Distributed Mode Scheduling for Coordinated Power Balancing

Hiroaki Kawashima
Graduate School of Informatics
Kyoto University
Kyoto 6068501, Japan
Email: kawashima@i.kyoto-u.ac.jp

Takekazu Kato
Graduate School of Informatics
Kyoto University
Kyoto 6068501, Japan
Email: t.kato@ieee.org

Takashi Matsuyama
Graduate School of Informatics
Kyoto University
Kyoto 6068501, Japan
Email: tm@i.kyoto-u.ac.jp

***Abstract*—This paper proposes a distributed scheduling algorithm of demands of multiple households/consumers in order to achieve a power balancing in total. Assume that an autonomous energy management system is installed in each household and that those households are capable of communicating with an aggregator. Then, it becomes possible to negotiate via the autonomous energy management systems to find an agreement point that takes both the users' demands and the aggregator's objective into account. The key aspect of the algorithm is that it enables us to encapsulate the particular control and objective in each household, and realizes the negotiation based on power profiles. We also show the proposed framework can be smoothly integrated with a probabilistic generative model of power profiles.**

I. Introduction

Autonomous demand-side energy management is attracting great attention as it realizes machine to machine interaction among households/consumers and aggregators/providers (e.g., [1], [2], [3], [4], [5], [6]). By automated energy management systems (EMS), participants of demand response (DR) programs can respond to the information from the utility (or an aggregator) and manage the scheduling of loads/generators automatically. This trend is now increasing the variety of DR programs.

When DR is implemented as a price-based program [7] such as real-time pricing, in which customers are charged short-term fluctuating prices, participants can automatically change the on-off timing of appliances within an acceptable range [6]. Meanwhile, in an incentive-based program (IBP), participating customers receive some kind of rewards for the achievement of utility-side requests or for the participation in the programs. A classical IBP strategy is a direct load control, where utility companies directly turn off participants' devices. Although this classical strategy satisfies the demands of the aggregator side, it often cannot take account of the demands of the participants side flexibly. To deal with the quality of life (QoL) of participants, a recent trend is to develop a system that handles the time shifting of appliances while minimizing the deviation from their preference [2]. Indeed, recent IBP introduces market-based bidding strategies, where participants bid load reduction in electricity wholesale market [7]. As a result, the automated EMS is expected to be a negotiator between aggregators and users.

In this paper, we formulate this two-sided (i.e., households and aggregators) demands directly and provide an algorithm to optimize the power profiles of households in a distributed manner (Fig. 1). We here use the term "household" to denote a unit of group/area in which an installed EMS is capable of controlling the devices inside the group, and therefore the framework can be applied not only to home EMS (HEMS) but also to the management systems for building, office, etc. The present paper mainly focuses on providing an algorithmic framework. In particular, similar to [6], [8], we assume the situation of an IBP in which users have already been given an incentive to join the program. That is, the participants inside the territory of an aggregator are required to coordinate and negotiate their day-ahead schedules of power consumption so as to achieve the aggregator's objective (e.g., balancing the total power consumption/generation).

The contribution of this paper is twofold: (1) we introduce a distributed optimization technique which enables us to encapsulate the actual control and objective in each household; and, (2) we propose to use a probabilistic generative model, called the *hidden-semi Markov model (HSMM)*, to represent user-side demands and the flexibility of changing his/her schedules. In particular, (1) is quite useful property since, albeit beyond the scope of this paper, it can be applied to the coordination of heterogeneous types of EMS; and, the point of (2) is that a detailed model of profiles can be trained from real data.

While there are successful distributed optimization techniques based on game theories [1], [3], they assume each player (household) collects power profiles from all the other players. This may not only cause a privacy issue but assume a centralized communication topology in terms of the way of sharing information even though the players' decisions are made in a decentralized way. In contrast, this paper exploits the methods called the *dual decomposition* and the *alternating direction method of multipliers (ADMM)*[9]. This is the key to realize the decoupling between households and an aggregator and achieve a profile-based interaction characterized by (1).

In the next section we show how the problem can be formulated and solved by the distributed optimization. In Section III, we introduce a particular probabilistic model and show the model can be smoothly integrated with the distributed optimization framework. Simulation results, discussions, and conclusions are given in Section IV, V, and VI, respectively.

II. Distributed Optimization

In this section, we consider a general formulation of coordinating EMS via an aggregator in order to achieve

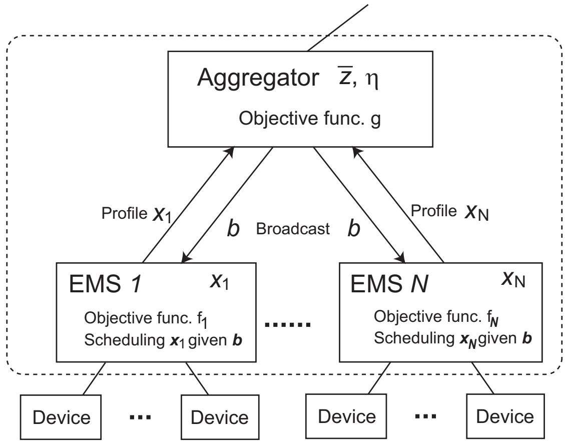

Fig. 1. Architecture of the proposed framework. Iterative negotiation is conducted in the box (dashed line) to determine the day-ahead power profiles of the participants.

both household objectives, such as QoL of users, and the aggregator's objective, such as power balancing, and we do not consider a particular EMS.

A. Problem Formulation

Consider N households are coordinated through an aggregator. Let the scheduled power profile (sequence) of household $i \in \mathcal{N} \triangleq \{1, ..., N\}$ be $x_i \in \mathbb{R}^T$, where T is the number of time slots in a day, and the t-th element of x_i represents the (averaged) power consumption in the t-th time slot. While the following discussion focuses on power consumption profiles, the framework has no limitation of dealing with power generation from distributed power supply.

Here, we introduce two types of objective functions. Let $f_i(x_i)$ be a local objective function that measures the cost of achieving profile x_i at household i, and let $g(v)$ be a global objective function of the total energy of N households, where $v = \sum_{i\in\mathcal{N}} x_i$. Then, we formulate the optimization problem as

$$\min_{x} \quad L(x) \triangleq \sum_{i\in\mathcal{N}} f_i(x_i) + g(\sum_{i\in\mathcal{N}} x_i), \tag{1}$$

where $x = [x_1^\top, ..., x_N^\top]^\top \in \mathbb{R}^{NT}$ is a stacked vector form of $\{x_i\}_{i\in\mathcal{N}}$. For example, one may use

$$g(v) = \alpha \frac{T||v||_\infty}{\mathbf{1}_T^\top v} \tag{2}$$

to minimize the peak-to-average ratio (PAR), where $\mathbf{1}_n$ is a n-dimensional column vector whose entries are all 1, and $\alpha > 0$ is a weight parameter. Another example of function g to flatten (and also to reduce) the total power consumption is

$$g(v) = \alpha ||v||^2, \tag{3}$$

where we denote a Euclidean norm as $||\cdot||$ otherwise noted. On the other hand, the local objective function f_i ($\forall i \in \mathcal{N}$) basically encodes user's QoL. In Subsection II-C, we will see how this function can be related to the control of each EMS.

To solve Eq. (1) in a centralized manner, the aggregator needs to know all the local objective functions $f_i, \forall i \in \mathcal{N}$. However, this is often not desirable in terms of scalability and privacy issues, since the local objective of each household strongly depends on the user's preference, e.g., types of appliances and timing of their usage, and it is not realistic for the aggregator to collect and manage all the households' information in addition to their controllable ranges every day.

As such, we here try to decompose the total objective function $L(x)$ in Eq. (1) so as to allow each household to upload only their schedule of power consumption. Indeed, this can be achieved by introducing a method called the dual decomposition, which introduces a set of duplicated variables. We note that the problem in Eq. (1) is called a "sharing problem" in a context of distributed optimization [9]. In the next subsection we briefly explain its distributed algorithm as a preliminary.

B. Dual Decomposition and ADMM of Sharing Problem [9]

Let z_i be a duplicated variable of x_i for all $i \in \mathcal{N}$. Then the original problem can be rewritten as

$$\begin{aligned} \min_{x,z} \quad & \sum_{i\in\mathcal{N}} f_i(x_i) + g(\sum_{i\in\mathcal{N}} z_i) \\ \text{subject to} \quad & x_i = z_i, \quad \forall i \in \mathcal{N}, \end{aligned}$$

and therefore the Lagrangian becomes

$$L(x, z, \lambda) \triangleq \sum_{i\in\mathcal{N}} f_i(x_i) + g(\sum_{i\in\mathcal{N}} z_i) + \sum_{i\in\mathcal{N}} \lambda_i(x_i - z_i), \tag{4}$$

where the elements of $\lambda \in \mathbb{R}^T$ are called the dual variables (Lagrange multipliers). Note that the above Lagrangian can be solved iteratively, i.e., a standard dual ascent [10]. In each iteration, we can solve $N + 1$ minimization, each of which is relative to f_i ($i \in \mathcal{N}$) and g, independently (this is the reason of the name "dual decomposition"). The dual variables $\{\lambda_i\}_{i\in\mathcal{N}}$ are also updated together.

However, the update of λ_i in the dual ascent algorithm requires a careful tuning of the step size. For this reason, more robust methods, which exploit the following augmented Lagrangian, have been introduced:

$$L_\rho(x, z, \lambda) \triangleq \sum_{i\in\mathcal{N}} f_i(x_i) + g(\sum_{i\in\mathcal{N}} z_i) + \lambda^\top(x - z) + \frac{\rho}{2}||x - z||^2, \tag{5}$$

where λ and z are the stacked vector form similar to x. The benefit of the additional penalty term is to improve the robustness of the algorithm; that is, it converges under rather mild conditions, e.g., without assumptions such as strict convexity. Letting $\eta = \frac{1}{\rho}\lambda$, Eq. (5) can be further rewritten as a convenient form

$$\tilde{L}_\rho(x, z, \eta) \triangleq \sum_{i\in\mathcal{N}} f_i(x_i) + g(\sum_{i\in\mathcal{N}} z_i) + \frac{\rho}{2}||x - z + \eta||^2 - \frac{\rho}{2}||\eta||^2, \tag{6}$$

which is often referred to as the "scaled form" since the dual variables are scaled by $\frac{1}{\rho}$.

Using the scaled form, the algorithm of ADMM becomes

$$x_i^{(k+1)} := \operatorname*{argmin}_{x_i} \left(f_i(x_i) + \frac{\rho}{2}||x_i - z_i^{(k)} + \eta_i^{(k)}||^2 \right)$$

$$z^{(k+1)} := \operatorname*{argmin}_{z} \left(g(\sum_{i\in\mathcal{N}} z_i) + \frac{\rho}{2} \sum_{i\in\mathcal{N}} ||z_i - x_i^{(k+1)} - \eta_i^{(k)}||^2 \right)$$

$$\eta_i^{(k+1)} := \eta_i^{(k)} + x_i^{(k+1)} - z_i^{(k+1)},$$

where k is the index of iterations, and one can use $\eta_i^{(-1)} = 0$ and $x_i^{(0)} = \arg\min_{x_i} f_i(x_i)$ for the initialization. Here, the minimization of the first and the last equations can be carried out independently for each $i \in \mathcal{N}$, i.e., solved in a distributed manner. Considering the averages $\bar{x}$ and $\bar{z}$, where we denote $\bar{q}$ the average of $q_1, ..., q_N$, and using the fact that $\min_z \sum_i ||z_i - a_i||^2$ with $\bar{z}$ fixed is given by $N||\bar{z} - \bar{a}||^2$, where $a_i = x_i^{(k+1)} + \eta_i^{(k)}$, the above algorithm can be rewritten as

$$x_i^{(k+1)} := \underset{x_i}{\operatorname{argmin}} \left(f_i(x_i) + \frac{\rho}{2} ||x_i - x_i^{(k)} + b^{(k)}||^2 \right) \tag{7}$$

$$\text{where} \quad b^{(k)} \triangleq \bar{x}^{(k)} - \bar{z}^{(k)} + \eta^{(k)} \tag{8}$$

$$\bar{z}^{(k+1)} := \underset{\bar{z}}{\operatorname{argmin}} \left(g(N\bar{z}) + \frac{N\rho}{2} ||\bar{z} - (\bar{x}^{(k+1)} + \eta^{(k)})||^2 \right) \tag{9}$$

$$\eta^{(k+1)} := \eta^{(k)} + \bar{x}^{(k+1)} - \bar{z}^{(k+1)}, \tag{10}$$

where we denote all the dual variables as $\eta^{(k)}$ since it can be shown that $\eta_i^{(k)}$ takes the same value for all $i \in \mathcal{N}$.

The flow of this algorithm is as follows (see also Fig. 1). In each iteration, participant $i \in \mathcal{N}$ determines the most desirable profile $x_i^{(k+1)}$ given information $b^{(k)}$ from the aggregator, while the aggregator collects $x_i^{(k+1)}$ from all the participants $i \in \mathcal{N}$ in order to calculate $\bar{x}^{(k+1)}$ which is required in Eq. (9) and Eq. (10). Then, new information $b^{(k+1)}$ calculated by Eq. (8) is broadcasted to all the participants again. Therefore, this algorithm can be seen as an iterative negotiation process to find an agreeable compromise of all the participants and the aggregator through the communication of $x_i^{(k)}$ and $b^{(k)}$. If the objective functions f_i $(i \in \mathcal{N})$ and g are strictly convex, and the augmented Lagrangian Eq. (4) has a saddle point, $x_i^\star$ at the convergence point minimizes the original objective function $L(x)$ in Eq. (1). Non-convex case will be discussed in Section V-A.

C. Control in Each Household

We here closely examine the structure of objective function $f_i : \mathbb{R}^T \to \mathbb{R}$ in our scenario. By $f_i(x_i)$, household i evaluates the difficulty of realizing x_i, for example, how large the gap from his/her comfortable usage of appliances (dissatisfaction) to realize x_i, and how large the risk of declaring to use schedule x_i (uncertainty). And, the measure of these quantities must involve the control in the household.

Let $u_i \in \mathcal{U}_i$ be the control of appliances in the household i, where $\mathcal{U}_i$ is the set of possible control candidates in the household. For example, u_i can be a mode scheduling of appliances in the household, as will be explained in the next section. The control u_i is tightly related to the QoL of the user, and it is natural for the user to evaluate u_i itself instead of x_i. Therefore, we introduce a dissatisfaction function $f_i^u : \mathcal{U}_i \to \mathbb{R}$ to evaluate the cost of taking the control u_i as $f_i^u(u_i)$.

Once the variable u_i is decided, profile x_i should be determined by u_i. For example, if the household i has only two possible control patterns, $\mathcal{U}_i = \{U_{i,1}, U_{i,2}\}$, then x_i takes either $\chi_i(U_{i,1})$ or $\chi_i(U_{i,2})$ with some mapping $\chi_i : \mathcal{U}_i \to \mathbb{R}^T$. Thus, the control variable serves as a compact factor (compression) for specifying possible x_i in the vector space $\mathbb{R}^T$. However, there always exists the uncertainty of realizing profile x_i. Therefore, we let $f_i^{x|u}(x_i, u_i)$ be an uncertainty measure of achieving x_i given control u_i. Taking the above aspects into account, we define $f_i(x_i)$ as

$$f_i(x_i) \triangleq \min_{u_i \in \mathcal{U}_i} \left(f_i^u(u_i) + f_i^{x|u}(x_i, u_i) \right). \tag{11}$$

That is, to evaluate the cost of x_i, we use the optimal control u_i in terms of minimizing both dissatisfaction and uncertainty.

Using the above definition of $f_i(x_i)$, Eq. (7) becomes

$$\begin{aligned} u_i^* &:= \underset{u_i \in \mathcal{U}_i}{\operatorname{argmin}} \left(f_i^u(u_i) + \min_{x_i} \left(f_i^{x|u}(x_i, u_i) + F_i^{(k)}(x_i) \right) \right) \\ x_i^{(k+1)} &:= \underset{x_i}{\operatorname{argmin}} \left(f_i^{x|u}(x_i, u_i^*) + F_i^{(k)}(x_i) \right) \\ &\text{where} \quad F_i^{(k)}(x_i) \triangleq \frac{\rho}{2} ||x_i - x_i^{(k)} + b^{(k)}||^2 \end{aligned} \tag{12}$$

If the function $f_i^{x|u}$ is given as a particular form, e.g., quadratic function of x_i, the minimization over x_i can be solved analytically, and the optimization in each household only involves the minimization over u_i.

The key of this form is that now the algorithm does not depend on a particular form of the control and optimization in each household. Therefore, this gives us a general framework of aggregating different types of EMS. Finally, we give some important examples regarding the design of $f_i^u(u_i)$ and $f_i^{x|u}(x_i, u_i)$.

1) Indicator function: Consider an ideal case that the power profile under control u_i is determined as $\chi(u_i) \in \mathbb{R}^T$ without uncertainty. In this case, $f_i^{x|u}(x_i, u_i)$ is given as

$$f^{x|u}(x_i, u_i) = \begin{cases} 0 & x_i = \chi(u_i) \\ +\infty & x_i \neq \chi(u_i) \end{cases}. \tag{13}$$

Note that this is an indicator function that defines a set of constraints on x_i. Namely, the set of possible power profiles is given as $\{\chi(u_i)\}_{u_i \in \mathcal{U}_i} \subset \mathbb{R}^T$ since x_i is uniquely determined by u_i. Substituting Eq. (13) into Eq. (12) yields

$$\begin{aligned} u_i^* &:= \underset{u_i \in \mathcal{U}_i}{\operatorname{argmin}} \left(f_i^u(u_i) + F_i^{(k)}(\chi(u_i)) \right) \\ x_i^{(k+1)} &:= \chi(u_i^*), \end{aligned} \tag{14}$$

which requires each EMS the minimization over only the control variable u_i.

2) Probabilistic model: We also note that one of useful implementations of the measures $f_i^u(u_i)$ and $f_i^{x|u}(x_i, u_i)$ is a probabilistic generative model $P(x_i, u_i) = P(u_i)P(x_i|u_i)$. Here, $P(u_i)$ and $P(x_i|u_i)$ can be trained via machine learning techniques (e.g., the maximum likelihood or Bayesian estimation) from daily sensing data of power consumption and/or some prior knowledge. Once the model is given or trained, the probability $P(u_i)$ tells us how natural the control u_i is for the household i, and it is plausible that the larger the probability the higher the user's satisfaction. Meanwhile, the probability $P(x_i|u_i)$ tells us the confidence/certainty of achieving power profile x_i given control u_i. Therefore, an example of $f_i^u(u_i)$ and $f_i^{x|u}(x_i, u_i)$ can be

$$f_i^u(u_i) = -\log P(u_i), \quad f_i^{x|u}(x_i, u_i) = -\log P(x_i|u_i),$$

and in this case Eq. (11) becomes

$$f_i(x_i) = -\log \max_{u_i \in \mathcal{U}_i} P(x_i, u_i).$$

If the probability becomes zero at some x_i, $f_i(x_i)$ takes $+\infty$ and those profiles are excluded from the candidates of x_i. Indeed, if $P(x_i|u_i)$ is given by the delta function

$$\delta(x_i - \chi(u_i)) = \begin{cases} 1 & x_i = \chi(u_i) \\ 0 & x_i \neq \chi(u_i) \end{cases}, \tag{15}$$

$f_i^{x|u}(x_i, u_i)$ becomes the indicator function shown in the previous paragraph. On the other hand, if $P(x_i|u_i)$ is given by a Gaussian distribution, the minimization over x_i in Eq. (12) has a unique analytical solution.

III. Probabilistic Model of Power Profiles

Based on the discussion in the previous section, this section introduces a particular probabilistic generative model of power profiles as an example of users' objective functions expected to be useful. We here consider a model for the total power profile in a household, while it can also be used for the model of individual appliances (see Subsection V-B).

As discussed in the literature (e.g., [7], [2], [3], [5]), important aspects of the control of consumption patterns is to alter the "timing" and "level" of demands. Therefore, to estimate and model power profiles, we propose to use the probabilistic model called the hidden semi-Markov model (HSMM) (or the explicit-duration hidden Markov model). The HSMM is a particular instance of "segment models" [11], and therefore it models time-varying signals as a sequence of segments or intervals. As the model has been studied long years since 1980s [12], it now has several types of efficient algorithms for model learning and state estimation. In particular, we extend the probabilistic inference algorithm proposed in [13] to solve the optimization in Eq. (12), i.e.,

$$\begin{aligned} &\min_{u_i \in \mathcal{U}_i} \left(-\log P(u_i) + \min_{x_i} \left(-\log P(x_i|u_i) + F_i^{(k)}(x_i) \right) \right) \\ &= -\max_{u_i \in \mathcal{U}_i} \left(\log P(u_i) + \max_{x_i} \left(\log P(x_i|u_i) - F_i^{(k)}(x_i) \right) \right) \end{aligned} \tag{16}$$

A. Hidden Semi-Markov Model

Consider that the profile x_i can be modeled by a transition of discrete states, which we refer to as *modes*. Let $s_{i,t} \in \mathcal{Q}_i \triangleq \{q_{i,1}, ..., q_{i,M_i}\}$ be the mode at time t. We assume that control variables of household i are these modes at time $t = 1, ..., T$, i.e., $u_i \triangleq s_{i,1:T} \in \mathcal{Q}_i^T$, where, and in what follows, we denote $a_{b:e}$ the sequence $(a_t)_{t=b}^{e} = (a_b, a_{b+1}, ..., a_e)$. To simplify the notations, we will omit the index i from subscripts, and we use y_t instead of $x_{i,t}$, i.e., $x_i = (y_1, ..., y_T) = y_{1:T}$, to avoid the confusion between indices i and t.

In the HSMM, y_t is called an output at time t, and assume that its distribution is determined only by s_t, the mode at t. Thus, the probability $P(x_i|u_i)$ becomes

$$P(x_i|u_i) = P(y_{1:T}|s_{1:T}) = \prod_{t=1}^{T} P(y_t|s_t),$$

where $P(y_t|s_t)$ is called an output probability distribution. Meanwhile, as for $P(u_i) = P(s_{1:T})$, the model assumes a segment-based process. In accordance with [13], let $\tau_t \geq 1$ be a random variable denoting the remaining (or residual) time of the current mode. Assume a situation the pair (s_t, τ_t) takes value (q_m, d). If $d > 1$ then $(s_{t+1}, \tau_{t+1}) = (q_m, d-1)$, otherwise (i.e., if $d = 1$), a mode transition occurs and $(s_{t+1}, \tau_{t+1}) = (q_n, d')$. This process is modeled by the mode transition probability $P(s_{t+1} = q_n | s_t = q_m, \tau_t = 1)$ and duration distribution $P(\tau_t = d | s_t = q_m, \tau_{t-1} = 1)$.

In summary, the parameters of the HSMM become

Initial mode: $\pi_m \triangleq P(s_1 = q_m)$
Mode transition: $a_{mn} \triangleq P(s_{t+1} = q_n | s_t = q_m, \tau_t = 1)$
Duration: $p_m(d) \triangleq P(\tau_t = d | s_t = q_m, \tau_{t-1} = 1)$
Output : $b_m(y_t) \triangleq P(y_t | s_t = q_m)$

where $q_m, q_n \in \mathcal{Q}$, $d \geq 1$, and $y_t \in \mathbb{R}$. A typical parameter estimation (learning) method is the maximum likelihood estimation via the expectation-maximization algorithm (see, for example, [12], [13]). Although the model can be trained from actual sensing data, we will use manually determined parameters in Section IV since our main objective is to demonstrate the distributed optimization framework.

B. Optimal Mode Scheduling

Since the full search to find the optimal mode sequence for Eq. (16) requires comparing $|Q^T| = M^T$ sequences, the computational cost grows exponentially depending on the resolution of the time slots, and thus some efficient algorithm is required. Here, we note an important fact that the second term of the minimization over u_i in Eq. (16) can be decomposed as

$$\max_{y_{1:T}} \left(\sum_{t=1}^{T} \log P(y_t|s_t) - \sum_{t=1}^{T} F_{i,t}^{(k)}(y_t) \right) = \sum_{t=1}^{T} G_t(s_t), \tag{17}$$

where, recalling the definition of $F_i^{(k)}$ in Eq. (12) and letting $b_t^{(k)}$ be the t-th element of $b^{(k)}$,

$$\begin{aligned} G_t(s_t) &\triangleq \max_{y_t} \left(\log P(y_t|s_t) - F_{i,t}^{(k)}(y_t) \right) \\ F_{i,t}^{(k)}(y_t) &= F_{i,t}^{(k)}(x_{i,t}) \triangleq \frac{\rho}{2}(x_{i,t} - x_{i,t}^{(k)} + b_t^{(k)})^2. \end{aligned}$$

Therefore, it is possible to apply the dynamic programming (DP) technique to Eq. (16).

Another key fact is from the observation in [13] that the probability $P(s_t = q_m, \tau_t = d)$ can be calculated as the summation of $P(s_{t-1} = q_m, \tau_{t-1} = d{+}1)$ (mode continues at time t) and $P(\tau_{t-1} = 1, s_t = q_m) P(\tau_t = d | \tau_{t-1} = 1, s_t = q_m)$ (mode changes at time t) (see Fig. 2 for reference). Besides, $P(\tau_{t-1} = 1, s_t = q_m) = \sum_{n \neq m} a_{nm} P(\tau_{t-1} = 1, s_{t-1} = q_n)$. In other words, to arrive at $(s_t, \tau_t) = (q_m, d)$ from s_{t-1}, there exists $M (= 1 + (M-1))$ paths depending on s_{t-1}.

Hence, letting

$$\varphi_t(m, d) \triangleq \max_{s_{1:t-1}} \left(\log P(s_{1:t-1}, s_t = q_m, \tau_t = d) + \sum_{t'=1}^{t-1} G_{t'}(s_{t'}) \right),$$

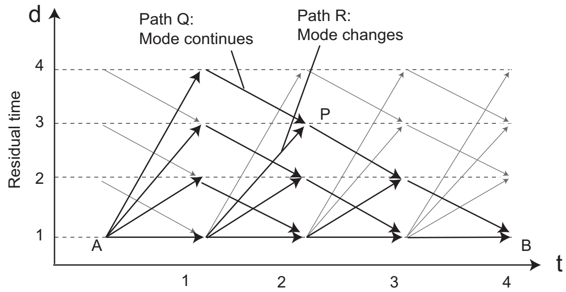

Fig. 2. Trellis diagram of mode transition timing (the axis of mode itself is omitted). Possible paths to get from A to B are depicted by thick lines. In the dynamic programming, the value of P is calculated from path Q and R.

we can use the following recursive algorithm:

$$\varphi_t(m,d) = \max\Bigg(\varphi_{t-1}(m,d+1) + G_{t-1}(q_m), \\ \log p_m(d) + \max_n \Big(\varphi_{t-1}(n,1) + \log a_{nm} + G_{t-1}(q_n)\Big)\Bigg).$$

Note that this is the maximization over all the possible s_{t-1}. From the definition of $\varphi_t(m,d)$, the initialization is given by

$$\varphi_1(m,d) = \log P(s_1 = q_m, \tau_1 = d) = \log p_m(d) + \log \pi_m.$$

The optimal value of Eq. (16) becomes $\max_{m,d}(\varphi_T(m,d) + G_T(q_m))$ and the optimal mode sequence is given by tracing back the optimal path of the recursion.

Using this optimization of individual households for Eq. (12) with Eq. (9) and Eq. (10), we finally have a distributed mode scheduling algorithm.

IV. Simulation Results

To demonstrate the effectiveness of the proposed algorithm, we present simulation results of flattening the power consumption of multiple residences. Specifically, we used Eq. (3) for the global objective function with $\alpha = \nu/(2N)$, where $\nu = 2 \times 10^{-6}$. In this case, Eq. (9) can be solved analytically, and the update of $\bar{z}$ is given by $\bar{z}^{(k+1)} = \frac{\rho}{\nu+\rho}(\bar{x}^{(k+1)} + \eta^{(k)})$, where $\rho = 0.2 \times 10^{-6}$ was used. The scale of ρ and ν was decided so as to balance with the local objective functions.

In this simulation, we used the HSMM explained in the previous section to model the power profile in each residence. To investigate the basic properties of the proposed algorithm, we used simple parameters as explained below. First, we assume that each of the residences owns the same type of a plug-in hybrid electric vehicle (PHEV) as a controllable device but has his/her own preference on the start time of charging. For simplicity, we consider the consumption profile of only the PHEV in each household as its required power is dominant in many residences. Specifically, we assume that the power consumption profile can be represented by three modes: $\mathcal{Q} = \{q_1, q_2, q_3\}$ and that the initial mode, the mode transition probabilities, and the output distributions are identical for all the residences. In particular, we consider the deterministic (fixed) transition pattern $q_1 \to q_2 \to q_3$. To focus on the aspect of mode scheduling, we used simple output distributions given by $P(y_t|s_t) = \delta(y_t - \chi(s_t))$, where $\chi(s_t) = 0, 1000, 0[W]$ for $s_t = q_1, q_2, q_3$, respectively. As for the time slots of a day, we divided one day into ten-minute slots (i.e., $T = 144$). In addition, we assume that each PHEV requires about 3kWh charge (i.e., three hours) in a day.

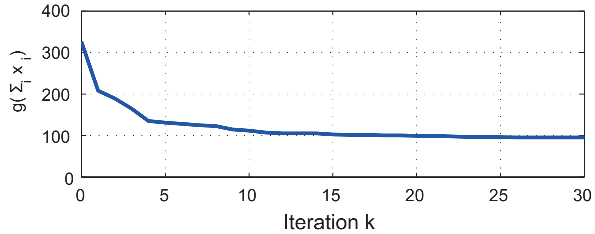

Fig. 3. The change of the values of the global objective function (scenario 1). In this scenario, peak-to-average ratio (PAR), not shown here, was reduced from 8.0 to 2.9 in 30 iterations.

Using this setting, we conducted two scenarios. In both scenarios, N households negotiate with an aggregator on their day-ahead scheduling of power profiles. In the first scenario (scenario 1), $N = 20$ households have some flexibility to determine the starting time of PHEV charging. This can be represented as follows. Consider that the duration distribution of mode q_m is given by a Gaussian with mean μ_m and standard deviation σ_m. In scenario 1, we used the following parameters:

- μ_1 differs depending on a user (uniformly distributed in the time interval $[50, 55]$); $\sigma_1 = 10$.
- $\mu_2 = 18$ (three hours); $\sigma_2 = 1$.
- $\mu_3 = T - (\mu_1 + \mu_2)$; $\sigma_3 = 10$.

Note that μ_1 determines the starting time of the PHEV charging and μ_2 represents the charging time (duration), while σ_m $(m = 1, 2, 3)$ determines the flexibility of these durations (e.g., the charging time has less flexibility).

Figure 3 shows the value of the global objective function $g(\sum_i(x_i))$ as the distributed optimization algorithm proceeds. We observe that the value decreased enough in approximately 20 iterations. To see the details of the total power consumption, as for the second scenario (scenario 2), we added another 20 households (group 2) which have much less flexibility than the original 20 households (group 1), and applied the same algorithm to the $N = 40$ households. For group 2, μ_1 was uniformly chosen in the time interval $[100, 105]$, and $\sigma_1 = 1$ was used (i.e., less flexible than group 1). As shown in Fig. 3, the total power consumption of group 1 was flatten drastically while that of group 2 was almost remained at the original profiles. This observation shows the characteristics of using the probabilistic model; that is, if the models are properly trained to encode users' preference, the distributed mode scheduling algorithm successfully finds an agreement point between households and an aggregator.

V. Limitations and Extension

A. *Non-convexity of Objective Functions*

If the objective functions f_i $(i \in \mathcal{N})$ and g are non-convex, the convergence and the optimality are not always guaranteed. This is often the case when users have complicated preference patterns. Nevertheless, if the parameters such as ρ is chosen appropriately, the algorithm still converges in many cases due

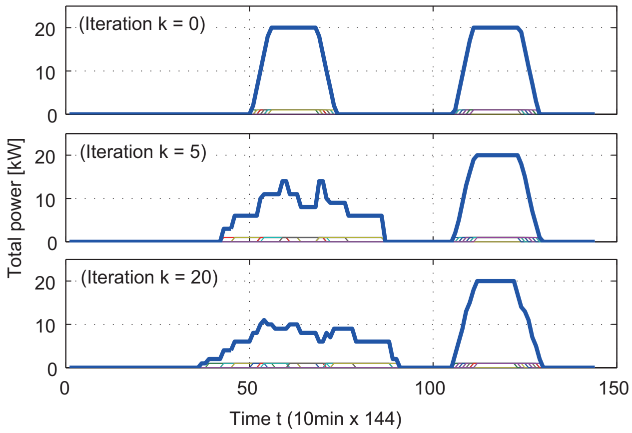

Fig. 4. Total power consumption of 40 households (thick lines) (scenario 2). Each of thin lines close to zero axes depicts a profile of each household. In this scenario, 20 households (group 2, right peak) are designed to have much less flexibility than the remaining 20 households (group 1, left peak).

to the additional penalty term of the augmented Lagrangian in Eq. (5) even though the convergence point is possibly a local minimum [9].

B. Multiple Devices

While this paper has considered two-layer architecture (the dashed box in Fig. 1), one can extend the architecture to have more than three layers. Let $\mathcal{A}_i$ be the set of devices in household i. Letting $e_{i,a}$, $a \in \mathcal{A}_i$, be the cost function of each of devices, the following objective function can be used:

$$\sum_{i\in\mathcal{N}}\sum_{a\in\mathcal{A}_i} e_{i,a}(x_{i,a}) + \sum_{i\in\mathcal{N}} f_i(\sum_{a\in\mathcal{A}_i} x_{i,a}) + g(\sum_{i\in\mathcal{N}}\sum_{a\in\mathcal{A}_i} x_{i,a}).$$

By introducing duplicated variables again such that $x_{i,a} = y_{i,a}$ and $y_{i,a} = z_{i,a}$, it is not difficult to show that a similar profile-based distributed optimization can be derived (see also [14], for ADMM with multiple agents). Our future work includes the detailed analysis of such hierarchical architectures with multiple devices and/or multiple aggregators.

C. Incentive Design of Demand Response Program

Although this paper has focused on an algorithmic aspect of coordination, the design of incentives such as rewards/penalties and prices is crucial to implement the framework as a practical demand response program. It is worth to seek the possibility of extending the proposed framework by including some factors of rewards/penalties and prices into the design of the objective functions, and to see how the method is effective for important challenges such as preventing rebound peaks [4].

D. Conditional Probability Models

In the simulation, all the parameters were given manually, and were fixed in a day. However, the trend of the usage of devices can be affected by many other factors such as time of day, temperature, season, the day of week, and users' special events. To take these factors into account, one can train the model with many types of conditional probabilities. Obviously, to find such detailed conditional probabilities requires a large amount of training data together with any available prior knowledge, and this will be investigated in the future work.

VI. Conclusion

This paper proposes a distributed scheduling algorithm of demands of multiple households in order to balance the total power in the territory of an aggregator. By communicating via profiles, the proposed algorithm finds the agreement point of the users' and the aggregator's objectives. The key aspect of the algorithm is that it enables us to encapsulate a particular control in each household and hide the objective functions each other. We proposed to use the probabilistic model called the hidden semi-Markov model as a particular model for power profiles in each household, and we have shown that the model can be smoothly integrated with the proposed framework.

Acknowledgment

This work was supported in part by CREST program of the Japan Science and Technology Agency (JST).

References

[1] S. Caron and G. Kesidis, "Incentive-Based Energy Consumption Scheduling Algorithms for the Smart Grid," *IEEE Int. Conf. on Smart Grid Communications*, pp. 391–396, Oct. 2010.

[2] A. Schülke, J. Bauknecht, and H. Johannes, "Power Demand Shifting with Smart Consumers," *IEEE Int. Conf. on Smart Grid Communications*, pp. 437–442, Oct. 2010.

[3] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous Demand-Side Management Based on Game-Theoretic Energy Consumption Scheduling for the Future Smart Grid," *IEEE Trans. on Smart Grid*, vol. 1, no. 3, pp. 320–331, 2010.

[4] S. Kishore and L. V. Snyder, "Control Mechanisms for Residential Electricity Demand in SmartGrids," *IEEE Int. Conf. on Smart Grid Communications*, pp. 443–448, Oct. 2010.

[5] T. Kato, K. Yuasa, and T. Matsuyama, "Energy on Demand: Efficient and Versatile Energy Control System for Home Energy Management," *IEEE Int. Conf. on Smart Grid Communications*, pp. 392–397, Oct. 2011.

[6] T.-H. Chang, M. Alizadeh, and A. Scaglione, "Coordinated Home Energy Management for Real-Time Power Balancing," *IEEE Power and Energy Society General Meeting*, pp. 1–8, Jul. 2012.

[7] M. H. Albadi and E. F. El-Saadany, "Demand Response in Electricity Markets : An Overview," *IEEE Power and Energy Society General Meeting*, pp. 1–5, 2007.

[8] M. Alizadeh, T.-h. Chang, and A. Scaglione, "Grid Integration of Distributed Renewables through Coordinated Demand Response," *51st IEEE Conf. on Decision and Control (CDC)*, pp. 3666–3671, Dec. 2012.

[9] S. Boyd, N. Parikh, E. Chu, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[10] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[11] M. Ostendorf, V. Digalakis, and O. A. Kimball, "From HMMs to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition," *IEEE Trans. on Speech and Audio Process*, vol. 4, no. 5, pp. 360–378, 1996.

[12] S. E. Levinson, "Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition," *Computer Speech and Language*, vol. 1, pp. 29–45, 1986.

[13] S.-Z. Yu and H. Kobayashi, "Practical Implementation of an Efficient Forward-Backward Algorithm for an Explicit Duration Hidden Markov Model," *IEEE Trans. on Signal Processing*, vol. 54, no. 5, pp. 1947–1951, 2006.

[14] E. Wei and A. Ozdaglar, "Distributed Alternating Direction Method of Multipliers," *51st IEEE Conf. on Decision and Control (CDC)*, pp. 5445–5450, Dec. 2012.