Reduced-order Load Models for Large Populations of Flexible Appliances

Mahnoosh Alizadeh, Anna Scaglione, Andy Applebaum, George Kesidis, and Karl Levitt

Abstract—To respond to volatility and congestion in the power grid, demand response (DR) mechanisms allow for shaping the load compared to a base load profile. When tapping on a large population of heterogeneous appliances as a DR resource, the challenge is in modeling the dimensions available for control. Such models need to strike the right balance between accuracy of the model and tractability. The goal of this paper is to provide a medium-grained stochastic hybrid model to represent a population of appliances that belong to two classes: deferrable or thermostatically controlled loads. We preserve quantized information regarding individual load constraints, while discarding information about the identity of appliance owners. The advantages of our proposed population model are 1) it allows us to model and control load in a scalable fashion, useful for ex-ante planning by an aggregator or for real-time load control; 2) it allows for the preservation of the privacy of end-use customers that own submetered or directly controlled appliances.

I. INTRODUCTION

Significant efficiency loss is experienced in the electricity market since demand is considered mostly inelastic. Demand response (DR) programs allow participants to modify their load in response to economic incentives. Dynamic pricing experiments seem to be promising [1], but the architecture for DR is not yet settled.

Irrespective of implementations, heterogeneous DR functionalities provided through the commercial and residential sectors are commonly subsumed in entities referred to as aggregators. Any aggregator with flexible demand will need to have day-ahead, hour-ahead and shorter term forecasts and risk models for the flexible loads it serves (ex-ante models). It will also need a real-time hybrid control model to execute the optimal planned control in real-time. This need for a scalable model of load flexibility is an inherent property of any type of end-use demand management, e.g., direct load control/scheduling and dynamic retail pricing. However, when controlling a large population of heterogeneous appliances, developing such models can be a burden. Consequently, oversimplification of aggregate demand characteristics and constraints is common.

The quest of this paper is to mathematically capture the ability of load profiles resulting from disparate, random uses to change into different shapes. We refer to the set of possible load shapes of an appliance or a population of appliances as load plasticity. Our idea is to build medium-grained models for the load plasticity of a population based on quantizing the intrinsic controllability of a few archetypes. These accurate yet scalable load models can be beneficial in multiple stages. As we will see in this paper, ex-ante, an aggregator can use stochastic load plasticities to plan forward energy market purchases, make DR decisions, and quantify any associated risks in a computationally-friendly fashion. They can also be used to derive a compatible real-time control framework that is accurate and privacy-preserving.

A. Contribution and prior art

There has been extensive work on aggregate load modeling at the transmission level to forecast inelastic demand. However, as loads start to become responsive at large scales, the need for aggregate models of flexible demand becomes more significant. Without sufficiently accurate models, widespread DR can become a reliability hazard instead of a resource.

Thus, the literature on DR includes a wide range of modeling options, corresponding to different resolutions in describing flexible load populations. One approach is to preserve all details about the appliance state and constraints. The electric power consumption properties are either idealized (often as a battery) or described realistically via hybrid dynamical systems equations. Examples of adopters of such models for market analysis are [2]–[6]. Detailed models are naturally advocated to shape the load in response to a price, using Home Energy Management Systems protocols [7], [8], or to follow a desired load profile via centralized controllers or task scheduling rules that are optimal in some conditions; many papers considered Electric Vehicles (EV) [9]–[13] and explored the optimality of simple policies such as Least Laxity First (LLF) and Earliest Deadline First (EDF).

Another series of papers propose to capture the total flexibility of deferrable loads for planning and market interactions as a tank that needs to be fully charged by a certain deadline [14]–[16]. All the specific characteristics of individual appliances except for their total energy consumption are discarded. Thus, the model has minimal computational cost for ex-ante planning, but it is unsuitable for scheduling due to its coarseness and lack of information about individual appliances.

These two groups of previous work sit at the two opposite sides of the spectrum in terms of accuracy and scalability. The goal of our work is to provide a medium-grained model that can be shifted across the spectrum as required for the application. The literature on flexible load modeling that most relates to our paper dates back to work on predictive modeling...
of the rebound peak due to emergency interruption of a homogeneous population of Thermostatically Controlled Loads (TCL) in [17]. The challenge in population modeling is that different TCLs are at different states (temperatures) at a certain point of time and have different comfort requirements. The important idea to capture their load in a single mathematical model is to classify these TCLs into groups with similar states, resulting in a discrete formulation. The concept of grouping similar TCLs in [17] was later further refined by [18]–[20]. The temperature range of devices is quantized in bins, and the occupancy of each of these bins is tracked using a linear state space model. This approach leads to promising results in providing ancillary services. Recently [21], [22] used a similar model for heterogeneous TCLs with a control approach inspired by processor scheduling, and proposed LLVM as a possible simple heuristic load control policy. The authors in [23] and our previous work [24] discretized state of charge, rate and deadline constraints to map the problem of charge scheduling of heterogeneous electric vehicles into a task scheduling problem. Here we take this work a step further by providing a unified framework to describe a large population of heterogeneous loads that can be deferred, interrupted, and whose consumption rate can be controlled, as well as TCLs.

A population load model is the basis for modeling the response of an appliance population to DR signals and alter the load in the desired fashion. We will see that to build a population load model, the aggregator must gather information about the energy usage time and pattern of different customers. Obtaining this information through the Smart Meters, however, has raised significant concerns about customer privacy. Finding good compromises (see e.g. [25]) may be hard, as the need to ensure grid safety usually overrules arguments for confidentiality. One of the byproducts of the reduced-order load model that we propose is that it can allow us to design a specific communication scheme to collect the data which is appliance identity-blind; that is, we require no individual identification for monitoring and direct scheduling of appliances. Previous work on privacy for the Smart Grid, however, has raised significant concerns about customer privacy. Finding good compromises (see e.g. [25]) may be hard, as the need to ensure grid safety usually overrules arguments for confidentiality. One of the byproducts of the reduced-order load model that we propose is that it can allow us to design a specific communication scheme to collect the data which is appliance identity-blind; that is, we require no individual identification for monitoring and direct scheduling of appliances. Previous work on privacy for the Smart Grid, however, has raised significant concerns about customer privacy. Finding good compromises (see e.g. [25]) may be hard, as the need to ensure grid safety usually overrules arguments for confidentiality.

Synopsis: We introduce the basic elements of making scalable population models using the simplest type of appliances, i.e., ideal batteries, in Secion II. In Section III, we generalize our models to realistic appliance categories. In Section IV, we have a high-level discussion of the potentials of population load plasticities to help with DR planning and control. In Section V, we discuss the communication requirements of submetering and direct scheduling frameworks that are based on hybrid population load models. Finally, in Section VI, we demonstrate the usefulness of accurate yet scalable population models for an aggregator controlling EVs and TCLs.

II. MODELING AGGREGATE DEMAND

Most appliances are hybrid systems whose evolution is described by both continuous and discrete state variables [32]. One such continuous variable is the electric power that the appliance consumes, which we denote by \( L(t) \in \mathbb{R}^+ \) for an appliance indexed by \( i \), and one such discrete variables is the power-switch state.

To address the aggregate demand modeling problem systematically we introduce the concept of load plasticity. We show how the plasticity can be represented as an integer linear model in some important appliance categories. Then, building on these models for the individual appliances, we propose the idea of clustering appliances by quantizing the load plasticity into discrete classes. This leads to computationally scalable aggregate load plasticity models.

Notation: We use roman font for continuous variables \( x(t) \) and italic for discrete variables \( x(t) \). Boldface is used for vectors \( x \). We denote finite differences with respect to time as \( \partial x(t) = x(t+1) - x(t) \). The unit step is denoted by \( u(t) \), and the Kronecker delta function is \( \delta(t) \) (equal to 1 if the argument is equal to zero and equal to 0 otherwise). The symbol \( \mathbb{E} \) denotes the discrete time convolution. We use the notation \( \mathbb{E} \{ x \} \) to indicate the expected value of a random variable \( x \).

Vocabulary: We say that an appliance arrivers in the grid when it is first available to consume electricity, e.g., when an EV is parked and plugged in, or when a TCL has to condition ambient temperature. The appliance may be immediately provided with electricity, as is the current service paradigm of the power grid. Alternatively, the service may be shifted to a later time under some type of demand management strategy, in which case we refer to the load as deferrable. The maximum amount of delay that an appliance can tolerate is captured by a deadline to finish its job, e.g., charge a battery in full, or it can be captured by a slack-time, defined as the amount of time left between the end of an appliance’s task and its deadline. For example, if a dryer cycle starts immediately after it is requested, the amount of time between the end of the cycle and its deadline represents the slack time.

A. A Modular Model for Demand Elasticity

Short-run elasticity of residential elasticity demand can be captured in two layers:
1) The first layer captures the elasticity in the overall demand level. This is manifested through the customer’s initial decision to use an appliance and the accompanying preferences, and will be modeled in this work through appliance arrival processes in the grid. This is always an economic decision in response to prices;

2) The second layer captures the inter-temporal elasticity of electricity demand once a request arrives. After the customer has decided to use a certain appliance, but he/she is flexible in the timing of electricity use, possible load shifting decisions mirror this layer of elasticity, and the effects are imperceptible to the customer. Here, the response can be controlled via economic prices or direct scheduling, depending on the type of DR employed.

These layers of demand flexibility are represented as two serial modules in the top and bottom diagrams in Fig. 1. In Section IV we further comment on the two diagrams Fig. 1 and suggest how the models introduced in this section can be used to design pricing strategies. However, the emphasis of this paper is not solving the price design problem. Rather, we mainly focus on the subproblem of modeling the second layer flexibility for a large population of heterogeneous appliances, and propose a model to manage the underlying heterogeneity of the response in a scalable fashion.

In fact, to approach the problem in a modular fashion, in this section, we take the decisions of customers to use electric appliances (arrival processes) as exogenous, and solely focus on the second module. Given the heterogeneous load characteristics and customer preferences for appliances that arrive in the grid, our goal is to have a reduced-order expression for the set of possible load shapes that any control scheme can extract out of an appliance population, which we refer to as load plasticity.

### B. Load Plasticity for the Ideal Battery

We refer to the potential of $L_i(t)$ for being modified by control actions as load plasticity. Mathematically, the load plasticity is a set of functions $L(t): t \rightarrow \mathbb{R}^+$ representing the load profiles that satisfy the service requirements specified by the end-user(s). We denote this set by $\mathcal{L}(t)$. Note that $\mathcal{L}(t)$ is not the set of possible load values for each specific time instance $t$. Rather, it includes a number of functions $L(t)$ with argument $t$, as individual members of a set.

1) The ideal battery: The simplest type of load plasticity is that offered by an ideal battery that remains on indefinitely. An ideal battery indexed by $i$ starts to charge at a certain time $t_i$, it has an initial charge of $S_i$ energy units, and a total charge capacity of $E_i$ energy units. Denoting the state of charge at time $t$ by $x_i(t)$, the load plasticity is the set of load profiles:

$$\mathcal{L}_i(t) = \{L_i(t) | L_i(t) = x_i(t), x_i(t_i) = S_i, 0 \leq x_i(t) \leq E_i, t \geq t_i \}.$$  \hspace{1cm} (1)

This model is analog and continuous and, thus, computationally infeasible. A natural step commonly taken in many control and communication applications that involve continuous values is to quantize signals, which is the approach we take in this paper. The reader will see that quantizing continuous values and signals is the basic principle which allows us to provide a medium-grained scalable model for the load plasticity of a population of non-homogeneous appliances.

2) Reduced-order model for a single ideal battery: To reduce the order of the load model, we start by quantizing time. If discrete time indices are separated by $\delta T$, we can use the index $t \in \mathbb{Z}$ such that $t = \delta t$. The second step consists of quantizing $E_i$ and $S_i$. We write the discrete counterparts of these variables as $(E_i, S_i) = (Q(E_i), Q(S_i))$, where $Q(\cdot)$ represents a uniform quantizer with step of $\delta E$ energy units. For brevity of notation, in the analytical part of this paper, we assume that both quantization steps are normalized to $\delta T = 1$ and $\delta E = 1$.

Thus, the discrete version of the load plasticity for an ideal battery becomes:

$$\mathcal{L}_i(t) = \{L_i(t) | L_i(t) = \delta x_i(t), x_i(t_i) = S_i, x_i(t) \in \{0, 1, \ldots, E_i\}, t \geq t_i \}.$$  \hspace{1cm} (2)

Note that $\mathcal{L}_i(t)$ is uniquely specified by the appliance type $v$ (ideal battery here), and the following parameters:

$$\vartheta_i = (t_i, S_i, E_i) \in \mathbb{T}^v.$$  \hspace{1cm} (3)

Hence, ex-ante, i.e., before the arrival of an appliance, the aggregator can model each load’s plasticity, $\mathcal{L}_i(t)$, by characterizing the randomness of these three parameters. The randomness of $t_i$ is captured by a random arrival process:

$$a_i(t) = u(t - t_i),$$  \hspace{1cm} (3)

with $a_i(t)$ equal to 1 if appliance $i$ is present in the system. The randomness of $S_i$ and $E_i$ is captured by a probability law $p(S_i, E_i | t_i)$ that reflects historical statistical information about requests. For a real-time controller, the state and thus the future load plasticity of an appliance changes as control actions are taken, and the residual set $\mathcal{L}_i(t)$ at time $t > t_i$ is given by replacing $S_i$ with $x_i(t)$ and $t_i$ with $t$.

### C. Reduced-order population model for ideal batteries

The difference between modeling a single ideal battery and a population of them ($i \in \mathbb{P}^v$ ) is in capturing the non-homogeneous individual parameters $\vartheta_i = (t_i, S_i, E_i)$ in the description of the aggregate load plasticity. After quantizing $\vartheta_i$ to reduce the order of the model, we cluster together all batteries that fall in the same quantization bin. We denote the plasticity of each member of these groups as $\mathcal{L}^v_\vartheta(t)$.

We define the following operations on plasticities $A_1, A_2$:

$$A_1 + A_2 = \{x | \exists (x_1, x_2) \in A_1 \times A_2, \text{ such that } x = x_1 + x_2 \} \hspace{1cm} (nA_1 = \{x | \exists x_i \in A_1, i = 1, \ldots, n, \text{ such that } x = \sum_{i=1}^n x_i \}.$$  \hspace{1cm} (4)

1Changing $\delta x$ and $\delta T$ to a generalized value is trivial and amounts to a scaling in all equations that map energy, e.g., $x(t)$, into power, e.g., $L(t)$. We avoid carrying on this scale as it unnecessarily burdens the notation.
where \( n \in \mathbb{N} \) and \( 0, A_1 = \{0\} \).

We denote as \( n_\vartheta^v \) the total number of batteries with \( \vartheta \in \mathcal{T}^v \) Then, the aggregate plasticity of the population \( \mathcal{P}^v \) is:

\[
\mathcal{L}^v(t) = \sum_{\vartheta \in \mathcal{T}^v} n_\vartheta^v \mathcal{L}^\vartheta_\vartheta(t). \tag{4}
\]

Next, we wish to simplify the description of plasticity \( \mathcal{L}^v(t) \) by decreasing the number of variables that define it. We do so by exploiting similarities in the sets \( \mathcal{L}^\vartheta_\vartheta \). Generally, the parameters that describe an individual appliance load can be divided in two subsets: one set of parameters, denoted by \( \kappa_i \), describe the initial state of control variables. For example, \( S_i \) and \( t_i \) here define the initial arrival event of each battery. Changing these quantities affects the load plasticity of an appliance only in a transient fashion, but does not affect the underlying structure of the plasticity. Appliances that only differ in terms of these initial parameters can be bundled together in a single population model for the load plasticity, as we will see next. Parameters in \( \vartheta_i \) not included in \( \kappa_i \) are denoted by \( \theta_i \), and are assumed to be homogenous across the population for now.

1) Clustering Batteries with Homogeneous \( E_i \): To start, suppose that we are modeling a population of batteries indexed by \( i \in \mathcal{P}_E \) that all have the same capacity \( E \); but have non-homogeneous \( S_i \) and \( t_i \). Using (2) and (3), we can write the total load of the batteries as:

\[
L(t) = \sum_{i \in \mathcal{P}_E} L_i(t) = \sum_{i \in \mathcal{P}_E} \partial x_i(t) a_i(t). \tag{5}
\]

To avoid tracking individual appliances and lower the complexity of (5), the basic idea first used in [33] is to track the number of appliances \( n_x(t) \) that are present in the system at time \( t \) and are at quantized state \( x \), discarding information that identifies individual appliances. This aggregation is what enables our anonymous modeling and control techniques. Mathematically, we can write \( n_x(t) \) as

\[
n_x(t) = \sum_{i \in \mathcal{P}_E} \delta(x_i(t) - x) a_i(t), \quad x = 1, \ldots, E \tag{6}
\]

Similarly, we denote as \( a_x(t) \) the total number of batteries that arrive in the system with an initial state of charge equal to \( x \) at, or before, time \( t \). The value of \( a_x(t) \) should be tracked in real-time as an input to the population model and can be written in terms of individual arrival processes \( a_i(t) \) as:

\[
a_x(t) = \sum_{i \in \mathcal{P}_E} \delta(S_i - x) a_i(t). \tag{7}
\]

We refer to \( a_x(t) \) as the \textit{arrival process for state} \( x \).

Next, we directly tie the evolution of \( n_x(t) \) and \( a_x(t) \) to the total load \( L(t) \), removing all dependence on \( x_i(t) \) and \( a_i(t) \).

\textbf{Lemma 2.1:} The following relationship holds between \( n_x(t) \) and the load \( L(t) \):

\[
L(t) = \sum_{x=0}^{E} \left[ \left( \sum_{x' = x}^{E} \partial n_{x'}(t) \right) - (x+1) \partial a_x(t) \right]. \tag{8}
\]

See Appendix A for proof.

The movement of appliances from one state to another is what determines \( \partial n_x(t) \). We capture this next.

\textbf{Definition 2.2:} We denote by \( d_{x,x'}(t) \) the number of batteries that go from state \( x \) to state \( x' \) at time \( t \). We refer to this number as the switch process from state \( x \) to \( x' \). We define \( d_{x,x}(t) = 0, \forall t, x, \) and \( d_{x,x}(0) = 0, \forall x, x' \).

\textbf{Corollary 2.3:} The occupancy \( n_x(t) \) and aggregate load \( L(t) \) in terms of \( d_{x,x'}(t) \) are:

\[
n_x(t + 1) = a_x(t + 1) + \sum_{x' = 0}^{E} [d_{x,x'}(t) - d_{x,x'}(t)] \tag{9}
\]

\[
L(t) = \sum_{x=0}^{E} \sum_{x'=0}^{E} (x' - x) \partial d_{x,x'}(t) \tag{10}
\]

\textbf{Proof:} The occupancy at time \( t + 1 \) should include the previous occupancy plus new arrivals from other states or from outside, minus the population that exits the state:

\[
\partial n_x(t) = \partial a_x(t) + \sum_{x' = 0}^{E} \partial [d_{x,x'}(t) - d_{x,x'}(t)] \tag{11}
\]

which leads to (9) if summed over time. If we substitute the value of \( \partial n_x(t) \) in (11) into (8), we get (10).

Thus, the load plasticity of a \( L(t) \) of a population of ideal batteries can be presented in terms of the \( d_{x,x'}(t) \)’s, under appropriate constraints:

\[
\mathcal{L}^v(t) = \{ L(t) | L(t) = \sum_{x=0}^{E} \sum_{x'=0}^{E} (x' - x) \partial d_{x,x'}(t), \partial d_{x,x'}(t) \in \mathbb{Z}^+, \sum_{x'=1}^{E} \partial d_{x,x'}(t) \leq n_x(t) \}, \tag{12}
\]

where \( n_x(t) \) is given by (9). The second constraint ensures that the number of appliances that leave state \( x \) at time \( t \) is less than or equal to the number of appliances present in state \( x \) at time \( t \), i.e., \( n_x(t) \).

One advantage of aggregating constraints for appliance populations into one model is that integrality constraints on \( d_{x,x'}(t) \)’s can be safely relaxed at large scales. This makes the population model linear and less computationally expensive.

Now, we go back to address non-homogeneity of the battery capacity \( E_i \) across the population. As we discussed, a set of parameters, denoted by \( \theta_i \), define the constraints that change the underlying structure of load plasticity. These parameters can include the physical constraints of the device, e.g., capacity of a battery or the wattage of a TCL, or the quality of service required by the user, e.g., the need to fully charge the battery by 8AM, or the need to keep the temperature inside a certain comfort band. Next, we address how we handle this change in nature of load plasticity by grouping loads in clusters.

\textbf{D. Load Clusters for Modeling Non-homogenous Populations}

The proposed hybrid scalable load model can be naturally generalized to handle a population of appliances with non-homogeneous \( \Theta_i \) by quantizing \( \Theta_i \). Thereby, we assume that appliances convey the parameters that capture their inherent constraints and the users’ consumption preferences from a finite number of choices. To provide an example, this means
that we can quantize the battery capacity to levels 1kWh apart. Thus, our model will not distinguish between two batteries with capacities 5.4 and 5.25 kWhs.

We bundle requests with similar constraints in clusters indexed by \( q = 1, \ldots, Q \):

\[
\theta_i \overset{\text{Quantize}}{\rightarrow} \theta_q \quad \text{Cluster index } \quad \tau \rightarrow q.
\] (13)

The level of quantization error can be controlled by modifying \( Q \) or \( \theta_q \)'s, and is the knob that controls the complexity and accuracy of the aggregate model. This amounts to quantizing both the feature space and constraints of each appliance.

For the ideal battery with no charging deadline, \( \theta_i = (E_i). \) This means that two batteries that share the same quantized capacity (\( E^q \)) and set of possible states \( \mathcal{X}^q = \{0, 1, \ldots, E^q\} \), but may vary in \( \kappa_i = (t_i, S_i) \), belong to the same cluster \( q \). However, if two batteries have different capacities, they are not banded together in the same cluster, since their load plasticities are different in nature. This is due to the fact that \( \mathcal{X}^q \) differs from one cluster to the next.

We use a superscript \( q \) to refer to any previously defined quantity for cluster \( q \). Thus, generalizing (12), the load plasticity of a non-homogeneous population of ideal batteries is,

\[
L^q(t) = \left\{ L(t) | L(t) = \sum_{q=1}^{Q} \sum_{x=0}^{E^q} (x' - x) \partial d_{x,x'}^{q}(t) \right\}
\]

\[
\partial d_{x,x'}^{q}(t) \in \mathbb{Z}^+, \sum_{x'=1}^{E^q} \partial d_{x,x'}^{q}(t) \leq n_x^q(t)
\]

with

\[
n_x^q(t) = a_x^q(t) + \left[ \sum_{x'=0}^{E^q} [d_{x,x'}^{q}(t-1) - d_{x,x'}^{q}(t-1)] \right]
\] (14)

This gives us a hybrid stochastic model for the load plasticity of a population of ideal batteries with no charge deadline. In Section III, we use the clustering concept introduced here to provide population models for appliances with different load shapes and user preferences, e.g., charging deadlines.

E. Stochastic Decision Models

The objective of this paper is not to propose a specific approach for demand management, e.g., [7]–[13]. Rather, we offer scalable and stochastic aggregate load models that can be used in conjunction with different DR schemes. The DR designer can use load plasticities with any preferred multistage stochastic optimization framework or be myopic. Obviously the computational complexity varies among different frameworks. Notice that the arrivals of appliances are naturally going to be non-stationary random processes. Thus, if the population is large, Monte Carlo sampling and Sample Average Approximation (SAA) on the random arrivals [34] are likely choices to reduce the stochastic optimization size.

Note that one would need to capture the statistics of the random arrival processes \( a_x^q(t) \) in order to model stochastic load. As an example of how one can gather and forecast these arrival statistics, the reader can see e.g. [35], which looks at forecasting and modeling Electric Vehicle (EV) charging requests arrivals using non-homogeneous Poisson processes.

III. REDUCED-ORDER MODELS FOR REALISTIC LOADS

Real appliance loads are more constrained in terms of controllability than the ideal battery. Some loads are merely deferrable but uninterruptible. Many loads can be only turned ON and OFF, i.e., \( L_i(t) \in \{0, G_i\} \). Some loads resemble the ideal battery, e.g., electric vehicles, but often include limitations on the rate of charge (change of state) and deadlines.

Here we build on the canonical battery population model to tackle other categories of real appliances. Appliances in each category \( v \) are bundled together in a single population model, whose plasticity set is \( L^v(t) \). There are \( Q^v \) clusters for each category \( v \). In general, an Aggregator can serve \( V \) different categories of loads. Given the population load plasticity of each category \( L^v(t) \), the load plasticity of the total demand served by an Aggregator as the set:

\[
L(t) = L^I(t) + \sum_{v=1}^{V} L^v(t)
\] (15)

where \( L^I(t) \) denotes the plasticity of the inflexible demand served by the Aggregator, which is the sample space containing the realizations of the random uncontrolled portion of the load process.

A. Rate-Constrained Instantaneous Consumption (RIC)

The first category looks at a general battery with a maximum charging rate and a deadline to be charged by a certain amount. This could capture electric vehicle charging requests, including Vehicle to the Grid (V2G) applications. We consider a non-homogeneous population of batteries, each characterized by the vector \((t_i, \chi_i, E_i, \rho_i, G_i)\). The new parameters over those of the ideal battery are as follows: \( \chi_i \) denotes the deadline for battery \( i \) to receive at least \( \rho_i \) percent of full charge, and \( G_i \) denotes the maximum rate at which battery \( i \) can be charged/discharged. The load plasticity of battery \( i \) is:

\[
L_i(t) = \left\{ L_i(t) | L_i(t) = \partial x_i(t), x_i(t_i) = S_i, x_i(t) \in \{0, 1, \ldots, E_i\}, x_i(\chi_i) \geq \rho_i E_i, -G_i \leq \partial x_i(t) \leq G_i, t \geq t_i \right\}
\] (16)

By setting \( \chi_i = \infty, \rho_i = 1, \) and \( G_i = E_i + 1 \) the model is equivalent to the ideal battery. Mapping this type of load plasticity into a population model is straightforward following the discussion in Sections II-C and II-D. We cluster batteries using \( \theta_i = (E_i, \chi_i, \rho_i, G_i) \), and denote the parameters associated with cluster \( q \) as \((E^q, \chi^q, \rho^q, G^q)\). Then, we need to modify the population model to handle the additional two constraints in (16) over that of the ideal battery in (2).

In each cluster \( q \):

1) Batteries cannot move from state \( x \) to any state \( x' \) in one time step. Due to rate limitations, at each time step, a battery can move by a maximum of \( G^q \) states, i.e. it can draw or deliver power at a rate \( |\partial x(t)| \leq G^q \). Thus, the switch processes from and to state \( x \) are only defined for \( x' \in S_x \) with

\[
S_x = \{x - G^q, x - G^q + 1, \ldots, x + G^q\};
\] (17)
2) All appliances in cluster $q$ should be in a state $x \geq \rho E^q$ by time $\chi$. This translates into $n_2^q(\chi^q) = 0, \forall x < \rho E^q$. Consequently, the load plasticity of the population is:

$$L^u(t) = \left\{ L(t) | L(t) = \sum_{q=1}^{Q} \sum_{x=0}^{E^q} (x' - x) \partial \alpha_{x,x'}^q(t), \right\}$$

$$\partial \alpha_{x,x'}^q(t) \in \mathbb{Z}^+, \sum_{x' \in S_x} \partial \alpha_{x,x'}^q(t) \leq n_2^q(t),$$

$$n_2^q(\chi^q) = 0, \forall x < \rho E^q \right\},$$

with

$$n_2^q(t) = a_2^q(t) + \sum_{x' \in S_x} [d_x^q(x(t-1) - d_x^{x'}(t-1)].$$

B. Interruptible service (IS)

This category is more constrained since it only allows batteries to charge at a constant rate $G_i$. The charge can however be interrupted multiple times. This category best models pool pumps or EVs that can only be charged at certain charging levels, e.g., 1.1 kW or 3.3 kW. Due to the similar nature of this category with the RIC, we simply present the population model. We cluster loads based on $\theta_i = (E_i, \chi_i, b_i, G_i)$, and the population plasticity is:

$$L^u(t) = \left\{ L(t) | L(t) = \sum_{q=1}^{Q} \sum_{x=0}^{E^q} (x' - x) \partial \alpha_{x,x'}^q(t), \right\}$$

$$\partial \alpha_{x,x'}^q(t) \in \mathbb{Z}^+, \sum_{x' \in S_x} \partial \alpha_{x,x'}^q(t) \leq n_2^q(t),$$

$$n_2^q(\chi^q) = 0, \forall x < \rho E^q \right\},$$

C. Thermostatically controlled loads (TCLs)

TCLs keep a representative temperature $x_i(t)$ within the boundaries of a comfort or safety band $[x_i^* - B_i/2, x_i^* + B_i/2]$ in the time window $\chi_i^*$. The day. We denote by $x_i^*$ the center and by $B_i$ the width of the comfort band. The information on $\chi_i^*$ can be used for preconditioning (precooling and preheating). Thus, the time frame at which the thermostat operates the appliance can be larger than $[\chi_i^*, \chi_i^*]$. We assume that a TCL can be operated within a time frame $[t_i^*, t_i^*]$. The availability of a TCL for control can be modeled through defining an arrival and departure process for TCL $i$:

$$a_i(t) = u(t - t_i^*), \quad r_i(t) = u(t - t_i^*).$$

TCLs are controlled with a cycling switch that turns them off/on, i.e., $L_i(t) \in \{0, G_i\}$. Emulating the approach proposed in [33], we take a quantized version of $x_i(t)$ to capture the state of TCL $i$ at time $t$. Using the model proposed in [36], we capture any random effects through a noise term denoted by $\alpha_i(t)$.

$$L_i(t) = \left\{ L_i(t) | L_i(t) = -k_i x_i(t) + \alpha_i(t) + b_i(t) G_i \right\},$$

$$b_i(t) \in \{0, 1\}, L_i(t) = b_i(t) G_i, \forall t \in [t_i^*, t_i^*]$$

$$|x_i(t) - x_i^*| \leq B_i/2, \forall [t_i^*, t_i^*] \in [\chi_i^*, \chi_i^*].$$

2 The additive noise term in the original model proposed in [36] is subsumed in the random process $\alpha_i(t)$.

where $b_i(t) = \{0, 1\}$ denotes the off/on status of TCL $i$. The rate of heat gain is taken to be $G_i$ Btu/h. The noise expected value is $\mathbb{E}[\alpha_i(t)] = x_{amb}(t)k_i$ where $x_{amb}(t)$ is the ambient temperature. We later assume that $x_{amb}(t)$ varies slowly in time, and can be approximated with a constant in each hour.

TCL load control, mainly through direct setpoint changes, has been widely studied, e.g., [37]. Aggregate TCL models are fairly complex compared to the models introduced thus far in this work, because: 1) state transitions are random due to the interaction with the randomly changing ambient temperature and customer activities; 2) the control action can only indirectly affect the number of appliances that migrate from one state to the other. In fact, let $p_{a_i}(\alpha; t)$ be the PMF of the quantized random process $\alpha_i(t)$ at time $t$. According to (22), a transition from state $x$ to state $x'$ occurs at time $t$ if and only if

$$\alpha_i(t) = x' - (1 - k_i)x - b_i(t)G_i$$

The impact of switching the control $b_i(t) \in \{0, 1\}$ is in changing the PMF that dictates how appliances move from one state $x$ to $x'$, denoted by $P_i(x'|x; t; b)$:

$$P_i(x'|x; t; b_i(t)) = p_{a_i}(x' - x(1 - k_i) - b_i(t)G_i).$$

This models the probability that TCL $i$ goes from state $x$ to $x'$ at time $t$ as the sum of two mutually exclusive random events:

1) If the appliance is on, i.e., $b = 1$, it can go from state $x$ to state $x'$ with success probability $P_i(x'|x; t; 1)$;

2) If the appliance is off, i.e., $b = 0$, it can go from state $x$ to state $x'$ with success probability $P_i(x'|x; t; 0)$;

To generalize this individual load plasticity to a population model, let us introduce:

Definition 3.1: Let $D_{x,x'}(t)$ denote the random number of TCLs that switch from state $x$ to state $x'$ in the interval of time $t$. Let $n_{x,x'}$ indicate the number of TCLs in state $x$ that have a status of $b = \{0, 1\}$ at time $t$.

Note that $n_{x,b}(t)$ is exactly analogous to the model in [38] and to the priority stack in [21]. What our model incorporates is the concept of clustering characteristics, which other authors commonly replaced with simplifying mean field assumptions. The discussion above implies that $D_{x,x'}(t)$ is the total number of TCLs that go from state $x$ to $x'$ under one of two Bernoulli random trials with success probability $P_i(x'|x; t; b_i(t))$.

In general, the switching probabilities $P_i(x'|x; t; b_i(t))$ are different for heterogeneous TCLs. Thus, following our previous quantization approach, we cluster the TCLs based on their basic characteristic tuple $\theta_q = \{G^q, x^q, B^q, \chi^q, \chi^q\}$, but also based on the statistics $p_{a,q}$. Consequently, we can define per-cluster switching PMFs $P^q(x'|x; t; b)$, which gives the probability of TCLs with switch position $b$ in cluster $q$ going from state $x$ to $x'$. Consequently, $D_{x,x'}(t)$ is the sum of two random components:

1) The number of TCLs among $n_{x,1}(t)$ TCLs in the ON position that go from state $x$ to state $x'$ with probability $P^q(x'|x; t; 1)$;

2) The number of TCLs among $n_{x,0}(t)$ TCLs in the OFF position that go from state $x$ to state $x'$ with probability $P^q(x'|x; t; 0)$;
Thus, $D_{x,x'}(t)$ at time $t$ is the sum of two Binomial random variables $\mathcal{B}(n_{x,b}(t), P^o(x'|x; t; b))$, with $b \in \{0, 1\}$. Hence,

$$\mathbb{E} \{ D_{x,x'}(t) \} = \sum_{b=0}^{1} n_{x,b}(t) P^o(x'|x; t; b)$$  \hspace{1cm} (25)

Observe that the control action available here is picking the number of appliances that are turned on/off in each state $x$, i.e., $n_{x,0}(t)$ and $n_{x,1}(t)$, subject to:

$$n_{x,0}(t) + n_{x,1}(t) = n_x(t).$$  \hspace{1cm} (26)

If we denote the set of all temperature bins in cluster $q$ by $S^q$, the total number of appliances in state $x$ in cluster $q$, $n^q_x(t)$, is governed by the dynamics:

$$n^q_x(t) = a^q_x(t) - r^q_x(t) + \sum_{x' \in S^q} D^q_{x,x'}(t-1) - D^q_{x,x'}(t-1).$$  \hspace{1cm} (27)

where $a^q_x(t)$ and $r^q_x(t)$ are the arrival and departure processes for state $x$ of cluster $q$. They respectively count the number of TCLs that first become available and unavailable for control before or at time $t$, when they are at state $x$ of cluster $q$.

The comfort band constraint translates into

$$\forall t: |x - x^q| > B^q/2 \rightarrow \Pr(n_x(t) = 0) \geq \eta,$$  \hspace{1cm} (28)

where $\eta$ is close to one, indicating that violations are rare if outright impossible.

Consequently, the stochastic aggregate plasticity of the population of heterogeneous TCLs is:

$$\mathcal{L}^o(t) = \{L(t)|L(t) = \sum_{q=1}^{Q^o} \sum_{x \in S^q} n_{x,1}(t), n_{x,1}(t) \in \mathbb{Z}^+ \}$$

$$n_{x,0}(t) + n_{x,1}(t) = n^q_x(t), n^q_x(t) = \text{see (27)}$$

$$\mathbb{E} \{ D_{x,x'}(t) \} = \sum_{b=0}^{1} n_{x,b}(t) P^o(x'|x; t; b);$$

$$\forall x : |x - x^q| > B^q/2, \forall t: \forall [x^q, \ldots, x^q] \rightarrow \Pr(n^q_x(t) = 0) \geq \eta \}$$

Real-time Coarse Clustering: Contrary to all other population models discussed in the paper, (29) is rather complex, mainly due to the fact that energy storage in a TCL is lossy. Note that (29) should naturally be paired with a model-predictive control strategy since its essence is in modeling the future random changes in temperature $x$ due to external noise and heat loss. The noise term $\alpha^q_x(t)$ in (22) can be replaced with its expectation $x_{amb}(t)\mu^q$ when using the model ex-ante, and this could considerably simplify planning decisions by lowering the number of clusters $Q^o$. However, for real-time direct scheduling, using (29) can be cumbersome.

Thus, we propose a coarser real-time clustering method that is considerably less computational and communication intensive compared to (29). Suppose that we limit ourselves to myopic policies, which is common practice in the field of routing and scheduling for large number of tasks. Note that the ultimate goal of a real-time direct scheduler would be to plan the switching events of each TCL under the constraint that the temperature does not go outside of the comfort band. Thus, one could envision that each TCL switching event can be scheduled by a deadline that can be predicted locally using (22). In this case, a good myopic on-line scheduling technique can be derived by assigning an index that describes the imminence of each switching deadline communicated to the scheduler by the TCLs. Then, the scheduler can cluster the population based on the quantized deadline index, rather than all of the parameters in (29).

Thus, in the real-time coarse clustering approach, TCLs are clustered based on the current state of their switch (ON or OFF) and based on how imminent is their next switching event. Assume that $\alpha^q_x(t)$ is known to the $i$th customer and it varies slowly. Note that the two boundaries that force the appliance to switch state can be written as a function of the switch state $b_i(t)$ in equation (22):

$$x_i^+ - (-1)^b_i(t)\frac{B_i}{2}.$$  \hspace{1cm} (25)

Then, solving the equation $x_i(\tau_i + t) = x_i^+ - (-1)^b_i(t)\frac{B_i}{2}$ using the linear dynamics in (22), the time at which appliance $i$ will reach its upper or lower boundary is:

$$\tau_i(t) = \frac{1}{\ln(n)} \ln \left( \frac{x_i(t) - b_i(t)\frac{G_i}{k_i} - \frac{\alpha^q_x(t)}{k_i}}{x_i^+ - (-1)^b_i(t)\frac{B_i}{2} - b_i(t)\frac{G_i}{k_i} - \frac{\alpha^q_x(t)}{k_i}} \right).$$  \hspace{1cm} (30)

Given that each TCL can locally calculate $\tau_i(t)$, we propose to use as the state of the TCL the pair $(\tau_i(t), b_i(t))$ that captures the switching deadline for each TCL, irrespective of all other cluster parameters. Note that the power consumption $G$ is also ignored in this abstraction. Thus, the scheduler will have to use the average power consumption $G$ of all TCLs in the population as an approximation of how much the load will be affected by each turn on/off event. Just as before, $\tau_i$ is quantized and replaced with the closest point $Q(\tau_i)$ in the set $\{0, \delta \tau, \ldots, (N-1)\delta \tau\}$. Therefore, if we now use $\tau$ as a discrete index, the population model is:

$$n_{\tau,b}(t) = \sum_{i \in P} \delta(\tau - Q(\tau_i))\delta(b - b_i(t))(a_i(t) - r_i(t)).$$  \hspace{1cm} (31)

We will see an example of how these two models can be used in Sections VI-E and VI-F respectively.

D. Non-interruptible Deferrable Service (NID)

This for category of loads we assume that, once on, the appliance needs to complete a cycle and will automatically follow a preset load profile. However, the starting time can be shifted across hours within customer-specified limits. This best models appliances such as washer/dryers, and non-interruptible EV charging. The load is the output of a hybrid system captured by two hybrid states, one being the system at rest, and the other is the system evolution once ON, captured by a continuous consumption profile.

Each appliance $i$ in this category is characterized by $(t_i, \chi_i, \xi_i(t))$, where $t_i$ is the arrival time, $\chi_i$ is the maximum tolerable delay to start consumption, and $\xi_i(t)$ is a pulse that captures the load profile of appliance $i$ (if it is turned on at $t = 0$). Hence, the only control available is shifting the load
to start at time $t_i \leq \tau_i \leq t_i + \chi_i$. Let $\ell_i(t)$ be the load signal if it starts at time 0. The load plasticity is simply

$$\mathcal{L}_i(t) = \{L_i(t) | L_i(t) = \ell_i(t) - \tau_i, t_i \leq \tau_i \leq t + \chi_i\} \quad (32)$$

The description above can be replaced with following integer linear model, based on the state $x_i(t)$ of the ON switch:

$$\mathcal{L}_i(t) = \{L_i(t) | L_i(t) = \ell_i(t) \ast d^x_i(t), x_i(t) \in \{0, 1\}\}, \quad (33)$$

where $x_i(t) \in \{0, 1\}$ denotes the state of the appliance, with 0 and 1 respectively corresponding to off and on status, and $\ast$ denotes the convolution operation. Note that an appliance can only be turned on after it arrives, i.e., $x_i(t) = 1$ at time $\tau_i$, if it has to remain 1 for the remainder of time due to the constraint $x_i(t - 1) \leq x_i(t)$. The convolution with $d^x_i(t) = \delta(t - \tau_i)$ yields $L_i(t) = \ell_i(t - \tau_i)$. The constraints ensure that an appliance can only be turned on after it arrives, i.e., when $a_i(t) = 1$, and that it cannot be turned off once on, i.e., appliances do not move from state $x = 1$ to state $x = 0$.

To build a population model, we cluster appliances based on quantized versions of the load profiles $\ell^q(t)$, plus the maximum tolerable delay $\chi^q$. They can tolerate the total number of arrivals and activations of cluster $q$ up to time $t$ are denoted by $a^q(t)$ and $d^q(t)$, respectively. Similar to the individual model, the load of cluster $q$ is the convolution of the number of activations with the pulse $\ell^q(t)$.

The population plasticity is the sum of contributions from each cluster $q$:

$$\mathcal{L}^q(t) = \left\{ \left( L(t) | L(t) = \sum_{q=1}^{Q^q} \ell^q(t) \ast d^q(t), d^q(t) \in \mathbb{Z}^+ \right) \right\} \quad (34)$$

$$d^q(t) \geq a^q(t - \chi^q), \quad d^q(t - 1) \leq d^q(t) \leq a^q(t)$$

E. General Hybrid System (GHS)

The modeling for the NID can be extended to include more hybrid states $x \in \mathcal{X}^q$ than just a single switch. We can extend the associated dynamics of the system as it enters in a certain hybrid state to state specific load injection profiles $\ell^q_x(t)$. This is useful for modeling non-interruptible tasks that follow each other with a single shared deadline. A simple example for residential use is the washer and dryer cycle.

In the general case, there will be a directed graph $\mathcal{G} = (\mathcal{S}^q, \mathcal{E}^q)$ with discrete states $x \in \mathcal{S}^q$ and edges $\mathcal{E}^q$ describing the hybrid automaton and constraining the switching $d^q_{x,x'}(t)$ to be between state pairs $(x, x') \in \mathcal{E}^q$, i.e., neighbors of state $x$ in $\mathcal{G}$. Hybrid states can be characterized by switching conditions that depend on the evolution of the hybrid state dynamics hitting a boundary.

In this case, the load plasticity is in scheduling transitions from hybrid state $x$ to hybrid state $x'$ before or after the boundary is hit. If the automaton can switch the discrete state every $\delta T$ and the injection rate is constant $G^q$, irrespective of the underlying job done in the state, the GHS model will end up resembling the RIC plasticity. We do not further elaborate for lack of space and worthwhile examples to explore.

IV. USING REDUCED-ORDER MODELS FOR DR PLANNING AND CONTROL

As mentioned in Section II-A, up to now, our efforts were focused on mathematically capturing the set of inherent load shapes that can be extracted from a large appliance populations by any DR scheme, taking the decisions of customers to use appliances as given. However, the real value of such models is realized when they are coupled with DR functionalities. Thus, in this section, we present benefits of having reduced-order load plasticity models when designing DR schemes.

As mentioned in the introduction, hybrid stochastic load models can help the planning and the real-time control decisions of an aggregator. For example, in a two-settlement structure:

**Ex-ante:** the aggregator plans how much power $B(t)$ to purchase and how much ancillary service capacity $M(t)$ to offer in the forward market. The aggregator picks $(B(t), M(t))$ to minimize an expected cost $C^F(L(t), B(t), M(t))$, which includes costs and benefits of buying energy and selling ancillary services over a certain time horizon $\Omega$, i.e.,

$$\min_{B(t),M(t)} \sum_{t \in \Omega} \mathbb{E}\{C^F(L(t), B(t), M(t))\} \quad \text{s.t.} \quad L(t) \in \mathcal{L}^{DR}(t), \quad (35)$$

where $\mathcal{L}^{DR}(t) \subseteq \mathcal{L}(t)$ denotes the set of possible load shapes that can be extracted from a population with load plasticity $\mathcal{L}(t)$ under a specific demand response strategy exercised by the aggregator. The expected value is over all future random realizations of load flexibility in $\mathcal{L}^{DR}(t)$, given the random nature of the appliance arrivals, captured through $a^q(t)$ in our model. For brevity, we assume that $(B(t), M(t))$ will be cleared by the market operator at the marginal price.

**Real-time:** the aggregator is committed to control $L(t)$ to follow the schedule $(B(t), M(t))$ for the current time $t$ and minimize its real-time cost. The aggregator can be myopic:

$$\min_{L(t)} C^R(L(t), B(t), M(t)) \quad \text{s.t.} \quad L(t) \in \mathcal{L}^{DR}(t), \quad (36)$$

or be farsighted, solving a model-predictive problem similar to (35) with the real-time cost $C^R$ as opposed to $C^F$. At time $t$, the cost includes $H$ dummy future decisions for a time horizon $\Omega = \{t, t+1, \ldots, t+H\}$, constantly revised and updated in real-time. Note that such model-predictive strategies will be very hard to implement if the aggregator considers the continuous characteristics of every single load in the population. Thus, without clustering, the aggregator may need to resort to myopic policies for online control, the consequences of which are studied in the numerical example (Section VI), wherein we solve (35) and (36) to showcase the usefulness of our proposed paradigm.

Note also that the cost functions $C^R$ and $C^F$ grow with the wholesale market clearing costs but they also depend on the retail market pricing strategy. In fact, a complex aspect in solving this problem is that the feasible set $\mathcal{L}^{DR}(t)$ depends on the response to the price incentives offered by the aggregator and the cost $C^R$ also depends on the price incentives.

Next we discuss the set $\mathcal{L}^{DR}(t)$ for the two most prominently discussed types of continuous DR, i.e., direct load scheduling and dynamic retail pricing (see Fig. 1).
A. Price-Based Direct Load Scheduling (DLS)

For this targeted type of DR, the aggregator directly observes the arrivals of the recruited flexible appliances and decides their exact load shape, choosing from the set $L(t)$. This makes DLS the most reliable form of load management possible in terms of real-time control because the aggregator has no uncertainty regarding the load response.

However, the set of possible load shapes $L(t)$ depends on the initial choice of end-use customers to participate in the DLS program and allow the aggregator to schedule their consumption, which is naturally an economic decision. A rational customer will not allow an aggregator to exploit load plasticity for free on a regular basis. Thus, we assume that the aggregator provides recruited appliances with differentiated participation incentives based on their level of load flexibility. So this is a form of targeted dynamic retail pricing.

Let us denote by $v^v(t)$ the price incentives available to appliances in category $v$ with parameters $v$ participating in the DLS program. The vector that contains all incentives for $v$ in category $v$ is $i^v(t)$. Then,

$$L(t) = L^I(t; i^v) + \sum_{v=1}^V \sum_{v \in T^v} n_v^v(i^v) L_v^v (t).$$

where the number of flexible appliances arriving in each category is modeled as a function of the incentives. Note that the incentives $i^v(t)$ have two main effects: 1) they could affect the participation rate of customers in the DLS program. Given higher incentives, customers will more easily agree to relinquish control to the aggregator; 2) they could affect the consumption habits of customers. For example, customers may run their dishwasher more often that usual when they can get lower tariffs by participating in DLS programs.

The design of economic incentives for DLS recruitment given the varying levels of elasticity for different clusters is the subject of ongoing research, see, e.g., our previous work [39], and the work in [40], [41], and is not discussed here.

B. Dynamic Retail Pricing

Here a single (general-purpose) price signal is used to bill all loads. This price will constitute the only knob available to the aggregator. Non-intrusive load monitoring aims at disaggregating the consumption of different appliances in a household based on meter information.

Due to these extra layers of complexity, previous works that study the price-elasticity of residential customers are commonly model-free [43]–[45], or rely on iterative communication that requires customers to specify their response to different potential prices before the price is actually set [46]–[48]. Due to lack of access to real-world data that captures the price-response of customers in using different clusters of appliances, we leave this problem to future work, and focus on DLS approaches in our numerical test cases.

V. SCALABLE AND ANONYMOUS TELEMETRY

Fig. 2 shows the steps that the aggregator should take to perform DLS. As the reader can see, the aggregator will need two-way communications with participating customers. Here we discuss how this communication can happen in an anonymous fashion. A reader not interested in anonymous communication and submetering can simply skip this section.

Under DLS, retaining anonymity means that the aggregator can be blind to the recruited customers’ identity and still be able to directly schedule their appliances. As a simple example, if John and Anna each submit a request to turn on their dishwasher, and Mary submits a request to turn on her pool pump, all the aggregator will be informed about is that there are two dishwasher requests and one pool pump request; neither will the aggregator learn the identities of John, Mary or Anna, nor will it be able to link the requests to individual users. However, the aggregator will still be able to communicate its scheduling decisions back to the appliances through the scheme discussed in this section.

The components of the telemetry architecture for DLS are:

1) Uplink traffic: The value of the arrival process $a^{q,v}(t)$ for each state $x$, cluster $q$ and category of appliance $v$ can be communicated anonymously to the aggregator;
2) **Downlink traffic:** To activate appliances, information regarding the decided schedule, e.g., $d_{q,v}(t)$ should be relayed to each waiting appliance in cluster $q$ in an appliance-blind fashion;

3) **Measurement and verification:** A slower channel is necessary to measure that the DLS control actions are correctly executed by the recruited appliances that received participation incentives.

We note that a Dynamic Pricing scheme can be supported by the current AMI setup. In the uplink, household meter information would be processed to produce estimates of $\partial a_{x,v}(t)$ [42], while the control is merely a price signal. Obfuscating meter information to protect privacy would corrupt the estimates of $\partial a_{x,v}(t)$, thereby having direct consequences on the control reliability.

1) **Downlink:** First, we shall clarify how the downlink works for DLS, i.e., how appliances are made aware of scheduling decisions. We omit the category index $v$ for brevity here. Since the aggregator’s scheduling decisions do not involve any individual appliance identifiers, the downlink communication should also be designed accordingly to function in an appliance-blind fashion as a broadcast message. Next, we discuss our proposed solution for each of the four appliance categories discussed in Sections III-A to III-D.

- **Categories III-A and III-B:** Here appliances in the same cluster $q$ all share the same deadline $\chi^q$. Thus, if the aggregator decides to move $d_{q,x,v}(t)$ from state $x$ to state $x'$, it can broadcast the following list of ratios to all appliances in state $x$:

$$\kappa^q_{x,x'}(t) = \frac{d^q_{x,x'}(t)}{n^q_x(t)}, \ x' \in S_x$$

Upon receiving these values, if an appliance happens to be in cluster $q$ at state $x$, they have to move to state $x'$ with probability $\kappa^q_{x,x'}(t)$. Given a large enough population size, this randomized scheme will perform well in executing the scheduling decisions.

- **Category III-C:** The decision that needs to be broadcast is $n^q_{x,1}(t)$, i.e., how many TCLs in cluster $q$ at state $x$ can be on at time $t$. Given the current value of $n^q_{x,1}(t)$, this activation can be executed using a simple randomized policy with

$$\kappa^q_{2,1}(t) = \frac{n^q_{2,1}(t)}{n^q_2(t)}$$

- **Category III-D:** Here we assumed that the population in cluster $q$ share the same slack time value $\chi^q$. However, since some appliances arrive earlier than others and should naturally be scheduled earlier, we can assume that arriving appliances are queued in a first-come-first-out (FIFO) discipline, i.e., if you arrived earlier in cluster $q$, you are scheduled earlier. But how can the aggregator address appliances with higher priorities in the queue when their identity is unknown? The idea is simple. At time $t$, the aggregator needs to find the time epoch $\tau^q \leq t$ at which the number of appliances that arrived in cluster $q$ since the origin of time, i.e., $a^q(\tau^q)$, was marginally higher than the number of appliances the aggregator plans to have activated since time epoch 1 until time $t$, i.e., $d^q(t)$:

$$\tau^q(t) = \min \{t' \leq t: a^q(t') \geq d^q(t)\}.$$  

If this time index $\tau^q(t)$ is broadcast to all appliances in cluster $q$, then the appliances that submitted requests before time $\tau^q(t)$ can start their task while the rest wait until authorized. If the difference between $a^q(t')$ and $d^q(t)$ is not negligible, a randomization ratio

$$\kappa^q(t) = \frac{a^q(t') - d^q(t)}{\partial a^q(t')}$$

can also be broadcast along with $\tau^q(t)$ to ensure that nearly $d^q(t)$ appliances are turned on.

2) **Uplink:** In the uplink, the aggregator just needs to gather the values of the arrival processes $\partial a^q_{x,v}(t)$ from the population. These values are aggregated across customers, but disaggregated across consumption clusters $q$, state $x$ and appliance categories $v$. In other words, while it is true that the appliances’ specific use modality is visible, no explicit information on appliance owners is required for the aggregator’s decisions. This makes any DR scheme using this model anonymous.

Each arrival event, i.e., every instance of $\partial a_{x}(t) = 1$, is marked by the transmission of a packet, including the appliance type $v_i$, its appropriate cluster index $q_i$ (associated to $\theta^q$), and potentially the state $x_i$. This results in a short packet, and can be easily integrated with simple application layer protocols used in industrial control and AMI networks. Because of this, the total communication traffic, in terms of number of average packets per $\delta T$ that enter the network, scales exactly as $\sum_{x,q,v} E\{\partial a^q_{x,v}(t)\}$. In each $\delta T$, the probability of transmission of a packet per appliance is a Bernoulli random variable with success probability equal to $p_i(t)$, i.e.:

$$p_i(t) = E\{\partial a_i(t)\}.$$  

Note that the discrete time bin $\delta T$ needs to be greater than the network end to end delay. This is the delay of getting a request forwarded from a HEMS upstream to the point where final delivery to the aggregator is made. This delay will vary widely depending on whether the AMI network or a conventional network provider serves this traffic. Current AMI mesh network solutions have significant latency but transmit a long record of raw samples of meter data. This is a much heavier task compared to forwarding the payload $\{v_i, q_i, x_i\}$ associated with the appliances’ requests. For instance, IEEE 802.15.4 (Zigbee), considering the packet acknowledgment, has a latency per packet bounded below by about $\delta T = 7$msec. However, this is assuming that the channel is idle. Denoting the ratio $\delta T / \delta T$ by $\alpha$, in a certain neighborhood $N$, the packet throughput per second is

$$\rho_N(t) = \delta T^{-1} \sum_{i \in N} p_i(t) \prod_{j \in N / i} (1 - \alpha p_j(t)).$$

To ensure that the data presented to the aggregator can be anonymous, a trusted third party has to act as an intermediary, placing neighborhood collector stations throughout the population. These third-party collectors manage their own neighborhood AMIs and tally each received payload $\{v_i, q_i, x_i\}$ to compute a local $a^q_{x,v}(t)$ that is then sent to the aggregator.

3For TCLs, the switch process $D^q_{x,v}(t)$ is communicated as well.
While collectors hide the identities of participating customers, their updates on \(a^{\text{Q}}_i(t)\) reveal some information about their possibly small neighborhoods. To mitigate this, we assume that the collectors have access to an out-of-band channel that can be used to deliver updates to the aggregator. Using this channel, anonymous routing protocols (e.g., onion routing between collectors [49]) can obfuscate the true sender of a message, allowing anonymous delivery of updates.

Note that the presence of neighborhood collectors is necessary to preserve customer anonymity when an otherwise significant portion of the AMI would be owned and operated by the aggregator. As we will explain in Section V-3, collectors also have the added benefit of providing verification mechanisms for DLS and align well with our goals.

While with these check and balances, it is possible to ensure anonymous transactions, the question that remains is whether anonymity is sufficient to ensure privacy. The problem that should be investigated is whether knowing aggregated \(a^{\text{Q}}_i(t)\) across the population can reveal information about individual users. This can be viewed a special case of the broader problem posed in differential privacy papers [25], [50]. We leave the study of this topic to future work.

3) Measurement & Verification (M&V) for DLS: Naturally, if DLS is performed, a verification mechanism must be in place to ensure that the control actions take place and economic benefits or penalties for the customer are appropriately accounted for. Furthermore, the customer can have real-time reports signaling if something goes wrong with the control system. The aggregator can also receive appropriately filtered reports to monitor the status and bill its appliance fleet.

During the day, the customer will have a specific household daily load profile \(L_i = (L_i(0), \ldots, L_i(N))\), where \(N\delta T\) is equal to the 24h period. This vector will have two components, \(L_i = L^i_1 + L^F_i\) one inflexible and random part \(L^i_1\), and another flexible portion \(L^F_i\) that has to follow the aggregator’s dispatch in a DLS program. If the control took place correctly, \(L^F_i\) should be entirely predictable ex-post based on the control actions. More precisely, \(L^F_i\) needs to be equal to a dispatched value \(L^DR_i\), whose value is known to the collector. The most precise approach to verify that \(L^F_i\) is equal to \(L^DR_i\) is to submeter the controllable appliances through a trusted Application Programming Interface (API) in the HEMS. The API can then forward the submetered load to the collector.

Theoretically, submetering can be replaced by a strategy that allows the collector to search for the presence of the desired profile in the aggregate household consumption. The M&V detection problem can be then posed as the detection of the profile in the aggregate household consumption. The M&V allows the collector to search for the presence of the desired profile in the aggregate load profile, allowing anonymous delivery of updates.

For instance, if \(L^i_1\) is a normal vector with mean \(\mu L^i_1\) and covariance \(W_{L^i_1}\), the test is equivalent to

\[
\|L_i - L^DR_i - \mu L^i_1\|_{W^{-1}_{L^i_1}}^2 \leq \eta^i.
\]

This means that the wider the variance of \(L^i_1\) is compared to \(L^DR_i\), the more the decision is plagued by uncertainty. This could make it possible for the customer to deceive the aggregator, even without tampering with the sensors. However, since flexible appliances are likely to be the most significant components of the household consumption pattern, it is highly likely that the error will be limited.

VI. NUMERICAL CASE STUDY

To showcase the benefits of our scalable framework for modeling and scheduling heterogeneous loads, we simulate the operation of an aggregator dealing with large populations of EVs and TCLs. As discussed in Section II-D, the operations of the aggregator can be divided into two steps: ex-ante planning to determine the optimal energy market purchase and ancillary service capacity to provide, and real-time operations to manage the load. For load management, we choose to perform DLS on both TCLs and EVs. The reasons for this choice were discussed in Section IV-B. In the ex-ante planning we make a significant simplification and assume that the recruitment of the appliances is given, ignoring the entanglement of the cost and feasible loads with the decision of price incentives that truly characterize this problem (see e.g. [39]).

A. Non-interruptible Plug-in Hybrid Electric Vehicles - Setting

We test our proposed model on a simulated population of 40,000 Plug-in Hybrid EVs (PHEV) that receive home charging. The operation of the aggregator is simulated for one full day, and the effects of shifting the load forward to the next day is captured by extending the duration of this study to 32 hours. Participating EVs in the DLS program arrive randomly, according to a non-homogeneous Poisson arrival process. The arrival rate, as well as the SoC and tolerable delay of each request, is simulated using real-world statistics in [35]. We divide arriving vehicles between 15 different clusters, representing different discrete SoCs between 0 to 4 hours and slack times between 1 to 3 hours (full charge corresponds to 5 hours of charging from a zero SoC). Charge requests are taken to be non-interruptible once started. We add the simulated charging load of this population on top of a typical daily base load scaled to have PHEV demand as 10% of peak load [51].

To numerically showcase the benefits of our reduced order, yet accurate, load model, we pick the aggregator objective to be a simple one: minimize the expected cost of serving the load given perfect forecasts of market prices but uncertainty in load arrival statistics \(a^\delta(t)\). The aggregator’s decision structure follows that of a two-settlement energy market (Section IV), which we describe next.

B. Non-interruptible PHEVs - Ex-ante Planning

The ex-ante cost captures the cost of buying energy in the forward market, plus the expected cost in the real-time market.
load plasticity will be captured by an extension of (12) to capture 32 different possible deadlines through 32 clusters:
\[
\mathcal{L}^v(h) = \left\{ \begin{array}{ll}
L(h) | L(h) = \sum_{q=1}^{32} \sum_{x=0}^{5} (x' - x) \partial d^q_{x,x'}(h),
& \frac{\partial d^q_{x,x'}(h)}{\partial d^q_{x,x'}(h)} \in \mathbb{Z}^+, \sum_{x=0}^{5} \partial d^q_{x,x'}(h) \leq n^q_x(h),
\end{array} \right\}
\]
with \( n^q_x(h) = a^q_x(h) + \sum_{x=0}^{5} [d^q_{x,x'}(h - 1) - d^q_{x,x'}(h - 1)] \).

Both population models (50) and (34) are inherently linear, allowing the cost-minimizing optimization (48) to be linear. Thus, given the stochastic nature of vehicle arrival patterns captured through \( a^q_x(h) \), we can find \( B(h) \) under both models by solving instances of a stochastic Integer Linear Program (ILP). The good news is that, if needed, integrality constraints on the decision variables \( d^q(h) \) and \( d^q_{x,x'}(h) \) can be relaxed at high aggregation levels without degrading performance.

The stochastic nature of vehicle arrivals in clusters is captured by a non-homogeneous Poisson process, which we show to be an appropriate choice based on real data in [35]. Thus, we apply sample average approximation (SAA) to the random vehicle arrivals and compute the expected cost considering multiple scenarios for the vehicle arrival rates. Since the cost under any scenario is linear with \( d^q(h) \), we solve a deterministic ILP. See the \( B(h) \) determined under the tank model and our hybrid model in Figures 5 and 4 respectively. Notice that the \( B(h) \) determined under the tank model is bouncier, due to the higher plasticity of ideal batteries. We will see why this is not a good choice for \( B(h) \) next.

C. Non-interruptible PHEVs - Online Scheduling

We simulate the performance of a direct scheduler whose objective is to make real-time scheduling decisions \( d^q(t) \) in (34) to minimize (49) in real-time, with different \( B(h) \) given by the two models. We approach the real-time scheduling problem in two ways:

1) Model-predictive Control (MPC) Based on Load Plasticity: Here we minimize the real-time cost in a model-predictive fashion, given the low complexity of solving such problems under our proposed clustered load model. Note that in real-time, the arrival counts of EVs in each cluster \( q \), i.e., \( a^q(h) \), are only causally observed. Thus, at each \( h \), the
scheduler has to solve a model predictive optimization to make scheduling decisions \( d^q(t) \) while taking into account the possible scenarios for future arrivals. Here, we take a certainty equivalent approach at each \( h \), replacing future arrival numbers of EVs for each cluster \( q \) with their expected value \( \mathbb{E}[a^q(h')] = \lambda^q(h'), h' > h \). See [35] for a discussion of how \( \lambda^q(h') \) can be predicted.

2) Myopic Approach: Without load clustering, the problem of scheduling 40,000 loads in real-time can be computationally prohibitive. Thus, here we simulate the performance of a myopic scheduler that, at each time \( q \), simply picks the PHEVs that are closest to their deadline and schedules those vehicles for charge. The number of vehicles picked at each time is such that the scheduled load is as close as possible to the forward purchase. We refer to this myopic scheme as Earliest Deadline First (EDF). Many authors have previously investigated EDF for EV scheduling, e.g., [10], [11].

To compare the performance of population models (50) and (34), we use the MPC method to schedule load in real-time, with the goal of minimizing (49) under the forward purchase determined from these two models. Figures 4 and 5 compare the actual hourly loads of the aggregator in these two scenarios. The reader can observe that the coarse nature of the tank model leads to large deviations of the real-time load from the bulk purchase. With the clustering method, on the other hand, the deviation of the real-time load from the bulk purchase is nearly negligible. To give an intuitive explanation for the poor performance of the tank model, the reader should note that non-interruptible PHEVs have a more restricted load plasticity than the ideal battery.

To compare the effectiveness of MPC to that of LLF, we simulate the performance of both real-time scheduling schemes over the forward purchase determined through our model in Fig. 6. Notice the considerable deviations from the forward purchase for the LLF scheduler.

D. Thermostatically Controlled Loads - Setting

Another useful property of scalable hybrid models is in determining the ancillary service capacity that a large population of flexible appliances can collectively provide. Here we simulate the performance of an aggregator that directly controls the on/off state of 10,000 residential TCLs in heating mode for a duration of 6 hours. The ambient temperature for each hour \( h \), \( x_{amb}(h) \), in the simulated interval mirrors that of Jan 29th, 2012 in Davis, California between 12 to 6 am.

The comfort band, wattage \( G_i \), initial temperature \( x_i^{\text{init}} \), loss rate \( k_i \), and initial state \( b_i \) of each device were generated randomly. The parameters \( G_i \) and \( k_i \) were randomly chosen to be independent and identically distributed (iid) from uniform distributions \( U([2000, 4000]) \) Btu/h and \( U([50, 200]) \) W/C, and the heat gain was simply picked to be equal to \( G_i \). The desired temperature of each device, \( x_i^* \), came from an iid uniform distribution \( U([69, 75]) \) °F, and \( B_i \) from \( U([2, 4]) \) °F.

Remember that for population models, we needed to quantize the individual parameters of each device and assign them to clusters. Here, \( G_i, k_i, x_i^* \) and \( B_i \) were quantized into 3, 4, 4, and 2 levels respectively. Thus, the TCLs were divided into 96 clusters to form the population model. For simulation purposes, the initial temperature was uniformly chosen to be in the comfort band, while initial on/off states were chosen from a Bernoulli distribution with success probability of \( s = 0.15 \).

In what we refer to as the autonomous switching scheme, a heating unit is turned on when the temperature dips below the band and turned off when it gets above the band. This describes the normal switching behavior of a TCL. The resulting load is what we take as the baseline load, which has the following expected value in the forward market:

\[
\mathbb{E}[L_{\text{base}}(h)] = \sum_{q=1}^Q n^q(h)G^q \frac{x_{amb}^q(h)}{\tau_{\text{on}}^q(h) + \tau_{\text{off}}^q(h)},
\]

(51)

\[
n^q(h) = \mathbb{E} \left[ \sum_{x \in \mathcal{S}} \sum_{\ell=1}^T n_x^\ell((h-1)T + \ell) \right],
\]

(52)

and \( \tau_{\text{on}}^q \) and \( \tau_{\text{off}}^q \) respectively denote the expected on and off times of a TCL in cluster \( q \) on the day-ahead. They are:

\[
\tau_{\text{on}}^q(h) = \frac{1}{k_i} \ln \left( \frac{x^{\text{amb}} - x^{q} - \frac{G^q}{k_i} - \frac{x_{amb}^q}{k_i}}{x^{\text{amb}} + \frac{G^q}{k_i} - \frac{x_{amb}^q}{k_i}} \right),
\]

(53)

\[
\tau_{\text{off}}^q(h) = \frac{1}{k_i} \ln \left( \frac{x^{\text{amb}} - x^{q} + \frac{B^q}{k_i} - \frac{x_{amb}^q}{k_i}}{x^{\text{amb}} + \frac{B^q}{k_i} - \frac{x_{amb}^q}{k_i}} \right).
\]

(54)

To simplify the compilation of these these average values, we take the noise term \( \alpha_t(t) \) in (22) to be equal to its expected value, \( \mathbb{E}[\alpha_t(t)] = x_{amb}(h) a^\delta \) (c.f. [22]).

E. Thermostatically Controlled Loads - Ex-ante Planning

Ex-ante, the objective of the aggregator is to find the maximum regulation capacity that it can safely provide using a population. In real-time, the status of the TCL devices has
to be scheduled to follow the dispatch as closely as possible. Here we are focused on the ex-ante modeling.

The AGC signal determines how much the TCL aggregate load is required to deviate from the baseline load. AGC signals are claimed to resemble independent random variables with a zero mean. In practice, this is not the case and the signals are correlated in time. Hence, to be reliable, the aggregator must be able to increase/decrease demand from the baseline and hold the demand at that value for a certain duration. The natural question that comes to mind is: how much can the demand be increased/decreased from the baseline? For TCLs in different clusters, the answer is different.

Let us discuss the extreme case in which a TCL has to increase/decrease its load from the baseline in a stationary setting (i.e. for an indefinitely long duration). In this case, the change in consumption from base load is no longer considered as a transient state. This stationary load can be thought of as the load of a TCL that autonomously switches in a comfort band \([\theta_{q_{\text{on}}}^{\text{max}}, \theta_{q_{\text{off}}}^{\text{max}}]\) \(\subseteq [x_{q_{\text{on}}}^{\text{on}} - B_{q_{\text{on}}} / 2, x_{q_{\text{off}}}^{\text{off}} + B_{q_{\text{off}}} / 2]\). There is a limit to the highest and lowest amount of stationary consumption that can be achieved by a population of TCLs. If we choose to have a minimum width of \(\Theta\) units for the comfort band, the scenario corresponding to the maximum amount of stationary power consumption for cluster \(q\) is \(\theta_{q_{\text{on}}}^{\text{max}} = x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2\) and \(\theta_{q_{\text{off}}}^{\text{min}} = x_{q_{\text{off}}}^{\text{off}} + B_{q_{\text{off}}}^2 / 2 - \Theta\). This is because, in this case, the temperature respectively drops and increases at its quickest and slowest possible rate (see \(\partial x_i(t)\) in (22)). This leads to the maximum possible amount of time that the TCL is on. Since TCLs, when on, have constant power consumption \(C_q\), the maximum stationary expected load of the population is:

\[
\mathbb{E}[L_{\text{max}}(h)] = \sum_{q=1}^{Q} n_q^q(h) C_q \frac{\theta_{q_{\text{on}}}^{\text{max}}(h) + \theta_{q_{\text{off}}}^{\text{min}}(h)}{\tau_{q_{\text{on}}}^{\text{max}}(h) + \tau_{q_{\text{off}}}^{\text{min}}(h)},
\]

where \(\tau_{q_{\text{on}}}^{\text{max}}\) and \(\tau_{q_{\text{off}}}^{\text{min}}\) respectively denote the expected on and off times of a TCL in cluster \(q\) operating in the comfort band \([x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2 - \Theta, x_{q_{\text{off}}}^{\text{off}} + B_{q_{\text{off}}}^2 / 2\]. They are:

\[
\tau_{q_{\text{on}}}^{\text{max}}(h) = \frac{1}{k^q} \ln \left( \frac{x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2 - \Theta - C_q \theta_{q_{\text{amb}}}^q}{x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2 - \Theta - \theta_{q_{\text{amb}}}^q (h)} \right),
\]

\[
\tau_{q_{\text{off}}}^{\text{min}}(h) = \frac{1}{k^q} \ln \left( \frac{x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2 - \Theta - \theta_{q_{\text{amb}}}^q (h)}{x_{q_{\text{on}}}^{\text{on}} + B_{q_{\text{on}}}^2 / 2 - \Theta - C_q \theta_{q_{\text{amb}}}^q} \right).
\]

Similarly,

\[
\mathbb{E}[L_{\text{min}}(h)] = \sum_{q=1}^{Q} n_q^q(h) C_q \frac{\tau_{q_{\text{on}}}^{\text{min}}(h) + \tau_{q_{\text{off}}}^{\text{max}}(h)}{\tau_{q_{\text{on}}}^{\text{min}}(h) + \tau_{q_{\text{off}}}^{\text{max}}(h)},
\]

where \(\tau_{q_{\text{on}}}^{\text{min}}\) and \(\tau_{q_{\text{off}}}^{\text{max}}\) respectively denote the average on and off times of a TCL in cluster \(q\) operating in the comfort band \([x_{q_{\text{on}}}^{\text{on}} - B_{q_{\text{on}}} / 2, x_{q_{\text{off}}}^{\text{off}} - B_{q_{\text{off}}} / 2 + \Theta]\).

This leads to a very conservative estimate of the regulation capacity that can be safely provided by the TCL population:

\[
M = \min_h \mathbb{E}\{L_{\text{base}}(h) - L_{\text{min}}(h), L_{\text{max}}(h) - L_{\text{base}}(h)\}.
\]

In reality, to provide regulation services, the TCLs do not have to hold their power consumption at \(L_{\text{max}}(h)\) and \(L_{\text{min}}(h)\) for a very long time. One way to get a less conservative and more profitable estimate of the capacity, \(M'\), is to simulate the response of each cluster to a control signal that increases or decreases the load from its baseline by a certain step of variable height \(m\). The population should be able to increase and hold the consumption at baseline plus \(m\) for a certain amount of time \(\xi\). Here we choose \(\xi\) to be the 97 % quantile of the zero-crossing time of historical regulation signals. \(M'\) is equal to the maximum step size \(m\) for which the response has an acceptable error. When simulating the response, the initial temperature of all units is chosen uniformly at random within their comfort band. Here we take a response to be acceptable if the deviation from the target load is less than 0.05\(M'\) for 95 percent of the duration \(\xi\).

The value of \(\xi\) is 19 minutes based on 18 days of regulation signal data available through PJM. By simulating the response of each cluster to different step sizes at different hours \(h\), we determine the value of maximum safe deviation from base load for each \(q\), which we dub as \(M^q(h)\). Consequently,

\[
M' = \sum_{q=1}^{Q} \min_h n_q^q(h) M^q(h),
\]

which is equal to 2.05 MWs for our simulated TCL population.

F. Thermostatically Controlled Loads - Online Scheduling

In real-time, the status of the TCL devices has to be directly scheduled so that the load follows the regulation dispatch signal as closely as possible. We use the coarse clustering method in (31) to perform this scheduling. Clusters are assigned to different deadlines one minute apart within the next 30 minutes, resulting in 60 separate clusters for both on and off TCLs. After each switching event, each TCL waits for a short time (courtesy period). Then, they submit their next switching deadline to the scheduler and wait. If anytime during this wait the temperature goes out of the comfort band, the TCL autonomously switches on/off. These are the events that can result in deviations from the dispatch, and are mainly due to forecast errors in the deadlines.

Normalized regulation signals available online through PJM were used, scaled to the regulation capacity offered by the aggregator. Fig. 7 shows the performance of our proposed algorithm for a duration of 6 hours. The aggregate consumption follows the dispatched deviation from the baseline consumption very closely. Individual representative temperatures of 3 random participating units is shown concurrently. We believe that the choice of \(M'\) was near-optimal, since the population load response was not acceptable when the regulation capacity was increased to 2.4 MWs.

G. Communication requirements

Here we estimate the communication requirements of implementing such DLS programs. We assume that every 1000 residential households are connected via a multiple-access channel (MAC) to a collector node, who talks to the aggregator. Each house has 2 electric vehicles and 3 TCLs. Given the frequency of request arrivals for these appliance categories, the maximum
rates on market decisions and scheduling.

The rate requirement of the MAC channel in (45) is 4.9 packets per second. This is a very low rate for a ZigBee collector, which can accommodate up to 1/0.007 ≈ 142 packets per second. What is most likely going to limit the number of houses covered by one collector is the range at which the collector can communicate, which is 100 meters for ZigBee. Inside this diameter, a ZigBee collector can cover 0.0142 packets per second per square meter, which can cover a 284 story building with 3 TCLs and 2 EVs in each residence. This limit will obviously not be reached in realistic scenarios. The collectors forward the \( a^q(t) \)'s to the aggregator every one minute through the internet. This includes 15 values for the PHEVs and 60 for the TCLs (equal to the number of clusters), amounting to less than a byte of information. The same negligible rate requirement holds for the downlink broadcast channel.

VII. FUTURE WORK

We presented a stochastic hybrid model for large populations of heterogeneous appliances, and showcased the benefits of having such a model when making market decisions and in scheduling. Future work will explore: 1) the optimal clustering method (number of clusters and bundling rule) given consumption statistics; 2) the impact of different clustering levels on market decisions and scheduling.

APPENDIX A

PROOF OF LEMMA 2.1

Proof: Note that \( x_i(t+1) = x_i(t) + \partial x_i(t) \) and \( a_i(t+1) = a_i(t) + \partial a_i(t) \). Using (6), we can write:

\[
\begin{align*}
\frac{E}{E} \sum_{x=0}^{E} \frac{E}{E} \delta (x_i(t+1) - x') a_i(t+1) \\
- \sum_{x=0}^{E} \sum_{x'=x}^{E} \delta (x_i(t) - x') a_i(t) \\
= \sum_{i \in P} \sum_{x=0}^{E} \delta (x_i(t) + \partial x_i(t) - x) (a_i(t) + \partial a_i(t)) \\
- \sum_{i \in P} \sum_{x=0}^{E} \delta (x_i(t) - x) a_i(t) \\
= \sum_{i \in P} \sum_{x=0}^{E} \delta (x_i(t) + \partial x_i(t) - x) \delta (x_i(t) - x) a_i(t) \\
+ \sum_{i \in P} \sum_{x=0}^{E} \delta (x_i(t) + \partial x_i(t) - x) \delta a_i(t) \\
= \sum_{i \in P} L_i(t) + \sum_{i \in P} (S_i + 1) \delta a_i(t) \\
= L(t) + \sum_{x=0}^{E} x \cdot \delta a_x(t).
\end{align*}
\]

The second to last equality holds following (5) and the fact that \( \partial a_i(t) \) is only non zero when \( t_i = t + 1 \), i.e., \( x_i(t+1) = S_i \).

REFERENCES


