Empirical Characterization, Modeling, and Analysis of Smart Meter Data

Sean Barker, *Student Member, IEEE*, Sandeep Kalra, *Student Member, IEEE*, David Irwin, *Member, IEEE*, and Prashant Shenoy, *Fellow, IEEE*

***Abstract*—Smart meter deployments are spurring renewed interest in analysis techniques for electricity usage data. However, an important prerequisite for data analysis is characterizing and modeling how electrical loads use power. While prior work has made significant progress in deriving insights from electricity data, one issue that limits accuracy is the use of general and often simplistic load models. Prior models often associate a fixed power level with an "on" state and either no power, or some minimal amount, with an "off" state. This paper's goal is to develop a new methodology for modeling electric loads that is both simple *and* accurate. Our approach is empirical in nature: we monitor a wide variety of common loads to distill a small number of common usage characteristics, which we then leverage to construct accurate load-specific models. We show that our models are significantly more accurate than binary on–off models, decreasing the root mean square error by as much as 8× for representative loads. Finally, we demonstrate three novel applications that use our empirical load models to analyze and derive insights from smart meter data, including i) generating device-accurate synthetic traces of building electricity usage; ii) filtering out loads that generate rapid and random power variations in smart meter data; and iii) detecting the presence of specific load models in time-series power data.**

***Index Terms*—Power system measurements, meter reading power system modeling, load modeling home automation, smart homes.**

I. Introduction

COMPUTING for sustainability—where real-world physical infrastructure leverages sensing, networking, and computation to mitigate the negative environmental and economic effects of society's energy use—has emerged as an important new research area. As a result, in addition to improving the energy efficiency of information technology (IT) infrastructure, such as smartphones, servers, network devices, and data centers, computing researchers are now expanding their focus to include building energy efficiency. Since buildings account for nearly 40% of society's energy use [1], compared to an estimated 1–2% for IT infrastructure [2], this research has the potential to make a significant impact. In particular, efficiently managing electricity is critical because buildings consume the vast majority (73%) of their energy in the form of electricity [1]. Existing management techniques typically employ *sense-analyze-respond control loops*: various *sensors* monitor the building's environment (including electricity) via a smart meter, and transmit collected data in real-time to servers, which *analyze* it to reveal detailed building usage and occupancy patterns, and finally *respond* by automatically *controlling* electrical loads[1] to optimize energy consumption.

Research challenges exist at each stage of this control loop. For example, despite much prior research [3]–[6] accurate, fine-grained, e.g., $\geq$ 1 Hz, *in situ* sensing of electricity use in real time at large scales remains impractical, as it is prohibitively expensive, invasive, and unreliable. Unfortunately, timely and detailed knowledge of per-load electricity use is a prerequisite for implementing many sophisticated automated load control policies that increase energy efficiency. Since, even modest-sized, residential homes operate hundreds of individual loads, providing per-load data using existing sensors would require a large-scale sensing system [7]. One promising approach to address this problem is to use fewer sensors that generate less data, and compensate by employing more intelligence in the analysis phase to infer rich information from the data. For example, prior research indicates that analyzing changes in a building's aggregate electricity usage at small time granularities, e.g., every 15 minutes or less, reveals a wealth of information: Non-Intrusive Load Monitoring (NILM) techniques use the data to infer electricity usage for individual loads [8], while recent systems use it to infer building occupancy patterns [9], [10]. These inferences then inform control policies: NILM might enable buildings to identify opportunities for reducing peak demand by scheduling elastic background loads [11], such as air conditioners and heaters, while occupancy patterns are critical in determining when to turn loads off without disturbing people's lives [12].

Since electric utilities are rapidly deploying digital smart meters capable of measuring and transmitting a building's aggregate electricity usage in real time, a substantial amount of fine-grained electricity data for buildings is already available. For example, Pacific Gas and Electric now operates over nine million smart meters in California [13]. While today's deployed smart meters typically measure average power usage at intervals ranging from fifteen minutes to an hour, the granularity of data is trending downwards (e.g., some utilities already provide 5-minute data [14]), and commodity meters are available that measure and transmit, via the Internet, energy usage at intervals

Manuscript received March 27, 2014; accepted May 10, 2014. Date of publication June 20, 2014; date of current version August 13, 2014. This work was supported by the NSF under Grants CNS-1253063, CNS-1143655, CNS-0916577, CNS-0855128, CNS-0834243, and CNS-0845349.

The authors are with the University of Massachusetts Amherst, Amherst, MA 01003 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSAC.2014.2332107

[1]We use the term electrical load, or simply *load*, to refer to any distinct, self-contained appliance or device that consumes electricity.

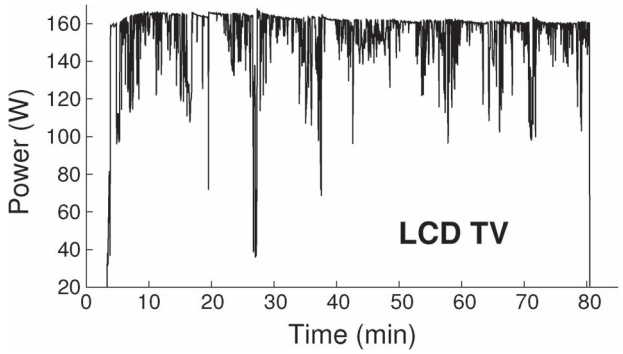

Fig. 1. An LCD TV's power usage varies rapidly, significantly, and unpredictably while on, and does not conform to a simple on-off load model.

as small as every second [15], [16]. Combined with the emergence of "big data" cloud storage systems, these smart meter deployments are spurring renewed interest in analysis techniques for smart meter data. While prior research has made significant progress in deriving insights from smart meter data [17], one issue that often limits accuracy is the use of general, and often simplistic, load models. In particular, many prior techniques for analyzing and modeling building electricity data characterize loads using simple on-off models, which associate a small number of fixed power levels with the "on" state (often just one) and either no power usage, or some minimal amount, with the "off" state.

On-off models do have a number of advantages. For instance, they exactly capture many simple loads, including light bulbs and other low-power resistive devices with mechanical switches. In addition, on-off models allow researchers to describe buildings as state machines that associate each building state with a fixed power level (implying the set of loads that are on), and where state transitions occur whenever a load turns on or off. Characterizing buildings as state machines admits a plethora of analysis techniques. For instance, much prior work maps building state machines to Hidden Markov Models (HMMs), and applies HMM-based techniques, such as Viterbi's algorithm [18], [19], to determine which loads are on in each state. In this case, using only a few (often two) power states per load is advantageous, since it minimizes the number of distinct power states for the entire building and reduces the complexity of analyzing the resulting state machine. Of course, even with only two power states per load, the number of building power states is still exponential in the number of loads, i.e., 2^n for n loads. Thus, even assuming simplistic on-off load models, precise analysis may still be intractable, i.e., require enumerating an exponential number of states.

Unfortunately, while on-off load models are simple, they are often inaccurate, since they fail to capture the complex power usage patterns common to many loads. As a simple motivating example, Fig. 1 shows a time-series of an LCD TV's electricity usage each second. In this case, the TV's switched mode power supply (SMPS) causes power variations as large as 120 watts (W) by rapidly switching between a full-on and full-off state to minimize wasted energy. The magnitude of these variations is effectively random—determined by the color and intensity of the TV's pixels. An on-off model clearly does not accurately capture the TV's power usage. As a result, modeling the TV as an on-off load may complicate higher-level analysis techniques for smart meter data. For example, the TV may obscure the use of low-power loads, such as a 60 W light bulb, since its power usage varies rapidly by > 60 W.

Our premise is that simple on-off models discard a significant amount of information that is potentially useful in analyzing data. As a result, in this paper, we focus explicitly on accurately characterizing and modeling a variety of common household loads. Our methodology is empirical: we i) gather fine-grained electricity usage data from dozens of loads across multiple homes, ii) characterize their behavior by distilling a small number of common usage attributes, and then iii) derive accurate load-specific models based on these attributes. One of our contributions is to show that a small number of model types, stemming from basic knowledge of power systems, accurately describe nearly all household loads. Thus, one of our goals is to highlight how many identifiable load attributes, which are well-known in power systems, manifest themselves in electricity data collected by smart meters. *Our hypothesis is that accurate load models, which leverage domain knowledge from power systems, provide a foundation for designing new electricity data analysis techniques*. In evaluating our hypothesis, this paper makes the following contributions.

Empirical Data Collection and Characterization. We instrument a wide variety of common electrical loads in multiple homes, and collect electricity usage data for each load, every second, for over two years. We show empirically that homes operate similar types of loads, e.g., lighting, AC motors, heating elements, electronic devices, etc., which results in significant commonality in power usage profiles across loads. We then characterize the data to identify distinguishing attributes in per-load power usage, forming the building blocks of our models. While many of these attributes are well-known in power systems, we show how they manifest in sensor data.

Data Modeling Methodology. We use our empirical characterization to construct a small number of load-specific model types. We show that our basic models, or a composition of them, capture nearly all household loads. Our models go beyond on-off models, by capturing power usage characteristics that i) decay or grow over time, ii) have frequent variations (as with the TV in Fig. 1), iii) exhibit complex repetitive patterns of simpler internal loads, and iv) are composites of two or more simpler loads. We show that our models are significantly more accurate than on-off models, decreasing the root mean square error by as much as $8\times$ for representative loads. Since our methodology is general, it applicable to modeling other types of loads beyond those in this paper.

Model-based Data Analysis. Finally, while we expect our models to have numerous uses, we illustrate three specific examples of novel applications using these models. First, we consider generating device-accurate synthetic traces of building electricity usage for use by NILM researchers. One barrier to evaluating NILM techniques is ground-truth data collection, which requires deploying sensors to every building load. Using our models, NILM researchers can quickly generate different types of synthetic building traces by composing collections of load models, and the resulting synthetic trace is more representative of real-world data (e.g., in terms of number and sizes

of power steps) than when using trivial models. Second, we design simple filters capable of identifying and removing loads that exhibit rapid power variations (such as an LCD TV). Since these loads introduce numerous spurious building power states, removing them improves the accuracy of analysis techniques that describe buildings as state machines (e.g., by 25% in our experiments). Finally, we use our models as queries to a variable-length subsequence matching algorithm to detect the presence of specific loads in smart meter data.

II. Empirical Data Collection and Characterization

A typical home consists of dozens of electrical loads, including heating and cooling equipment, lights, various appliances and electronic equipment. A partial list includes:

- *Heating, cooling, and climate control equipment* such as a central air conditioner, window air conditioner, space heater, electric water heater, dehumidifier, fan, air purifier;
- *Kitchen appliances* such as an electric range, microwave, refrigerator, coffee maker, toaster, blender, dishwasher;
- *Laundry appliances* such as a washer and dryer;
- *Lighting* including incandescent and fluorescent lights;
- *Miscellaneous electronic devices* such as a television, battery charger, computer, and gaming console; and
- *Other appliances* such as a vacuum and carpet cleaner.

Below, we briefly describe the data collection infrastructure we use to gather data from these common household loads. We then derive several insights from our data, which we use to design different types of load models in the next section.

A. Data Collection Infrastructure

Since our methodology is empirical, we instrument three homes with a large number of energy monitoring sensors to gather ground truth electricity usage data from a wide variety of loads. Each instrumented home consists of a smart home gateway in the form of an embedded Linux server that queries each sensor to collect data. We have deployed several different types of energy monitoring sensors, as described below.

We use current transducer (CT) sensors to monitor electricity usage for large loads wired to dedicated circuits, such as air conditioners, washing machines, dryers, dishwashers, and refrigerators. These sensors connect to power meters installed in the homes' electrical panel, such as The Energy Detective (TED) [16] or eGauge [15], which sample per-circuit electricity usage each second. The sensors transmit data from inside the electrical panel using powerline networking protocols, such as X10, Insteon, and HomePlug Ethernet-over-Powerline. We use the eGauge in our testbeds due to its support for HomePlug, which provides multiple orders of magnitude higher bandwidth ($>$ 100 Mbit/s) than X10 and supports reliable transport protocols, including TCP. We use plug-level energy sensors to track energy use for smaller loads plugged into wall outlets. Our plug-level sensors are commodity Insteon iMeters [20] and Z-Wave Smart Energy Switch meters [21], which use powerline and wireless communication, respectively, to transmit readings to our smart home gateway.

The iMeter plug meters do not support the higher bandwidth HomePlug powerline protocol, which severely limits the frequency at which our gateway is able to poll outlet power usage. To prevent saturating the powerline (at the Insteon transmit frequency), our initial prototype polled each iMeter-connected outlet's power usage once every 10 seconds. Since our testbed homes employ more than 30 iMeters, a naïve round-robin polling strategy results in a power reading once every 300 seconds [22]. As a result, we designed a smart polling strategy [22] that monitors the per-second eGauge circuit data and only polls outlets on a selected circuit if the eGauge circuit registers a change in power. Thus, we now only poll the iMeters to see which outlet's power changed. We also use Insteon-enabled wall switches to monitor switched loads wired directly into the electrical system. These switches replace normal wall switches, and transmit on-off-dim events to the gateway whenever a user manually toggles the switch.

The Z-Wave meters enable more rapid polling of power data every second from each outlet, although we have experienced numerous issues in our testbed with wireless range and coverage. For example, many sensors must be placed behind large metal appliances that severely attenuates wireless signals. Further, even though our testbed homes are not large (roughly 1700 ft^2), ensuring that each sensor is within range of the gateway is challenging, and requires careful gateway placement. Larger homes would likely require multiple wireless receivers or mesh networking protocols, both of which significantly increase the complexity and cost of the deployment.

While there have been numerous challenges in deploying our measurement testbeds, as described above, they have now been continually monitoring hundreds of individual loads for nearly two years in each of our three instrumented homes. Since the level of instrumentation in our measurement testbed is time-consuming and expensive to replicate, we have made much of our collected data available to benefit other researchers [7]. More details about our testbed deployment and multi-year data collection efforts are available in recent work [7]. We leverage our data to characterize various loads based on a few elemental types, described below. In addition, the challenges in deploying and operating our measurement testbed serve as important motivation for the modeling and analysis techniques we describe in Sections III and IV. These techniques enable users to analyze smart meter data to infer information about individual loads without requiring a power meter attached to each load. One attractive scenario is for all manufacturers to release load models once at design time, thereby enabling anyone to employ the type of model-based analysis techniques we propose in Section IV.

B. Characterizing Different Types of Loads

Despite their tremendous variety, most residential loads fall into one of a few elemental load types based on how they consume power in an alternating current (AC) system. In particular, loads are categorized as either resistive, inductive, capacitive, or non-linear based on how they draw current in relation to voltage, which in an AC system varies along a smooth sinusoidal pattern. These categories reveal properties of the loads

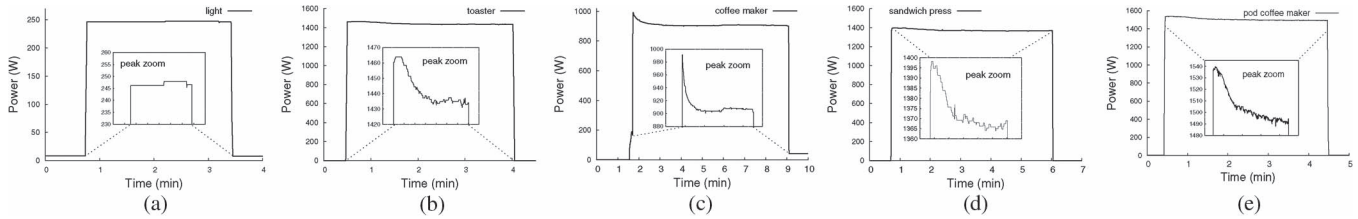

Fig. 2. Example resistive loads, demonstrating "step" behavior with a possible initial surge and slow decay to a stable power level. (a) Light bulb. (b) Toaster. (c) Coffee maker. (d) Sandwich press. (e) Pod Coffee.

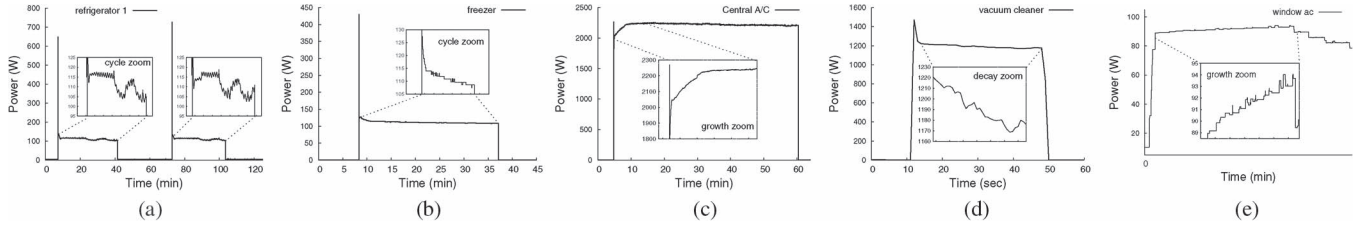

Fig. 3. Example inductive loads, demonstrating significant surge current followed by steady power growth or decay. (a) Refrigerator. (b) Freezer. (c) Central A/C. (d) Vacuum cleaner. (e) Window A/C.

that we leverage in our models. Since many researchers outside of power systems may be unfamiliar with these load types, for each type of load we first review its salient characteristics. We then empirically characterize data from multiple representative loads of each type to observe how their specific characteristics manifest themselves in the data.

Resistive Loads. Loads that consist of any type of heating element are resistive. Incandescent lights, toasters, ovens, space heaters, coffee makers, etc., are examples of common resistive loads in a home. Formally, if a load draws current along a sinusoidal pattern in the same phase as the voltage, i.e., the maximum, minimum, and zero points of the voltage and current sine waves align, then the load is purely *resistive*.

Fig. 2 depicts a time-series of the power usage for five different resistive loads with heating elements: an incandescent light bulb, a toaster oven, a coffee maker, a sandwich press, and a pod coffee maker, e.g., a Keurig or Tassimo. In general, the power usage of these loads resembles a "step" when turned on, with usage that remains relatively stable and flat. The incandescent light acts as a nearly perfect resistive load with a power usage equal to the bulb's wattage. While the toaster oven, coffee maker, sandwich press, and pod coffee maker act similar to the light bulb, they experience an initial higher power usage that slowly decays to a relatively stable usage, highlighted in Fig. 2. The initial high power is due to the large inrush (or surge) current that occurs as the device warms up and the resistance decreases, after which it stabilizes.

Observation 1: Resistive loads exhibit stable power usage when turned on, with high-power heating elements exhibiting an initial surge followed by a slow decay to stable power.

Inductive Loads. AC motors are the most common and widely-used examples of inductive loads. Motors are the primary component of many household devices, including fans, vacuum cleaners, dishwashers, washing machines, and compressors in refrigerators and air conditioners. Formally, if a load draws current along a sinusoidal pattern that peaks *after* the voltage sine wave, i.e., the current waveform lags the voltage waveform, then the load is purely *inductive*.

Fig. 3 depicts a time-series of the power usage for five inductive loads: a refrigerator, a freezer, a central air conditioner (A/C), a vacuum cleaner, and a window A/C unit. All five loads operate AC motors. Unlike the resistive loads above, each inductive load experiences a significant, but brief, initial power usage. The surge is also due to inrush current that occurs when starting an AC motor, although it is typically much higher than for heating elements. Intuitively, the underlying reason is that, while heating elements heat up slowly, the rotor inside a motor must transition from completely idle to full speed within seconds. Power usage then exhibits either a decay or growth, depending on the motor's operation, that eventually stabilizes. In contrast to resistive loads, motors exhibit small variations even during this "stable" phase. For instance, the refrigerator shown in Fig. 3(a) exhibits small fluctuations that repeat during each cycle of the compressor. The freezer, central A/C, vacuum cleaner, and window A/C depicted in Figs. 3(b), (c), (d), and (e) all show an initial spike followed by a sharper, smoother growth (central and window A/C) or decay (freezer and vacuum), with small variations as the usage stabilizes. These patterns demonstrate that, unlike resistive loads, modeling inductive loads using simple on-off step functions is problematic.

Observation 2: Inductive loads with AC motors exhibit an initial power spike followed by a growth or decay to a stable power level. The growth/decay rate is load-dependent, with the stable power level also exhibiting fine-grained variations.

Capacitive Loads. Capacitive loads are the dual of inductive loads. Formally, if a load draws current along a sinusoidal pattern that peaks *before* the voltage sine wave, i.e., the current waveform leads the voltage waveform, then the load is purely *capacitive*. While many loads have capacitive elements, they generally occur in addition to other resistive and inductive elements which dominate their overall behavior. Thus, there are no significant capacitive loads in buildings, particularly when considering real (as opposed to reactive) power.

Non-linear Loads. Finally, any load that does not draw current along a sinusoidal pattern is called *non-linear*. Non-linear loads may also be resistive, inductive, or capacitive based

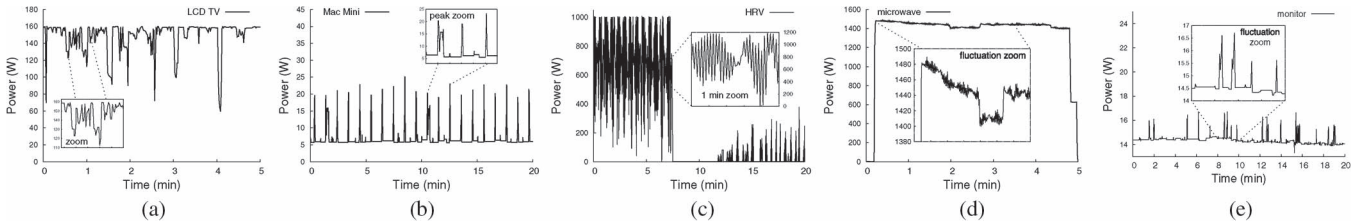

Fig. 4. Example non-linear loads, demonstrating rapid and significant random variations with possible ceilings and/or floors. (a) LCD TV. (b) Desktop PC. (c) Duct Heater. (d) Microwave. (e) Monitor.

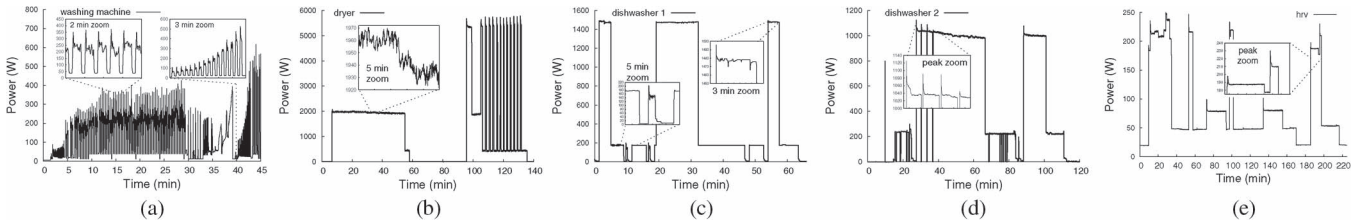

Fig. 5. Example composite loads, demonstrating combinations of the simpler loads above arranged in phases. (a) Washing machine. (b) Dryer. (c) Dishwasher 1. (d) Dishwasher 2. (e) HRV.

on when their current waveform peaks. The most predominant non-linear (and largely inductive) loads are electronic devices, including computers and TVs. The non-linear nature of these loads is primarily due to the use of switched-mode power supplies (SMPSs). Fluorescent lights are another example of a non-linear (inductive) load. Smaller electronic devices that convert AC to low-voltage DC, such as battery chargers for portable devices and digital clocks, are also non-linear.

Fig. 4 shows the power usage of five different non-linear loads: an LCD TV, a Mac Mini desktop computer, a microwave oven, a duct heater for a heat recovery ventilator (HRV), and a computer monitor. These loads exhibit significant power fluctuations when active, but also have a stable floor or ceiling from which these fluctuations derive. The LCD TV shown in Fig. 4(a) exhibits a stable maximum usage with random power reductions from this ceiling. These fluctuations result from displaying a variety of color and pixel intensities on the screen. Not surprisingly, the computer monitor in Fig. 4(e) has a similar pattern of power usage. In contrast, the desktop computer shown in Fig. 4(b) has a stable minimum power draw, with random power spikes above this floor depending on its workload, e.g., causing the CPU to ramp up, etc. Both the TV and desktop computer consist of a switched mode power supply (SMPS) that regulate the power usage of the device and switch between a full-on and full-off state to minimize wasted energy. The duct heater shown in Fig. 4(c) demonstrates two regular modes of operation; an active heating mode—with instantaneous intensity managed by the HRV controller—and a passive mode. In both modes, there are large, random variations in power usage. In the active state, there is also a clear stable maximum usage. Finally, the microwave shown in Fig. 4(d) has what initially appears to be a straightforward step, similar to the resistive loads. However, zooming in shows the microwave's non-linear behavior, with rapid, albeit small, variations in the second-to-second usage, along with larger periodic power shifts. These examples show that on-off models are inappropriate for non-linear loads, since two power states cannot capture their wide range of power variations.

Observation 3: Non-linear loads exhibit significant random variations in power usage. These fluctuations are often range-bound and capped by a floor or ceiling in the power level.

C. Composite Loads and Reactive Power

Composite Loads. Many household loads, particularly large appliances, are not purely resistive, inductive or non-linear. Instead, these loads consist of multiple components, each of which may be one of the simpler load types. For instance, a central air conditioner may consist of a compressor, a fan to blow air into ducts, duct dampeners to control air flow, and central humidifiers to control humidity. A refrigerator, which has a compressor that is an inductive load, may also consist of door lights, an ice maker, and a water dispenser. Similarly, electric dryers, washing machines, and dishwashers also consist of a motor—to spin clothes and circulate water via a pump—and a heating element—to dry clothes or warm water. In addition, these appliances often operate in repetitive cycles that activate each of their constituent loads differently, such as washing, draining, and then drying for a dishwasher. Fig. 5 depicts the power usage of a washing machine, a dryer, two dishwashers, and the HRV. As shown, these loads exhibit distinct behavior in different parts of their cycle depending on which appliance component is in use. For example, based on the observations above, distinguishing when a complex load, such as the dryer, activates its heating element versus its motor is straightforward. Finally, an appliance may activate its various components in sequence, in parallel, or both. For instance, a central air conditioner may operate the compressor, the fan and the dampeners concurrently, while a dishwasher may operate its motor, pump and heater in sequence.

Observation 4: Composite loads consist of simpler resistive, inductive and non-linear loads that operate in parallel, in sequence, or both. As a result, composite loads exhibit distinct behaviors in different operating regions of their active cycle.

Reactive Power. Finally, another important characteristic of the elemental load types above is how they consume *reactive*

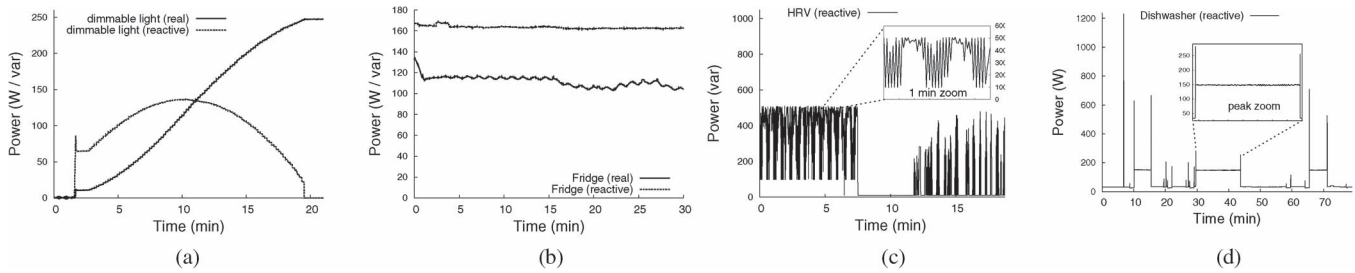

Fig. 6. Reactive power demonstrates the same types of patterns as real power and can help in identifying different types of electrical loads. (a) Dimmable light (0% to 100%). (b) Refrigerator. (c) Duct Heater. (d) Dishwasher 1.

power. While real power is the amount of power delivered to a load, and is often referred to as simply electricity or power (without the qualifier), reactive power is the amount of power generated, but not delivered, to the load; it is also measured in units of watts, but written as voltage-amperes reactive (VAR) to distinguish it from real power. Reactive power arises when a load draws current *out of phase* with the voltage. Thus, only non-resistive loads generate reactive power. At a high level, reactive power is the result of the instantaneous power (the product of current and voltage) occasionally becoming negative within each AC cycle, due to out-of-phase current and voltage. This state causes power to flow towards the generator and away from the load. Reactive power is typically dissipated as heat in power lines. For our purposes, reactive power provides additional useful information for modeling, and many commodity power meters are capable of measuring it. As a result, our models include both real and reactive power.

Fig. 6 depicts companion graphs for selected loads that shows their reactive, rather than real, power usage. Fig. 6(a) shows that, similar to real power, a resistive dimmable incandescent light produces a stable—zero if not dimmed—amount of reactive power when on, although the magnitude of the draw peaks at 50% dim level and decreases as the light approaches either 0% or 100% dim level. Likewise, an inductive load like the refrigerator in Fig. 6(b) exhibits a spike followed by a flat reactive draw; a non-linear load like the duct heater in Fig. 6(c) has a rapidly varying power usage; and composite load like dishwasher 1 in Fig. 6(d) operates a sequence of simpler internal loads. In each case, the pattern of a load's reactive power usage follows its pattern of real power usage.

Observation 5: While the magnitude of reactive power differs from real power, a load's pattern of reactive power consumption is qualitatively similar to its real power consumption.

D. Summary

In classifying loads in terms of the elemental load types above, we observe that *nearly every common household electric load is a composition of one or more of the small number of resistive, inductive, and non-linear loads described above*, with heating elements and AC motors consuming the majority of electricity in homes. Further, each type of elemental load exhibits similar characteristics when active: heating elements have a stable power usage or one that decays slowly over time, AC motors have a spike in power on startup and then vary their power usage smoothly over time, while SMPSs exhibit rapid and significant power variations. As we discuss in the next section, the presence of only a few elemental load types in homes simplifies model design, enabling us to accurately capture their behavior using a few basic types of models.

III. Load Modeling Methodology

Based on our empirical observations from the previous section, we develop models to capture key characteristics of each load type. We first present four basic model types—*on-off*, *on-off growth/decay*, *stable min-max*, and *random range*— to describe simple loads, and then use these models as building blocks to form compound *cyclic* and *composite* models that describe more complex loads. Ideal models describe i) *how much* real and reactive power a load uses when active, ii) *how long* a load is active, and iii) *when* a load is active. However, in many cases, users manually control loads, such that *when* a load is active and for *how long* is non-deterministic. For example, a user may run a microwave any time for either ten seconds or ten minutes. For these loads, we assume a random variable captures this non-determinism, and focus our efforts, instead, on modeling *how* each load behaves when active.

Given each model type, we employ an empirical methodology to construct accurate load-specific models: we leverage our observations of the load's power usage as a training set, and employ curve-fitting methods to map one of the model types onto the time-series data. If the best model type is not clearly evident *a priori*, we fit multiple models and then choose the one that yields the best fit. As described below, depending on the model type, we may employ simple regression or more complex curve-fitting methods, such as LMA [23], to construct a load-specific model for a given model type. As discussed in Section II, reactive power for loads exhibits similar behavior as real power, and thus constructing a model of a load's reactive power consumption uses the same methodology as above.

A. Basic Model Types

On-off Model. As discussed in Section I, prior work often uses simple on-off load models. An on-off model includes two states—an *on* state that draws some fixed power p_{active} and an *off* state that draws zero, or some minimal amount of, power p_{off}. Conventional, non-dimmable incandescent lights are the canonical example of an on-off load. Dimmable lights also

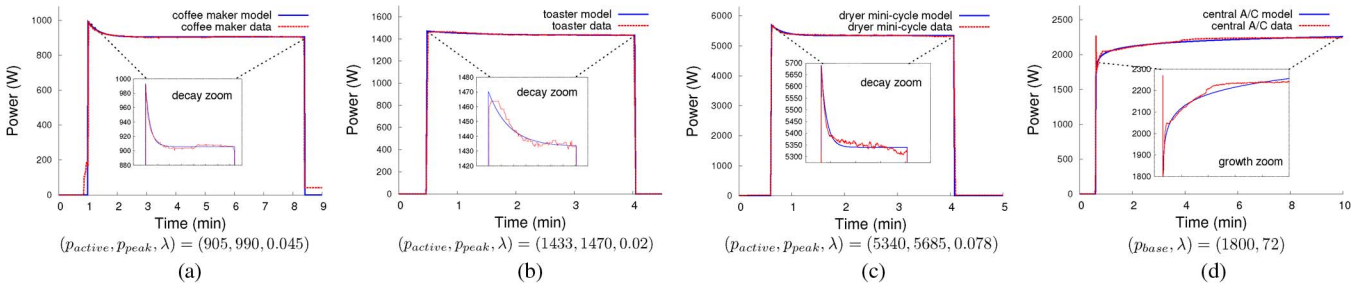

Fig. 7. An on-off growth/decay model closely matches the average power usage measured each second for a variety of resistive and inductive loads. (a) Coffee maker. (b) Toaster. (c) Single dryer mini-cycle. (d) Central A/C.

conform to on-off models, although p_{active} depends on the dim level. As shown in Fig. 6(a), a $N\%$ dim level yields a proportionate reduction in real power usage. In addition, while real power is a simple linear function of dim level, reactive power is a quadratic function that peaks at 50% dim level. Constructing an on-off model is simple—we use regression to determine appropriate values p_{active} and p_{off}. In particular, we partition the time series of load power usage into two mutually exclusive time-series, with data for the on and off periods, to determine the best values of p_{active} and p_{off}.

On-off Growth/Decay Model. An on-off growth/decay model is a variant of the on-off model that accounts for an initial power surge when a load starts, followed by a smooth increase or decrease in power usage over time. As discussed in Section II, AC motors are the most common example of a load exhibiting this behavior, e.g., refrigerator, central A/C, vacuum. Resistive loads with high-power heating elements, such as the toaster or coffee maker, also conform to an on-off growth/decay model, although the surge and the decay in these devices is far less prominent than in AC motors. We characterize on-off decay models using four parameters: p_{active}, p_{off}, p_{peak}, and λ. The first two parameters are the same as in on-off models, while p_{peak} represents the level of inrush current when a device starts up and λ represents the rate of growth or decay to the stable p_{active} power level. We model decay using an exponential function as follows, where t_{active} is the length of the active interval.

$$p(t) = \begin{cases} p_{active} + (p_{peak} - p_{active})e^{-\lambda t}, & 0 \le t < t_{active} \\ p_{off}, & t \ge t_{active}. \end{cases}$$

Similarly, we model on-off growth as a logarithmic function (i.e., the inverse of the exponential function) using starting power level p_{base} and growth parameter λ:

$$p(t) = \begin{cases} p_{base} + \lambda \ln t, & 0 < t < t_{active} \\ p_{off}, & t \ge t_{active}. \end{cases}$$

We can optionally augment the growth model with an additional parameter p_{ceil} to prevent unbounded growth that simply caps the maximum output of the model. In the growth model, the surge current must also be modeled separately (such as in the central A/C shown in Fig. 7(d)). Here, we can simply add a parameter p_{spike} specifying the power at $t = 0$.

As with the on-off models above, the length of the active interval for on-off growth/decay models is often not known *a priori* since it may depend on user behavior. However, we have observed that in many cases users repeatedly operate devices in the same way, e.g., a toaster that toasts a bagel every morning. In many cases, the device determines t_{active} automatically, e.g., the compressor for a refrigerator or freezer may turn on for an average of 20 minutes in each cycle. In these cases, we incorporate the mean value of t_{active} into the model. Constructing an on-off growth/decay model requires fitting an exponentially decaying (or logarithmically growing) function onto the time-series data, in addition to determining p_{peak}, p_{active}, and p_{off}. We employ the LMA algorithm [23] to numerically find the exponential or logarithmic function that best fits the data, i.e., based on a least-squares nonlinear fit.

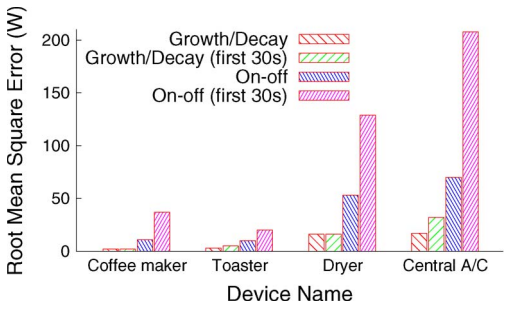

Fig. 8. On-off growth/decay models are more accurate than on-off models.

Fig. 7(a) shows the specific on-off decay model for a coffee maker in parallel with its real power data. The figure demonstrates that the exponential decay is a highly accurate approximation of the coffee maker's power usage. In this case, $p_{active} = 905$, $p_{peak} = 990$, $p_{off} = 0$, and $\lambda = 0.045$. Likewise, Fig. 7(b) and (c) show on-off decay models and real power data for a toaster and a portion of a dryer cycle. Finally, Fig. 7(d) shows how well an on-off growth model fits the real power data for the central A/C from Fig. 3(c). For comparison, we also fitted the lowest-error on-off model for each of the four representative device cycles pictured in Fig. 7. We then calculated the root mean square error (RMSE) for both the on-off and on-off decay models for (a) each load's duration, and (b) its first 30 seconds of activity (including the "on" event). As seen in Fig. 8, the growth/decay model decreases the error in the on-off model by as much as 8×, particularly in the first 30 seconds where the on-off model is unable to capture the rapid decay behavior.

Stable Min-Max Model. While on-off and on-off decay models accurately capture the behavior of resistive and

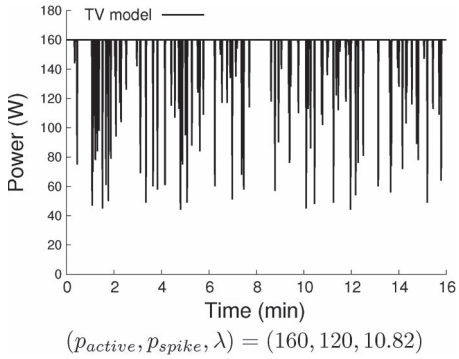

Fig. 9. A stable-max model of the LCD TV from Fig. 1.

inductive loads, they are inadequate for modeling non-linear loads. As seen in Section II, many non-linear loads maintain a stable maximum or minimum power draw when active, but often vary randomly and frequently from this stable state. These variations are due to the device rapidly regulating their electricity usage to "match" the current needs of the device. Our stable min-max model captures this behavior by first specifying a stable maximum or minimum power when active, denoted by p_{active}. The power usage then deviates, or "spikes," up or down from this stable value at some frequency. The magnitude of each spike is chosen uniformly at random between p_{active} and a specified maximum deviation, denoted p_{spike}. The inter-arrival times of the spikes are exponentially distributed with mean λ. Thus, the stable min-max model is specified by the choice of p_{active}, p_{spike}, and λ (as well as whether p_{active} denotes a stable minimum or a stable maximum).

Empirically constructing a load-specific stable min-max model requires determining the stable power level p_{active} and characterizing the magnitude and frequency of the power spikes. We employ simple regression to determine the stable power level p_{active} from the data, e.g., after filtering out the data for spikes and finding the fit for p_{active}. The mean observed duration between spikes then yields the parameter λ. Fig. 9 shows our stable-max model for the LCD TV (from Fig. 1) using a maximum p_{active} of 160 W and a λ of 10.82, which we derive from the TV's real power usage data. Importantly, as we discuss in Section IV, both the model and the raw data have similar statistical properties, which simple filters can recognize by detecting when power variations are significant, frequent, and symmetric, e.g., a decrease and then immediate increase in power of similar magnitude.

Random Range. Finally, we found that some devices draw a seemingly random amount of power within a fixed range when active. This is likely due to the fact that taking average power readings each second is too coarse a frequency to capture the device's repetitive behavior. We model such loads by determining upper and lower power usage bounds, denoted by $p_{\max}$ and $p_{\min}$. When active, our model randomly varies power within these bounds using a random walk. Note that the random range model is similar to the stable min-max model in that both employ upper and lower bounds on power usage. However, while the deviations in the stable min-max model are spikes from a stable value, those in the random range model are power variations within a range. The microwave is an example of a load that exhibits this behavior. As shown in Fig. 4(d), when turned on, the power usage of the microwave fluctuates continuously between 1400 W and 1480 W.

Random range models require determining the minimum and maximum of the load's range of power usage. We determine these values by simply choosing the minimum and maximum power values observed in training data, or by deriving a distribution of power values from the data and choosing a high and low percentile of the distribution to be the minimum and maximum, $p_{\min}$ and $p_{\max}$. We then model the variations with a random walk within the range.

B. Compound Model Types

While the models above accurately capture the behavior of simple loads, many loads, including large appliances, exhibit complex behavior from operating a variety of smaller constituent loads. We devise two types of compound models for complex loads that use the basic building blocks above.

Cyclic Model. Cyclic loads repeat one of the basic model types in a regular pattern, often driven by timers or sensors. For example, the HRV heater employs a timer that activates for 20 minutes each hour. Similarly, a refrigerator duty-cycle is based on sensing its internal temperature, which rises and falls at regular intervals and fits our model well, as shown in Fig. 3(a) and (b). A cyclic model augments a basic model by specifying the length of the active and inactive period, t_{active} and $t_{inactive}$, each cycle. Constructing cyclic models is straightforward, since it only requires extracting the duration of the active and inactive periods from the empirical data. We currently use the mean of the active and inactive periods from the time-series observations to model t_{active} and $t_{inactive}$.

Composite Model. Composite loads exhibit characteristics of multiple basic model types either in sequence or parallel. Example composite loads include dryers, washing machines, and dishwashers, as shown in Fig. 5. *Sequential composite loads* operate a set of basic load types in sequence; we model them as simple piecewise functions that encode the sequence of basic load models, including how long each load operates. For instance, a model for a dishwasher is a sequence of stages: modeled as the operation of the motor (wash stage), pump (drain stage), motor (rinse stage), pump (drain stage) and heater (dry stage), where each individual stage uses an inductive or resistive load. Some loads also exhibit characteristics of two or more basic models in *parallel* if two basic loads operate simultaneously. For example, a refrigerator may simultaneously activate both a compressor and an interior light. We model *parallel composite loads* by summing the power usage for two or more of the basic model types. Finally, composite loads may also be cyclic, referred to as *cyclic composite loads*, which repeat a pattern of individual model types at regular intervals. Our methodology permits arbitrary compositions of sequential, parallel or cyclic loads.

Constructing load-specific composite models is more complex and requires additional manual inputs. For example, constructing a *sequential composite* model requires manually partitioning and isolating load time-series data into individual sequences that reflect the activation of the various load

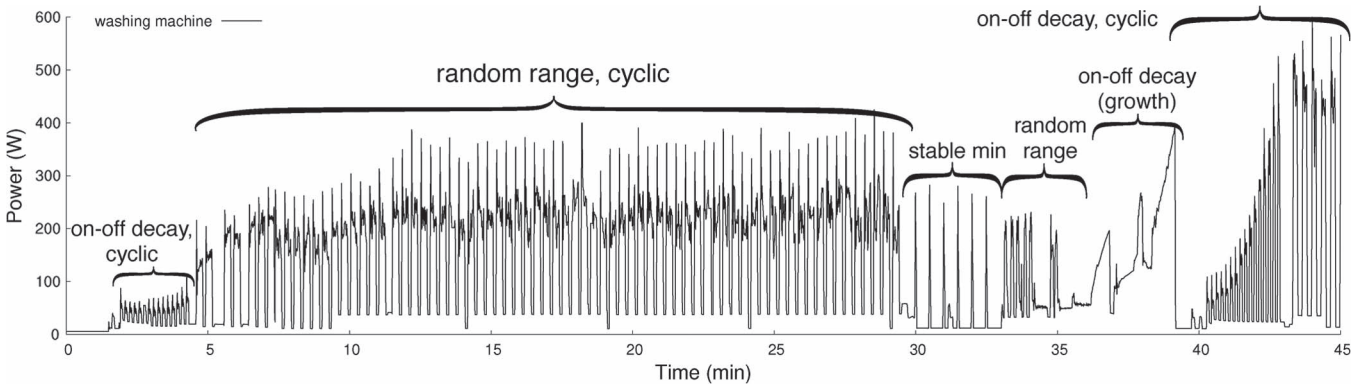

Fig. 10. A single complete cycle of a washing machinenannotated with the model types for the operation of simpler internal loads.

components. Each individual component of the composite load is modeled using a basic model type. The composite model is then simply a concatenation of these piecewise models in sequence (the duration of each component may be specified in the model or left as a variable).

As an example, Fig. 10 shows an extended operating cycle of the washing machine with the annotations for different basic load model types in the sequence. We represent these models as large piecewise functions of the basic models describing each constituent load. In addition, many of the large appliances that have composite models also have numerous operating states. For example, the washing machine and dryer in one of our homes has over 25 different types of cycles. Ideally, a model includes a different piecewise function for each cycle type. However, in the homes we monitor we have found that most residents operate devices using only a few states—in most cases one.

Constructing parallel composite models poses additional challenges. Since the time-series data for a load captures the power usage for all components that are concurrently active, there is no straightforward general-purpose technique to extract individual models from the composite time-series data. In practice, however, extracting basic models is often possible through exogenous means. For instance, many loads permit operating individual components to isolate them for profiling, e.g., such as a running a dryer on tumble mode without any heat or using an air conditioner's fan without any cooling. After separately profiling a constituent load, such as the tumbler or fan, it is possible to operate the compressor and the fan, and then infer the compressor power usage by "filtering out" the tumbler or fan usage from the aggregate. In some devices, such as a refrigerator, it also might be possible to deploy additional sensors that monitor important events, such as a door opening that triggers lights, to filter them out. Ideally, the model of a complex composite device would be provided by the device manufacturer, as the problem of identifying the *components* of a composite device is largely orthogonal to the problem of *modeling* each component. However, using the techniques described previously, it is generally possible to identify the key components even without detailed knowledge of the internals of the device (though for many devices, substantial information on the components and operation of the device is readily available, e.g., in an owner's manual).

IV. Model-Based Data Analysis

While we expect our models to have numerous uses in a variety of analysis and optimization tasks in building energy management, below, we illustrate three examples of novel applications we have designed: i) generating device-accurate synthetic data of a building's aggregate electricity usage, ii) designing filters to identify and remove random and frequent power variations from stable min-max loads, and iii) detecting the presence of specific load models in filtered power data using an existing variable-length subsequence matching algorithm for time-series data. Each of these applications focuses on improving the analysis of smart meter data that records a residential home's average electricity usage each second. We assume the smart meter not only transmits data to the utility for billing purposes, but also provides home users an interface to access the data. In this architecture, similar to our measurement testbeds, home users may employ a programmable gateway (either on-site or cloud-based) under their control to record and analyze the data to inform either manual or automated energy-efficiency improvements, e.g., via programmatic load control and scheduling. As we show, our applications should prove useful in both i) developing new techniques for analyzing smart meter data and ii) supplementing existing well-studied techniques, e.g., NILM, already developed by researchers and in broad use.

A. Device-Accurate Synthetic Building Data

Evaluating new techniques for analyzing electricity data requires actual building data for testing. Unfortunately, while recording a building's aggregate electricity usage is simple, requiring only a single smart meter, recording detailed aspects of the building's environment is not. For instance, evaluating the accuracy of a NILM algorithm, which disaggregates building electricity data into power data for individual loads, requires power data from *both* the entire building and each of its constituent loads. However, NILM's entire purpose is to prevent the need for recording such ground truth data at each load. As our own experience from Section II-A indicates, setting up even a test infrastructure for collecting ground truth data is expensive, invasive, and time-consuming, since it requires a power meter attached to each load in the home. While there are a few data sets for select buildings available for NILM researchers to use

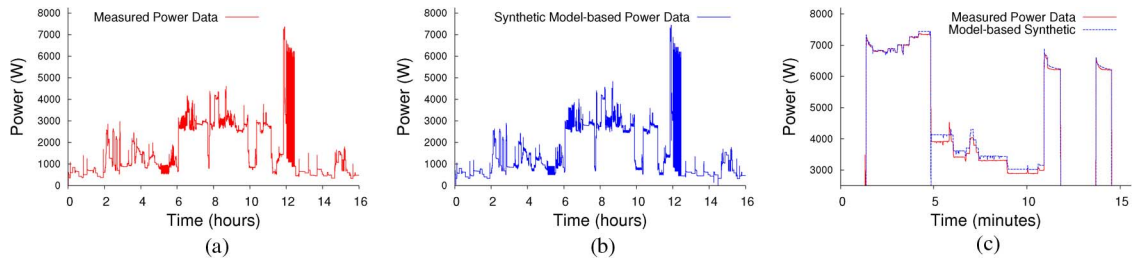

Fig. 11. Aggregate home power from measured data (a) and from our corresponding device models (b), along with a zoom-in comparison over 15 minutes. (a) Raw power data. (b) Synthetic model-based power data. (c) Zoom-in comparison.

in evaluation [7], [24], [25], they typically do not instrument *every* load nor do they cover a wide range of building types or load characteristics.

To address this problem, our first application uses our models to automatically generate device-accurate synthetic electricity data for buildings. Being device-accurate means that the synthetic trace data includes *both the synthetic aggregate time-series power data for a building, as well as time-series power data for each of the constituent loads* in the building generated using our models. While prior work targets generating synthetic traces of the power usage for entire buildings [26], we are not aware of any previous work that focuses on being device-accurate. Unlike real-world trace data collected from specific buildings, the synthetic traces generated using our models will provide researchers explicit control over the number and types of loads present in the data, enabling them to control the statistical properties of the dataset and discover which properties have the most influence on their results. Importantly, synthetic data does not not require researchers to deploy a large number of per-load power meters.

Fig. 11 shows an example of how our device-accurate synthetic building data compares with data collected from a real building. To generate the synthetic trace shown in Fig. 11(b), we replace each occurrence of a given device in the ground-truth data shown in Fig. 11(a) (i.e., each period when a device is using power) with our model of that device over the same time period. Fig. 11(c) shows a zoomed-in comparison of the two traces over a 15 minute period to graphically illustrate their similarity. The ground-truth and model-based traces look qualitatively similar, but they also have similar statistical properties: the real data (a) has an average power of 1200 W, a standard deviation of 1072 W, and 5591 changes in power $>$ 15 W, while the synthetic data (b) has an average power of 1165 W, a standard deviation of 1073 W, and 5833 changes in power $>$ 15 W. Since the synthetic data is composed of data from models of individual loads, it is useful for analysis techniques that look for patterns in the aggregate usage data. By comparison, if we generate on-off models that include at most 4 power states per load (as in recent work [24]), there are only 1985 changes in power $>$ 15 W, which eliminates many identifiable load-specific characteristics useful in analysis.

In addition to comparing our model-based trace data with the real building data, we also show that a state-of-the-art NILM disaggregation algorithm produces similar results using both datasets. Formally, a NILM algorithm analyzes changes in smart meter time-series power data, $P(t)$, to compute a separate power time-series $p_i(t)$ for each $i = 1 \ldots n$ loads in a home. NILM is a well-studied problem first proposed over 20 years ago [27]. We use the same NILM algorithm as Kolter and Johnson [24] to evaluate their Reference Energy Disaggregation Dataset (REDD), which was based on the technique by Kim *et al.* [28] and models building power data as a Factorial Hidden Markov Model (FHMM). We quantify the algorithm's accuracy using the *dissaggregation error factor*. For n loads with predicted power $\tilde{p}_t^i$ and actual power p_t^i at time t for all times T, the disaggregation error factor is computed as follows:

$$Error\ Factor = \frac{\sum_{t=1}^{T}\sum_{i=1}^{n}\left|p_t^i - \tilde{p}_t^i\right|}{\sum_{t=1}^{T}\sum_{i=1}^{n} p_t^i} \quad (1)$$

The numerator is the sum of the absolute value of the difference between the predicted and actual power for each load for all time periods T, while the denominator is two times the aggregate energy. An error factor of zero indicates precise second-to-second disaggregation, while an error factor of one indicates that the second-to-second errors equal the energy use of the home. Thus, simply predicting power for all loads to be zero every second results in an error factor of one. Note that, since we monitor average power each second, the numerator and the denominator are in units of watt-seconds or joules. Based on the equation above, we can also compute a disaggregation error factor for each individual load, called the *load error factor*. In this case, the numerator and denominator do not sum over all n loads, but only over the single load.

The FHMM-based NILM algorithm requires a training phase that uses data from each load to learn a power usage model for each load. Unlike our empirical models, the per-load models used by the FHMM include only 4 power states, although we do not explore how our models might apply to NILM in this paper. We train the FHMM on the same per-load data to generate the 4-state models, and then run the NILM algorithm on both the real building data and our synthetically generated trace data. Our results show that the NILM algorithm has a disaggregation error factor of 0.796 when using the real building data and 0.735 when using our synthetically generated data—a difference of only 7.5%. This similarity in error factors demonstrates that our synthetic data produces similar results for NILM algorithms as real data. In addition, Fig. 12 shows both the disaggregated refrigerator and actual refrigerator trace using the real building data (a) and the disaggregated refrigerator and model-derived refrigerator trace using our synthetic trace data (b). The figure graphically illustrates the similarity in results from the NILM algorithm operating on the real versus the synthetic data, and demonstrates that our synthetic traces are device-accurate.

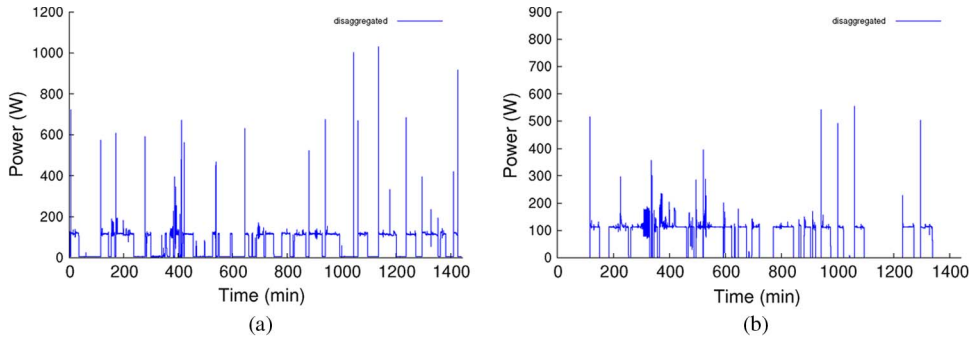

Fig. 12. Disaggregation of the refrigerator's power usage when performing NILM on both real power data and synthetic data. (a) NILM on actual data. (b) NILM on synthetic data.

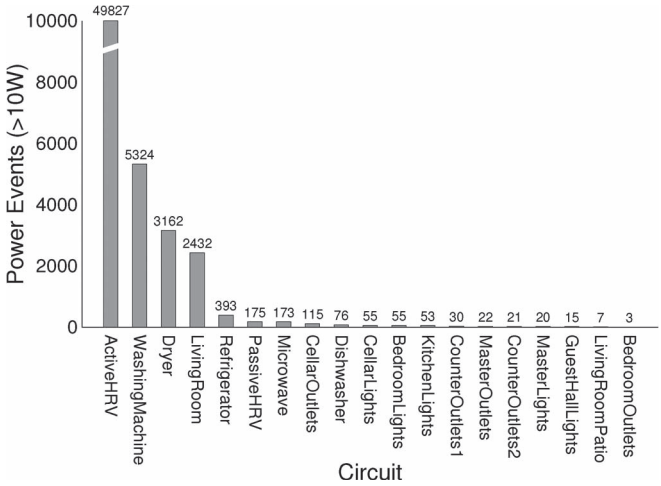

Fig. 13. A few highly variable (non-linear) loads are responsible for the vast majority of power variations in a home's per-second smart meter data.

Result: Our load models are useful in generating device-accurate synthetic datasets that enable researchers to test and validate new analysis techniques for smart meter data.

B. Stable Min-Max Filters

Since stable min-max loads account for the large majority of power variations in a home, filtering out their variations can significantly improve subsequent data analysis techniques, including the HMM-based NILM algorithms above that models a building as a state machine composed of on-off loads. To illustrate this point, Fig. 13 shows a bar graph of the number of second-to-second power variations in a single day for each of 19 active circuits in one of the homes we monitor. The graph demonstrates that a small number of circuits account for the vast majority of the power variations in the home—notably resulting from two non-linear loads (the HRV's duct heater and the living room TV) and two composite loads with non-linear components (the washing machine and dryer), each of which is highlighted in Fig. 4. In total, there were 61958 variations in power (> 1 W) out of a total of 86400 data points over the 24-hour period, out of which 49827 (or 80.4%) were caused by the HRV's non-linear duct heater from Fig. 4(c). The washing machine and dryer, combined, caused another 8486 variations (or 13.7%), followed by the living room TV, which caused 2432 variations (or 3.9%). Collectively, 98% of the second-to-second power variations over the course of the day derived from the four loads above, while only 2% of the variations derived from the home's other 90 loads.

The large number of power variations caused by non-linear loads makes data analysis challenging. For instance, it may be difficult to discern if a 60 W increase in power is due to a light bulb turning on or simply one of many random variations caused by a large non-linear load. Unfortunately, existing techniques, such as the NILM algorithm from the previous section, that model loads using a few discrete power states are not equipped to detect these rapid and seemingly random variations. However, our modeling indicates that these loads exhibit highly identifiable features, namely by maintaining stable minimum or maximum power levels. Thus, we design a simple filter to identify rapid variations from a stable minimum or maximum power level and remove them from the aggregate trace. Filtering these loads out of the smart meter data not only enables a new way to identify this special class of load, it also simplifies the data by removing a large number of power variations, which, as we show below, improves the results of the existing NILM algorithm from the previous section.

Our stable min-max filter works by simply scanning through the power data time-series for a home and maintaining a *stable power* parameter, which only updates if power deviates from the current parameter setting for more than time T by some threshold power $P_{threshold}$, both of which are chosen based on the load's model. The idea is to update the stable power when *other* devices operate (e.g., a lamp that is turned on) and leave these steps unchanged. Power steps that do *not* update the stable power are attributed to the stable min-max device and filtered—the settings of $P_{threshold}$ and T determine how large and how long a power change may be to be considered a transient change stemming from the stable min-max device.

To clearly illustrate the filter's effect, Fig. 14 applies the filter to aggregate power data that includes only the TV and duct heater (from Fig. 4) combined with a 60 W light bulb. Since the TV and the duct heater exhibit rapid power variations every few seconds (see Fig. 4), the value of T need only be slightly greater than the typical frequency of the variations. In both cases we set $T = 5$ seconds. In addition, we select a value of 10 W for $P_{threshold}$ since both loads have a narrow maximum power usage that varies by less than 10 W. As the figure shows, the filter makes the duct heater and TV appear to be simpler loads

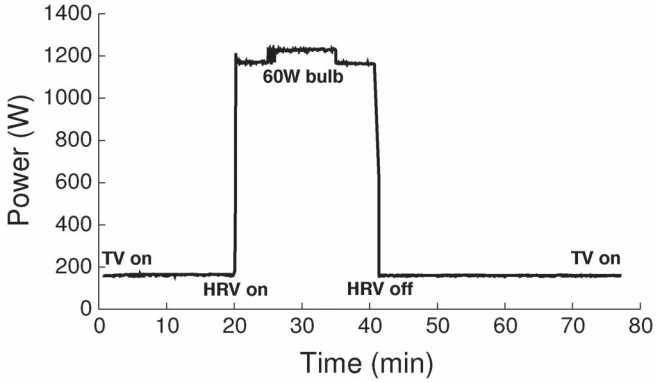

Fig. 14. The stable maximum power enables a filter that removes power variations in smart meter data, making it easier to detect on-off transitions.

with discrete power states, making it possible to easily identify a 60 W light bulb turning on or off by observing changes in power in the filtered data. However, without filtering, any data analysis technique, e.g., using HMMs, that uses changes in aggregate power usage to identify when the light bulb turns on or off would have difficulty, since operating either the TV or duct heater obscures changes in the light bulb's power by introducing significant and frequent power variations. In fact, the duct heater's power variations alone are large enough—greater than 800 W in some cases—to obscure all but the largest loads in the home.

Finally, we also ran the NILM algorithm described in the previous subsection on the unfiltered smart meter data and the data with the duct heater filtered out. We then computed the load error factor for each load in both cases, and found that they were generally less when the duct heater had been filtered out. As an example, Fig. 15 shows time-series power data for the home's refrigerator (a), as well as the disaggregated refrigerator power trace using the unfiltered (b) and filtered (c) smart meter data. The figures illustrate that the refrigerator power trace disaggregated after filtering the smart meter data more closely resembles the ground truth power usage of the refrigerator. The load error factor for the disaggregated refrigerator trace quantifies this resemblance to ground truth: it is 1.328 from the unfiltered data (b), while it is 0.987 (or 25% less) from the filtered data (c).

Result: Uniquely identifiable properties in our load models enable new analysis techniques, such as our filter, and improves existing techniques, such as the NILM algorithm above.

C. Load Detection

Finally, a third use of our models is detecting the presence of specific loads within a building's aggregate power trace. Load detection can reveal many useful pieces of information, such as when devices turn on, how long they tend to operate, and how much power they consume. Load detection is distinct from the well-studied NILM (or load disaggregation) problem [8], [29], in that the latter attempts to attribute a building's entire power usage across a known set of loads, while the former only attempts to detect the presence of a single load. Here, we study a simple distance-based matching technique as one representative example of using our models for load detection.

The core idea in distance-based matching is to use load-specific models to generate representative time-series of devices, which can then be compared against real-world usage data. By computing the "distance" between the model-generated time-series and the observed data, we can identify close "matches" corresponding to the operation of specific devices. Performing this matching requires the use of a specific distance measure. The most natural distance measure is standard Euclidean distance, computed as the square root of the sum of squared differences between energy values (i.e., predicted and actual) at the same times—i.e., for time-series a and b each with N points, the Euclidean distance is given by $D_{Euclidean} = \sqrt{(\sum_{i=1}^{n} |a_i - b_i|^2)}$. The primary advantage of Euclidean distance is simplicity and computational efficiency (i.e., $O(N)$ complexity). However, matching accuracy may suffer when two time-series are similar in shape, but are not well-aligned in time. This issue tends to arise with devices where the length of the active period is not known *a priori*, but rather determined by users at runtime, such as a light bulb that might be active either for ten minutes or an hour.

Results of matching model-generated time-series against aggregate data with Euclidean distance are shown in Figs. 16, 17, and 18, for three different devices. Here we simply construct models of each device and specify the model parameters (e.g., magnitudes) manually, though we envision a better scenario in which models are drawn from a global database containing (known) models for many devices. Note that we match second-to-second power *deltas*, i.e., changes in power, rather than absolute power values. The absolute values of the aggregate energy trace are typically much greater than the power usage of a specific device, which makes them largely meaningless when comparing with a model for a specific device. Matches are observed when the distance measure experiences a drop, indicating a closer similarity to the device model time-series. For example, the distance measure in Fig. 16(b) experiences significant drops (circled) whenever the refrigerator compressor cycles on. These drops are due to the close similarity of the refrigerator model to the actual refrigerator operating within the aggregate trace. Furthermore, the drops make it relatively straightforward to closely approximate the actual refrigerator operation; for example, we can simply insert the refrigerator model when specified by local minima in the distance measure. Circled in Fig. 16 are drops exceeding a fixed threshold, which is a simple but concrete way to detect device "on" events.

We see similar drops in Figs. 17(c) and 18(b); however, using our naïve threshold-based method to identify device operation leads to a significant number of false positives, e.g., roughly 200 false positives for the washing machine for only three actual device cycles. However, most of these false positives are clustered around the true positives, so simple locality-based heuristics, e.g., such as averaging within a window and accounting for knowledge of the approximate cycle length, can still result in accurate load detection.

Some of the shortcomings above with using Euclidean distance matching can also be mitigated by incorporating a notion of temporal stretching and amplitude transformation when matching traces. For example, we may know the consistent

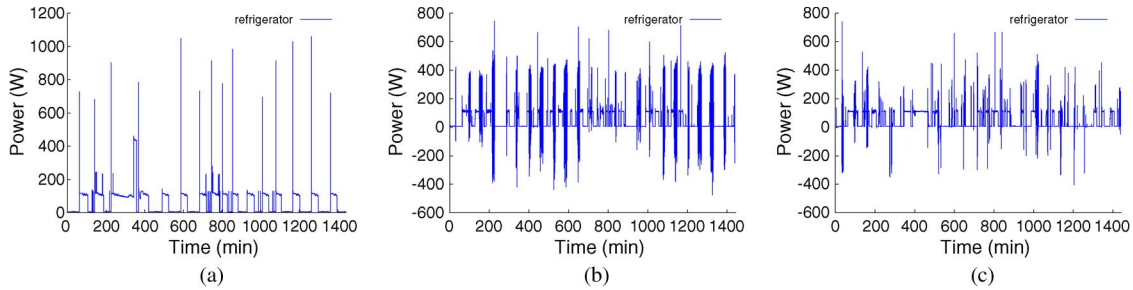

Fig. 15. Disaggregating refrigerator trace with the duct heater included and with the duct heater filtered out. (a) Actual refrigerator trace. (b) With duct heater included. (c) With duct heater filtered out.

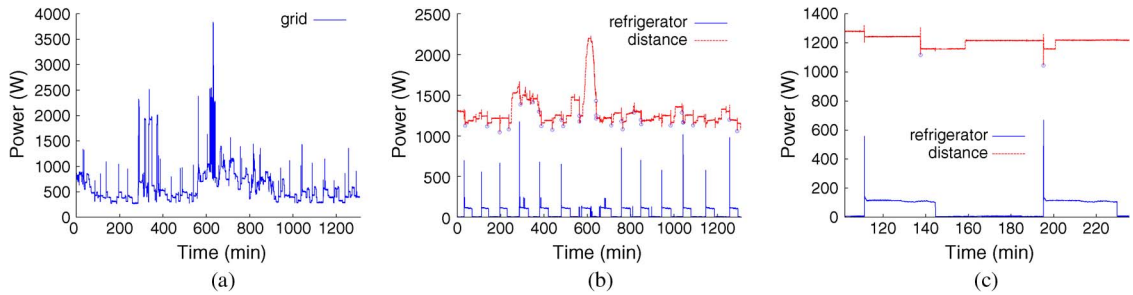

Fig. 16. Refrigerator matches using Euclidean distances. (a) Aggregate trace. (b) Refrigerator Euclidean distances. (c) Zoomed-in.

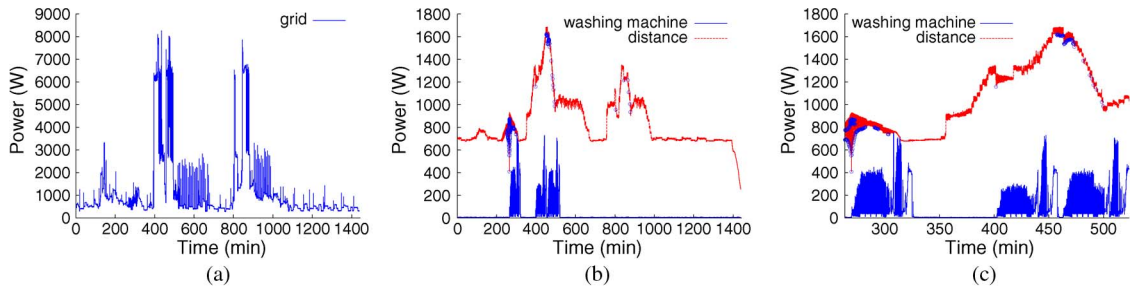

Fig. 17. Washing Machine matches using Euclidean distances where the model used is one entire cycle of the washing machine turning on. (a) Aggregate trace. (b) Washing machine Euclidean distances. (c) Zoomed-in.

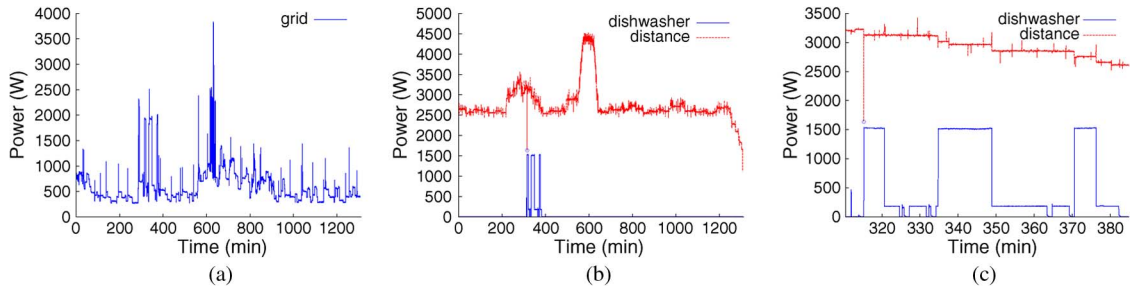

Fig. 18. Dishwasher matches using Euclidean distances. (a) Aggregate trace. (b) Dishwasher Euclidean distances. (c) Zoomed-in.

shape of a given device's usage, but not how long it operates (i.e., the temporal dimension) or exactly how much power it uses at a given time (i.e., its amplitude). Despite the ease with which we may be able to visually identify such devices within a trace (based on their shape alone), Euclidean distance may perform quite poorly on these devices due to slight differences between measured data and the device's model.

We explore an alternate matching technique detailed in [30] that aims to account for these issues, which we refer to as "augmented Euclidean matching." The idea behind this technique is to account for amplitude scaling by searching for matches of a linear function of the input, i.e., searching for a and b such that $aX + b$ matches the aggregate, and to account for temporal scaling by averaging consecutive points, i.e. changing the effective length of the time-series. After applying these changes, multiple candidate time-series windows are searched to determine matches, using an optimization to limit the number of windows that must be searched. Detailed discussion of this technique is omitted for brevity's sake but can be found in [30]. The technique is one of many within the time-series database community that looks for patterns in time-series data that are similar to a query time-series.

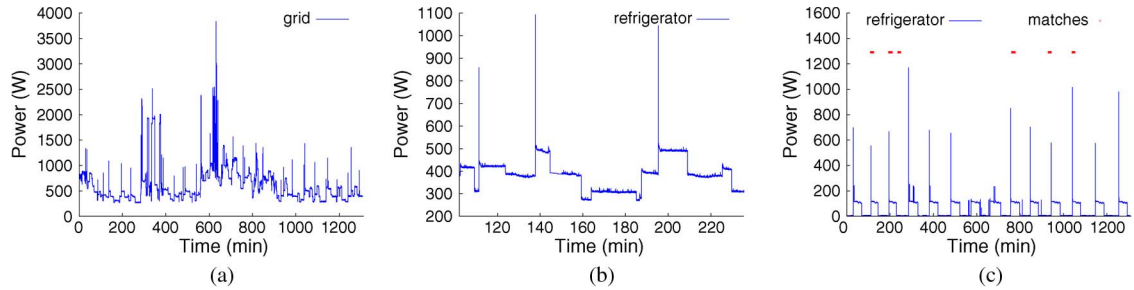

Fig. 19. Refrigerator matches using augmented Euclidean matching. (a) Aggregate trace. (b) Zoomed-in aggregate. (c) Refrigerator matches.

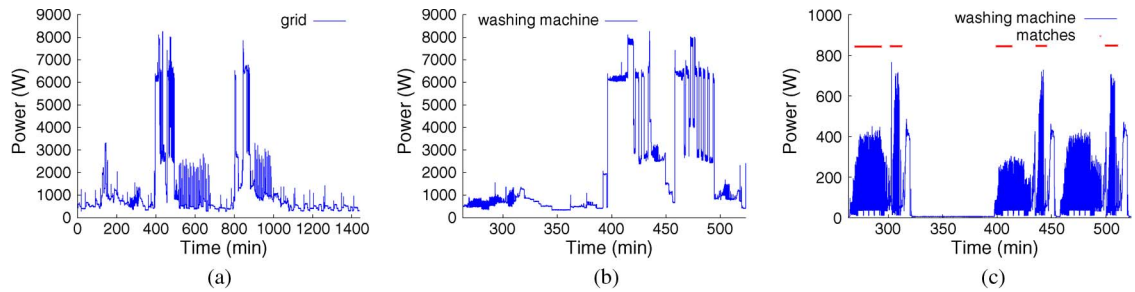

Fig. 20. Washing Machine matches using augmented Euclidean matching where the model used to match is the on-off decay cyclic part at the end. (a) Aggregate trace. (b) Zoomed-in aggregate. (c) Washing machine matches.

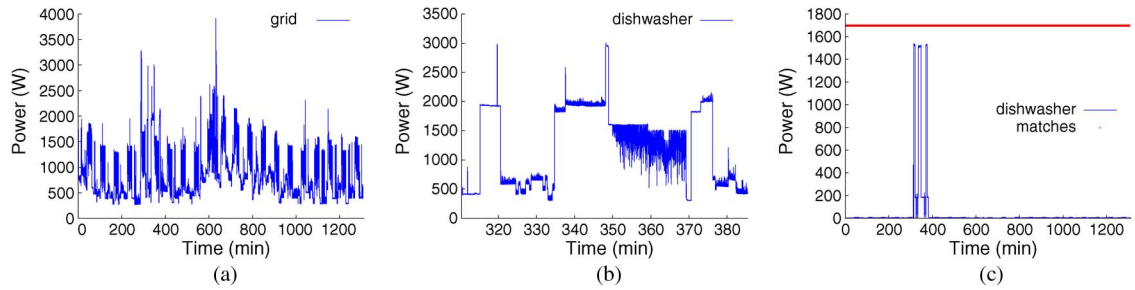

Fig. 21. Dishwasher matches using augmented Euclidean matching. (a) Aggregate trace. (b) Zoomed-in aggregate. (c) Dishwasher matches.

Fig. 19 shows the aggregate trace (a) and the refrigerator trace (b) with matches marked as before when given by the augmented Euclidean technique. We see that the technique accurately identifies matches towards the start and end of the trace when fewer devices are operating (and thus the device to match is least obscured) but has difficulty in the interim time periods when many other devices are operating. This is not surprising, since this technique, as with most in the time-series database community, was not designed for matching against data with variable amounts of noise from other sources. One interesting area of future research is applying these techniques to load detection by adapting them to account for such noise.

Similarly, Fig. 20 shows the results of matching the on-off decay and cyclic components of the washing machine trace shown in Fig. 10(a). We see in Fig. 20(c) that matches are found in the initial as well as latter cyclic phases of the washing machine, despite the multiple amplitudes and durations of the device. However, we see that the initial phases of the third device cycle are not matched, as the match is obscured by another large device operating during this period, which we can see around $t = 480$ min in Fig. 20(a).

Finally, we tried the augmented Euclidean matching on the dishwasher cycle as shown in Fig. 21, but here, no real matches were located at all—the technique simply "matches" the entire trace, which provides no useful information. This behavior is likely due to the fact that the dishwasher exhibits fairly simple stepping behavior between static states as shown in Fig. 5(c)—as a result, it can be morphed in the time and amplitude domains to match the entire aggregate trace. Despite this, augmented Euclidean matching tends to be more conservative in matching than straight Euclidean distance, which tends to result in more false positives.

Both strategies demonstrate that matching within aggregate traces is a difficult problem given the noisiness and complexity of typical aggregate traces. Despite this, we believe that our models can be useful across a variety of matching techniques, as demonstrated in the examples above.

Result: Our models are useful in detecting the presence of specific loads in smart meter data by matching them against a home's aggregate time-series power data.

V. Related Work

In this paper, we focus explicitly on modeling the power usage of common electrical loads. While recent work targets modeling for specific appliances, e.g., a particular brand of

refrigerator [31], it does not generalize to a broad range of devices. Much of the prior research on modeling power usage for individual loads has been done in the context of Non-Intrusive Load Monitoring (NILM). While we expect our models to be broadly useful for data analysis, including, but not limited to, NILM, we survey related work in NILM below.

NILM techniques differ significantly based on the granularity of the current, voltage, and power readings. For instance, fine-grain readings that sample current or voltage frequently within each cycle, e.g., $\gg 60$ Hz, differ significantly from the models we present, since they attempt to capture the behavior of the AC current and voltage waveform. In addition, gathering data at such high frequencies presents challenges: it requires expensive and highly calibrated equipment, while storing and transmitting the data in real time is beyond the capabilities of today's embedded power meters. While other types of device models may be supported by higher-frequency data (such as turn-on transient power signatures in [32]), our models target a data granularity of one reading per second, since this is the finest granularity that commodity low-cost power meters support. We could also build load models using the coarse-grain data, e.g., 5 minutes to an hour, supported by today's utility-installed smart meters [33]. Unfortunately, coarse-grain data measured every minute or more eliminates important details of each load's operation that are useful in analysis.

Recently, a number of researchers have focused on NILM approaches for the per-second power data we use to build our models. Most of these approaches employ generic on-off load models that, as we show, are not accurate. The techniques generally use these simple models to either i) detect changes in load power states by observing changes in building power, [8] or ii) use Viterbi-style algorithms [18] to determine the most likely set of "hidden" states, e.g., combinations of power states for multiple loads, from a sequence of changes in building power [28]. These prior techniques generally do not scale to the large numbers of, often low-power, loads found in typical buildings. For instance, we are not aware of any prior approach that focuses on large-scale scenarios—$>$ 100 loads—with many low-power loads $<$ 50 W, which is a common characteristic of many homes. The lack of research may be due to the inaccuracy of the underlying load models.

VI. Conclusion

This paper presents a new methodology for modeling common electric loads. We derive our methodology empirically by collecting data from a variety of loads and showing the commonalities between them. Finally, we illustrate examples of how to use our models for data analysis.

References

[1] J. Kelso, Ed., *2011 Buildings Energy Data Book*. Washington, DC, USA: U.S. Department Energy, Mar. 2012.

[2] J. Koomey, "Data center electricity use 2005 to 2010," Analytics Press, Burlingame, CA, USA, Tech. Rep., Aug. 2011.

[3] S. Dawson-Haggerty, S. Lanzisera, J. Taneja, R. Brown, and D. Culler, "@scale: Insights from a large, long-lived appliance energy WSN," in *Proc. IPSN*, Apr. 2012, pp. 37–48.

[4] T. Hnat *et al.*, "The Hitchhiker's guide to successful residential sensing deployments," in *Proc. SenSys*, Nov. 2011, pp. 232–245.

[5] X. Jiang, M. V. Ly, J. Taneja, P. Dutta, and D. Culler, "Experiences with a high-fidelity building energy auditing network," in *Proc. SenSys*, Nov. 2009, pp. 113–126.

[6] Y. Kim, T. Schmid, Z. Charbiwala, and M. Srivastava, "ViridiScope: Design and implementation of a fine grained power monitoring system for homes," in *Proc. Ubicomp*, Sep. 2009, pp. 245–254.

[7] S. Barker *et al.*, "Smart∗: An open data set and tools for enabling research in sustainable homes," in *Proc. SustKDD*, Aug. 2012, pp. 1–6.

[8] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 76–84, Feb. 2011.

[9] W. Kleiminger, C. Beckel, T. Staake, and S. Santini, "Occupancy detection from electricity consumption data," in *Proc. BuildSys*, Nov. 2013, pp. 1–8.

[10] D. Chen, S. Barker, A. Subbaswamy, D. Irwin, and P. Shenoy, "Non-intrusive occupancy monitoring using smart meters," in *Proc. BuildSys*, Nov. 2013, pp. 1–8.

[11] S. Barker, A. Mishra, D. Irwin, P. Shenoy, and J. Albrecht, "SmartCap: Flattening Peak Electricity Demand in Smart Homes," in *Proc. PerCom*, Mar. 2012, pp. 67–75.

[12] J. Lu *et al.*, "The smart thermostat: Using occupancy sensors to save energy in homes," in *Proc. SenSys*, Nov. 2010, pp. 211–224.

[13] Pacific Gas and Electric, Leveraging Smart Meter Installation Progress, Feb. 2013. [Online]. Available: http://www.pge.com/myhome/customerservice/smartmeter/deployment/

[14] M. Plaisance, Holyoke Gas and Electric Customers Could Net Savings From Ongoing Meter Study Involving High Performance Computing Center. [Online]. Available: http://www.masslive.com/news/index.ssf/2014/04/holyoke_gas_and_electric_custo.html

[15] eGauge Energy Monitoring Solutions, 2013. [Online]. Available: http://www.egauge.net/

[16] Energy, Inc., Jul. 2012. [Online]. Available: http://www.theenergydetective.com/

[17] S. Barker, S. Kalra, D. Irwin, and P. Shenoy, "Empirical characterization and modeling of electrical loads in smart homes," in *Proc. IGCC*, Jun. 2013, pp. 1–10.

[18] G. Forney, "The Viterbi Algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

[19] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[20] iMeter Solo. [Online]. Available: http://www.insteon.net/2423A1-iMeter-Solo.html

[21] [Online]. Available: http://aeotec.com/z-wave-plug-in-switch

[22] D. Irwin *et al.*, "Exploiting home automation protocols for load monitoring in smart buildings," in *Proc. BuildSys*, Nov. 2011, pp. 7–12.

[23] K. Levenberg, "A Method for the solution of certain non-linear problems in least squares," *Quart. Appl. Math.*, vol. 2, no. 2, pp. 164–168, 1944.

[24] J. Kolter and M. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. SustKDD*, Aug. 2011, pp. 1–6.

[25] K. Anderson *et al.*, "BLUED: A fully labeled public dataset for event-based non-intrusive load monitoring research," in *Proc. SustKDD*, Aug. 2012, pp. 1–5.

[26] O. Ardakanian, S. Keshav, and C. Rosenberg, "Markovian models for home electricity consumption," in *Proc. GreenNets*, Aug. 2011, pp. 31–36.

[27] G. Hart, "Residential energy monitoring and computerized surveillance via utility power flows," *IEEE Technol. Soc. Mag.*, vol. 8, no. 2, pp. 12–16, Jun. 1989.

[28] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," in *Proc. SDM*, Apr. 2011, pp. 747–7586.

[29] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? The case of electricity," *Energy Policy*, vol. 52, no. C, pp. 213–234, 2012.

[30] T. Argyros and C. Ermopoulos, "Efficient subsequence matching in time series databases under time and amplitude transformations," in *Proc. ICDM*, Nov. 2003, pp. 481–484.

[31] J. Taneja, D. Culler, and P. Dutta, "Towards cooperative grids: Sensor/actuator networks for renewables integration," in *Proc. SmartGridComm*, 2010, pp. 1–6.

[32] H.-H. Chang, K.-L. Chen, Y.-P. Tsai, and W.-J. Lee, "A new measurement method for power signatures of non-intrusive demand monitoring and load identification," in *Conf. Rec. IEEE IAS Annu. Meeting*, Oct. 2011, pp. 1–7.

[33] J. Kolter and A. Ng, "Energy disaggregation via discriminative sparse coding," in *Proc. NIPS*, Dec. 2010, pp. 1–9.

Sean Barker (S'12) received the M.S. degree in computer science from the University of Massachusetts, Boston, MA, USA, in 2012 and the B.A. degree in computer science from Williams College, Williamstown, MA, in 2009. He is currently working toward the Ph.D. degree in computer science with the University of Massachusetts Amherst, Amherst, MA. His research interests are in the areas of distributed systems, cloud computing, smart buildings, and sustainability.

Sandeep Kalra (S'06) received the B.E. degree in information technology from Sardar Patel University, Vallabh Vidyanagar, India, in 2009. He is currently working toward the M.S. and Ph.D. degrees in computer science with the University of Massachusetts Amherst, Amherst, MA, USA. His research interests lie in the areas of smart grids and connected homes, sustainability, data science, and analytics.

David Irwin (M'03) received the Ph.D. and M.S. degrees in computer science from Duke University, Durham, NC, USA, in 2007 and 2005, respectively, and the B.S. degree in computer science and mathematics from Vanderbilt University, Nashville, TN, USA, in 2001. He is an Assistant Professor with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst, Amherst, MA, USA. His research interests are broadly in experimental computing systems with a particular emphasis on sustainability.

Prashant Shenoy (F'13) received the B.Tech. degree in computer science and engineering from the Indian Institute of Technology, Bombay, India, in 1993, and the M.S. and Ph.D. degrees in computer science from the University of Texas at Austin, Austin, TX, USA, in 1994 and 1998, respectively. He is currently a Professor of computer science with the University of Massachusetts, Boston, MA, USA. His current research focuses on cloud computing and green computing. Dr. Shenoy is a distinguished member of the ACM.