Tackling the Load Uncertainty Challenges for Energy Consumption Scheduling in Smart Grid

Pedram Samadi, Student Member, IEEE, Hamed Mohsenian-Rad, Member, IEEE, and Vincent W.S. Wong, Senior Member, IEEE, and Robert Schober, Fellow, IEEE

Abstract—In this paper, we propose a novel optimization-based real-time residential load management algorithm that takes into account load uncertainty in order to minimize the energy payment for each user. Unlike most existing demand side management algorithms that assume perfect knowledge of users’ energy needs, our design only requires knowing some statistical estimates of the future load demand. Moreover, we consider real-time pricing combined with inclining block rate tariffs. In our problem formulation, we take into account different types of constraints on the operation of different appliances such as must-run appliances, controllable appliances that are interruptible, and controllable appliances that are not interruptible. Our design is multi-stage. As the demand information of the appliances is gradually revealed over time, the operation schedule of controllable appliances is updated accordingly. Simulation results confirm that the proposed energy consumption scheduling algorithm can benefit both users, by reducing their energy expenses, and utility companies, by improving the peak-to-average ratio of the aggregate load demand.

Keywords: Demand side management, energy consumption control, cost minimization, load uncertainties, smart power grid.

I. LIST OF VARIABLES USED IN THIS PAPER

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>Set of appliances</td>
</tr>
<tr>
<td>$T$</td>
<td>Number of time slots</td>
</tr>
<tr>
<td>$\gamma_a$</td>
<td>Nominal power of appliance $a$</td>
</tr>
<tr>
<td>$E_a$</td>
<td>Total required energy of appliance $a$</td>
</tr>
<tr>
<td>$\beta_a$</td>
<td>Operating deadline of appliance $a$</td>
</tr>
<tr>
<td>$x^0_a$</td>
<td>State of power consumption of appliance $a$ at time slot $t$</td>
</tr>
<tr>
<td>$E^0_a$</td>
<td>Remaining required energy of appliance $a$ at time slot $t$</td>
</tr>
<tr>
<td>$l_t$</td>
<td>Total household power consumption at time slot $t$</td>
</tr>
<tr>
<td>$\lambda_t$</td>
<td>Price function at time slot $t$</td>
</tr>
<tr>
<td>$m_k$</td>
<td>Price parameter at time slot $t$</td>
</tr>
<tr>
<td>$n_k$</td>
<td>Price parameter at time slot $t$</td>
</tr>
<tr>
<td>$b_k$</td>
<td>Price parameter at time slot $t$</td>
</tr>
<tr>
<td>$\mathcal{M}_{k,t}$</td>
<td>Set of must-run appliances that are awake at time slot $t$ and remain awake at time slot $k \geq t$</td>
</tr>
<tr>
<td>$\mathcal{M}_{k,t}$</td>
<td>Set of must-run appliances that are asleep at time slot $t$ and will be awake at time slot $k \geq t$</td>
</tr>
<tr>
<td>$\mathcal{N}_k$</td>
<td>Set of all must-run appliances that are awake at time slot $k$ ($\mathcal{M}<em>{k,t} \cup \mathcal{M}</em>{k,t}$)</td>
</tr>
<tr>
<td>$C_{k,t}$</td>
<td>Set of controllable appliances that are asleep at time slot $t$ and will be awake at time slot $k \geq t$</td>
</tr>
<tr>
<td>$\tilde{C}_{k,t}$</td>
<td>Set of controllable appliances that are asleep at time slot $t$ and will be awake at time slot $k \geq t$</td>
</tr>
<tr>
<td>$\tilde{C}_k$</td>
<td>Set of all controllable appliances that are awake at time slot $k$ ($\tilde{C}<em>{k,t} \cup \tilde{C}</em>{k,t}$)</td>
</tr>
<tr>
<td>$\tilde{N}_k$</td>
<td>Set of all non-interruptible appliances of $\tilde{C}_{k,t}$</td>
</tr>
<tr>
<td>$\tilde{N}_k$</td>
<td>Set of all non-interruptible appliances of $\tilde{C}_{k,t}$</td>
</tr>
<tr>
<td>$N_{k,t}$</td>
<td>Set of all non-interruptible appliances of $C_{k,t}$</td>
</tr>
<tr>
<td>$s_t$</td>
<td>Set of all appliances that are sleeping at time slot $t$</td>
</tr>
<tr>
<td>$y^k_a$</td>
<td>Auxiliary variable for each non-interruptible appliance $a$ at each time slot $k$</td>
</tr>
<tr>
<td>$\nu_k$</td>
<td>Auxiliary variable for each time slot $k$</td>
</tr>
<tr>
<td>$p^k_t$</td>
<td>Probability with which appliance $a$ becomes awake at time slot $t$</td>
</tr>
<tr>
<td>$q^k_a$</td>
<td>Probability that appliance $a$ does not become awake at any time slot $t$</td>
</tr>
<tr>
<td>$p^k_{\gamma,t}$</td>
<td>Probability that appliance $a$ becomes awake at time slot $t \gamma &gt; t$ such that it has not become awake until time slot $t$</td>
</tr>
<tr>
<td>$\delta^k_{\gamma,t}$</td>
<td>Probability that must-run appliance $a$ which is sleeping at time slot $t$ will be active in time slot $\gamma &gt; t$</td>
</tr>
<tr>
<td>$T_a$</td>
<td>Number of time slots required to finish the operation of appliance $a$</td>
</tr>
</tbody>
</table>

II. INTRODUCTION

Demand side management (DSM) programs are employed to better utilize the available power generation capacity and to circumvent installing new generation and transmission infrastructures [1]. DSM programs encourage users to shift their usage of high-power appliances to off-peak hours by providing economic incentives to consumers. Among different techniques considered for DSM (e.g., voluntary load management programs [2]–[7] and direct load control [8]–[10]), smart pricing is known as an effective means to encourage users to consume wisely and more efficiently. By reflecting the hourly changes in the wholesale electricity price to the demand side, users pay what the electricity is worth at different times of day and are consequently more willing to reduce their load at peak hours.

In time differentiating pricing tariffs, the intended operation period is divided into several time slots, and the price of electricity varies across different time slots. For example, the prices may correspond to off-peak, mid-peak, and on-peak hours. The prices are usually higher in the afternoon on hot days in the summer and cold days in the winter [11]. Several time differentiating pricing methods have already been proposed in the literature. Examples include real-time pricing (RTP), time-of-use (TOU) pricing, and day-ahead pricing (DAP) [12]–[16]. These methods mainly differ in how frequently the utility company changes the pricing tariffs.
which may vary from once or twice a year in TOU pricing to hourly changes in RTP. The level of success of different pricing methods depends on various factors such as the amount of information provided to each user, the effectiveness of the mapping of the hourly wholesale prices to the retail prices, and the ability of users to respond to price signals [17].

In general, it is difficult for consumers to follow the real-time prices and respond to their variations accordingly. This aspect and some other disadvantages of manual load control are discussed in [18]. An alternative approach is to equip users with automated control units that respond to real-time price signals to improve the level of rationality of users. Different approaches of the users in responding to price values are studied in [19], [20]. The necessity of more advanced methods to avoid efficiency loss in the system due to enhanced rationality levels of the users has been discussed in [19]. The effect of load synchronization, i.e., the concentration of a large portion of energy consumption in low-price hours, has been studied in [18]. It is shown that load synchronization can be prevented by using pricing tariffs with inclining block rates (IBR). For the IBR tariffs, the marginal price increases with the total consumed power [21]. That is, beyond a predetermined power consumption threshold, electricity is offered at higher rates. This provides incentives for the users to distribute their load across different times of the day. Southern California Edison and Pacific Gas & Electric in United States, and British Columbia Hydro in Canada currently use IBR with various two-level conservation rate structures [22], [23].

Despite its importance, the effect of load uncertainties on DSM has not been well-studied in the smart grid literature [6], [11], [19], [20], [24]–[29]. Therefore, in this paper, we focus on developing a novel automated optimization-based residential load scheduling algorithm in a retail electricity market with load uncertainties. We aim to minimize each user’s electricity payment by optimally scheduling the operation of its appliances in real-time, subject to the operational constraints defined by the users. As in [18], we adopt RTP combined with IBR to better reflect the fluctuation of the wholesale electricity prices and to avoid load synchronization. Our design can be partly compared with [27]. The problem tackled in this paper is different from that in [27] in two aspects. First, the work in [27] addresses uncertainty in price values while we tackle uncertainty in load and users’ energy consumption needs. Second, the key assumption in [27] is that the price values are independent from the load level in each time slot. Here, we relax this assumption. Our work is also different from the heuristic home automation schemes in [30], [31], as we use an optimization-based approach with elaborate mathematical modeling and take into account estimates of the future load to make better decisions. The contributions of this paper can be summarized as follows:

- We propose a real-time residential load management algorithm with load uncertainty for DSM purposes. Our algorithm is based on solving an optimization problem that aims to minimize the electricity payment of residential users. In our system model, each appliance sends an admission request to the energy consumption control unit to start operation. The operation of each appliance is subject to acceptance of its request. By running a centralized algorithm, the control unit determines the optimal operation schedule of each appliance in each time slot.
- We study operation constraints to model a variety of appliances including must-run appliances, and interruptible and non-interruptible controllable appliances. The last item refers to those appliances whose operation can be postponed, but once they start operation, they should stay on until they finish their task.
- Simulation results show that our proposed scheduling algorithm with load uncertainty reduces the energy payment of users compared to the case where no scheduling algorithm is adopted. Our proposed scheme also improves the overall power system performance by reducing the peak-to-average ratio (PAR) in aggregate load demand.

The rest of this paper is organized as follows. The system model is introduced in Section III. The problem formulation and algorithm description are presented in Section IV. Simulation results are provided in Section V. The paper is concluded in Section VI.

III. System Model

In this section, we present a mathematical model for real-time residential load scheduling when combined RTP and IBR tariffs are implemented. We assume that price values are informed by the retailer to end users through a digital communication infrastructure. Furthermore, we assume that each user is equipped with a smart meter, which has an energy consumption control (ECC) unit capable of scheduling and adjusting the household energy consumption.

A. Residential Consumers

Consider a residential unit that participates in a DSM program. Let \( A \) denote the set of all appliances in this unit. Each appliance \( a \in A \) can work either as must-run or controllable. Must-run appliances need to start working immediately. For example, we can classify TV and personal computer (PC) as must-run appliances. Clearly, the user should have the freedom to turn on or turn off the TV whenever he wants without the interference of the ECC. In contrast, the operation of controllable appliances can be delayed or interrupted if necessary. Each controllable appliance can be either interruptible or non-interruptible. For a controllable appliance \( a \), if it is non-interruptible, then the ECC may only delay its operation. However, for interruptible appliances, it is not only possible to postpone the operation but also to interrupt the operation when needed and then restore the operation later on. Plug-in electric vehicle (PEV) and washing machine are examples of interruptible and non-interruptible controllable appliances, respectively. We assume that based on the demand requirements of the user, each appliance can be set as must-run or controllable. This setting is decided by the user and can vary from time to time. That is, depending on the preferences of the user, an appliance can be set as a must-run appliance in one day and as a controllable appliance in another day.
We divide the intended operation cycle into $T$ time slots. Each time slot begins with an admission control phase. In this phase, to start the operation of an appliance, an admission request is sent to the ECC unit. Once an admission request is submitted, the state of the appliance changes from sleep to awake. The appliance remains awake until its operation is finished. However, the operation of an awake appliance is subject to acceptance of its admission request and specification of its operation schedule by the ECC unit. The decisions regarding the admission of the requests and the adjustment of the operation of different awake appliances are updated periodically in each admission control phase.

An awake appliance $a$ can be either inactive (with zero power consumption) or active (operating at nominal power $\gamma_a$). We note that the power consumption of each appliance could be different at different cycles of its operation due to the changes in the amount of current being absorbed. However, considering the exact load profile of each appliance adds to the complexity of the model and makes real-time implementation difficult. To tackle this implementation difficulty, similar to [24], [25], and [27], we consider an average power consumption $\gamma_a$ for each appliance. Different operating states of must-run and controllable appliances are shown in Fig. 1.

We note that the operation of different appliances is influenced by the preferences of the user. Different parameters of our model may be considered to capture different types of preferences. For example, our model takes into account the time and the frequency at which each appliance sends admission requests to the ECC unit. Furthermore, we assume that the mode of operation of each appliance, i.e., whether it is must-run or controllable, is not pre-determined. That is, based on the preference of the user, each appliance can work either as must-run or controllable. Moreover, for controllable appliances, the deadline before which the operation of the appliance has to be finished is also determined based on the preference of the user. Other aspects of user preferences, such as the desirable room temperature, can also be considered to enhance energy consumption scheduling; however, adding those aspects will also make the design more complex and less appealing for real-time implementation in practice.

The admission request of each appliance $a$ specifies the total energy $E_a$ needed to finish the operation of the appliance, the operating power $\gamma_a$, and whether the appliance is must-run or controllable. For controllable appliances, the deadline before which the operation of the appliance has to be finished is also determined based on the admission request. For a controllable appliance $a$, if it is not interruptible, the ECC may only delay its operation. However, for interruptible appliances, it is not only possible to postpone the operation but also to interrupt the operation when needed.

We define binary variable $x^a_t \in \{0, 1\}$ as the state of power consumption of appliance $a \in A$ at time slot $t \in \{1, \ldots, T\}$. We set $x^a_t = 1$ if appliance $a$ is admitted to operate in time slot $t$ (i.e., active), otherwise, we set $x^a_t = 0$ (i.e., inactive). Let $E^a_t$ denote the amount of energy required to finish the operation of appliance $a$ while the current time slot is $t$. Note that given $E^a_t$, for each future time slot $k > t > 0$, we have

$$E^a_k = \left[ E^a_t - \gamma_a \sum_{i=t}^{k-1} x^a_i \right]^+. \quad (1)$$

For controllable appliances that are non-interruptible, their operation can be delayed, but once they become active, they must remain active until the end of their operation. Thus, for each non-interruptible controllable appliance $a$, we have

$$x^a_k = 1, \quad \forall k \in \{t, \ldots, \beta_a\}, \quad 0 < E^a_k < E_a. \quad (2)$$

**B. Real-time Pricing**

Let $l_t \triangleq \sum_{a \in A} \gamma_a x^a_t$ denote the total household power consumption at time slot $t$. We consider a pricing function $\lambda_t(l_t)$ which represents the price of electricity in each time slot $t$ as a function of the user’s power consumption in that time slot. For combined RTP and IBR pricing tariffs, the price function $\lambda_t(l_t)$ is defined as [18]:

$$\lambda_t(l_t) = \begin{cases} m_t & \text{if } 0 \leq l_t \leq b_t, \\ n_t & \text{if } l_t > b_t, \end{cases} \quad (3)$$

where $m_t$, $n_t$, and $b_t$ are pre-determined parameters, and we have $m_t \leq n_t$. Recall from Section II that a combined RTP and IBR pricing model can effectively avoid load synchronization [18].

**IV. PROBLEM FORMULATION AND ALGORITHM DESCRIPTION**

In this section, we consider the problem of efficient power scheduling such that the electricity payment of each user is minimized. We assume that only some statistical demand information are known ahead of time. The exact information about the list of appliances that are awake in each time slot, whether they are must-run or controllable, and the deadline by which the operation of each appliance should be finished is revealed only gradually over time. We note that different sets of awake appliances, i.e., must-run and controllable, at future time slot $k > t$, can be separated into two disjoint subsets. The first subset includes those appliances that are awake at the current time slot $t$ and remain awake at time slot $k > t$, i.e., the information about this subset of appliances is known at the current time slot. The second subset consists of those appliances that are asleep at the current time slot $t$ and will
be awake at time slot $k > t$. However, at the current time slot $t$, only some statistical information about this subset of appliances is available. An update is received by the ECC unit at the beginning of each time slot, and the energy consumption schedule of each controllable appliance is adapted accordingly.

### A. Problem Formulation

The optimum operation schedule can be determined if the demand information of all appliances is available at the beginning of the scheduling horizon. However, we assume here that the demand information of the appliances is not known and instead only stochastic information regarding the demand is available a priori. Thus, we formulate a scheduling problem that minimizes the expected energy payment of the user with respect to demand uncertainties. In each time slot $t$, as the demand information of the appliances is updated, the operation schedule of each controllable appliance can be rescheduled, and the optimum power scheduling can be identified in real-time as the solution of the following optimization problem for minimization of the expected cost from the current time slot $t$ onwards:

$$
\minimize_{x_k^a, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\}} \mathbb{E} \left\{ L_t \lambda_t(L_t) + \sum_{k=t+1}^{T} L_{k,t} \lambda_k(L_{k,t}) \right\} \quad (4)
$$

subject to

$$
x_k^a \in \{0, 1\}, \quad \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\},
$$

$$
\sum_{k=t}^{\beta_a} x_k^a = E_t^a, \quad \forall a \in \mathcal{C}_t,
$$

$$
x_k^a = 1, \quad \forall a \in \mathcal{N}_t, \forall k \in \{t, \ldots, \beta_a\},
$$

$$
0 < E_k^a < E_a,
$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation,

$$
L_t = \sum_{a \in \mathcal{M}_t} \gamma_a + \sum_{a \in \mathcal{C}_t} \gamma_a x_t^a,
$$

$$
L_{k,t} = \sum_{a \in \mathcal{M}_{k,t}} \gamma_a + \sum_{a \in \mathcal{C}_{k,t}} \gamma_a x_{k,t}^a + \sum_{a \in \mathcal{G}_{k,t}} \gamma_a x_{k,t}^a, \quad (6)
$$

$x_t^a \triangleq (x_{t,1}^a, \ldots, x_{t,T}^a)$. $E_t^a$ is as in (1), and the definitions of the different sets of appliances $\mathcal{M}_{k,t}$, $\mathcal{M}_t$, $\mathcal{C}_t$, $\mathcal{C}_{k,t}$, $\mathcal{G}_{k,t}$, and $\mathcal{N}_t$ are presented in Section I. We note that $E_t^a$ for a known load $L_t$, while the second term is the expected cost of energy in the upcoming time slots. Each appliance can be either on or off. This is indicated by the first constraint. The second constraint implies that the operation of each appliance should be finished by its deadline. The last constraint guarantees that the operation of non-interruptible appliances will continue after they become active until they finish their job.

### B. Optimal Solution

In our stochastic model, it is possible to devise different objectives and different strategies to schedule the operation of different appliances. The performances of different scheduling strategies are different. However, their different performances may be compared based on their average performance and their worst case performance for different demand requirements of the user. Problem (4) in its current form is difficult to solve as it requires the computation of the expected schedule for currently sleeping appliances$^1$. To tackle this problem, we minimize an upper bound of the objective function. That is, we assume all appliances that become awake in future time slots are must-run appliances. By minimizing the upper bound of the objective function, the risk of loss for the user is minimized. That is, from the user’s electricity payment point of view, the worst performance of the ECC unit for different scheduling strategies is minimized.

$$
\minimize_{x_k^a, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\}} \mathbb{E} \left\{ L_t \lambda_t(L_t) + \sum_{k=t+1}^{T} \mathbb{E} \{ L_{k,t} \lambda_k(L_{k,t}) \} \right\} \quad (7)
$$

subject to

$$
x_k^a \in \{0, 1\}, \quad \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\},
$$

$$
\gamma_a \sum_{k=t}^{\beta_a} x_k^a = E_t^a, \quad \forall a \in \mathcal{C}_t,
$$

$$
x_k^a = 1, \quad \forall a \in \mathcal{N}_t, \forall k \in \{t, \ldots, \beta_a\},
$$

$$
0 < E_k^a < E_a,
$$

where

$$
\tilde{L}_{k,t} = \sum_{a \in \mathcal{M}_{k,t}} \gamma_a + \sum_{a \in \mathcal{C}_{k,t}} \gamma_a x_{k,t}^a + \sum_{a \in \mathcal{G}_{k,t}} \gamma_a x_{k,t}^a, \quad (8)
$$

denotes the load at time slot $k > t$.

Problem (7) is still difficult to solve in its current form since the last constraint is conditioned on the value of $E_k^a$ for $k \in \{t+1, \ldots, \beta_a\}$. From (1), for $k \in \{t+1, \ldots, \beta_a\}$, $E_k^a$ depends on variable $x_i^a$ for $i \in \{t, \ldots, k-1\}$, which is unknown and should be determined. However, by introducing an auxiliary variable $y_k^a \in \{0, 1\}$ for each appliance $a \in \mathcal{N}_t$ and at each time slot $k \in \{t+1, \ldots, \beta_a\}$, the problem formulation in (7) can be re-written in a more tractable form. Here, the auxiliary variable $y_k^a$ indicates whether the operation of appliance $a$ is finished ($y_k^a = 1$) or not ($y_k^a = 0$) at a particular time slot.

$^1$One option to solve problem (4) is to formulate it as a dynamic programming problem. Considering the amount of information required to describe the state of each appliance, i.e., whether the appliance is awake or not, the remaining energy requirements, and the number of time slots remaining to reach the deadline, we may be faced with a huge state space of outcomes. Dynamic programming may suffer from the curse of dimensionality [32].
where \( y_k^a \triangleq (y_1^a, \ldots, y_T^a) \), \( M \) is a large number to give a high weight to the term added to the objective, and \( 0 < \epsilon < \min \left\{ \frac{1}{E_t} \right\} \) is a small constant. We can justify the new constraints as follows. In (13), when \( E_k^a = E_t - \gamma_a \sum_{i=t}^{k-1} x_i^a = 0 \), \( y_k^a \) becomes 1. However, as long as the operation is not finished, i.e., \( E_k^a > 0 \), since \( y_k^a \) appears in the objective of the minimization problem, we have \( y_k^a = 0 \). This is true, since for any value \( E_k^a > 0 \), we have \( \frac{1}{E_k^a} > \epsilon \). In (14), when \( E_k^a = E_a \), we have \( y_k^a = 0 \), and \( x_k^a \) could be either 0 or 1, when \( E_a > E_k^a > 0 \), we have \( y_k^a = 0 \) and \( x_k^a \) has to be 1. However, when \( E_k^a = 0 \), we have \( y_k^a = 1 \), and since \( x_k^a \) appears in the objective of the minimization problem, it has to be 0.

For the price function in (3), since \( m_t \leq n_t \), for a total load \( l_t \) at time slot \( t \), the user’s payment \( l_t \times \lambda_t(l_t) \) is determined as the maximum of the two intersecting lines [18]:

\[
l_t \times \lambda_t(l_t) = \max \left\{ m_t l_t, n_t l_t + (m_t - n_t) b_t \right\}.
\]

Therefore, problem (9) can be reformulated as

\[
\min_{x_t^a, y_t^a, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, \beta_a\}} L_t \lambda_t(L_t) + \sum_{k=t+1}^T \mathbb{E} \left\{ \tilde{L}_{k,t} \lambda_k(\tilde{L}_{k,t}) \right\} + M \sum_{k=t}^T \sum_{a \in \mathcal{C}_t} y_k^a \tag{9}
\]

subject to

\[
x_k^a \in \{0, 1\}, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\}, \tag{10}
\]

\[
y_k^a \in \{0, 1\}, \forall a \in \mathcal{N}_t, \forall k \in \{t, \ldots, T\}, \tag{11}
\]

\[
\gamma_a \sum_{k=t}^{\beta_a} x_k^a = E_t^a, \quad \forall a \in \mathcal{C}_t, \tag{12}
\]

\[
y_k^a + \frac{E_t^a - \gamma_a \sum_{i=t}^{k-1} x_i^a}{E_a} \geq \epsilon, \quad \forall a \in \mathcal{N}_t, \forall k \in \{t, \ldots, \beta_a\}, \tag{13}
\]

\[
x_k^a + y_k^a + \frac{E_t^a - \gamma_a \sum_{i=t}^{k-1} x_i^a}{E_a} \geq 1, \quad \forall a \in \mathcal{N}_t, \forall k \in \{t, \ldots, \beta_a\}, \tag{14}
\]

Finally, by introducing another auxiliary variable, \( \nu_k \), for each time slot \( k \), and by adopting the certainty equivalent approximation technique, i.e., all uncertainties are fixed at their expected value [33], we can re-write problem (16) as

\[
\min_{\nu_t, x_t^a, y_t^a, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\}} \sum_{k=t}^T \nu_k + M \sum_{k=t}^T \sum_{a \in \mathcal{C}_t} y_k^a \tag{18}
\]

subject to (10) – (14),

\[
\nu_t, x_t^a, y_t^a, \forall a \in \mathcal{C}_t, \forall k \in \{t, \ldots, T\}, \tag{19}
\]

where

\[
l_{k,t} \triangleq \sum_{a \in \mathcal{M}_{k,t}} \gamma_a + \sum_{a \in \mathcal{M}_{k,t} \cup \mathcal{C}_{k,t}} \gamma_a. \tag{17}
\]

The solution of optimization problem (18) determines the appropriate scheduling for the operation of controllable appliances. However, for interruptible appliances, only the operation schedule of the current time slot \( t \) will be executed, and the schedule of the future time slots \( t + 1, \ldots, T \) may change when the optimization problem is solved again in the next time slot as new information about the future load becomes available.
C. Load Estimation

In our system model, we assume that the demand information of the appliances is not known ahead of time, i.e., in (17), the set of awake appliances in the upcoming time slots \( k > t \) that are currently sleeping, \( \mathcal{M}_{k,t} \cup \mathcal{C}_{k,t} \), is not known. Instead, only the probability with which each appliance becomes awake at each time slot \( t \), \( p_{k,t}^{a} \), is known before the operation cycle begins. Such information can be calculated, for example, based on the sleep and awake history of each appliance. For this purpose, we can observe a window of \( N \) consecutive days and mark those days in which appliance \( a \) becomes awake in a particular time slot \( t \). The ratio of the number of marked days to the total number of observed days determines the probability with which appliance \( a \) becomes awake in time slot \( t \), \( p_{k,t}^{a} \triangleq \mathbb{P}(\Delta_{t}^{a} = 1) \), where \( \Delta_{t}^{a} \) is a random variable that is equal to one if appliance \( a \) becomes awake in time slot \( t \), and equal to zero otherwise. In our model, each appliance can become awake only once. If an appliance becomes awake more often, we can simply introduce virtual appliances to deal with this issue. Therefore, we have

\[
\sum_{t=1}^{T} p_{k,t}^{a} + q_{a} = 1,
\]

where \( q_{a} \) denotes the probability that appliance \( a \) does not become awake at any time within the DSM’s operation period \([1,T]\). We define \( p_{k,t}^{a} \) as the probability that appliance \( a \) becomes awake in time slot \( \tau > t \) given that it has not become awake until time slot \( t \). That is,

\[
p_{k,t}^{a} = \mathbb{P}(\Delta_{t}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0) = \frac{\mathbb{P}(\Delta_{t}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0) \mathbb{P}(\Delta_{t}^{a} = 1)}{\mathbb{P}(\Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0)}.
\]

Based on Bayes rule, \( p_{k,t}^{a} \) can be calculated as

\[
p_{k,t}^{a} = \frac{\mathbb{P}(\Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0 | \Delta_{\tau}^{a} = 1) \mathbb{P}(\Delta_{\tau}^{a} = 1)}{\mathbb{P}(\Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0)}.
\]

If appliance \( a \) becomes awake at time slot \( \tau > t \), it implies that it has not become awake in previous time slots. Therefore, \( \mathbb{P}(\Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0 | \Delta_{\tau}^{a} = 1) = 1 \). On the other hand, we obtain \( \mathbb{P}(\Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0) = \sum_{k=t+1}^{T} p_{k}^{a} + q_{a} \) based on (19). We also have \( \mathbb{P}(\Delta_{\tau}^{a} = 1) = p_{k,t}^{a} \). Therefore, (21) becomes

\[
\frac{p_{k,t}^{a}}{\sum_{k=t+1}^{T} p_{k}^{a} + q_{a}}.
\]

Next, assume that all appliances that become awake in future time slots are must-run appliances, and must-run appliances start operation once they become awake, see Section IV-A. Let \( \Lambda_{\tau}^{a} \) denote the random variable that indicates whether must-run appliance \( a \) is active (\( \Lambda_{\tau}^{a} = 1 \)) or not active (\( \Lambda_{\tau}^{a} = 0 \)) in time slot \( \tau \). Also, let \( \delta_{\tau,t}^{a} \) denote the probability that a must-run appliance \( a \) which is sleeping in time slot \( t \) will be active in time slot \( \tau > t \). By conditioning on the time slot in which must-run appliance \( a \) becomes awake, \( \delta_{\tau,t}^{a} \) can be calculated as

\[
\delta_{\tau,t}^{a} = \mathbb{P}(\Lambda_{\tau}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0) = \sum_{k=t+1}^{T} \mathbb{P}(\Lambda_{\tau}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0, \Delta_{k}^{a} = 1, \delta_{k,t}^{a} = 0),
\]

Fig. 2. Must-run appliance \( a \in S_{t} \) with \( T_{a} = 3 \) will be active in \( \tau > t \) if it starts operating within time interval \([\max\{t+1, \tau - T_{a} + 1\}, \tau]\).

where \( p_{k,t}^{a} \) is defined in (20). As illustrated in Fig. 2, a currently sleeping appliance will be active in time slot \( \tau > t \), if it starts operation within time frame \([\max\{t+1, \tau - T_{a} + 1\}, \tau]\), where \( T_{a} \triangleq \frac{\tilde{T}_{a}}{\gamma_{a}} \) is defined as the number of time slots required to finish the operation of appliance \( a \) while operating at power level \( \gamma_{a} \). For simplicity, we assume \( T_{a} \) is integer. Therefore, \( \mathbb{P}(\Lambda_{\tau}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0, \Delta_{k}^{a} = 1) = 1 \) if \( k \in [\max\{t+1, \tau - T_{a} + 1\}, \tau] \), and \( \mathbb{P}(\Lambda_{\tau}^{a} = 1 | \Delta_{t}^{a} = 0, \ldots, \Delta_{\tau-1}^{a} = 0, \Delta_{k}^{a} = 0) = 0 \) otherwise. Thus, we have

\[
\delta_{\tau,t}^{a} = \sum_{k=\max\{t+1, \tau - T_{a} + 1\}}^{\tau} p_{k,t}^{a}.
\]

Finally, by conditioning on the event of observing a currently sleeping appliance active in an upcoming time slot \( \tau \), while the system is at time slot \( t \), the estimate of the power consumption required in (18) becomes:

\[
\hat{\gamma}_{\tau,t} = \mathbb{E}\{\lambda_{\tau,t}\} = \sum_{a \in \mathcal{M}_{t+1}} \gamma_{a} + \sum_{a \in S_{t}} \gamma_{a} \delta_{\tau,t}^{a},
\]

where \( S_{t} \) is defined in Section I.}

D. Algorithm Description

In this section, we explain the different steps of the proposed energy consumption scheduling algorithm in presence of load uncertainty (Algorithm 1) executed at each time slot \( t \).

Step 1: At the beginning of the admission control phase at each time slot, all received admission requests are labeled as either must-run or controllable, c.f. Lines 1 and 2.

Step 2: Activate must-run appliances \( a \in \mathcal{M}_{t} \) right away, c.f. Line 3. That is, start or continue their operation at the requested power \( \gamma_{a} \). Their operation will not be interrupted, and they remain must-run until the end of their operation.

Step 3: In Line 4, considering the list of appliances that have already become awake, update the probabilities at which other appliances will send an admission request in the upcoming time slots as in (22). Adopt (24) to update the probabilities with which sleeping devices become active in upcoming time slots, c.f. Line 5.

Step 4: Use the current information to calculate the expected load in the upcoming time slots using (25) as indicated in Line 6. Update the remaining required energy of each appliance at the beginning of the current time slot, i.e., \( E_{t}^{a} \), using (1), c.f. Line 7.

Step 5: Next, set the “on” / “off” state of each awake controllable appliance for the rest of the time slots by solving optimization problem (18), c.f. Line 8.
Algorithm 1: Energy consumption scheduling algorithm in presence of load uncertainty executed at the beginning of each time slot $t$.

1. Receive admission requests.
2. Label received requests either as must-run or controllable.
3. Activate must-run appliances (start / continue operation).
4. Update $p^a_{t}$ according to (22).
5. Update $\delta^a_{t}$ according to (24).
6. Update $\lambda_{t}$ according to (25).
7. Update $E^a_{t}$ according to (1).
8. Solve (18) to activate / deactivate controllable appliances.
9. if activated device is non-interruptible
   10:  Mark it as must-run.
11:  end if

Step 6: In Lines 9 to 11, if any non-interruptible controllable appliance became active (i.e., it switched from off to on) in Step 5, remove it from the list of controllable appliances and add it to list of must-run devices as it should remain on until it finishes its operation.

V. PERFORMANCE EVALUATION

In this section, we present simulation results and assess the performance of our proposed DSM algorithm. We run the simulation multiple times with different patterns for the times at which the appliances become awake. We then present the average results. Unless stated otherwise, the simulation setting is as follows. We assume that the general RTP method combined with IBR is adopted as described in (3). In our system model, the retail price parameters, $m_{t}$, $n_{t}$, and $b_{t}$, are set by the retail energy provider to compensate the cost of providing energy and to shape the daily energy consumption of the user. However, these parameters are different from the load cost profile of the energy provider, as the load cost profile of the energy provider is determined in the wholesale electricity market. The exact load cost profile of the retailer is usually not known to the end users. Fig. 3 illustrates the variation of parameters $m_{t}$ and $n_{t}$ of the price function over one day. We consider a single household with various must-run and controllable appliances. Controllable appliances can be either interruptible or non-interruptible. Non-interruptible appliances include: electric stove ($E_a = 4.5$ kWh, $\gamma_a = 1.5$ kW), clothes dryer ($E_a = 1$ kWh, $\gamma_a = 0.5$ kW), and vacuum cleaner ($E_a = 3$ kWh, $\gamma_a = 1.5$ kW). Interruptible appliances include: Refrigerator ($E_a = 2.5$ kWh, $\gamma_a = 0.125$ kW), air conditioner ($E_a = 6$ kWh, $\gamma_a = 1.5$ kW), dishwasher ($E_a = 2$ kWh, $\gamma_a = 1$ kW), heater ($E_a = 4$ kWh, $\gamma_a = 1$ kW), water heater ($E_a = 2$ kWh, $\gamma_a = 1$ kW), pool pump ($E_a = 4$ kWh, $\gamma_a = 2$ kW), and PEV ($E_a = 10$ kWh, $\gamma_a = 2.5$ kW). Must-run appliances include: Lightning ($E_a = 3$ kWh, $\gamma_a = 0.5$ kW), TV ($E_a = 1$ kWh, $\gamma_a = 0.25$ kW), PC ($E_a = 1.5$ kWh, $\gamma_a = 0.25$ kW), ironing appliance ($E_a = 2$ kWh, $\gamma_a = 1$ kW), hairdryer ($E_a = 1$ kWh, $\gamma_a = 1$ kW), and others ($E_a = 6$ kWh, $\gamma_a = 1.5$ kW). The details of the average annual energy consumption of different appliances and the average monthly energy consumption of residential users in the US can be found in [35] and [36]. The time slot at which each appliance becomes awake is selected randomly from a pre-determined time interval, e.g. [6:00, 14:00] for electric stove and [16:00, 24:00] for PEV.

A. Performance Gains of Users and Utility Company

To have a baseline to compare with, we consider a system without ECC deployment, where each appliance $a$ is assumed to start operation right after it becomes awake at its nominal power $\gamma_a$. As an upper bound, we also consider a system with ECC deployment in which all the demand information of the appliances is available ahead of time. Simulation results for the average total power consumption for the proposed residential load control algorithm, the system without ECC deployment, and the system in which complete demand information is available ahead of time are depicted in Fig. 4. In our simulation model, we set $b_{t} = 3.5$ kW in (3) for all time slots. As illustrated in Fig. 4, to reduce electricity payment, the ECC unit shifts the load to time slots with lower prices such as the after midnight hours. However, the high price penalty for exceeding the $b_{t}$ threshold prevents load synchronization as discussed in Section II. The simulation results show that exploiting the use of the ECC unit reduces the average daily payment of the user from 4.76 Dollars/day to 4.01 Dollars/day. For the case, where complete information about the demand of each appliance is available ahead of time, the average daily payment of the user is 3.92 Dollars/day. To evaluate the PAR, the user’s daily peak load is divided by his daily average load. That is, after running the algorithm, at the end of the operation period, we compute

$$PAR = \frac{T \max\{l_1, \ldots, l_T\}}{\sum_{k=1}^{T} l_k},$$

where $l_k$ is the total power consumption of the user at time slot $k$. The proposed algorithm also helps to reduce the average PAR of the system from 2.66 to 1.98 (25.5% PAR reduction) compared to the system without ECC deployment. The PAR of the system with ECC deployment if complete demand information is known a priori is 1.89.

B. Computational Complexity

In general, integer linear programs with $n$ integer variables and $m$ constraints are known to be NP-complete [37]. How-
ever, there exist pseudo-polynomial algorithms for solving $m \times n$ integer programs with fixed $m$ which have an order of complexity of $O(n^{2m+2}(ma)^{m+1}(2m+1)\log(n^2(ma^2)^{2m+3}))$, where $\alpha$ is the maximum coefficient in the set of constraints [38]. A complete discussion of the complexity of such algorithms is out of the scope of this paper. However, to illustrate the complexity of our proposed algorithm, simulation results for the average run time of the algorithm, the number of integer variables, and the number of constraints for different numbers of appliances and different time granularities are given for one time slot in Table I. The order of complexity of the algorithm determines the maximum run-time or the maximum number of elementary operation required to solve the problem for any input scenarios. However, in practice, the times at which different appliances become awake are distributed over the operation horizon, and it is unlikely that all appliances become awake at the same time. Therefore, at each time slot, the number of awake appliances required to be scheduled is limited. This can significantly reduces the average run time of the algorithm in most practical scenarios. By increasing the time granularity, the number of integer variables and the number of constraints are increased, since the number of time slots at which the operation of each appliance should be scheduled is increased. However, the effect of this increase is mitigated, since the times at which different appliances become awake are distributed over a larger number of time slots, and the number of awake appliances in each time slot is reduced.

C. The Impact of Price Control Parameter $b_t$

Considering the price function as described in (3), in each time slot, if the power consumption of the user exceeds a certain threshold defined as $b_t$, the user will be penalized by paying a higher price. The choice of parameter $b_t$ has a significant impact on users’ payments and the PAR.

To have a baseline to compare with, similar to [27], we consider a system in which the effect of IBR is ignored and only the basic price in each time slot is taken into account to schedule the operation of different appliances in order to minimize the electricity payment of the user. Simulation results for the average payment of the user and the PAR of the system for different values of parameter $b_t$ in shown in Figs. 5 and 6, respectively. Intuitively, increasing the price parameter $b_t$ for each time slot results in a reduction of the user’s payment as shown in Fig. 5. Considering the average PAR, for the system without ECC deployment and the system without IBR consideration, the PAR does not change as the user does not respond to changes of parameter $b_t$. For the system with ECC deployment, for low values of parameter $b_t$, even the load of must-run appliances in most time slots exceeds this threshold. Thus, the ECC unit mainly considers the $n_t$ price parameter to schedule the operation of controllable appliances. However, by increasing parameter $b_t$, since the load of must-run appliances lies below threshold $b_t$ in some time slots, the user is encouraged to shift the controllable portion of its load to avoid paying higher price $n_t$ rather than lower price $m_t$, which initially results in reducing the PAR. Later on, a further increase of parameter $b_t$ reduces the effect of IBR, entails load synchronization effects, and increases the PAR of the system.

D. The Impact of Adopting Inclining Block Rates

In this section, we examine how changes of the two parameters $m_t$ and $n_t$ of the price function will affect the performance of the system. In our simulation model, parameter $m_t$ changes as illustrated in Fig. 3 and we set $b_t = 3$ kW for all time slots. However, parameter $n_t$ is given by

$$n_t = \theta m_t, \quad \forall t \in \{1, \ldots, T\}. \quad (27)$$

| Time granularity | $|\mathcal{A}|=20$ | $|\mathcal{A}|=25$ | $|\mathcal{A}|=35$ |
|------------------|----------------|----------------|----------------|
| 1 hour           | 0.0287         | 0.0308         | 0.0350         |
| 30 minutes       | 0.0294         | 0.0316         | 0.0422         |
| 15 minutes       | 0.0302         | 0.0318         | 0.0988         |

| Average number of integer variables. |
|-------------------------------|----------------|----------------|----------------|
| Time granularity | $|\mathcal{A}|=20$ | $|\mathcal{A}|=25$ | $|\mathcal{A}|=35$ |
| 1 hour           | 80             | 156            | 296            |
| 30 minutes       | 121            | 230            | 440            |
| 15 minutes       | 168            | 316            | 619            |

| Average number of constraints. |
|-------------------------------|----------------|----------------|----------------|
| Time granularity | $|\mathcal{A}|=20$ | $|\mathcal{A}|=25$ | $|\mathcal{A}|=35$ |
| 1 hour           | 43             | 79             | 139            |
| 30 minutes       | 46             | 81             | 142            |
| 15 minutes       | 54             | 87             | 145            |

Table I: Performance Measures and Complexity Analysis of the Proposed Algorithm.
Simulation results for the average daily payment of the user as well as the average PAR of the system for different values of parameter \( \theta \) are depicted in Figs. 7 and 8, respectively. Intuitively, when \( \theta \) is equal to one, i.e., when \( m_t = n_t \) for all \( t \), the performance of our proposed method is the same as the one in which the effect of IBR is ignored. However, by increasing parameter \( \theta \), the payment of the user will be increased, as the user has to pay more every time that its load exceeds threshold \( b_t \) as shown in Fig. 7. As indicated in Fig. 8, increasing parameter \( \theta \) improves the PAR of the system, as load synchronization is prevented. That is, to avoid paying at higher price \( n_t \), the ECC unit tries to distribute the load such that it does not exceed the \( b_t \) threshold. However, for the system without IBR consideration, changes of parameter \( \theta \) do not affect the PAR.

VI. CONCLUSIONS

In this paper, we proposed an optimal residential load control algorithm for DSM in presence of load uncertainty. We formulated an optimization problem to minimize the electricity payment of the users in situations where only an estimate of the future demand is available. We focused on a scenario where real-time pricing is combined with IBRs to balance residential load to achieve a low PAR. Simulation results show that the proposed algorithm reduces the energy cost of users, encouraging them to participate in DSM. Exploiting IBR with RTP tariffs can help to avoid load synchronization, and the combination of the general RTP method with our algorithm reduces the PAR of the total load. The latter provides incentives for utilities to support implementing the proposed algorithm.

REFERENCES

Robert Schober (S’98, M’01, SM’08, F’10) was born in Neuendettelsau, Germany, in 1971. He received the Diplom (Univ.) and the Ph.D. degrees in electrical engineering from the University of Erlangen-Nuernberg in 1997 and 2000, respectively. From May 2001 to April 2002 he was a Postdoctoral Fellow at the University of Toronto, Canada, sponsored by the German Academic Exchange Service (DAAD). Since May 2002 he has been with the University of British Columbia (UBC), Vancouver, Canada, where he is now a Full Professor and Canada Research Chair (Tier II) in Wireless Communications. Since January 2012 he is an Alexander von Humboldt Professor and the Chair for Digital Communication at the Friedrich Alexander University (FAU), Erlangen, Germany. His research interests fall into the broad areas of Communication Theory, Wireless Communications, and Statistical Signal Processing.

Dr. Schober received the 2002 Heinz Maier-Leibnitz Award of the German Science Foundation (DFG), the 2004 Innovations Award of the Vodafone Foundation for Research in Mobile Communications, the 2006 UBC Killam Research Prize, the 2007 Wilhelm Friedrich Bessel Research Award of the Alexander von Humboldt Foundation, the 2008 Charles McDowell Award for Excellence in Research from UBC, a 2011 Alexander von Humboldt Professorship, and a 2012 NSERC E.W.R. Steacie Fellowship. In addition, he received best paper awards from the German Information Technology Society (ITG), the European Association for Signal, Speech and Image Processing (EURASIP), IEEE WCNC 2012, IEEE Globecom 2011, IEEE ICUWB 2006, the International Zurich Seminar on Broadband Communications, and European Wireless 2000. Dr. Schober is a Fellow of the Canadian Academy of Engineering and a Fellow of the Engineering Institute of Canada. He is currently the Editor-in-Chief of the IEEE Transactions on Communications.