Chapter 2
ORCHARD Algorithm for PEV Charging

In this chapter, we consider the PEV charging problem in a community, where the power consumption consists of the load of a PEV charging station and the other inelastic base load, as shown in Fig. 2.1. By controlling the charging rates of PEVs, we aim to minimize total cost on electricity bill paid by the charging station. PEVs arrive at the charging station at random instants with random charging demands that must be fulfilled before their departure time.

The optimal PEV charging scheduling problem has been widely studied in the literature. Many of the existing PEV charging algorithms are "offline" in the sense that they rely on the non-causal information of future PEV charging profiles when deciding the charging schedules. That is, the arrival time and charging demand of a PEV are assumed to be known to the charging station prior to the arrival of the PEV. For instance, [1] requires all PEVs to negotiate with the charging station about their charging schedules one day ahead. However, this assumption does not hold in practice. A PEV's charging profile is revealed only after it arrives at the charging station or connects to the charging pole. Considering the most conservative case where neither the future PEV arrival instants, charging demands, nor their distributions are known *a priori*, we are interested in developing an *online* charging algorithm that schedules PEV charging based only on the information of the PEVs that have already arrived at the charging station.

2.1 Problem Formulation

In this section, we first introduce the offline PEV charging problem by assuming the knowledge of future information. We then formulate the online PEV charging problem without future knowledge. The optimal offline PEV charging scheme

W. Tang, Y.J.A. Zhang, *Optimal Charging Control of Electric Vehicles in Smart Grids*, SpringerBriefs in Electrical and Computer Engineering,
DOI 10.1007/978-3-319-45862-5_2

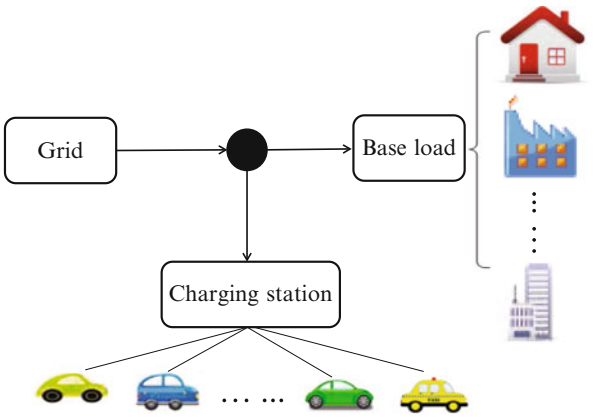

Fig. 2.1 Illustration of PEV charging scenario

will be used as a benchmark to evaluate the performance of the proposed online algorithm.

2.1.1 Optimal Offline PEV Charging Problem

Suppose that N PEVs arrive during a time period T, indexed from 1 to N according to their arrival order. Notice that for a given time period T, N itself is a random variable due to the random arrival of PEVs. Let $D_i, t_i^{(s)}, t_i^{(e)}$ denote the charging demand, arrival time, and departure time of PEV i, respectively, which will be known by the charging station once the PEV arrives. In order to capture the key characteristic of the online charging problem, we assume that a PEV will not depart unless its charging demand is fulfilled. Nonetheless, later we will show that our online algorithm is not affected by early charging terminations.

Due to the battery constraint, PEV i can only be charged at a rate $x_{it} \in [0, U_i]$, where U_i is the maximum charging rate. For the formulation to be meaningful, we assume that all the charging demands are feasible. That is,

$$D_i \leq \min\{U_i(t_i^{(e)} - t_i^{(s)}), \zeta_i\} \tag{2.1}$$

holds for all i, where ζ_i is the battery capacity of PEV i. For simplicity, we omit the upper bound of the total charging rate that can be provided by the charging station. Let $\mathscr{I}_t$ be the set of PEVs parking in the station at time t. The charging station has the control of the charging rate x_{it} for each PEV i. We define s_t as the total charging rate at time t, i.e.,

$$s_t = \sum_{i \in \mathscr{I}_t} x_{it}, \tag{2.2}$$

which is also called charging load at time t. The total load consists of the charging load and the inelastic base load. The base load, denoted by l_t, represents the load of other electricity consumptions at time t except for PEV charging. Here, we assume that the base load does not change continuously with time. Rather, it remains constant for a duration of time (usually in the unit of seconds or minutes) and varies to another value afterwards (see Fig. 2.2 for the illustration). Then, the total load at time t, denoted by y_t, is given by

$$y_t = s_t + l_t = \sum_{i \in \mathscr{I}_t} x_{it} + l_t. \tag{2.3}$$

In this chapter, we assume that the community pays a wholesale electricity price that is time-varying and determined by the total power consumption rate in the system. This often corresponds to a generator supporting a small geographic area with only the temporal variation but no spatial variation of the price [1, 2]. The electricity price is modeled as a linear function of the instant load [1, 3], which is given as follows:

$$a + 2bz_t, \tag{2.4}$$

where a and b are non-negative real numbers, z_t is the instant load. Similar to [3], the electricity cost paid by the charging station at time t is given by

$$\int_{l_t}^{y_t} (a + 2bz_t)\mathrm{d}z_t = (a(\sum_{i \in \mathscr{I}_t} x_{it} + l_t) + b(\sum_{i \in \mathscr{I}_t} x_{it} + l_t)^2) - (al_t + bl_t^2), \tag{2.5}$$

which indicates that the charging station should be responsible for the increased electricity cost caused by the PEV charging. The total cost paid by the charging station for the electricity bill within $[0, T]$ is denoted by Ψ and computed by

$$\Psi = \int_0^T \left(a(\sum_{i \in \mathscr{I}_t} x_{it} + l_t) + b(\sum_{i \in \mathscr{I}_t} x_{it} + l_t)^2 - (al_t + bl_t^2) \right) \mathrm{d}t. \tag{2.6}$$

The optimal charging scheduling problem that minimizes the total energy cost is then formulated as (2.7).

$$\min_{x_{it}} \quad \int_0^T \left(a(\sum_{i \in \mathscr{I}_t} x_{it} + l_t) + b(\sum_{i \in \mathscr{I}_t} x_{it} + l_t)^2 - (al_t + bl_t^2) \right) \mathrm{d}t \tag{2.7a}$$

$$\text{s. t.} \quad \int_{t_i^{(s)}}^{t_i^{(e)}} x_{it}\mathrm{d}t = D_i, i = 1, 2, \ldots, N, \tag{2.7b}$$

$$0 \le x_{it} \le U_i, i = 1, 2, \ldots, N, t \in \left[t_i^{(s)}, t_i^{(e)} \right]. \tag{2.7c}$$

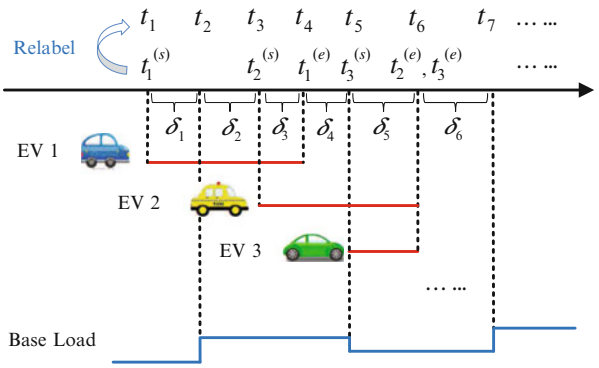

Fig. 2.2 Illustration of one offline case. The time instants are defined as the arrivals and departures of PEV $1, 2, 3, \cdots$, and the change times of base load. Then, relabel the time instants as $t_1, t_2, \cdots$ in a sequential order. In this case, both PEV2 and PEV3 leave at time t_6, and both PEV 3 arrives and base load changes at time t_5

As shown in [4], (2.7) also captures the intent of flattening the total load over time since the cost function is a convex function of total load. It is obvious that Problem (2.7) is a convex optimization problem. In the ideal case where the base load l_t and all PEVs' charging profiles, including $t_i^{(s)}$, $t_i^{(e)}$, U_i and D_i are known to the charging station noncausally at time 0, one can obtain the optimal x_{it} for all i and t by solving (2.7) before the start of system time. We refer to the optimal solution obtained with noncausal information as the offline optimal solution. In practice, however, a PEV's charging profile is revealed only after it arrives at the station. Meanwhile, the base load is also a time-varying random process that cannot be precisely predicted beforehand. In Sect. 2.3, we will investigate an online PEV charging problem that determines the charging rate at each time t based only on the current and past information.

2.1.2 Model Transformation

A close look at (2.7) suggests that there are infinite number of variables x_{it}, because the time t is continuous. In this subsection, we show that the problem (2.7) can be equivalently transformed to a discrete model that is easier to solve and more practical to implement, i.e., the optimal charging rate changes only once in a while.

The equivalence is established through transforming the original continuous problem (2.7) to an event-driven discrete time problem. Throughout this chapter, an *event* is defined by an PEV arrival, departure, or a change in the base load. Likewise, a time interval is defined as the time period between two adjacent events. As illustrated in Fig. 2.2, we relabel the time instants when the events occur as $t_1, t_2, \cdots$ in a sequential order. By doing so, neither the base load nor the set of

PEVs parked in the station changes in the middle of a time interval. Here, we does not exclude the possibility that more than one event occurs at the same time. For instance, in Fig. 2.2, both PEV2 and PEV3 leave at time t_6, and both PEV 3 arrives and base load changes at time t_5. Let $\mathscr{K}$ denote the set of indices of the time intervals, and $\delta_k (k \in \mathscr{K})$ denote the length of kth interval. Without loss of generality, we denote the base load during the kth interval $[t_k, t_{k+1})$ by l_k, since it does not change within a time interval. We show in Lemma 2.1 that there exists an optimal solution where the charging rates remain constant during each time interval.

Lemma 2.1. *Let x_{it}^* denote an optimal solution to (2.7) and $s_t^* = \sum_{i \in \mathscr{I}_t} x_{it}^*$. Then, the optimal total charging rate s_t^* remains constant in each time interval. Moreover, there exists an optimal solution where x_{it}^* remains constant during each time interval.*

The lemma can be easily proved by Jensen's inequality. The detailed proof please refer to Appendix "Proof of Lemma 2.1".

Due to Lemma 2.1, we can safely assume that x_{it} do not change during a time interval. Denote by x_{ik} the charging rate of PEV i during the kth interval. Likewise, denote $\mathscr{J}(i)$ as the set of indices of the time intervals during which PEV i parks in the station, $\mathscr{I}(k)$ as the set of PEVs that park in the kth interval. Based on Lemma 2.1, we can equivalently transform problem (2.7) to the following form that has finitely many variables:

$$\min_{x_{ik}} \quad \sum_{k \in \mathscr{K}} \Big(a\Big(\sum_{i \in \mathscr{I}(k)} x_{ik} + l_k\Big) + b\Big(\sum_{i \in \mathscr{I}(k)} x_{ik} + l_k\Big)^2 - (a l_k + b l_k^2) \Big) \delta_k \tag{2.8a}$$

$$\text{s.t.} \quad \sum_{k \in \mathscr{J}(i)} x_{ik} \delta_k = D_i, i = 1, 2, \ldots, N, \tag{2.8b}$$

$$0 \le x_{ik} \le U_i, i = 1, 2, \ldots, N, k \in \mathscr{J}(i). \tag{2.8c}$$

It is worth pointing out that the discrete time model in (2.8) is different from the traditional time-slotted models. The lengths of time slots are fixed in traditional time-slotted models, whereas the variables in (2.8) are defined by the random events. By doing so, the model in (2.8) successfully captures the dynamics in the system, which is not achievable by the traditional time-slotted models unless the time slots are set infinitesimally small.

2.1.3 Online PEV Charging Problem and Performance Metric

The online PEV charging problem assumes that, at any time instant t, the scheduler only knows the information that is available so far, including the charging profiles

of the PEVs that have arrived upon or before t, as well as the past and current base load. Based on the causal information, the scheduler makes an online decision of the charging rates x_{it} when an event occurs, and the charging rates remain unchanged until the occurrence of the next event. Notice that for practicality, a past decision that has already been implemented cannot be changed in the future. Thus, without knowing the future information, an online algorithm is forced to make decisions that may later turn out to be suboptimal. That is, we have $\Psi_{ON} \geq \Psi^*$, where Ψ_{ON} denotes the total cost induced by an online algorithm and Ψ^* denotes the optimal cost obtained by the offline optimization.

A standard metric to evaluate the performance of an online algorithm is the competitive ratio, which compares the relative performance of an online and the offline algorithm under the same sequence of inputs (e.g., the PEV charging profiles in our problem)[5]. In particular, the competitive ratio of an online algorithm is the maximum ratio between its performance and that of the offline optimal algorithm over all possible input sequences. The formal definition is given in the following Definition 2.1 [5].

Definition 2.1. An online algorithm is $c-$competitive if there exists a constant θ such that

$$\Psi_{ON} \leq c \cdot \Psi^* + \theta \tag{2.9}$$

holds for any input.

By definition, the competitive ratio is always greater than or equal to 1. Notice that the competitive ratio measures the performance ratio in the worst case. Very often, the average performance ratio is much smaller than c. This will be shown in the simulation section, where the proposed ORCHARD algorithm achieves an average performance ratio less than 1.06, although the competitive ratio is 2.39 when the cost function is a quadratic function of load demand.

2.2 Related Work

There have been some recent studies on online PEV charging [3, 4, 6–11]. Gerding et al. [6] proposes an online auction protocol that vehicle owners use agents to bid for the charging opportunities. Therein, it assumes that all the PEVs have the same fixed charging rate. In practice, however, the charging rate could vary among different types of PEVs. Masoum et al. [7] studies the coordinated charging of PEVs in residential distribution systems to reduce the power loss, by assuming that all the PEVs have the same charging period. In practice, the PEVs are very likely to be at the charging station during different time periods. In contrast to the algorithms proposed in [6–8], the proposed PEV charging algorithms in this chapter allow heterogeneity among PEVs. That is, PEVs can have arbitrary arrival (or plug-in time) and departure times, charging demands and maximum charging

rates. He et al. [3] considers the scheduling of PEV charging and discharging in a small geographic area and proposes an online charging algorithm based on an assumption that no future PEV will arrive when a charging schedule is made. The resulting charging schedule is suboptimal as it underestimates the actual charging load. More importantly, most of the existing work, including [3, 4, 6–8], do not provide theoretical analysis of their online algorithms. The few works that analyze the performance, e.g., [9], do not guarantee the satisfaction of PEVs' charging demands before their departures. Compared with [3, 4, 6–9], we have rigorously analyzed the performance of the proposed PEV charging algorithms and show that the proposed PEV charging algorithms are strictly feasible in the sense that they guarantees to fulfill all charging demands before the due time. In addition, the cost functions adopted in [10, 11] depend on the individual PEV's charging demand, whereas in this chapter we consider the price based on the aggregate load demand, including the charging demand of all PEV users as well as the base load.

The charging scheduling for PEV is similar to, but not the same as, the speed scaling problem, which is a power management technique that involves dynamically changing the speed of a processor [12–16]. Specifically, the processor must schedule in real-time a number of tasks and allocate a processing rate to each of them, given that all tasks can be completed before their predetermined deadlines. The processor tries to minimize the total energy cost, where the energy cost at each time t is a positive power function of the total processing rate $s(t)$ at that time, i.e. $s^{\alpha}(t)$ and $\alpha > 1$. The key difference from a PEV charging problem is that speed scaling studied in [12–14] does not place a constraint on the maximum processing rate of each individual job as the PEV charging problem, i.e., each PEV has a maximum charging rate. Another difference is that the cost function of PEV charging problem is a general polynomial instead of a positive power function. In other words, PEV charging schedule is a more general problem than the speed scaling problem, thus its competitive ratio is no less than 2.39, i.e., the best known ratio for speed scaling problem when the cost function is a quadratic form [14].

The first offline optimal algorithm to solve the speed scaling problem was proposed by Yao, Demers and Shenker (YDS) [12]. Later, [12] proposed two online algorithms, i.e. Average Rate (AVR) and Optimal Available (OA). Conceptually, AVR processes a task at a rate equals to its average work load within its specified starting time and deadline. The algorithm is proved to be $2^{\alpha-1}\alpha^{\alpha}$-competitive in [12]. OA uses YDS to calculate the current optimal processing rate by assuming no more tasks will be released in the future, and its competitive ratio was proved to be α^{α} [13]. Apparently, the OA solution is suboptimal, as it underestimates the future workload. To address the problem, [14] proposed a qOA algorithm that scales up the processing rate of OA by a factor $q > 1$. It also showed that qOA works better than OA and AVG in terms of competitive ratio. There are many follow-up works on extended topics, such as managing both temperature and power [13], minimizing the total flow plus energy [15, 16] , etc. Overall, the existing online algorithms for speed scaling cannot be directly applied to solve our problem, mainly because they do not consider the limits on the maximum processing speeds of tasks.

In contrast to the previous work, we propose an Online cooRdinated CHARging Decision (ORCHARD) algorithm, which minimizes the energy cost by mimicking the offline optimal charging decision. Note that ORCHARD relies on no assumptions or predictions of the future information. Through rigorous proof, we show that ORCHARD is strictly feasible in the sense that it guarantees to fulfill all charging demands before due time. Meanwhile, it achieves the best known competitive ratio of 2.39, when the cost function is a quadratic function of the load demand. By exploiting the problem structure, we propose a novel reduced-complexity algorithm to replace the standard convex optimization techniques used in ORCHARD. Through extensive simulations, we show that the average performance gap between ORCHARD and the offline optimal solution, which utilizes the complete future information, is as small as 6.5%. By setting a proper speeding factor, the average performance gap can be further reduced to 5%.

2.3 Online Algorithm

In this section, we present an efficient online algorithm ORCHARD. We show that ORCHARD achieves a competitive ratio that is the best known so far. Moreover, the algorithm is strictly feasible in the sense that it always ensures to satisfy all PEV charging demands.

The proposed ORCHARD algorithm could be easily implemented in a practical charging station. On one hand, it has low computational complexity. On the other hand, it only relies on the causal information of the vehicles and the base load rather than the schedule of the vehicles and the base load in the future. It is robust under any PEV traffic distribution and base load pattern, because it involves no predictions about the future information of PEVs and the base load.

2.3.1 Online Optimal Available (OA) Algorithm

In this subsection, we describe a simple online scheme called Optimal Available (OA) algorithm, which, although suboptimal, will be helpful later in understanding our proposed ORCHARD algorithm.

The OA algorithm works as follows. At a time instant t_j when an event occurs, the scheduler calculates the optimal charging schedule assuming that no more PEVs will arrive and base load is unchanged in the future. More specifically, the scheduler solves the following problem (2.11) at time instant t_j, where $\bar{\mathscr{I}}(t,t_j)$ denotes the set of PEVs who have arrived by time t_j and will be in the station at time $t \in (t_j, \bar{T}(t_j)]$, where

$$\bar{T}(t_j) = \max\{t_i^{(e)} : i \in \mathscr{I}_{t_j}\} \tag{2.10}$$

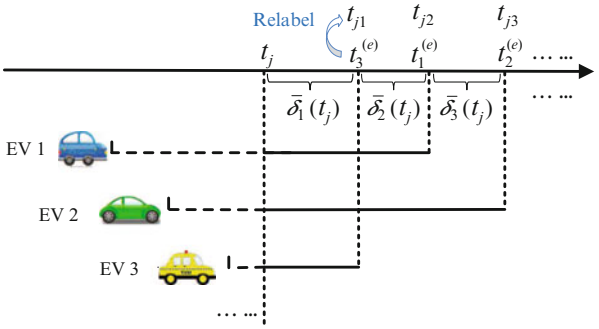

Fig. 2.3 Illustration of one online case. The time intervals are defined by change times of base load as well as the departures of PEVs that are parking in the station at current time t_j. Then, relabel the time instants as $t_{j1}, t_{j2}, \cdots$ in a sequential order

denotes the latest departure time of all PEVs that have already arrived by t_j(recall that $\mathscr{I}_{t_j}$ is the set of PEVs parking in the station at time t_j), $\bar{D}_i(t_j)$ denotes the residual demand to be satisfied for PEV i at time t_j, i.e., the unfinished charging demand of PEV i observed at time t_j.

$$\min_{x_{it}} \quad \int_{t_j}^{\bar{T}(t_j)} (a(\sum_{i\in\bar{\mathscr{I}}(t,t_j)} x_{it} + l_{t_j}) + b(\sum_{i\in\bar{\mathscr{I}}(t,t_j)} x_{it} + l_{t_j})^2 - (al_{t_j} + bl_{t_j}^2))\mathrm{d}t \tag{2.11a}$$

$$\text{s. t.} \quad \int_{t\in[t_j,t_i^{(e)}]} x_{it}\mathrm{d}t = \bar{D}_i(t_j), i \in \mathscr{I}_{t_j}, \tag{2.11b}$$

$$0 \le x_{it} \le U_i, i \in \mathscr{I}_{t_j}, t \in \left[t_j, t_i^{(e)}\right]. \tag{2.11c}$$

Having obtained the solution to (2.11), the scheduler charges the PEVs according to the solution until a new PEV arrives or the base load changes. Then, Problem (2.11) is re-solved with the updated set of charging profiles and base load level. Similar to the discussion in Section II-B, the time axis (from t_j to $\bar{T}(t_j)$]) in Problem (2.11) can be divided into intervals, which are defined only by the departures of the existing PEVs, since the events occurred after time t_j, i.e., the arrivals/departures of new PEVs or the changes of base load after time t_j are not known by the scheduler. By keeping a charging rate in each interval constant, Problem (2.11) can be equivalently transformed to one with finitely many variables. An example in Fig. 2.3 illustrates the intervals defined by change times of base load and the departures of PEVs that are parking in the station at time t_j. Denote $\bar{\mathscr{K}}(t_j)$ as the set of indices of the intervals seen at time t_j, $\bar{\delta}_k(t_j)$, where $k \in \bar{\mathscr{K}}(t_j)$ as the length of the kth interval, $\bar{\mathscr{I}}(k, t_j)$ as the set of PEVs who have arrived by time t_j

and will be in the station at interval $k, k \in \bar{\mathscr{K}}(t_j)$, and $\bar{\mathscr{J}}(i,t_j)$ as the set of indices of time intervals that PEV i will park in the station. It directly follows from Lemma 2.1 that there exists an optimal solution to (2.11) where the optimal charging rates are constants during each interval. Denote x_{ik} by the charging rate of PEV i in interval $k, k \in \bar{\mathscr{K}}(t_j)$. Then, (2.11) is equivalent to the following discrete time optimization problem

$$\min_{x_{ik}} \quad \sum_{k \in \bar{\mathscr{K}}(t_j)} (a(\sum_{i \in \bar{\mathscr{I}}(k,t_j)} x_{ik} + l_{t_j}) + b(\sum_{i \in \bar{\mathscr{I}}(k,t_j)} x_{ik} + l_{t_j})^2$$

$$- (a l_{t_j} + b l_{t_j}^2)) \bar{\delta}_k(t_j) \tag{2.12a}$$

$$\text{s.t.} \quad \sum_{k \in \bar{\mathscr{J}}(i,t_j)} x_{ik} \bar{\delta}_k(t_j) = \bar{D}_i(t_j), i \in \mathscr{I}_{t_j}, \tag{2.12b}$$

$$0 \le x_{ik} \le U_i, i \in \mathscr{I}_{t_j}, k \in \bar{\mathscr{J}}(i,t_j). \tag{2.12c}$$

In the next section, we will introduce our proposed ORCHARD algorithm. Note that ORCHARD also solves Problem (2.12), but only uses x_{i1}, $i \in \mathscr{I}(1,t_j)$, i.e., the charging solutions in the first (i.e., current) interval. As we will introduce later, (2.12) needs to be resolved again with the updated l_{t_j}, $\bar{\delta}_k(t_j)$, $\bar{\mathscr{I}}(k,t_j)$, $\bar{\mathscr{J}}(i,t_j)$, $\bar{D}_i(t_j)$ once a new PEV arrives, finishes charging, or the base load changes.

2.3.2 The ORCHARD Algorithm

The charging rate scheduled by OA tends to be smaller than the optimal offline solution due to the neglect of future demands. In ORCHARD, we speed up the charging schedule obtained from (2.12) by a speed-up factor q ($q \ge 1$). Roughly speaking, the total charging rate by ORCHARD is q times that of OA. The value of q determines the performance of ORCHARD, including both the competitive ratio and the average performance. We will discuss how to set a proper q to obtain the minimum competitive ratio in Sect. 2.3.3 and to obtain the best average performance in Sect. 2.5.3.

Due to the factor q, the charging rate of ORCHARD is larger than that of OA such that ORCHARD finishes charging PEVs earlier than OA does. Then ORCHARD always finishes charging PEVs before their departure time. Hence, ORCHARD recalculates the charging rate when there is a new PEV arrival, a PEV finishes charging, or the base load changes. We denote by $\bar{x}_{ik}(t_j)$ the charging rate of PEV $i \in \mathscr{I}_{t_j}$ in the kth interval computed by OA at time t_j, $\hat{x}_{it}$ the charging rate of PEV i at time t computed by ORCHARD, and $\hat{s}_t$ the sum of $\hat{x}_{it}$ at time t. When ORCHARD recalculates the charging rate, the right hand side of (2.12b) is updated as follows

$$\bar{D}_i(t_j) = \begin{cases} 0, & \text{if PEV } i \text{ finishes charging,} \\ D_i, & \text{if PEV } i \text{ arrives,} \\ \bar{D}_i(t_{j-1}) - \hat{x}_{it_{j-1}}(t_j - t_{j-1}), & \text{otherwise.} \end{cases} \tag{2.13}$$

Here, $\hat{x}_{it_{j-1}}$ denotes the constant charging rate between t_{j-1} and t_j. Moreover, we also need to update $\bar{\mathscr{J}}(i,t_j)$, $\bar{\delta}_k(t_j)$ and $\bar{\mathscr{I}}(k,t_j)$ due to the change of current PEVs at t_j. A pseudo code of ORCHARD is presented in Algorithm 1 and explained as follows:

Step 1: Once the base load changes, a PEV arrives, or a PEV finishes charging, let $j = j+1$, set t_j as the current starting time and update the current base load l_{t_j} as well as the parameters based on current PEVs (line 3).
Step 2: Solve (2.12) with the updated parameters. Denote by $\bar{x}_{i1}(t_j)$ the optimal charging solution of the first (i.e., current) time interval. (line 4).
Step 3: Determine the total charging rate, which is the minimum of q times of the total charging rate computed by OA, i.e., $\sum_{i\in\bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j)$, and the sum of maximum charging rates of current PEVs, i.e., $\sum_{i\in\bar{\mathscr{I}}_{t_j}} U_i$ (line 5).
Step 4: Determine the charging solution at time $[t_j, t_{j+1})$ by setting the charging rate of PEV i as in line 6 in Algorithm 1, where t_{j+1} is the next time that the base load changes, a PEV arrives, or a PEV finishes charging.

By doing so, we ensure that: (1) for each PEV, the charging rate does not exceed its maximum charging rate, i.e., $x_{it} \le u_i, i \in \mathscr{I}_{t_j}$; (2) the sum of the charging rates equals the total charging rate given by Step 3, i.e., $\sum_{i\in\mathscr{I}_{t_j}} \hat{x}_{it} = \hat{s}_t$; (3) for each PEV, the charging rate is no smaller than the solution given by OA in Step 2, i.e., $\hat{x}_{it} \ge \bar{x}_{i1}(t_j), \forall i \in \mathscr{I}_{t_j}$ (line 6).

Algorithm 1: ORCHARD

input : U_i, $t_i^{(e)}$, D_i of all parking PEVs, the base load l_t
output: $\hat{x}_{it}$
1 initialization $j = 0$;
2 **while** *the base load changes, a PEV arrives, or a PEV finishes charging* **do**
3 Let $j = j+1$, record current time t_j. Update l_{t_j}, $\bar{\delta}_k(t_j)$, $\bar{\mathscr{I}}(k,t_j)$, $k \in \bar{\mathscr{K}}(t_j)$, $\bar{\mathscr{J}}(i,t_j)$, $\bar{D}_i(t_j)$, $i \in \mathscr{I}_{t_j}$.
4 Solve problem (2.12) for the optimal solution $\bar{x}_{i1}(t_j) \forall i \in \bar{\mathscr{I}}_{t_j}$.
5 Set $\hat{s}_t = \min\{q \cdot \sum_{i\in\bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j), \sum_{i\in\bar{\mathscr{I}}_{t_j}} U_i\}$.
6 Set the charging rate of PEV i at the time $t \in [t_j, t_{j+1})$ as $\hat{x}_{it} = \min\{\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i\in\bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \cdot \frac{q-1}{q} \hat{s}_t, U_i\}$.
7 **end**

Since OA always guarantees a feasible solution, we can intuitively infer that ORCHARD also guarantees producing a feasible solution, simply because its charging rate is always no smaller than that of the OA. The feasibility of ORCHARD is proved in Lemma 2.2 below.

Lemma 2.2. *ORCHARD always outputs a feasible solution to problem (2.12).*

Proof. Please see the detailed proof in Appendix "Proof of Lemma 2.2". ■

2.3.3 Derivation of Competitive Ratio

In this subsection, we show that ORCHARD is 2.39-competitive when the cost function is a quadratic function of the load demand. Here, we consider an amortized local competitiveness analysis and a potential function $\Phi(t)$ as a function of time. In the following, we will construct a $\Phi(t)$ to prove the inequality (2.9) for a specific competitive ratio c. In particular, Φ is chosen to satisfy

$$\Phi(0) = \Phi(T) = 0. \tag{2.14}$$

We always denote the current time as τ_0. Let $l, \hat{s}$ and s^* be the current base load, total charging rate of ORCHARD and the optimal offline algorithm respectively. In order to establish that ORCHARD is c-competitive, it is sufficient to show that the following key equation

$$\begin{aligned} &(a(\hat{s}+l)+b(\hat{s}+l)^2-(al+bl^2))+\frac{\mathrm{d}\Phi}{\mathrm{d}\tau_0} \\ \leq& c\cdot(a(s^*+l)+b(s^*+l)^2-(al+bl^2)), \end{aligned} \tag{2.15}$$

holds for all $\tau_0 \leq T$, where $c \geq 1$. This is because the integral over the entire time T on both sides leads to

$$\begin{aligned} &\int_0^T (a(\hat{s}+l)+b(\hat{s}+l)^2-(al+bl^2))\mathrm{d}t \\ \leq& c\cdot\int_0^T (a(s^*+l)+b(s^*+l)^2-(al+bl^2))\mathrm{d}t, \end{aligned} \tag{2.16}$$

where $\int_0^T (a(\hat{s}+l)+b(\hat{s}+l)^2)\mathrm{d}t$ is the total cost of ORCHARD, $\int_0^T (a(s^*+l)+b(s^*+l)^2)\mathrm{d}t$ is the cost of optimal offline algorithm. In this sense, (2.16) is consistent with the definition of competitive ratio in (2.9). Before providing the proof of competitiveness, we introduce the following notations. At a current time τ_0 in the ORCHARD algorithm, let $\hat{w}(t',t''), \tau_0 \leq t' \leq t''$ denote the total residual demand of PEVs whose deadlines are between $[t',t'']$. Similarly, for offline optimal algorithm, let $w^*(t',t''), \tau_0 \leq t' \leq t''$ denote the total residual demand of PEVs whose deadlines

are between $[t', t'']$. Note that $\hat{w}(t', t'')$ and $\hat{w}(t', t'')$ are likely to be rather different since the charging solution before current time τ_0 of ORCHARD and offline optimal algorithm are very likely different. We further denote

$$\begin{aligned} d(t', t'') = \max \Big\{ 0, \min\{\hat{w}(t', t''), \frac{1}{q} \sum_{i \in \mathscr{I}(t')} U_i(t'' - t')\} \\ - \min\{w^*(t', t''), \sum_{i \in \mathscr{I}(t')} U_i(t'' - t')\} \Big\} \end{aligned} \tag{2.17}$$

as the amount of additional demand left for ORCHARD with deadline in $(t', t'']$. Then, we define a sequence of time points $\tau_1, \tau_2, \cdots$ as follows: let τ_1 be the time such that $d(\tau_0, \tau_1)/(\tau_1 - \tau_0)$ is maximized. If there are several such points, we choose the furthest one. Given τ_k, we let $\tau_{k+1} > \tau_k$ be the furthest point that maximizes $d(\tau_k, \tau_{k+1})/(\tau_{k+1} - \tau_k)$, i.e.,

$$\tau_{k+1} = \arg\max_{\tau > \tau_k} d(\tau_k, \tau)/(\tau - \tau_k). \tag{2.18}$$

The "load intensity gap" within $(\tau_k, \tau_{k+1}]$ is denoted as

$$g_k = d(\tau_k, \tau_{k+1})/(\tau_{k+1} - \tau_k), k = 1, 2, \cdots. \tag{2.19}$$

Evidently, g_k is a non-negative monotonically decreasing sequence.

We are now ready to define the potential function Φ as

$$\Phi = \beta_1 \cdot a \sum_{k=0}^{\infty} ((\tau_{k+1} - \tau_k) g_k) + \beta_2 \cdot b \sum_{k=0}^{\infty} ((\tau_{k+1} - \tau_k) g_k^2), \tag{2.20}$$

where β_1, β_2 will be assigned finite values later. We notice that $\Phi(0) = \Phi(T) = 0$ holds, since the load is clearly zero before any PEV arrives and after the last deadline.

In the Theorem 2.1 below, we derive the competitive ratio of ORCHARD. First, we provide the following Lemma to be used in proving Theorem 2.1.

Lemma 2.3.

$$q g_0 \le \hat{s} \le q g_0 + q s^*. \tag{2.21}$$

The detailed proof please see Appendix "Proof of Lemma 2.3".

In the Theorem 2.1 below, we derive the competitive ratio of ORCHARD.

Theorem 2.1. *ORCHARD is* 2.39*-competitive when the cost function is a quadratic function of the load demand by setting* $q = 1.46$.

Proof. Please see the detailed proof in Appendix "Proof of Theorem 2.1".

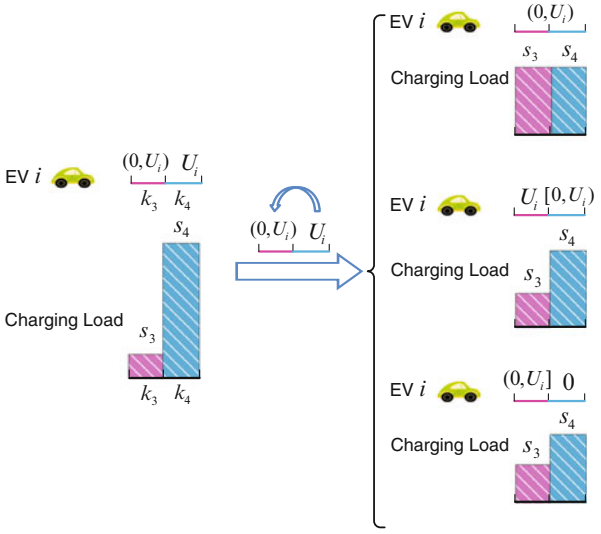

Fig. 2.4 Illustration of example based on KKT conditions. In case 1, we have $y^*_{k_1} = y^*_{k_2}$. In case 2 and 3, we cannot achieve the balanced charging rate, i.e., $y^*_{k_1} < y^*_{k_2}$, either because $x^*_{ik_1}$ has increased to the limit U_i (case 2) or $x^*_{ik_2}$ has decreased to 0 (case 3)

2.4 The Optimal Offline Algorithm with Low Complexity

The major complexity of Algorithm 1 lies in the computation involved in solving Problem (2.12) every time when a PEV arrives or finishes charging. By exploring the special structure of the optimal solution, we propose in this section a low-complexity solution algorithm to solve problem (2.12). Notice that Problem (2.12) and the offline optimization problem (2.8) have exactly the same structure. Both of them are to minimize a convex and additive objective function over a polyhedron. Thus, the algorithm proposed here can also apply to (2.12). The proposed algorithm is shown to have a much lower computational complexity than generic convex optimization algorithms, such as the interior point method.

2.4.1 Optimality Analysis

It is easy to verify that problem (2.8) is convex, and then we apply the Karush-Kuhn-Tucker (KKT) conditions to it [17]. We associate a dual variable λ_i with inequality (2.8a), a dual variable w_{ik} with inequality (2.8b), a dual variable v_{ik} with inequality (2.8c). Then, the Lagrangian is given by:

$$
\begin{aligned}
L = & \sum_{k\in\mathcal{K}} (a(\sum_{j\in\mathcal{I}(k)} x_{jk} + l_k) + b(\sum_{j\in\mathcal{I}(k)} x_{jk} + l_k)^2 \\
& - (al_k + bl_k^2))\delta_k - \sum_{i=1}^{N} \lambda_i (\sum_{k\in\mathcal{J}(i)} x_{ik}\delta_k - D_i) \\
& + \sum_{i=1}^{N} \sum_{k\in\mathcal{J}(i)} (-\omega_{ik}x_{ik} + \nu_{ik}(x_{ik} - U_i)).
\end{aligned} \tag{2.22}
$$

Let x_{ik}^* denote the optimal charging rate for EV i at interval $k \in \mathcal{I}(k)$. The necessary and sufficient *KKT* conditions are given by:

$$
a + 2b(\sum_{j\in\mathcal{I}(k)} x_{jk}^* + l_k) - \lambda_i + \nu_{ik} - \omega_{ik} = 0,
$$

$$
i = 1,\ldots,N, k \in \mathcal{J}(i), \tag{2.23a}
$$

$$
\lambda_i (D_i - \sum_{k\in\mathcal{J}(i)} x_{ik}^*) = 0, i = 1,\ldots,N, \tag{2.23b}
$$

$$
\omega_{ik} x_{ik}^* = 0, i = 1,\ldots,N, k \in \mathcal{J}(i), \tag{2.23c}
$$

$$
\nu_{ik}(x_{ik}^* - U_i) = 0, i = 1,\ldots,N, k \in \mathcal{J}(i), \tag{2.23d}
$$

where (2.23a) means that the differentiation of L should be 0 at x_{ik}^*, and (2.23b), (2.23c), (2.23d) are the complementary slackness conditions. We separate our analysis into the following three cases:

1. If $x_{ik_1}^* \in (0, U_i)$ for a particular PEV i in a time interval $k_1 \in \mathcal{J}(i)$, then, by complementary slackness, we have $\nu_{ik_1} = w_{ik_1} = 0$. From (2.23a), $y_{k_1} = \sum_{j\in\mathcal{I}(k_1)} x_{jk_1} + l_{k_1} = (\lambda_i - a)/2b$.
2. If $x_{ik_2}^* = 0$ for PEV i during a time interval $k_2 \in \mathcal{J}(i)$, we can infer from (2.23c) and (2.23d) that $\omega_{ik_2} > 0$ and $\nu_{ik_2} = 0$. Then, $y_{k_2} = \sum_{j\in\mathcal{I}(k_2)} x_{jk_2}^* + l_{k_2} = (\lambda_i - a)/2b + \omega_{ik_2}/2b$.
3. Similarly, if $x_{ik_3}^* = U_i$ for PEV i in interval $k_3 \in \mathcal{J}(i)$, then, we have $y_{k_3} = \sum_{j\in\mathcal{I}(k_3)} x_{jk_3}^* + l_{k_3} = (\lambda_i - a)/2b - \nu_{ik_3}/2b$.

Let y_k^* be the optimal total load that

$$
y_k^* = \sum_{i\in\mathcal{I}(k)} x_{ik}^* + l_k. \tag{2.24}
$$

From the above discussions, we can conclude that the necessary and sufficient conditions for the optimal total charging rate as follows:

1. y_k^* is the same for a set of intervals as long as there exists a PEV i that parks through this set of intervals with $x_{ik}^* \in (0, U_i)$.

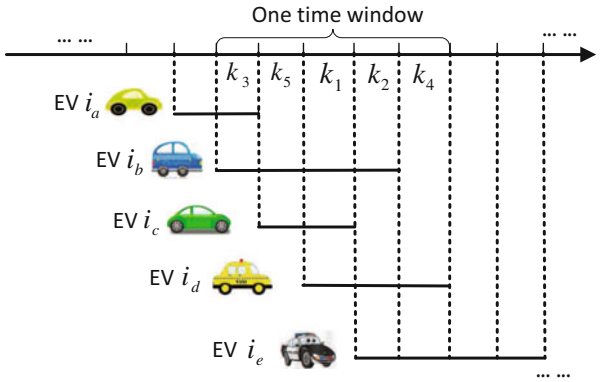

Fig. 2.5 Illustration of one time window. The time window starts from the arrival time of PEV i_b to the departure time of PEV i_d

2. If $x^*_{ik} = 0$ for a PEV i during an interval k that it parks in, then, y^*_k in that interval is no smaller than that of the other interval $k' \in \mathscr{J}(i)$ during which $x^*_{ik'} \in (0, U_i]$.
3. If $x^*_{ik} = U_i$ for PEV i during an interval k, then, y^*_k is no larger than that of the other interval $k' \in \mathscr{J}(i)$ whose charging rate $x^*_{ik'} \in [0, U_i)$.

The above conditions can be intuitively understood as follows. Due to the convexity the objective function, the optimal solution to (2.8) always tries to balance the total load y_k across different intervals. For example, as shown in Fig. 2.4, if there are two intervals k_1 and k_2 with $y^*_{k_1} < y^*_{k_2}$, and a PEV i such that $x^*_{ik_1} = 0$ and $0 < x^*_{ik_2} \leq U_i$, then we can always shift the charging load of PEV i from interval k_2 to k_1 to decrease the total load difference between k_1 and k_2. In other words, whenever possible, the charging load should be shifted from interval k_2 to k_1 until $y^*_{k_1} = y^*_{k_2}$ (case 1 in Fig. 2.4). However, such balanced charging rate at two intervals may not achievable, resulting $y^*_{k_1} < y^*_{k_2}$, either because $x^*_{ik_1}$ has increased to the limit U_i (case 2) or $x^*_{ik_2}$ has decreased to 0 (case 3). Based on these conditions, we present a low-complexity solution algorithm in the next subsection.

2.4.2 *Algorithm Description*

From the analysis of KKT optimality conditions, one should manage to balance the total load among all intervals under the constraints of each individual PEV's charging profiles. In this subsection, we present a charging rate allocation algorithm to achieve the objective of "load balancing". The optimality and complexity of the proposed algorithm will be discussed in the next subsection.

Intuitively, one should shift the demand from "heavily loaded" intervals to the others. To do this, we first introduce the concept of *intensity* of an interval k, denoted

by ρ_k, to quantify the heaviness of the load in the interval. Specifically, ρ_k is defined as the upper bound of the charging load of an interval, and is given by:

$$\rho_k = \sum_{i \in \mathscr{I}(k)} \min\left\{U_i, \frac{D_i}{\delta_k}\right\}. \tag{2.25}$$

This is because the charging rate of each PEV i in the interval k will not exceed the minimum between the charging rate bound U_i and the D_i/δ_k, i.e., PEV i only charges in the interval k. The basic idea of the proposed algorithm is to shift the demand of a set of intervals with high intensities to the others with lower intensities. Notice that the demand of an interval k_1 can only be transferred to its neighboring interval k_2 such that $k_1 \in J(i)$ and $k_2 \in J(i)$ hold for some PEV i. Therefore, we need to consider both the intensities of an interval set and their neighboring intervals to make the decision on "load balancing".

From the above discussion, we take into consideration a set of consecutive intervals, referred to as a "time window", starting from the arrival time of a PEV to the departure time of one, probably another PEV. If there are N PEVs, the maximum number of time windows is N^2. Within a tagged time window, we select a set of intervals of the highest intensities as the candidate interval set from which the load is to be transferred to the other intervals in the time window. In practice, we first consider the single interval with the highest intensity, then the top two intervals, top three intervals, etc. That is, for each time window, sort the intervals in descending order according to ρ. The index is denoted by $k_1, k_2, \ldots$, as illustrated in Fig. 2.5.

Evidently, a time window consisting of K' intervals contains K' such interval sets. For example, there are 5 interval sets in the time window shown in Fig. 2.5. We denote the interval sets obtained from all the time windows in the entire duration T as $\mathscr{K}_1, \mathscr{K}_2, \cdots$. Then, the following iterative algorithm determines the load transfer operation of intervals as well as the charging rate schedule of all PEVs.

Step 1: For each interval set $\mathscr{K}$, we first compute the residual demand of PEV i on $\mathscr{K}$. The residual demand of PEV i on $\mathscr{K}$, denoted by $D_i(\mathscr{K})$, is calculated by letting PEV i be charged at the upper bound U_i on its parking intervals non-overlapped with $\mathscr{K}$. That is

$$D_i(\mathscr{K}) = D_i - U_i \sum_{k \in \mathscr{J}(i) \setminus (\mathscr{J}(i) \cap \mathscr{K})} \delta_k. \tag{2.26}$$

The intuition is to transfer as much as possible the charging demand from intervals with high intensities to its neighboring intervals. Then, we can calculate the total load of the interval set $\mathscr{K}$ by balancing the residual demand over all the intervals in $\mathscr{K}$, i.e.,

$$y = \frac{\sum_{k \in \mathscr{K}} \left(\sum_{i \in \mathscr{J}(k)} (\max\{0, D_i(\mathscr{K})\}) + \hat{y}_k \delta_k\right)}{\sum_{k \in \mathscr{K}} \delta_k}, \tag{2.27}$$

where $\hat{y}_k$ is the total load after scheduling in previous iterations at the interval and initially set to be l_k.
Step 2: Find the interval set $\mathscr{K}^*$ with the highest total load y^*. Then the optimal total charging rate of interval in $\mathscr{K}^*$ is set to be s_k^*, where

$$s_k^* = y^* - l_k, \forall k \in \mathscr{K}^*. \tag{2.28}$$

We denote $\mathscr{I}^*$ by the set of PEVs of which the residual demand $D_i(\mathscr{K}^*)$ is non-negative, Δ^* by the total length of the intervals in the set $\mathscr{K}^*$, i.e., $\Delta^* = \sum_{k\in\mathscr{K}^*} \delta_k$. For each PEV $i \in \mathscr{I}^*$, we schedule the charging rate as

$$x_{ik}^* = \begin{cases} U_i - \frac{(U_i\Delta^* - D_i(\mathscr{K}^*))(\sum_i U_i - s_k^*)}{\sum_i (U_i\Delta^* - D_i(\mathscr{K}^*))}, & k \in \mathscr{K}^*, \\ U_i, & k \in \mathscr{J}(i) \setminus \mathscr{K}^*. \end{cases} \tag{2.29}$$

It is easy to verify that $\sum_{k\in\mathscr{K}^*} x_{ik}^* = s_k^*$ for $k \in \mathscr{K}^*$. Note that PEV $i \in \mathscr{I}^*$ has finished scheduled charging rate and will not be considered in the next iterations. Then the total charging rate at any interval $k \in \mathscr{J}(i) \setminus \mathscr{K}^*$ should be increased by U_i. We use $\hat{s}_k$ to denote the total rate scheduled in the interval $k \notin \mathscr{K}^*$ up to the current iteration, which is updated as.

$$\hat{s}_k = \hat{s}_k + \sum_{i\in\mathscr{I}^*\cap\mathscr{I}(k)} U_i. \tag{2.30}$$

For a PEV $i \notin I^*$ whose parking intervals overlaps with $\mathscr{K}^*$, the charging rate of its parking intervals overlapped with $\mathscr{K}^*$ is assigned to be 0, i.e.,

$$x_{ik}^* = 0, k \in \mathscr{J}(i) \cap \mathscr{K}^*. \tag{2.31}$$

Step 3: Exclude $\mathscr{I}^*$ and $\mathscr{K}^*$ from the PEV set and interval set, and merge the remaining intervals into a new time duration. Find all the interval sets in the newly formed time windows as in Fig. 2.5. Then, repeat from step 1 until the charging rates of all PEVs are scheduled.

2.4.3 Optimality and Complexity

We first provide the following Lemma 2.4 before proving the global optimality of the proposed algorithm. Denote $\mathscr{K}^*(m)$ by the interval set found in mth iteration, $\Delta^*(m)$ by the total length of intervals in $\mathscr{K}^*(m)$, i.e,

$$\Delta^*(m) = \sum_{k\in\mathscr{K}^*(m)} \delta_k, \tag{2.32}$$

$y^*(m)$ by the highest total load of interval set $\mathscr{K}^*(m)$ respectively.

Lemma 2.4. *In the proposed low-complexity solution algorithm, the highest total load found in mth iteration is no smaller than that found in* $(m+1)$*th iteration, i.e.,* $y^*(m) \geq y^*(m+1)$.

Please see the proof in Appendix "Proof of Lemma 2.4".

Theorem 2.2. *The proposed algorithm always outputs a globally optimal schedule.*

Proof. Please see the detailed proof in Appendix "Proof of Theorem 2.2".

Now we give a complexity analysis of the proposed algorithm. Consider the worst case where N PEVs lead to $2N-1$ intervals, N^2 variables and $2N^2+N$ constrains. The proposed low-complexity solution algorithm at least excludes one interval in each outer loop that leads to at most $2N-1$ iterations. In each iteration (step 1 - step 3), there are at most $N(N+1)/2$ time windows which contains at most N possible interval sets. Hence, the total number of iterations is in the order of $O(N^4)$. Since the operation complexity of intensity calculation for each sequence is $O(N)$ (we regard one addition, subtraction, multiplication and division as one operation), the upper bound of operation complexity is $O(N^5)$. On the other hand, the generic interior point algorithm has a complexity at the order of $O(n^{3.5})$ [18], where n is the number of variables. Note that $n = N^2$ in our problem, and thus the complexity of interior point algorithm is $O(N^7)$, which is much higher than that of the proposed algorithm.

2.5 Simulations

In this section, we evaluate the performance of ORCHARD and verify the iteration complexity of the low-complexity solution algorithm. Specially, we define the *average performance ratio* as the ratio of the average cost of online algorithm to that of offline optimal algorithm. Note that the variation of a and b, $(a, b > 0)$ will not change charging solution to ORCHARD while the variation of q will, so we only discuss how the average performance ratio changes by varying q, shown in Sect. 2.5.3.

2.5.1 Performance Ratio Evaluation

We consider a running time T of 24 hours. We choose the base load profile of one day in the service area of South California Edison from [4]. The coefficients of the cost function are set to $a = 10^{-4}$ \$/kWh and $b = 0.6 \times 10^{-4}$ \$/kWh/kW [3]. There are two types of PEVs in our simulation [19]: (1) maximum charging rate $U_i = 3.3$ kW, battery capacity $\zeta_i = 35$ kWh; (2) maximum charging rate $U_i = 1.4$ kW, battery capacity $\zeta_i = 16$ kWh. Each PEV is equally likely chosen from the two

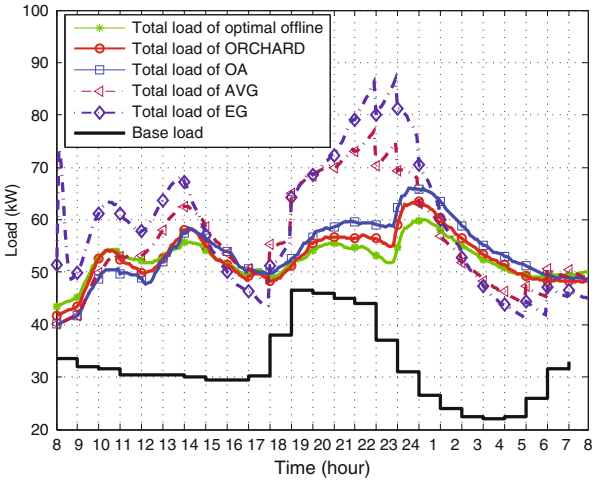

Fig. 2.6 Base load and total load of five algorithms

Table 2.1 Parameter settings of the arrival and parking durations

Time of day	Arrival rate (PEVs/hour)	Mean parking Time (hour)
08:00–10:00	7	10
10:00–12:00	5	1/2
12:00–14:00	10	2
14:00–18:00	5	1/2
18:00–20:00	10	2
20:00–24:00	5	10
24:00–08:00	0	0

types and the charging demand is uniformly chosen from $[0, \min\{U_i \cdot (t_i^{(e)} - t_i^{(s)}), \zeta_i\}]$ (this ensures that (2.8) is feasible). Each PEV's arrival follows a Poisson distribution and the parking time follows an exponential distribution [9]. The mean arrival and parking durations are listed in Table 2.1, where there are three peak hours with large arrival rates, i.e. 8 to 10, 12 : 00 to 14 : 00 and 18 : 00 to 20 : 00. The settings of the peak hour match with the realistic vehicle trips in National Household Travel Survey (NHTS) 2009 [20].

We compare ORCHARD to the optimal offline algorithm as well as other online algorithms. Unless otherwise specified, the speeding factor of ORCHARD, q, is set to be 1.46. Note that by Theorem 2.1 $q = 1.46$ achieves the best competitive ratio in the worst case, but may not be the best choice for average performance. We will discuss the effect of q in Sect. 2.5.3. We denote the cost of ORCHARD and the optimal offline algorithm by Ψ_{ORC} and Ψ^*, respectively. The other online algorithms for comparison are

Table 2.2 Average performance ratio of online algorithms

$\frac{\Psi_{ORC}}{\Psi^*}$	$\frac{\Psi_{OA}}{\Psi^*}$	$\frac{\Psi_{AVG}}{\Psi^*}$	$\frac{\Psi_{EG}}{\Psi^*}$
1.065	1.134	1.528	2.344

Table 2.3 Parameter settings of the three scenarios

Time of day	Arrival rate (PEVs/hour)			Mean parking
	S. 1	S. 2	S. 3	Time (hour)
08:00–10:00	7	7	7	10
10:00–12:00	5	5	5	1/2
12:00–14:00	10	30	50	2
14:00–18:00	5	5	5	1/2
18:00–20:00	10	30	50	2
20:00–24:00	5	5	5	10
24:00–08:00	0	0	0	0

1. online average charging (AVG): The charging demand is evenly distributed during the parking period, i.e. the charging rate is $D_i/(t_i^{(e)} - t_i^{(s)})$.
2. online eagerly charging (EG): PEV i is charged at the maximum charging rate U_i.
3. online optimal available information charging (OA) : Set $q = 1$ in ORCHARD.

Their costs are denoted by Ψ_{AVG}, Ψ_{EG} and Ψ_{OA}, respectively.

All the convex optimizations are solved by CVX [21]. We simulate 10^5 cases and plot the average base load as well as the total load over time in Fig. 2.6, where the total load represents the sum of the base load and the charging load, defined in Eq. (2.3). In addition, the average performance ratios normalized against the optimal offline solution are shown in Table 2.2. Figure 2.6 shows that the total load curve of ORCHARD follows closely with the optimal offline solution curve. In contrast, EG and AVG largely deviate from the optimal charging curve, being either too aggressive or too conservative depending on the arrival patterns. From Table 2.2, we can see that ORCHARD performs the best among the four online algorithms, which has on average less than 6.5 % extra cost compared with the optimal offline algorithm.

2.5.2 The Influence by the PEV Pattern

In this subsection, we ignore the base load that mainly discuss how PEV pattern affects the average performance ratio. We consider three different scenarios, whose mean arrival and parking durations are listed in Table 2.3. In particular, scenarios 1–3 represent light, moderate and heavy traffic, respectively. The main difference lies in the arrival rates at the two peak hours, i.e. 12 : 00 to 14 : 00 and 18 : 00 to 20 : 00.

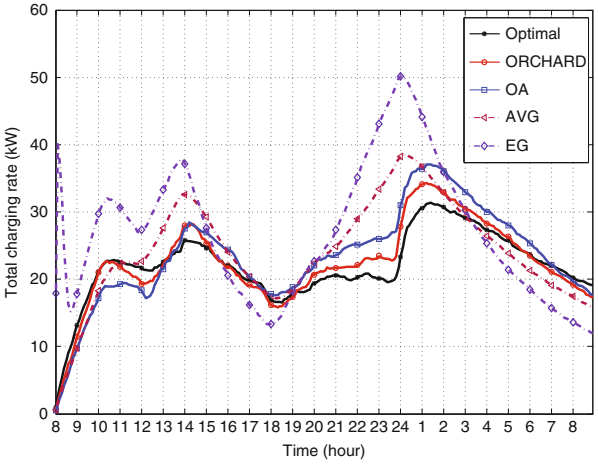

Fig. 2.7 PEV total charging rate of five algorithms in Scenario 1

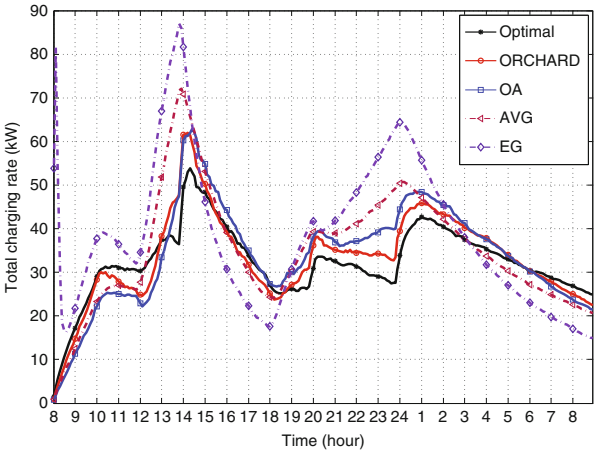

Fig. 2.8 Base load and total load of five algorithms in Scenario 2

For each scenario, we simulate 10^5 cases and plot the average total charging rate over time in Figs. 2.7, 2.8 and 2.9, respectively, where the vertical axis is the total charging rate of PEVs, defined in Eq. (2.2). In addition, the average performance ratios normalized against the optimal offline solution are shown in Table 2.4. In all scenarios, ORCHARD performs the best among the four online algorithms, which has on average less than 14% extra cost compared with the optimal offline algorithm. We also notice that ORCHARD has a 10% performance gain compared with the OA algorithm in the scenario with heavy traffic. We will discuss the proper setting of q in Sect. 2.5.3. The charging rate curve of the proposed online charging algorithm follows closely with the optimal offline solution curve. In

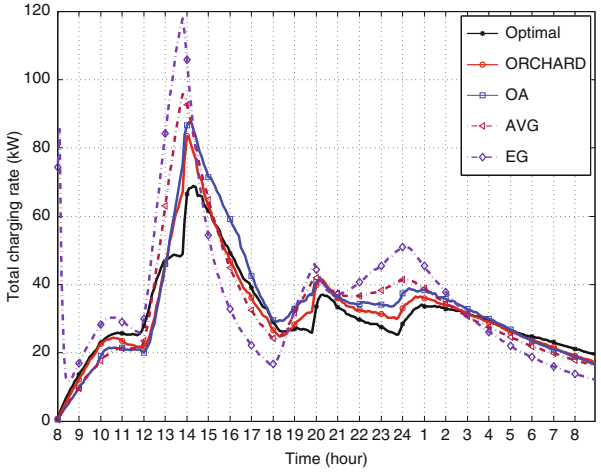

Fig. 2.9 Base load and total load of five algorithms in Scenario 3

Table 2.4 Average performance ratio of online algorithms

Scenario	$\frac{\Psi_{ORC}}{\Psi^*}$	$\frac{\Psi_{OA}}{\Psi^*}$	$\frac{\Psi_{AVG}}{\Psi^*}$	$\frac{\Psi_{EG}}{\Psi^*}$
1	1.068	1.135	1.530	2.346
2	1.104	1.197	1.645	2.309
3	1.133	1.240	1.701	2.273

contrast, EG and AVG largely deviate from the optimal charging curve, being either too aggressive or too conservative depending on the arrival patterns. In general, all charging algorithms perform better when the traffic is relatively light, except for EG. It produces even the worst performance ratio under light traffic. This is partly because its aggressive charging scheme somehow matches with the large traffic variations in scenario 3.

2.5.3 Setting a Proper q

Theoretically, setting q to be 1.46 will achieve the best ratio in the worst case. However, it does not achieve the best average performance in general. In this subsection, we discuss how q affects the normalized average performance ratio. For the three scenarios with different traffic, we plot the normalized average performance ratio in Fig. 2.10 by varying q from 1 to 5. For scenario 1, setting $q = 1.8$, $\frac{\Psi_{ORC}}{\Psi^*}$ achieves the lowest average ratio 1.053. For scenario 2, setting $q = 2.1$, $\frac{\Psi_{ORC}}{\Psi^*}$ achieves the lowest average ratio 1.052. For scenario 3, setting $q = 2.3$, $\frac{\Psi_{ORC}}{\Psi^*}$ achieves the lowest average ratio 1.050, which is about 8 % lower than that when $q = 1.46$. In general, the optimal q is larger when the traffic is heavy and unpredictable as in scenario 3. Intuitively, this is because the energy cost during

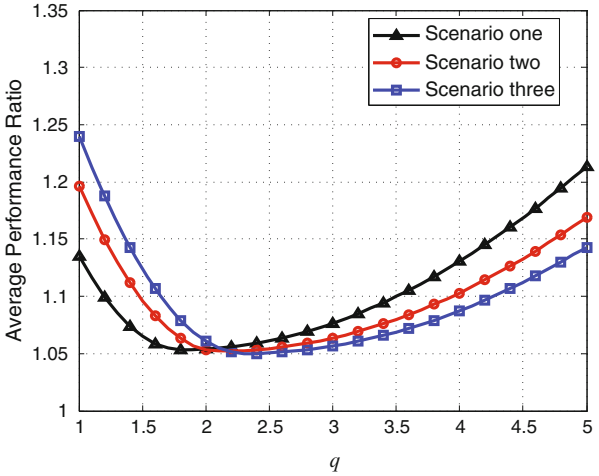

Fig. 2.10 Average performance ratios of ORCHARD in three scenarios with varied q

peak arrivals largely dominates the overall cost. A larger q is able to better utilize off-peak hour and to speed up charging when peak hours arrive. Here, we provide a simple method to achieve a better average performance by adjusting q. In practice, a charging station can collect the past data on PEV patterns, based on which, the value of q can be searched for the best average performance.

2.5.4 Complexity of Low-Complexity Solution Algorithm

To verify the iteration complexity of the low-complexity solution algorithm, we adopt it to solve problem (2.8). For the system parameter, we use the same settings with the default settings except the arrival rates, which are assumed to be the same during $8:00-18:00$ and 0 after $18:00$. We vary the arrival rate in $8:00-18:00$ from 1 to 10 (PEVs/hour) that leads to the mean of N(the number of total PEVs one day) varies from 10 to 100. For each specified mean of N, we simulate 10000 times and compute the average number of iterations and operations. The result is shown in Fig. 2.11. We also fit the data of iterations and operations with the polynomial function $f(x) = 2019.5x^4 - 297.9x^3 + 18.6x^2 - 0.2x$ and $f(x) = 166010x^5 - 26270x^4 + 1200x^3 - 10x^2$ with the *mean relative error* 0.039 and 0.054 respectively. It shows that both the iteration and operation complexity match our complexity analysis in Section IV.C.

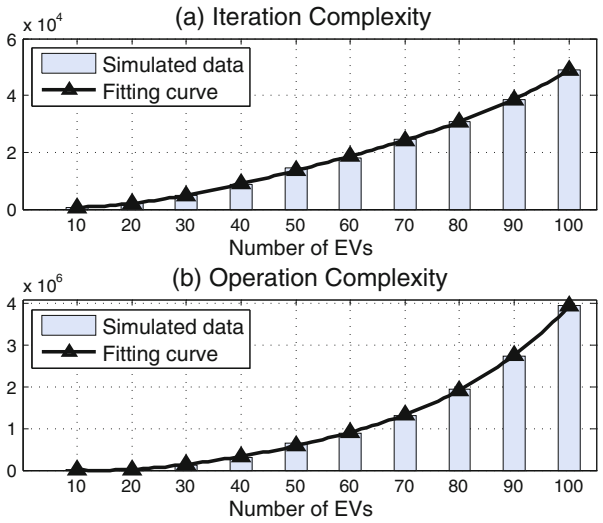

Fig. 2.11 Iterations and operations of low complexity algorithm

2.6 Conclusions

In this chapter, we have proposed an Online cooRdinated CHARging Decision (ORCHARD) algorithm, which minimizes the energy cost and flattenes the total electric load profile without knowing the future information. Through rigorous proof, we showed that ORCHARD is strictly feasible in the sense that it guarantees to fulfill all charging demands before due time. Meanwhile, it achieves the best known competitive ratio of 2.39 when the cost function is a quadratic function of the load demand. To further reduce the computational complexity of the algorithm, we proposed a novel reduced-complexity algorithm to replace the standard convex optimization techniques used in ORCHARD. Through extensive simulations, we showed that the average performance gap between ORCHARD and the optimal offline solution, which utilizes the complete future information, is as small as 6.5 %. By setting proper speeding factor, the average performance gap can be further reduced to less than 5 %.

Appendix

Proof of Lemma 2.1

Proof. The proof is given by contradiction. The optimal total charging rate at time $t \in [t_k, t_{k+1})$ is denoted by $\tilde{s}_k(t)$, where

$$\tilde{s}_k(t) = \sum_{i \in I(t_k)} x_{it}^*, k = 1, \ldots, K. \tag{2.33}$$

Let

$$s_k = \frac{\int_{t_k}^{t_{k+1}} \tilde{s}_k(t) dt}{t_{k+1} - t_k} \tag{2.34}$$

be the average charging rate in δ_k. Note that s_k is always achievable by setting the charging rate of each EV i as $\int_{t_k}^{t_{k+1}} x_{it}^* dt / (t_{k+1} - t_k)$. By Jensen's inequality, we have

$$\begin{aligned} &\int_{t_k}^{t_{k+1}} \frac{a(\hat{s}_k(t) + l_k) + b(\hat{s}_k(t) + l_k)^2}{t_{k+1} - t_k} dt \\ \geq & a \frac{\int_{t_k}^{t_{k+1}} \hat{s}_k(t) + l_k dt}{t_{k+1} - t_k} + b \left[\frac{\int_{t_k}^{t_{k+1}} \hat{s}_k(t) + l_k dt}{t_{k+1} - t_k} \right]^2 \\ = & a(s_k + l_k) + b(s_k + l_k)^2. \end{aligned} \tag{2.35}$$

Equivalently, we have

$$\begin{aligned} &\int_{t_k}^{t_{k+1}} \left[a(\hat{s}_k(t) + l_k) + b(\hat{s}_k(t) + l_k)^2 - (al_t + bl_t^2) \right] dt \\ &\geq (t_{k+1} - t_k) \left(a(s_k + l_k) + b(s_k + l_k)^2 - (al_k + bl_k^2) \right). \end{aligned} \tag{2.36}$$

From (2.36), the uniform total charging rate s_k incurs no higher cost than that of x_{it}^*, which contradicts with the assumption that x_{it}^* is the optimal charging schedule. Therefore, the optimal charging schedule must produce constant total charging rate in each interval δ_k, which completes the proof. ■

Proof of Lemma 2.2

To see this, the charging rate of EV i is

$$\hat{x}_{it} = \min\{\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i \in \bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \cdot \frac{q-1}{q} \hat{s}_t, U_i\}. \tag{2.37}$$

From step 5 of ORCHARD, the total charing rate is

$$\hat{s}_t = q \cdot \sum_{i \in \bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j), \tag{2.38}$$

if

$$q \cdot \sum_{i \in \bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j) \leq \sum_{i \in \bar{\mathscr{I}}_{t_j}} U_i. \tag{2.39}$$

In this case, the charging rate of PEV i is

$$\begin{aligned}
&\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i \in \bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \frac{q-1}{q} \hat{s}_t \\
=&\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i \in \bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \cdot (q-1) \sum_{i \in \bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j) \\
\leq&\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i \in \bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \cdot \Big(\sum_{i \in \bar{\mathscr{I}}_{t_j}} U_i - \sum_{i \in \bar{\mathscr{I}}_{t_j}} \bar{x}_{i1}(t_j) \Big) \\
=&U_i.
\end{aligned} \tag{2.40}$$

Otherwise if (2.39) does not hold, the total online charging rate is

$$\hat{s}_t = \sum_{i \in \bar{\mathscr{I}}_{t_j}} U_i. \tag{2.41}$$

Then, we have

$$\bar{x}_{i1}(t_j) + \frac{U_i - \bar{x}_{i1}(t_j)}{\sum_{i \in \bar{\mathscr{I}}_{t_j}} (U_i - \bar{x}_{i1}(t_j))} \cdot \frac{q-1}{q} \hat{s}_t. \tag{2.42}$$

By step 4, $\hat{x}_{it} = U_i$, and also it is easy to verify that $\sum_{i \in \bar{\mathscr{I}}_{t_j}} \hat{x}_{it} = \hat{s}_t$. To sum up, in any case, the following constraints are satisfied, i.e.

$$\bar{x}_{i1}(t_j) \leq \hat{x}_{it} \leq U_i, \quad \sum_{i \in \bar{\mathscr{I}}_{t_j}} \hat{x}_{it} = \hat{s}_t. \tag{2.43}$$

On the other hand, the charging schedule $\hat{x}_{it}$ can finish the charging of all EV's before their departures. This is because it is no slower than the optimal charging schedule $\bar{x}_{i1}(t_j)$, which guarantees the feasibility of (2.7). ■

Proof of Lemma 2.3

Based on the definition, we have following two inequalities

$$\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \le \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.44a}$$

$$\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} \le \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.44b}$$

which hold because all the PEVs with deadlines in $[\tau_0,\tau_1]$ must park in the station at current time τ_0 such that $\sum_{i\in\mathscr{I}(\tau_0)} U_i$ is larger or equal to $\sum_{i\in\mathscr{I}(t)} U_i$ for $t\in(\tau_0,\tau_1]$. Due to the setting of $\hat{s}$ in our online algorithm, either the inequality

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \le \hat{s} < \sum_{i\in\mathscr{I}(\tau_0)} U_i \tag{2.45}$$

or

$$\hat{s} = \sum_{i\in\mathscr{I}(\tau_0)} U_i \le q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \tag{2.46}$$

holds. Similarly, for optimal total charging rate s^* in offline algorithm, the inequality

$$\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} \le s^* \le \sum_{i\in\mathscr{I}(\tau_0)} U_i \tag{2.47}$$

holds since $w^*(\tau_0,t)$ does not include the demand of the future coming PEVs while s^* dose. From the definition of g_k, we get that

$$\begin{aligned} g_0 = \max\Big\{0, \min\{\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0}, \frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i\} - \\ \min\{\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}, \sum_{i\in\mathscr{I}(\tau_0)} U_i\}\Big\} \end{aligned} \tag{2.48}$$

To further reduce g_0, we need to discuss the following four cases.

Case 1: If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \ge \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} = \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.49}$$

then from (2.46), (2.47) we get

$$\hat{s} = s^* = \sum_{i\in\mathscr{I}(\tau_0)} U_i \tag{2.50}$$

and

$$g_0 = \max\left\{0, \frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i - \sum_{i\in\mathscr{I}(\tau_0)} U_i\right\} = 0. \tag{2.51}$$

Hence,

$$qg_0 = 0 \le \hat{s} = \sum_{i\in\mathscr{I}(\tau_0)} U_i \le q\sum_{i\in\mathscr{I}(\tau_0)} U_i = qg_0 + qs^*. \tag{2.52}$$

Case 2: If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} = \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.53}$$

then from (2.45), (2.47) we get

$$\hat{s} \le s^* = \sum_{i\in\mathscr{I}(\tau_0)} U_i \tag{2.54}$$

and

$$g_0 = \max\left\{0, \frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} - \sum_{i\in\mathscr{I}(\tau_0)} U_i\right\} = 0. \tag{2.55}$$

Hence,

$$qg_0 = 0 \le \hat{s} \le q\sum_{i\in\mathscr{I}(\tau_0)} U_i = qg_0 + qs^*. \tag{2.56}$$

Case 3: If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \ge \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.57}$$

then from (2.46), (2.47) we get

$$s^* \le \hat{s} = \sum_{i\in\mathscr{I}(\tau_0)} U_i \tag{2.58}$$

and

$$g_0 = \max\left\{0, \frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i - \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}\right\}. \tag{2.59}$$

If

$$\frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i \le \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}, \tag{2.60}$$

then we get $g_0 = 0$, and

$$qg_0 = 0 \le \hat{s} = \sum_{i\in\mathscr{I}(\tau_0)} U_i \le q\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} \le qs^* = qg_0 + qs^*. \tag{2.61}$$

If

$$\frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i > \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}, \tag{2.62}$$

then we get

$$g_0 = \frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i - \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}. \tag{2.63}$$

Hence, from (2.58) we have

$$qg_0 = \sum_{i\in\mathscr{I}(\tau_0)} U_i - q\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} \le \hat{s}, \tag{2.64a}$$

and from (2.47) we have

$$qg_0 + qs^* = \sum_{i\in\mathscr{I}(\tau_0)} U_i - q\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} + qs^* \ge \sum_{i\in\mathscr{I}(\tau_0)} U_i = \hat{s}. \tag{2.65a}$$

Since (2.21) holds for both cases, we see that (2.21) holds in Case 3.
Case 4: If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.66}$$

we get

$$g_0 = \max\left\{0, \frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} - \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}\right\}. \tag{2.67}$$

When $\hat{w}(\tau_0,\tau_1) \ge w^*(\tau_0,\tau_1)$, (2.48) is reduced to

$$g_0 = \frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} - \frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0}. \tag{2.68}$$

Recall that $\hat{s} = qs^{OA}$ where s^{OA} is the total charging rate of OA algorithm given the current demand $\hat{w}(\tau_0, \tau_1)$, and s^* is the charging rate given $w^*(\tau_0, \tau_1)$ and possible future arrivals of other PEVs. Notice that both $\hat{w}(\tau_0, \tau_1)$ and $w^*(\tau_0, \tau_1)$ do not include the charging demand of future coming PEVs, and the difference between s^{OA} and $\hat{w}(\tau_0, \tau_1)/(\tau_1 - \tau_0)$ is only resulted from the bounds of current PEVs, while the difference between s^* and $w^*(\tau_0, \tau_1)/(\tau_1 - \tau_0)$ is due to the bounds of current PEVs as well as the possible heavy load of future coming PEVs. Therefore,

$$s^{OA} - \frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} \leq s^* - \frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0} \tag{2.69}$$

As $\hat{s} = qs^{OA}$, we have

$$\hat{s} - q\frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} \leq qs^* - q\frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0}. \tag{2.70}$$

Hence we get the following inequalities:

$$qg_0 = q\frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} - q\frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0} \leq \hat{s}, \tag{2.71a}$$

$$\begin{aligned} qg_0 + qs^* &\geq \left(q\frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} - q\frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0}\right) \\ &+ \left(\hat{s} - q\frac{\hat{w}(\tau_0, \tau_1)}{\tau_1 - \tau_0} + q\frac{w^*(\tau_0, \tau_1)}{\tau_1 - \tau_0}\right) = \hat{s}, \end{aligned} \tag{2.71b}$$

where the last inequality of (2.71a) and the first inequality of (2.71b) are derived from (2.45) and (2.70) respectively. For the case that $\hat{w}(\tau_0, \tau_1) < w^*(\tau_0, \tau_1)$, we have $g_0 = 0$ and $\hat{s} \leq qs^*$ by adding on left hand side of (2.70) $q\hat{w}(\tau_0, \tau_1)/(\tau_1 - \tau_0)$ and right hand side $qw^*(\tau_0, \tau_1)/(\tau_1 - \tau_0)$. Therefore, (2.21) holds in case 4. Finally, inequality (2.21) holds in all the four cases. This completes the proof. ■

Proof of Theorem 2.1

Proof. We can derive from (2.20) that

$$\begin{aligned}\frac{\mathrm{d}\Phi}{\mathrm{d}\tau_0} =& \beta_1 \cdot a \sum_{k=0}^{\infty} \frac{\mathrm{d}\left[(\tau_{k+1}-\tau_k)g_k\right]}{\mathrm{d}\tau_0} \\ &+ \beta_2 \cdot b \sum_{k=0}^{\infty} \frac{\mathrm{d}\left[(\tau_{k+1}-\tau_k)g_k^2\right]}{\mathrm{d}\tau_0}.\end{aligned} \tag{2.72}$$

When $\hat{w}(\tau_0,\tau_1) < w^*(\tau_0,\tau_1)$, we divide into following four cases to prove that $g_0 = 0$, where τ_1 is infinity.

1. If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \geq \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} = \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.73}$$

   then (2.51) implies that $g_0 = 0$.
2. If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} = \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.74}$$

   then (2.55) implies that $g_0 = 0$.
3. If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \geq \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.75}$$

   then

$$q\frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} \geq q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} \geq \sum_{i\in\mathscr{I}(\tau_0)} U_i. \tag{2.76}$$

   Hence,

$$g_0 = \max\left\{0, \frac{1}{q}\sum_{i\in\mathscr{I}(\tau_0)} U_i - \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}\right\} = 0. \tag{2.77}$$

4. If

$$q\frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i \text{ and } \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0} < \sum_{i\in\mathscr{I}(\tau_0)} U_i, \tag{2.78}$$

   then

$$g_0 = \max\left\{0, \frac{\hat{w}(\tau_0,\tau_1)}{\tau_1-\tau_0} - \frac{w^*(\tau_0,\tau_1)}{\tau_1-\tau_0}\right\} = 0. \tag{2.79}$$

Hence, $g_0 = 0$ holds when $\hat{w}(\tau_0, \tau_1) < w^*(\tau_0, \tau_1)$. Then $\mathrm{d}\Phi/\mathrm{d}\tau_0$ remains zero and $\hat{s} \leq qs^*$ by Lemma 2.3. Then, (2.15) always holds by letting $q^2 \leq c$. Therefore, we only consider the case that $\hat{w}(\tau_0, \tau_1) \geq w^*(\tau_0, \tau_1)$ with $q^2 \leq c$. For the speed scaling problem in [15], since there is no constraint of scheduling rate for each individual job, both the online and offline algorithm can always have a solution that only schedule one job that the load intensity gap varies only in at most two time intervals. However, in our problem, since for any PEV, its charging rate can not exceed the maximum charging rate, this leads to that the scheduler should at least charging one PEV at time τ_0. Then we should compute the differential of intensity gap for all intervals and then combine them together. For the time interval $[\tau_0, \tau_1]$, we have

$$\begin{aligned} &\frac{\mathrm{d}(\tau_1 - \tau_0)g_0}{\mathrm{d}\tau_0} \\ =&(\tau_1 - \tau_0)\frac{(\tau_1 - \tau_0)\frac{\mathrm{d}d(\tau_0,\tau_1)}{\mathrm{d}\tau_0} + d(\tau_0, \tau_1)}{(\tau_1 - \tau_0)^2} - g_0^2 \\ =&\frac{\mathrm{d}d(\tau_0, \tau_1)}{\mathrm{d}\tau_0} \end{aligned} \tag{2.80}$$

and

$$\begin{aligned} &\frac{\mathrm{d}(\tau_1 - \tau_0)g_0^2}{\mathrm{d}\tau_0} \\ =&2g_0(\tau_1 - \tau_0)\frac{(\tau_1 - \tau_0)\frac{\mathrm{d}d(\tau_0,\tau_1)}{\mathrm{d}\tau_0} + d(\tau_0, \tau_1)}{(\tau_1 - \tau_0)^2} - g_0^2 \\ =&2g_0\frac{\mathrm{d}d(\tau_0, \tau_1)}{\mathrm{d}\tau_0} + g_0^2. \end{aligned} \tag{2.81}$$

For the time interval $(\tau_k, \tau_{k+1}], k = 1, 2, \ldots$, we have

$$\begin{aligned} \frac{\mathrm{d}((\tau_{k+1} - \tau_k)g_k)}{\mathrm{d}\tau_0} =&\frac{\mathrm{d}d(\tau_k, \tau_{k+1})}{\mathrm{d}\tau_0} \\ <&\frac{\mathrm{d}d(\tau_k, \tau_{k+1})}{\mathrm{d}\tau_0}, \end{aligned} \tag{2.82}$$

and

$$\begin{aligned} \frac{\mathrm{d}((\tau_{k+1} - \tau_k)g_k^2)}{\mathrm{d}\tau_0} =&2g_k\frac{\mathrm{d}d(\tau_k, \tau_{k+1})}{\mathrm{d}\tau_0} \\ <&2g_0\frac{\mathrm{d}d(\tau_k, \tau_{k+1})}{\mathrm{d}\tau_0}, \end{aligned} \tag{2.83}$$

where the last inequality holds because $g_k < g_0, \forall k > 0$. Summing up (2.80), (2.81), (2.82) and (2.83), $\mathrm{d}\Phi/\mathrm{d}\tau_0$ is upper bounded by

$$\begin{aligned}&\beta_1 a\left(\sum_{k=0}^{\infty}\frac{\mathrm{d}d(\tau_k,\tau_{k+1})}{\mathrm{d}\tau_0}\right)+\beta_2 b\left(2g_0\sum_{k=0}^{\infty}\frac{\mathrm{d}d(\tau_k,\tau_{k+1})}{\mathrm{d}\tau_0}+g_0^2\right)\\&=\beta_1 a(-\hat{s}+s^*)+\beta_2 b\left(2g_0(-\hat{s}+s^*)+g_0^2\right).\end{aligned}\tag{2.84}$$

Then, to prove (2.15), it suffices to show that the following inequality holds, where

$$\begin{aligned}&(a(\hat{s}+l)+b(\hat{s}+l)^2-(al+bl^2))+(\beta_2 b(2g_0(-\hat{s}+s^*)+g_0^2)\\&+\beta_1 a(-\hat{s}+s^*))-c(a(s^*+l)+b(s^*+l)^2-(al+bl^2))\le 0.\end{aligned}\tag{2.85}$$

It also suffices to show that the following two inequalities hold, where

$$(a+2bl)\hat{s}+\beta_1 a(-\hat{s}+s^*)-c\cdot(a+2bl)s^*\le 0,\tag{2.86a}$$

$$b(\hat{s})^2+\beta_2 b(2g_0(-\hat{s}+s^*)+g_0^2)-c\cdot b(s^*)^2\le 0.\tag{2.86b}$$

Notice that the LHS of (2.86a) is a linear function of $\hat{s}$. Therefore, it suffices to show that (2.86a) holds for all $s^*\ge 0$ and $g_0\ge 0$, when $\hat{s}=qg_0$ and $\hat{s}=q(s^*+g_0)$, i.e.,

$$(1+2bl/a-\beta_1)qg_0+(\beta_1-c(1+2bl/a))s^*\le 0\tag{2.87a}$$

$$(1-\beta_1)(qg_0+qs^*)+(\beta_1-c)s^*\le 0.\tag{2.87b}$$

Since $c\ge 1$ and $q\ge 1$, by setting $\beta_1=1+2bl/a$, (2.87) holds for all $s^*\ge 0$ and $g_0\ge 0$. Note that β_1 is finite since a,b, and l are finite number. Similarly, the LHS of (2.86b) is a convex function of $\hat{s}$. Therefore, it suffices to show that (2.86b) holds for all $s^*\ge 0$ and $g_0\ge 0$ when $\hat{s}=qg_0$ and $\hat{s}=q(s^*+g_0)$. To obtain the lowest competitive ratio, we need to determine the values of q where $1\le q^2\le c$ and β_2 that minimize c. This can be achieved by using the numerical method in [14]. We do not present the detailed steps but only the numerical results. That is, the optimal parameters are $q=1.46$ and $\beta_2=2.7$, where the lowest competitive ratio is 2.39. ■

Proof of Lemma 2.4

We give the proof by contradiction. Note that, in the mth iteration, one of the candidate interval sets is the union of intervals sets $\mathscr{K}^*(m)$ and $\mathscr{K}^*(m+1)$, denoted by $\mathscr{K}'$, i.e.,

$$\mathscr{K}'=\mathscr{K}^*(m)\cup\mathscr{K}^*(m+1).\tag{2.88}$$

Note that the interval set $\mathscr{K}^*(m)$ and $\mathscr{K}^*(m+1)$ have no intersections, i.e.,

$$\mathscr{K}^*(m) \cap \mathscr{K}^*(m+1) = \emptyset, \tag{2.89}$$

then the total energy of $\mathscr{K}'$ is the sum of energy of $\mathscr{K}^*(m)$ and $\mathscr{K}^*(m+1)$, i.e., $y^*(m)\Delta^*(m) + y^*(m+1)\Delta^*(m+1)$. Thus, the total load of the interval set $\mathscr{K}'$ is the balanced the energy over all the intervals in $\mathscr{K}'$, that is,

$$\begin{aligned} y &= \frac{y^*(m)\Delta^*(m) + y^*(m+1)\Delta^*(m+1)}{\Delta^*(m) + \Delta^*(m+1)} \\ &> y^*(m). \end{aligned} \tag{2.90}$$

where the last inequality holds because $y^*(m) < y^*(m+1)$. Thus, it makes a contradiction with $y \leq y^*(m)$ since $y^*(m)$ is the highest total charging rate over all candidate interval sets in iteration m. This completes the proof. ■

Proof of Theorem 2.2

Proof. For any PEV i, assume that there exist intervals $k_1, k_2, k_3 \in \mathscr{J}(i)$ where $x^*_{ik_1} = 0$, $x^*_{ik_2} \in (0, U_i)$ and $x^*_{ik_3} = U_i$. We separate the proof into the following three parts to match with the three cases of KKT optimality conditions:

1. Interval k_1 must be excluded before interval k_2 and interval k_3 since when schedule $x^*_{ik_1} = 0$ from (2.31), the considered PEV i has not been scheduled that interval k_2 and k_3 should be reserved and goto next iteration. By Lemma 2.4, we have $y^*_{k_1} \geq y^*_{k_2}$ and $y^*_{k_1} \geq y^*_{k_3}$.
2. Interval k_2 must be excluded before interval k_3 since when schedule $x^*_{ik_2}$ and $x^*_{ik_3}$ from (2.29), interval k_2 belongs to the interval set with highest total charging rate and will be excluded in the current iteration, while interval k_3 should be reserved to next iteration. Similarly, by Lemma 2.4 we have $y^*_{k_2} \geq y^*_{k_3}$.
3. For any other interval $k' \in \mathscr{J}(i)$ with $x^*_{ik'} \in (0, U_i)$, $y^*_{k'}$ is the same as $y^*_{k_2}$ because both k' and k_2 belongs to the set $\mathscr{K}^*$ in the same iteration by (2.29) and are assigned the same optimal total charging rate from (2.28).

Therefore, our algorithm satisfies the *KKT* conditions that the solution is always global optimal. ■

References

1. Z. Ma, D. Callaway, and I. Hiskens, "Decentralized charging control of large populations of plug-in electric vehicles," *IEEE Trans. on Control Systems Technology*, vol. 21, no. 1, pp. 67–78, 2013.
2. K. Mets, T. Verschueren, W. Haerick, C. Develder, and F. D. Turck, "Optimizing smart energy control strategies for plug-in hybrid electric vehicle charging," *in Proc. IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, pp. 293–299, Apr. 2010.
3. Y. He, B. Venkatesh, and L. Guan, "Optimal scheduling for charging and discharging of electric vehicles," *IEEE Trans. on Smart Grid*, vol.3, no.3, pp. 1095–1105, 2012.
4. L. Gan, U. Topcu, and S. H. Low, "Optimal decentralized protocol for electric vehicle charging," *IEEE Trans. on Power System*, vol.28, iss. 2, pp. 940–951, 2012.
5. B. Allan and E. Ran, *Online Computation and Competitive Analysis*, Cambridge, U.K.: Cambridge Univ. Press, 1998.
6. E. Gerding, V. Robu, S. Stein, D. Parkes, A. Rogers, and N. Jennings, "Online mechanism design for electric vehicle charging," *in Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, pp. 811–818, May 2011.
7. M. A. S. Masoum, P. S. Moses, and S. Hajforoosh, "Distribution transformer stress in smart grid with coordinated charging of plug-in electric vehicles," *IEEE Power Energy Syst. Innovative Smart Grid Tech. Conf.*, pp. 1–8, 2012.
8. 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, 2010.
9. S. Chen and L. Tong, "iEMS for large scale charging of electric vehicles architecture and optimal online scheduling," *in Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, pp. 629–634, Nov. 2012.
10. V. Robu, S. Stein, E. H. Gerding, D. C. Parkes, A. Rogers, and N. R. Jennings, "An online mechanism for multi-speed electric vehicle charging," *in Second International Conference on Auctions, Market Mechanisms and Their Applications (AMMA'11)*, pp. 100–112, 2012.
11. S. Stein, E. Gerding, V. Robu, and N. R. Jennings, "A model-based online mechanism with pre-commitment and its application to electric vehicle charging," *in Proce. of the 11th Int. Conf. on Autonomous Agents and Multiagent Systems,* pp. 669–676, 2012.
12. F. Yao, A. Demers, and S. Shenker, "A scheduling model for reduced CPU energy," *in Proc. IEEE Symp. Foundations of Computer Science*, pp. 374–382, 1995.
13. N. Bansal, T. Kimbrel, and K. Pruhs, "Speed scaling to manage energy and temperature," *Journal of the ACM (JACM)*, vol. 54, no. 1, pp. 1–39, 2007.
14. N. Bansal, H. L. Chan, K. Pruhs, and D. Katz, "Improved bounds for speed scaling in devices obeying the cube-root rule," *Proc. 36th Int. Colloqium on Automata, Languages and Programming: Part I*, pp. 144–155, Jul. 2009.
15. N. Bansal, H. L. Chan, and K. Pruhs, "Speed scaling with an arbitrary power function," *In Proc. of the 20th ACM-SIAM Symposium on Discrete Algorithm*, pp. 693–701, 2009.
16. T. W. Lam, L. K. Lee, Isaac K. K. To, and Prudence W. H. Wong, "Speed scaling functions for flow time scheduling based on active job count," *Algorithms-ESA 2008*, pp. 647–659, 2008.
17. S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge, U.K.: Cambridge Univ. Press, 2004.
18. Y. Ye, *Interior Point Algorithms: Theory and Analysis*, Wiley-Interscience Press, 1997.
19. A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power and Energy Mag.*, vol. 7, no. 2, pp. 52–62, 2009.
20. A. Santos, A. N. McGuckin, H. Y. Nakamoto, D. Gray, and S. Lis, *Summary of Travel Trends: 2009 National Household Travel Survey*, Federal Highway Administration, Washington, DC, 2011.
21. M. Grant and S. Boyd, CVX: Matlab Software for Disciplined Convex Programming [Online]. Available: http://cvxr.com/cvxMar. 2013, Version 2.0 (beta).

http://www.springer.com/978-3-319-45861-8

Optimal Charging Control of Electric Vehicles in Smart Grids

Tang, W.; Zhang, Y.-J.A.

2017, XI, 106 p. 24 illus., 23 illus. in color., Softcover
ISBN: 978-3-319-45861-8