)$ is the electricity price at time τ. The max operation actually models the integration of solar generation by comparing the total charging load with the solar output value for each future time interval. Inspired by the model predictive control paradigm, the optimal energy consumption schedules for all the remaining time intervals are computed, however, only the first element, $r_n^k(t)$, is implemented to control EVSE. As the scheduling proceeds, the scheduling horizon recedes to the maximum time step T, indicated by the name receding horizon control. The complete charging control algorithm is summarized in the following Algorithm 3.

Note that this control paradigm requires the scheduling service to be initiated every time step, which leads to the successive

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

operations for data retrieval, parameter estimation and optimization. In cases when the computing resources are limited or quality of communication network is not reliable, failure to update charging schedules may happen. Therefore, to overcome this drawback, we propose an event-based scheduling paradigm that minimizes the number of charging session controls.

Algorithm 3: Predictive EV Scheduling Algorithm(PESA)

Generate price data;
Retrieve forecast solar data;
$\tau = t_0$;
Do
  For each power source $k \in K$:
    Terminate charging sessions whose leave time $t_l \leq \tau$;
    For each vehicle $n \in N_k$:
      Estimate stay duration $\hat{d}_n$ and energy consumption $\hat{e}_n$ for vehicle n, according to **Algorithm 2**;
    End
    Solve problem (15) ,subject to (10) − (14);
    If solution infeasible:
      Relax constraint (14), and set $\eta = 1$ in equation (10) and (11);
    End
    For each vehicle $n \in N_k$:
      Implement $r_n^k(\tau)$;
    End
  End
  $\tau = \tau + \triangle t$;
While $\tau \leq T$

5) Event Trigger Scheme

In most cases, the continuous estimation of session parameters does not have large variations, which means the schedules obtained previously are still valid under current conditions and the re-computation is not necessary. Under the proposed event-based control paradigm, the scheduling services only initiate when the pre-defined events are detected, instead of being computed every time interval. The pre-defined events should represent obvious deviations of system states. Therefore, the do-while structure in Algorithm 3 needs to be updated with the event trigger structure based on the real-time monitoring of charging sessions and the following events are defined.

Event 1: New vehicle arrives with charging request;

Event 2: Vehicle leaves from EVSEs or terminates charging;

Event 3: Energy already consumed exceeds the estimated one, which is believed as an abnormal behavior and we infer that this user needs more energy than consumed;

Event 4: Leave time exceeds the estimated one, which might indicate the extended stay duration for this user;

Event 5: New estimated session parameters deviates the original estimations by a pre-defined value.

IV. Results and Discussion

A. Experiment setup

To evaluate the performance of our proposed scheduling framework, charging sessions of real-world users on UCLA campus are utilized to set up the simulation experiments. The dataset includes UCLA experiment data from August. 2013 to Mar. 2015. 20-fold cross validation, discussed in [29], section 5.3.3, is utilized to justify the performance of the proposed scheduling framework. Specifically, the total charging records of 588 days are randomly divided into 20 partitions, *i.e.* approximate 30 days in each partition, and the simulation will run for each partition as the test set, using the remaining 19 partitions as training datasets. The details of this method are shown in Fig. 7. Training sets provide the historical records for all users as basis for session parameter estimation, while test set is used to evaluate the scheduling performances, in terms of energy delivery rate, cost performance, etc. The properties for the datasets are displayed in Table 1.

Table 1. Dataset Properties

	Training Set	Test Set
Number of Partitions	19	1
Number of Days	558	30
Number of Sessions	≈4400	≈200
Number of Users	79	79

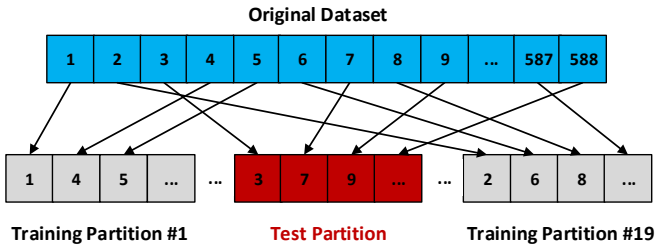

Fig. 7. 20-fold cross validation

TOU price list is generated based on the wholesale price signals from California Independent System Operator (CAISO) [30]. The original prices are modified with additional values for certain hours during the day to simulate the retail electricity prices in distribution networks, which are displayed in Fig. 8.

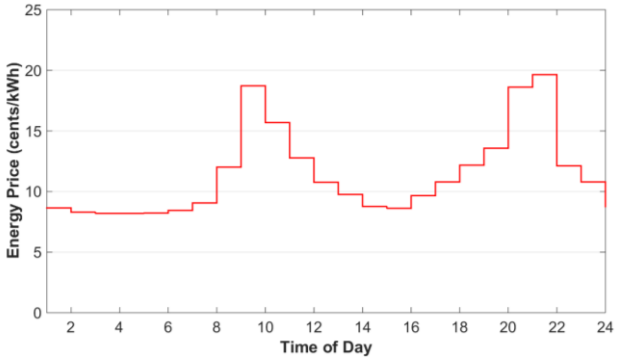

Fig. 8 Energy Price Used for Simulation

The solar generation data we used in simulation is from solar integration project [31], [32] on UCLA campus. We assume that each EVSE is equipped with 10 solar panels. Since the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

focus of this paper is on the uncertainty of user behaviors, details of solar prediction algorithm are not within the scope of this paper. We simply apply the smoothed solar curve as our forecast solar generation data, which is shown in *9*.

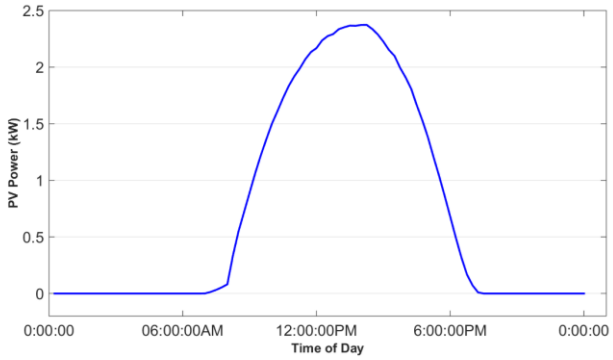

Fig. 9. Sample Solar Generation Data

Note that the time interval Δt for scheduling algorithm is set to 15 min. For each dimension, bandwidth h in kernel based estimator, i.e. equation (8), is set to $1.06\sigma \cdot N^{-\frac{1}{5}}$ according to [28], where σ is standard deviation of the values in that dimension. Adjustable values in session parameter estimator, i.e. Δd and Δe in Algorithm 2, are set to 0.5 hour and 2 kWh, respectively. For virtual load constraint in equation (14), horizontal length ΔH is set to 3 hours. The EVSE picked for simulation has 2 power sources and each one has the maximum output 6.6 kW. The safety coefficient η in equation (11) is set to 0.7. The virtual load constraint λ is set to 0.3. The package in [33] is employed to solve the schedule optimization problem.

B. Cost Saving and Load Shifting Effects

Since the primary objective of the scheduling framework is to optimize the overall cost performance for providing charging services and satisfy the charging demand from EV users, we randomly pick one day for the single day simulation, with the following records shown in Table 2.

Table 2. Charging Records on 17th, Marth, 2015

No.	User Index	Start Time	Duration (h)	Energy Demand (kWh)
1	CE1*	06:10:12	9.33	8.561
2	F42*	06:42:33	2.02	4.468
3	BFE*	07:07:44	6.87	12.207
4	155*	07:17:24	9.92	9.185
5	9CA*	14:08:58	7.3	6.154
6	8D5*	15:30:23	4.2	11.11
7	2E7*	18:31:56	1.05	5.722

The scheduling results from the original ESSA and our proposed PESA (virtual load constraint $\lambda = 0.3$) are shown in Fig. 10. The blue dot curve denotes the original EV load created by the charging sessions in Table 2, while the red curve is the new EV power consumption schedule generated by PESA. From the figure, one can see that a large portion of the load is shifted from early morning to early afternoon when there is abundant solar generation and the energy prices are lower, which can be found in Fig. 8 and 9. Another interesting phenomena is the solar generation following effects of this algorithm. Since the local solar generation can be utilized as alternative power source instead of purchasing electricity from grid, the allocation of EV charging energy tends to follow the curve of solar generation, as is shown in Fig. 10. Thus, due to the load shifting effects, the total energy cost by PESA has been reduced.

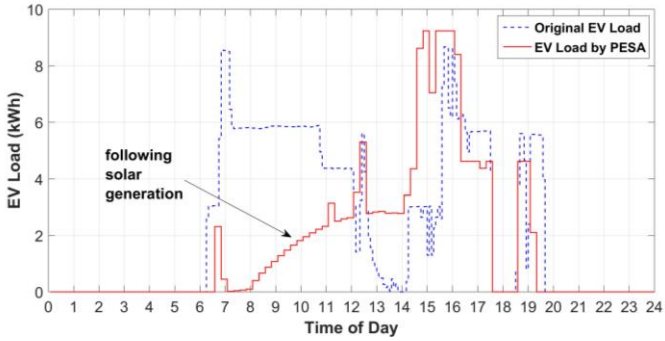

Fig. 10. EV Load Scheduling Results

However, it should be noted that the total delivered energy by the new PESA algorithm is not as much as the originally delivered energy, *i.e.* the area under the red curve is less than that under blue dot one. A close investigation reveals that this issue is caused by the uncertainty of session parameter estimation. In other words, there exist certain users in this EVSE who leave earlier than their estimated leave time so that not enough energy is allocated to their EVs. In the single day test, the energy delivered by PESA is 51.6 kWh, which is 10.12% less than the original value. On the other hand, the average unit energy cost (¢/kWh) is originally 11.23 ¢/kWh without optimization and solar integration, and it is then reduced to 5.72 ¢/kWh by PESA.

Thus, we define another criteria to evaluate the robustness of the scheduling framework over all the test samples in each partition, *i.e.* Average Schedule Error Rate (ASER), whose mathematical form is:

$$ASER = \frac{1}{M} \cdot \sum_{m=1}^{M} \left\{ \frac{1}{N_m} \cdot \sum_{i=1}^{N_m} \frac{e_i^m - min(e_i^{m,c}, e_i^m)}{e_i^m} \right\} \cdot 100\% \quad (16)$$

where e_i^m is the actual energy consumption from ESSA for *i-th* charging session on *m-th* test day of a particular partition in training sets. $e_i^{m,c}$ is the corresponding energy consumption from PESA. N_m here denotes the number of charging sessions in *m-th* test day. *M* is the total number of test days in *m-th* partition. Smaller ASER value represents a more robust solution with higher EV energy delivery rate, while the larger one indicates higher probability of failures to provide enough energy to EVs due to their uncertain charging behaviors. For a particular charging session, it is possible for the system to provide either more or less energy, *i.e.* $e_i^{m,c}$, than actual consumption value e_i^m, however, only the cases where $e_i^{m,c} < e_i^m$, are defined as schedule errors, which should be avoided or minimized.

To validate overall performance, the average unit energy cost and ASER values across all test partitions are computed and shown in Fig. 11 and Fig. 12, respectively. The scheduling results demonstrate that the PESA is more effective in cost

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

optimization compared to the original ESSA. With the solar integration and time-varying energy prices, PESA will adaptively minimize the total operational cost by searching for the optimal time ranges and charging power for each vehicle. The average reduction of unit energy cost across all partitions reaches 29.42% (blue vs. yellow bars) and it can be further improved to 66.71% (blue vs. green bars) by integrating renewable generations with EVSEs. PESA can also outperform ESSA when both are with solar integration (green vs. red bars). In our experiment, the maximum solar output is around 3 kW so the cost saving performance can be further optimized by increasing the capacity of solar integration, especially when the energy demand is high on the test days. Besides, the ASER values for all partitions are also computed to evaluate the energy delivery rate. The maximum ASER value, in Fig. 12, is approximately 12%, indicating that the energy delivery rates are acceptable across all test samples. Note that, ASER values can be improved by tuning the virtual load constraint (λ), which is discussed in later sections.

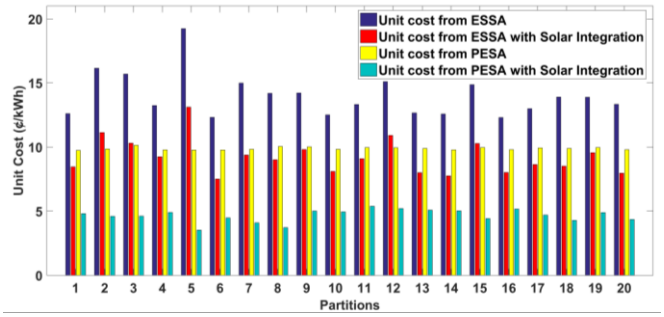

Fig. 11. Average unit energy cost on partitions

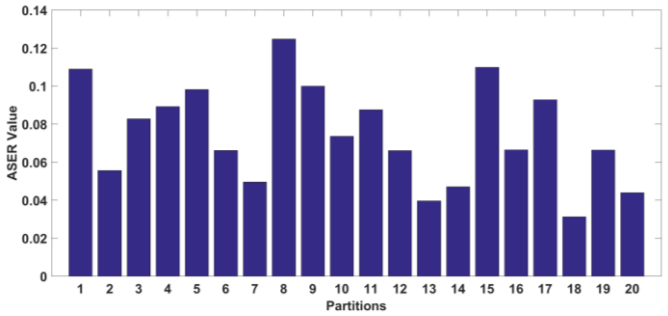

Fig. 12. ASER values for partitions

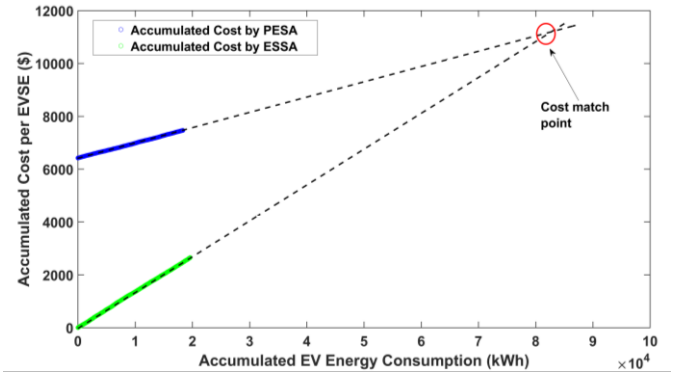

Fig. 13. Accumulated energy consumption and cost

Considering the initial investment on the solar Photovoltaics (PV) infrastructure, including fees of installation and maintenance, etc., the overall operational costs by one EVSE with and without solar installation, are visualized in Fig. 16, respectively, by plotting the accumulated values of energy consumption and operational cost by PESA and ESSA. According to [34], the solar installation and maintenance cost in California in 2015 Q1 is roughly 2.14 $/W, and lifetime of service is longer than 30 years. Therefore, it is estimated that after 82000 kWh energy delivery (approximate 6.6 years), the proposed EVSE with solar integration will provide more benefits than the traditional solution. As the cost of solar panel drops, the time it takes to reach the cost match point in Fig. 13 is becoming shorter.

C. Virtual Load Constraint on Scheduling Performance

Since the energy delivery deviations are caused by users' early departures which cannot be estimated with 100% accuracy, here we demonstrate the effects of virtual load factor on improving the ASER values. Intuitively, constraints on the future power supply renders the algorithm to allocate more energy as soon as possible in order to avoid infeasible schedules, *i.e.* the power source cannot provide enough energy to match the total demands from active charging sessions after ΔH. For one test day, the ASER value and the total operational cost are computed with different λ values and the results are shown in Fig. 14. As the value of constraint factor λ goes smaller, tighter restrictions after ΔH are applied on power sources, the algorithm tends to shift as much EV energy consumption as possible to time intervals before time $t + \Delta H$, so that energy delivery rate is improved even though users may leave unexpectedly earlier. However, as more energy is consumed in earlier non-preferable time ranges with higher prices or less solar generation, which is forced by virtual load constraints, the solution becomes less optimal. Thus, the total energy cost increases as λ goes smaller. Due to the uncertainties of user behaviors, this is a trade-off one has to consider. For example, in scenarios where requirements on schedule error rates are restrict, a smaller λ is minimize the ASER value down to 5%, however, it will not achieve the best overall cost.

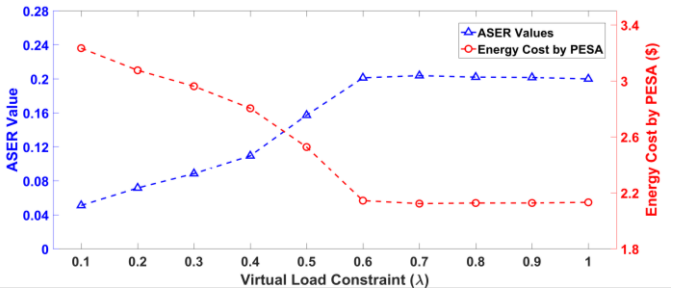

Fig. 14. Virtual load constraint factor effect

D. Estimation Accuracy

In preliminary work [25], a simpler estimator was utilized to estimate session parameters, including stay duration and energy consumption. The original solution is a simple mean estimator, which calculates the mean value across all qualified sessions extracted by $(6) - (9)$. At each time interval when the scheduling algorithm is initiated, the estimated values from

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2617314, IEEE Internet of Things Journal

both the simple mean estimator and kernel-based estimator are recorded for performance assessment. Due to the variety of session parameters, the number of estimations for each charging session may be varied so that we define another metric, *i.e.* sample estimation deviation to evaluate the overall estimation accuracy.

$$Dev_m = \sqrt{\frac{1}{N_m} \cdot \sum_{i=1}^{N_m} \frac{1}{L_n} \cdot \sum_{l=1}^{L_n} (v_l^i - v_T^i)^2} \qquad (17)$$

where L_n is the total number of estimations that belongs to *i-th* charging session in the *m-th* partition. Note there must be at least one estimation for each charging session. N_m denotes the total number of charging sessions on partition m. v_l^i is the *l-th* estimated value of *i-th* session and v_T^i is the actual parameter, *i.e.* true values for stay duration or energy consumption.

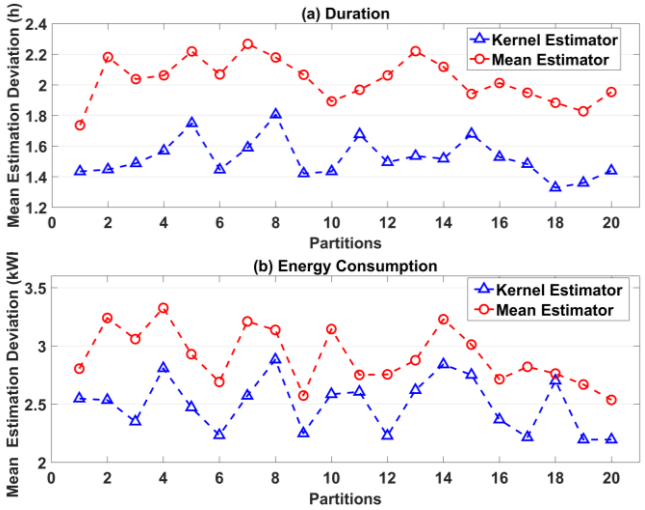

Fig. 15 Estimation deviation

According to equation (17), the estimation deviations for stay duration and energy consumption are both displayed in Fig. 15. The performance of kernel-based estimator is better than the mean estimator for both stay duration and energy consumption, with smaller deviation values. The advantage of Gaussian kernel estimator is the smoothing effect across all the variable space, which does not require a specific model. The averaged estimation deviation for stay duration by kernel-based estimator is 1.52 h, which is 26.05% less than that of mean estimator, while for energy consumption, the deviation value is reduced from 2.91 kWh to 2.50 kWh by 14.22%. The effectiveness of kernel-based estimator is demonstrated thereby.

E. Performance Evaluation for Event-based Scheduling

Event-based EV scheduling paradigm is first introduced in [25], which is believed to reduce the number of controls and computer resources, while maintaining system performances. Event-based Cost-optimal Scheduling Algorithm (ECSA) is compared with Predictive Energy Scheduling Algorithm(PESA) in terms of ASER values and unit energy cost. The only difference between these two algorithms is that ECSA is initiated by the pre-defined events while PESA runs periodically at fixed time interval. As Δt is set to 15 minutes, the results are compared in Fig. 16 and Fig. 17.For all partitions, PESA has better ASER values, which indicates the better performance on energy delivery rate, than ECSA, which fails to capture the most up-to-date system states by skipping over certain time steps and assuming the previous estimations are still right. The average ASER values across all partitions for PESA and ECSA are 7.5% and 11.65%, respectively, shown in Fig. 16. Interestingly, even though ECSA has a little worse ASER values, it has comparable average unit energy cost at 4.81 ¢/kWh, which is only ¢0.03 higher than that by PESA. Note that the maximum ASER value by ECSA is still less than 15%. Therefore, ECSA can serve as a cost-efficient solution with acceptable performance, if the requirements on energy delivery rate are not too strict.

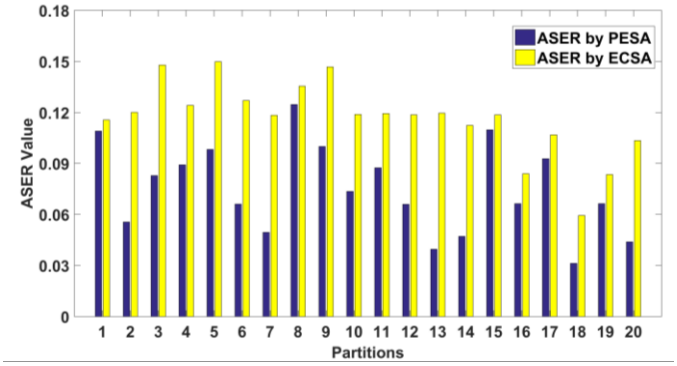

Fig. 16 ASER values for PESA and ECSA

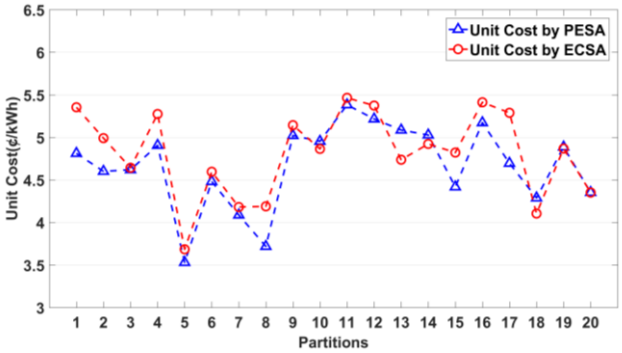

Fig. 17. Unit cost for PESA and ECSA

V. Conclusion

In this paper, we propose a predictive scheduling framework which takes into account the uncertainties of EV user behaviors. Specifically, Gaussian kernel estimator is designed to dynamically estimate the charging session parameters with improved estimation accuracies. In addition, virtual load constraint is also formulated to handle the unexpected EV energy demand arriving in the near future. Real-world data on UCLA campus is utilized for the cross validation of the proposed framework to demonstrate the improved cost performance and EV energy delivery rate.

References

[1] "Monthly Plug-In Sales Scorecard." [Online]. Available: http://insideevs.com/monthly-plug-in-sales-scorecard/. [Accessed: 03-Jan-2016].

[2] "Electric Vehicles Initiative (EVI) | Clean Energy Ministerial." [Online]. Available:

http://cleanenergyministerial.org/Our-Work/Initiatives/Electric-Vehicles. [Accessed: 03-Jan-2016].

[3] "California Statewide Plug-In Electric Vehicle Infrastructure Assessment - CEC-600-2014-003.pdf." [Online]. Available: http://www.energy.ca.gov/2014publications/CEC-600-2014-003/CEC-600-2014-003.pdf. [Accessed: 03-Jan-2016].

[4] R.-C. Leou, C.-L. Su, and C.-N. Lu, "Stochastic Analyses of Electric Vehicle Charging Impacts on Distribution Network," *IEEE Trans. Power Syst.*, vol. 29, no. 3, pp. 1055–1063, May 2014.

[5] K. Clement-Nyns, E. Haesen, and J. Driesen, "The Impact of Charging Plug-In Hybrid Electric Vehicles on a Residential Distribution Grid," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 371–380, Feb. 2010.

[6] S. Bashash and H. K. Fathy, "Cost-Optimal Charging of Plug-In Hybrid Electric Vehicles Under Time-Varying Electricity Price Signals," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 1958–1968, Oct. 2014.

[7] L. Gan, A. Wierman, U. Topcu, N. Chen, and S. H. Low, "Real-time deferrable load control: handling the uncertainties of renewable generation," in *Proceedings of the fourth international conference on Future energy systems*, 2013, pp. 113–124.

[8] N. Chen, L. Gan, S. H. Low, and A. Wierman, "Distributional analysis for model predictive deferrable load control," in *2014 IEEE 53rd Annual Conference on Decision and Control (CDC)*, 2014, pp. 6433–6438.

[9] L. Gan, U. Topcu, and S. H. Low, "Optimal decentralized protocol for electric vehicle charging," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 940–951, May 2013.

[10] L. Chen, N. Li, L. Jiang, and S. H. Low, "Optimal Demand Response: Problem Formulation and Deterministic Case," in *Control and Optimization Methods for Electric Smart Grids*, A. Chakrabortty and M. D. Ilić, Eds. Springer New York, 2012, pp. 63–85.

[11] N. G. Paterakis, O. Erdinc, A. G. Bakirtzis, and J. P. S. Catalao, "Optimal Household Appliances Scheduling Under Day-Ahead Pricing and Load-Shaping Demand Response Strategies," *IEEE Trans. Ind. Inform.*, vol. 11, no. 6, pp. 1509–1519, Dec. 2015.

[12] X. Xi and R. Sioshansi, "Using Price-Based Signals to Control Plug-in Electric Vehicle Fleet Charging," *IEEE Trans. Smart Grid*, vol. 5, no. 3, pp. 1451–1464, May 2014.

[13] 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 Trans. Smart Grid*, vol. 3, no. 1, pp. 388–393, Mar. 2012.

[14] A. Dubey, S. Santoso, M. P. Cloud, and M. Waclawiak, "Determining Time-of-Use Schedules for Electric Vehicle Loads: A Practical Perspective," *IEEE Power Energy Technol. Syst. J.*, vol. 2, no. 1, pp. 12–20, Mar. 2015.

[15] C. Jin, J. Tang, and P. Ghosh, "Optimizing Electric Vehicle Charging: A Customer's Perspective," *IEEE Trans. Veh. Technol.*, vol. 62, no. 7, pp. 2919–2927, Sep. 2013.

[16] B. Wang, B. Hu, C. Qiu, P. Chu, and R. Gadh, "EV charging algorithm implementation with user price preference," in *Innovative Smart Grid Technologies Conference (ISGT), 2015 IEEE Power Energy Society*, 2015, pp. 1–5.

[17] Y. Wang, B. Wang, C.-C. Chu, H. Pota, and R. Gadh, "Energy management for a commercial building microgrid with stationary and mobile battery storage," *Energy Build.*

[18] Y. Wang, O. Sheikh, B. Hu, C.-C. Chu, and R. Gadh, "Integration of V2H/V2G hybrid system for demand response in distribution network," in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014, pp. 812–817.

[19] J. Pan, R. Jain, S. Paul, T. Vu, A. Saifullah, and M. Sha, "An Internet of Things Framework for Smart Energy in Buildings: Designs, Prototype, and Experiments," *IEEE Internet Things J.*, vol. 2, no. 6, pp. 527–537, Dec. 2015.

[20] T. Zhang, W. Chen, Z. Han, and Z. Cao, "Charging Scheduling of Electric Vehicles With Local Renewable Energy Under Uncertain Electric Vehicle Arrival and Grid Power Price," *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2600–2612, Jul. 2014.

[21] P. Grahn, K. Alvehag, and L. Soder, "PHEV Utilization Model Considering Type-of-Trip and Recharging Flexibility," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 139–148, Jan. 2014.

[22] C. Chen and S. Duan, "Optimal Integration of Plug-In Hybrid Electric Vehicles in Microgrids," *IEEE Trans. Ind. Inform.*, vol. 10, no. 3, pp. 1917–1926, Aug. 2014.

[23] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, "Fast Prediction for Sparse Time Series: Demand Forecast of EV Charging Stations for Cell Phone Applications," *IEEE Trans. Ind. Inform.*, vol. 11, no. 1, pp. 242–250, Feb. 2015.

[24] M. Alizadeh, A. Scaglione, J. Davies, and K. S. Kurani, "A scalable stochastic model for the electricity demand of electric and plug-in hybrid vehicles," *Smart Grid IEEE Trans. On*, vol. 5, no. 2, pp. 848–860, 2014.

[25] B. Wang, Y. Wang, C. Qiu, C.-C. Chu, and R. Gadh, "Event-based Electric Vehicle Scheduling Considering Random User Behaviors."

[26] C.-Y. Chung, J. Chynoweth, C.-C. Chu, R. Gadh, C.-Y. Chung, J. Chynoweth, C.-C. Chu, and R. Gadh, "Master-Slave Control Scheme in Electric Vehicle Smart Charging Infrastructure, Master-Slave Control Scheme in Electric Vehicle Smart Charging Infrastructure," *Sci. World J. Sci. World J.*, vol. 2014, 2014, p. e462312, May 2014.

[27] J. Chynoweth, C.-Y. Chung, C. Qiu, P. Chu, and R. Gadh, "Smart electric vehicle charging infrastructure overview," in *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*, 2014, pp. 1–5.

[28] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[29] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 103. New York, NY: Springer New York, 2013.

[30] "Locational Marginal Prices." [Online]. Available: http://oasis.caiso.com. [Accessed: 03-Jan-2016].

[31] "Ackerkman Solar Rooftop Project 35kW." [Online]. Available: https://evsmartplug.net/smartgrid/ackerman_pv/. [Accessed: 03-Jan-2016].

[32] Y. Wang, B. Wang, R. Huang, C.-C. Chu, H. R. Pota, and R. Gadh, "Two-Tier Prediction of Solar Power Generation with Limited Sensing Resource," *ArXiv150802669 Stat*, Aug. 2015.

[33] "CVX: Matlab Software for Disciplined Convex Programming | CVX Research, Inc." [Online]. Available: http://cvxr.com/cvx/. [Accessed: 11-Jan-2016].

[34] "U.S. Photovoltaic Prices and Cost Breakdowns: Q1 2015 Benchmarks for Residential, Commercial, and Utility-Scale Systems." [Online]. Available: http://www.nrel.gov/docs/fy15osti/64746.pdf. [Accessed: 18-Jul-2016].

Bin Wang (S'13) received his B.S. degree in vehicle engineering from Jilin University, Changchun, China in 2012. He is currently working towards his Ph.D degree at Smart Grid Energy Research Center (SMERC), University of California, Los Angeles. His current research interests include demand response, EV integration and energy management system.

Yubo Wang (S'13) received his B.S degree from Southeast University, Nanjing, China in 2011 and M.S degree from University of California, Los Angeles in 2012, all in electrical engineering. He currently is a Ph.D student at Smart Grid Energy Research Center, University of California, Los Angeles. His research interests include demand side management and EV integration to smart grid.

Hamidreza Nazaripouya (S'11) received the B.S. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 2007, the M.S. degree in power electronics from the Sharif University of Technology, Tehran, in 2010, and the M.S. degree in power systems from Louisiana State University, Baton Rouge, LA, USA, in 2013. He is currently pursuing the Ph.D. degree with the Smart Grid Energy Research Center, University of California at Los Angeles, Los Angeles, CA, USA. He has conducted several projects for utility companies during his education. His current research interests include the application of power electronics in power system, renewable energy integration, power system stability, microgrid technologies, and electric vehicle.

Charlie Qiu is a senior researcher at Smart Grid Energy Research Center, University of California, Los Angeles.

Chi-cheng Chu received the B.S. degree from National Taiwan University, Taipei, Taiwan, and the Ph.D. degree from the University of Wisconsin–Madison, Madison, WI, USA, in 1990 and 2001, respectively. He is currently a Project Lead with the Smart Grid Energy Research Center, University of California, Los Angeles, CA, USA. He is a seasoned Research Manager who supervised and steered multiple industry and academia research projects in the field of smart grid, radio frequency identification technologies, mobile communication, media entertainment, 3-D/2-D visualization of scientific data, and computer aided design.

Rajit Gadh received the Bachelor's degree from the Indian Institute of Technology, Kanpur, India; the Master's degree from Cornell University, Ithaca, NY, USA; and the Ph.D. degree from Carnegie Mellon University, Pittsburgh, PA, USA, in 1984, 1986 and 1991, respectively. He is a Professor with the Henry Samueli School of Engineering and Applied Science, University of California, Los Angeles (UCLA), CA, USA, and the Founding Director of the UCLA Smart Grid Energy Research Center, Los Angeles. His current research interests include smart grid architectures, smart wireless communications, sense and control for demand response, microgrids and electric vehicle integration into the grid, and mobile multimedia.