# Energy Disaggregation via Learning ‘Powerlets’ and Sparse Coding

**Ehsan Elhamifar and Shankar Sastry**
Electrical Engineering and Computer Sciences Department
University of California, Berkeley

## Abstract

In this paper, we consider the problem of energy disaggregation, i.e., decomposing a whole home electricity signal into its component appliances. We propose a new supervised algorithm, which in the learning stage, automatically extracts signature consumption patterns of each device by modeling the device as a mixture of dynamical systems. In order to extract signature consumption patterns of a device corresponding to its different modes of operation, we define appropriate dissimilarities between energy snippets of the device and use them in a subset selection scheme, which we generalize to deal with time-series data. We then form a dictionary that consists of extracted power signatures across all devices. We cast the disaggregation problem as an optimization over a representation in the learned dictionary and incorporate several novel priors such as device-sparsity, knowledge about devices that do or do not work together as well as temporal consistency of the disaggregated solution. Real experiments on a publicly available energy dataset demonstrate that our proposed algorithm achieves promising results for energy disaggregation.

## Introduction

Energy disaggregation, also referred to as non-intrusive load monitoring (Hart 1992), is the task of separating the whole energy signal of a residential, commercial, or industrial building into the energy signals of individual appliances. Disaggregated electricity consumptions not only provide feedback to consumers in order to improve their consumption behavior, but also help to detect malfunctioning of electrical devices, design energy incentives, forecast demands and more (Froehlich et al. 2011). In particular, studies have shown that presenting such an energy breakdown to consumers can lead to energy-saving behavior that improves user efficiency by about $15\%$ (Darby 2006; Neenan and Robinson 2009).

**Prior Work:** Studies on energy disaggregation date back to about thirty years ago. However, recent energy and sustainability challenges facing the society have created renewed interest in this problem, see (Ziefman and Roth 2011; Froehlich et al. 2011) for a comprehensive review. The literature on energy disaggregation can be divided into two categories. The first group of algorithms focuses on classifying electrical events rather than the disaggregation task. Earlier work in this category model each device as a finite state machine and look for sharp edges in real and reactive power signals (Hart 1992). Subsequently, they cluster devices according to consumption changes. However, the drawback of these approaches is that for many devices with low power consumption, clusters corresponding to different devices are indistinguishable. To better distinguish devices, later work incorporate transient and harmonic information using very high-frequency sampling (Laughman and Leeb 2003; Gupta, Reynolds, and Patel 2010; Berges et al. 2010). However, sampling at high-frequency requires expensive hardware and installation of monitoring devices in a building.

The second group of algorithms directly addresses the disaggregation problem by decomposing the aggregate electricity signal into its component appliances over time (Kim et al. 2011; Kolter, Batra, and Ng 2010; Kolter and Jaakkola 2012; Wytock and Kolter 2014). Different approaches in this category can be divided into supervised and unsupervised disaggregation algorithms. The supervised sparse coding-based method in (Kolter, Batra, and Ng 2010) uses a training dataset of electricity signals from different devices across several homes. It models the entire signal of each device over a long period of time, such as a week, as a sparse linear combination of the atoms of an unknown dictionary. Both the sparse coefficients and the dictionary for each device are learned using a discriminative approach. However, the drawback of the algorithm is that it requires to have access to a large training dataset to capture all possible times that the same device may operate. The work of (Kim et al. 2011) uses a Factorial Hidden Markov Model (FHMM) (Ghahramani and Jordan 1997) with block Gibbs sampling to decompose the whole electricity signal into signals of individual devices. Building upon the FHMM framework, (Kolter and Jaakkola 2012) uses an approximate algorithm based on convex programming, which also takes into account unmodeled devices. FHMM can work in the supervised or unsupervised setting, depending on whether a training dataset for individual devices is used. However, the learning often involves EM (Dempster, Laird, and Rubin 1977), which depends on initialization and can get stuck in local optima.

**Paper Contributions:** In this paper, we propose a new supervised algorithm for the energy disaggregation problem. Given a training dataset of electricity consumptions from different devices, we automatically extract signature consumption patterns for each device using a convex programming scheme. To do so, we model the energy consumption of each device using a mixture of dynamical models corresponding to different operation modes of the device. We then find signature consumption patterns by defining an appropriate dissimilarity between pairs of energy snippets and selecting representative snippets, which we refer to a *powerlets*. To do so, we propose a subset selection framework by generalizing the state of the art to the case of dealing with time-series data. Unlike EM-based methods (Kim et al. 2011; Kolter and Jaakkola 2012), whose performance depends on initialization and degrades by increasing the number of states, our framework relies on convex programming which is free of local optima issues. We use the extracted signature consumptions in order to build a dictionary that we refer to as the *powerlets dictionary* for the device. Unlike (Kolter, Batra, and Ng 2010) we do not require large amounts of training data and as few as a single home would be sufficient to build the dictionary. Our method also does not suffer from different consumption shifts in the signal. Moreover, the elements of our learned dictionaries correspond to snippets from actual energy signals instead of arbitrary vectors that are learned using sparse dictionary learning or EM. We collect powerlets of different devices into a dictionary that we use in order to disaggregate a whole energy signal. To perform disaggregation, we propose an optimization algorithm that searches for a representation of the aggregate signal in the learned dictionary by incorporating several types of priors on the solution such as device-sparsity, knowledge about devices that do or do not work together, and temporal consistency of the disaggregation. Finally, our real experiments on a publicly available dataset show that our framework achieves promising results for energy disaggregation.

## Energy Disaggregation Framework

In this section, we discuss our proposed energy disaggregation framework for determining the component appliance consumption from an aggregated electricity signal. In the next two sections, we describe in details each of the two stages of our framework in more details.

We assume that there are $L$ electrical devices in a building, where $x_i(t)$ denotes the energy signal of device $i$ at time $t \in \{1, 2, \ldots, T\}$. Let $\bar{x}(t)$ denote the aggregate energy signal, recorded by a smart meter, at time $t$. Thus, we can write

$$\bar{x}(t) = \sum_{i=1}^{L} x_i(t). \tag{1}$$

Given only the whole power consumption $\{\bar{x}(t)\}_{t=1}^{T}$, the goal of energy disaggregation is to recover the power signal of each of the appliances, i.e., to estimate $\{x_i(t)\}_{t=1}^{T}$ for $i \in \{1, \ldots, L\}$.

In this paper, we take a supervised approach for energy disaggregation, where we assume that a training dataset of energy signals from different devices is available. We propose an energy disaggregation framework that consists of two steps. First, we learn a dictionary of power consumption signatures from different devices using the training dataset. Then, we decode the aggregate energy signal in the learned dictionary using an optimization scheme, which incorporates different types of priors on the estimated device consumptions.

To address the disaggregation problem, we take energy consumption windows of length $w \leq T$ (typically $w << T$) and denote the consumption of device $i$ and the aggregate consumption in the interval $[t, t+w-1]$ by $w$-dimensional vectors $\boldsymbol{y}_i(t)$ and $\bar{\boldsymbol{y}}(t)$, respectively. Our goal is to build a dictionary $\boldsymbol{B} \in \mathbb{R}^{w \times N}$ such that a solution of

$$\bar{\boldsymbol{y}}(t) = \boldsymbol{B}\boldsymbol{c}(t), \tag{2}$$

with the right prior on $\boldsymbol{c}(t)$, reveals the disaggregation of the whole energy signal. We form the dictionary $\boldsymbol{B}$ as

$$\boldsymbol{B} = [\boldsymbol{B}_1 \quad \boldsymbol{B}_2 \quad \cdots \quad \boldsymbol{B}_L] \in \mathbb{R}^{w \times N}, \tag{3}$$

where $\boldsymbol{B}_i \in \mathbb{R}^{w \times N_i}$ denotes the subdictionary associated with the device $i$. Ideally, $\boldsymbol{B}_i$ should efficiently represent signals $\boldsymbol{y}_i(t)$ generated by device $i$.

In order to learn $\boldsymbol{B}_i$, we use the fact that each device has several electricity consumption dynamics corresponding to different operation modes, as shown in Figure 1. Thus, our goal is to efficiently extract these consumption signatures in order to build a dictionary for each device. To do so, we model the energy signal of each device $i$ using a mixture of dynamical systems, where each dynamical model captures an operation mode of the device. We define appropriate dissimilarities between energy snippets of each device $i$ and use them in a subset selection scheme, which we generalize to deal with time-series data, to find the energy signatures. We use these signatures, which we refer to as powerlets, to form the dictionary $\boldsymbol{B}_i$, which we refer to as the powerlets dictionary for device $i$.

Once we learn powerlets for all devices and form the dictionary $\boldsymbol{B}$, we use the solution of (2) in order to decode a new aggregate energy signal. Notice that the solution of (2) is generally not unique since there are many combinations of powerlets that result in the same aggregate signal. Thus, we need to impose appropriate priors and constraints on the solution $\boldsymbol{c}(t)$. Moreover, since a new aggregate signal may contain noise and unmodeled energy components, we search for an approximate solution for (2) instead of an exact solution. As a result, we propose to solve the optimization program

$$\begin{aligned} \min_{\{\boldsymbol{c}(t)\}_{t=1}^{T}} \; & \lambda\, \rho(\boldsymbol{c}(1), \ldots, \boldsymbol{c}(T)) + \sum_{t=1}^{T} \ell(\bar{\boldsymbol{y}}(t) - \boldsymbol{B}\boldsymbol{c}(t)) \\ \text{s.t.} \; & \mathbf{1}^\top \boldsymbol{c}_i(t) \leq 1, \;\; \boldsymbol{c}_i(t) \in \{0,1\}^{N_i}, \; \forall t, i. \end{aligned} \tag{4}$$

The function $\rho(\cdot)$ incorporates several types of priors on the solution, such as device-sparsity, knowledge about devices that work simultaneous or sequentially and temporal smoothness. The function $\ell(\cdot)$ incorporates the appropriate loss function for representing the whole electricity signal in

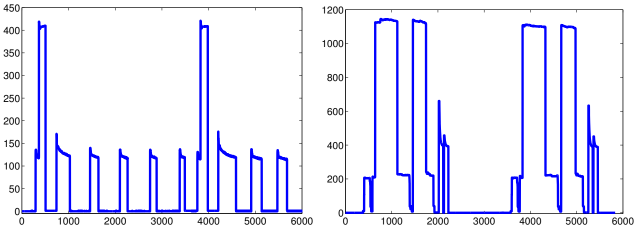

Figure 1: Energy consumption signals of different devices typically consist of distinct consumption patterns corresponding to different modes of the operation of the device. Left: energy signal of a fridge. Right: energy signal of a dishwasher. Horizontal and vertical axes correspond to time and power consumption, respectively.

the dictionary $\boldsymbol{B}$. Finally, the constraints ensure that, at each time instant, for operating devices we select one powerlet from the subdictionary associated with that device and for devices that are not working we select no powerlets.

## Learning Powerlets Dictionary

In this section, we consider the problem of learning a dictionary that captures consumption dynamics of different devices. We use a training dataset of energy signals $\{x_i(t)\}_{t=1}^{T_i}$ for each device $i = 1, \ldots, L$ to learn a dictionary $\boldsymbol{B}_i \in \mathbb{R}^{w \times N_i}$ that captures distinct consumption patterns of the device $i$. To do so, we model the electricity signal of each device $i$ using a mixture of dynamical systems, where each dynamical system represents a different operation mode of the device. For each device $i$ at each time instant $t$, we extract energy snippets of length $w$, defined as

$$\boldsymbol{y}_i(t) \triangleq [x_i(t) \quad x_i(t+1) \ \cdots \ x_i(t+w-1)]^\top \in \mathbb{R}^w. \quad (5)$$

Given the collection of vectors $\{\boldsymbol{y}_i(t)\}_{t=1}^{T_i-w+1}$, our goal is to find a compact dictionary $\boldsymbol{B}_i \in \mathbb{R}^{w \times N_i}$, with $N_i \ll T_i$, that efficiently represents the energy vectors from the device. To do so, we use a subset selection scheme to find representative vectors from $\{\boldsymbol{y}_i(t)\}$. Given the fact that $\{\boldsymbol{y}_i(t)\}_{t=1}^{T_i-w+1}$ correspond to sequential vectors, we generalize the subset selection algorithm in (Elhamifar, Sapiro, and Sastry 2014; Elhamifar, Sapiro, and Vidal 2012), which deals with static data, to time-series data.

**Remark 0.1** *While it is possible to use dictionary learning algorithms such as (Aharon, Elad, and Bruckstein 2006; Mairal et al. 2008), it is advantageous to learn the dictionary by finding representatives of $\{\boldsymbol{y}_i(t)\}_{t=1}^{T_i-w+1}$ instead of potentially arbitrary vectors in $\mathbb{R}^w$. This comes from the fact that each device often exhibits several distinct consumption patterns, as shown in Figure 1. In addition, while state-of-the-art dictionary learning solve the non-convex problem of simultaneously finding the dictionary and the representation, hence are prone to local minima, our method which selects representatives of a dataset is convex and free of local minima issues.*

### Sparse Subset Selection For Sequential Data

In this section, we review the Dissimilarity-based Sparse subset Selection (DS3) algorithm (Elhamifar, Sapiro, and Sastry 2014; Elhamifar, Sapiro, and Vidal 2012) for finding representatives of a dataset and extend the algorithm to deal with sequential observations.

Assume that we are given nonnegative dissimilarities $\{d_{ij}\}_{i,j=1,\ldots,N}$ between pairs of $N$ data points, where $d_{ij}$ denotes how well the data point $i$ represents the data point $j$. The smaller the value of $d_{ij}$ is, the better $i$ represents $j$. The dissimilarity matrix $\boldsymbol{D} \in \mathbb{R}^{N \times N}$ is then formed by collecting $d_{ij}$'s as its entries.

Given $\boldsymbol{D}$, the goal is to find a few data points that well represent the dataset. To do so, (Elhamifar, Sapiro, and Sastry 2014; Elhamifar, Sapiro, and Vidal 2012) propose a convex optimization framework by introducing a matrix $\boldsymbol{Z} \in \mathbb{R}^{N \times N}$ whose each entry $z_{ij} \in [0,1]$ indicates the probability that point $i$ becomes a representative of point $j$. In order to select a few representatives that well encode the collection of data points, the following minimization program is proposed

$$\min \ \gamma \, \|\boldsymbol{Z}\|_{\infty,1} + \operatorname{tr}(\boldsymbol{D}^\top \boldsymbol{Z}) \quad \text{s.t.} \quad \boldsymbol{Z} \geq 0, \ \boldsymbol{1}^\top \boldsymbol{Z} = \boldsymbol{1}^\top, \quad (6)$$

where $\|\boldsymbol{Z}\|_{\infty,1} \triangleq \sum_{i=1}^{N} \|\boldsymbol{Z}_{i*}\|_\infty$ is the convex surrogate for counting the number of nonzero rows of $\boldsymbol{Z}$, with $\boldsymbol{Z}_{i*}$ denoting the $i$-th row of $\boldsymbol{Z}$. Also, $\operatorname{tr}(\boldsymbol{D}^\top \boldsymbol{Z})$, with $\operatorname{tr}(\cdot)$ denoting the trace operator, is the encoding cost of data points via representatives. The constraints of the optimization program ensure that each column of $\boldsymbol{Z}$ is a probability vector. The regularization parameter $\gamma > 0$ puts a trade-off between the encoding cost and the number of representatives, where a smaller value of $\gamma$ results in obtaining more representatives and vice versa. One can find representatives from indices of nonzero rows of the solution of (6).

Notice that for problems such as energy disaggregation where the observations have a sequential nature, the formulation in (6) does not take into account such structure of the data. More specifically, one should notice that consecutive observations $j$ and $j+1$ should ideally have similar representatives except when a switch in the device dynamics occurs. Thus, denoting the $j$-th column of $\boldsymbol{Z}$ by $\boldsymbol{Z}_{*j}$, we would like to minimize the cost $\sum_j \mathrm{I}(\|\boldsymbol{Z}_{*j} - \boldsymbol{Z}_{*j+1}\|_2)$, where $\mathrm{I}(\cdot)$ denotes the indicator function, which is zero when its argument is zero and is one otherwise. In other words, we would like to obtain similar representatives for consecutive data points except at instances where a switch in the dynamics happens. Thus, to obtain representatives of sequential observations, we propose to solve the convex program

$$\begin{aligned} &\min \ \gamma \, \|\boldsymbol{Z}\|_{\infty,1} + \gamma' \sum_j \|\boldsymbol{Z}_{*j} - \boldsymbol{Z}_{*j+1}\|_2 + \operatorname{tr}(\boldsymbol{D}^\top \boldsymbol{Z}) \\ &\text{s.t.} \quad \boldsymbol{Z} \geq 0, \ \boldsymbol{1}^\top \boldsymbol{Z} = \boldsymbol{1}^\top, \end{aligned} \quad (7)$$

where $\gamma' > 0$ and we use the standard convex relaxation for $\sum_j \mathrm{I}(\|\boldsymbol{Z}_{*j} - \boldsymbol{Z}_{*j+1}\|_2)$ by dropping the indicator function.

### Extracting Powerlets

To capture electricity consumption signatures, we model each device $i$ using a mixture of dynamical systems, with each dynamical model representing a distinct consumption pattern. More specifically, for each vector $\boldsymbol{y}_i(t)$, we learn an ARX model with the parameter $\boldsymbol{\beta}_i(t)$ as

$$x_i(t') = \boldsymbol{\beta}_i(t)^\top \begin{bmatrix} \boldsymbol{r}_i(t') \\ 1 \end{bmatrix} + \varepsilon(t'), \;\; t' \in [t, t+w-1], \tag{8}$$

where $\boldsymbol{\beta}_i(t) \in \mathbb{R}^{m+1}$ denotes the parameter of an $m$-th order ARX model and $\boldsymbol{r}_i(t')$, called the regressor, is defined as

$$\boldsymbol{r}_i(t') \triangleq [x_i(t'-1) \;\; \cdots \;\; x_i(t'-m)]^\top \in \mathbb{R}^m. \tag{9}$$

The parameters $\boldsymbol{\beta}_i(t)$ are then learned using the least-squares approach. We define a dissimilarity between a pair of vectors $\boldsymbol{y}_i(t_1)$ and $\boldsymbol{y}_i(t_2)$ such that vectors that come from the same dynamical model are close and vectors from different models are far. To compute how well $\boldsymbol{y}_i(t_1)$ represents $\boldsymbol{y}_i(t_2)$, we use the model $\boldsymbol{\beta}_i(t_1)$ learned from $\boldsymbol{y}_i(t_1)$ and compute

$$d(\boldsymbol{y}_i(t_1), \boldsymbol{y}_i(t_2)) \triangleq \frac{1}{w} \sum_{t'=t_2}^{t_2+w-1} (x_i(t') - \boldsymbol{\beta}_i(t_1)^\top \begin{bmatrix} \boldsymbol{r}_i(t') \\ 1 \end{bmatrix})^2. \tag{10}$$

We then form the dissimilarity matrix for device $i$, which we denote by $\boldsymbol{D}_i$, and solve the optimization program in (7) to find representative consumption signatures, which we refer to as powerlets of device $i$. We then form the powerlets dictionary $\boldsymbol{B}_i \in \mathbb{R}^{w \times N_i}$, where $N_i$ denotes the number of representative vectors for device $i$.

## Aggregate Energy Decoding

In this section, we consider the problem of energy disaggregation in the powerlets dictionary learned from all devices. We consider the aggregate signal $\{\bar{x}(t)\}_{t=1}^T$ and define

$$\bar{\boldsymbol{y}}(t) \triangleq [\bar{x}(t) \quad \bar{x}(t) \quad \cdots \quad \bar{x}(t+w-1)]^\top \in \mathbb{R}^w. \tag{11}$$

Thus, the vector $\bar{\boldsymbol{y}}(t)$ contains the aggregate energy consumption of the building in the interval $[t, t+w-1]$. In order to disaggregate the energy signal, we search for a representation of $\bar{\boldsymbol{y}}(t)$ in the powerlets dictionary $\boldsymbol{B}$ as

$$\bar{\boldsymbol{y}}(t) = \boldsymbol{B}\boldsymbol{c}(t), \tag{12}$$

where $\boldsymbol{c}(t) \triangleq [\boldsymbol{c}_1(t) \;\; \cdots \;\; \boldsymbol{c}_L(t)]^\top \in \mathbb{R}^N$ and $\boldsymbol{c}_i(t) \in \mathbb{R}^{N_i}$ denotes the coefficient subvector associated with $\boldsymbol{B}_i$.

Notice that (12) often admits many solutions for a given $\bar{\boldsymbol{y}}(t)$, since different combinations of powerlets can lead to the same aggregate signal. Thus, we need to incorporate constraints and priors on the solution that capture realistic assumptions on the way the aggregate energy signal is generated. To do so, we first incorporate the constraint that since powerlets correspond to actual power consumption signatures, we should select at most one powerlet from each device. Thus, we impose the constraint

$$\mathbf{1}^\top \boldsymbol{c}_i(t) \leq 1, \;\; \boldsymbol{c}_i(t) \in \{0,1\}^{N_i}, \tag{13}$$

where $\mathbf{1}$ denotes a vector of all ones of appropriate dimension. Next, we impose priors on the structure of the coefficient vector $\boldsymbol{c}(t)$.

### Device-Sparsity Prior

It is often the case that different combinations of powerlets from different groups of devices lead to the same aggregate signal. For instance, a combination of lighting, fridge, dishwasher and heating as well as a combination of more than twenty different devices, which mostly consume small amounts of power, may construct the same aggregate energy signal at a particular time instant. Thus, it is natural to impose a device-sparsity constraint on the solution $\boldsymbol{c}(t)$, preferring representations that use smaller number of operating devices. This can be achieved by minimizing

$$\rho_1(\boldsymbol{c}(t)) \triangleq \sum_{i=1}^{L} \mathrm{I}(\|\boldsymbol{c}_i(t)\|_2), \tag{14}$$

which counts the number of devices that participate in the reconstruction of the aggregate signal. Given the constraints in (13), we can rewrite (14) as

$$\rho_1(\boldsymbol{c}(t)) = \sum_{i=1}^{L} (\mathbf{1}^\top \boldsymbol{c}_i(t))^2 = \boldsymbol{c}(t)^\top \mathbf{1}\mathbf{1}^\top \boldsymbol{c}(t), \tag{15}$$

since $\mathbf{1}^\top \boldsymbol{c}_i(t) = 1$ when $\|\boldsymbol{c}_i(t)\|_2$ is nonzero and $\mathbf{1}^\top \boldsymbol{c}_i(t) = 0$ otherwise.

### Co-occurrence Prior

It is often the case that some devices in a building work together around the same time, e.g., kitchen appliances, while some devices often do not work at the same time. In this section, we show that our framework can efficiently incorporate such priors. We discuss only two cases and the reader would notice that derivations for other cases would be similar.

**Case A:** device $i$ is on/off if and only if device $j$ is on/off. In this case, we should have $\mathbf{1}^\top \boldsymbol{c}_i(t) = \mathbf{1}^\top \boldsymbol{c}_j(t)$. As a result, we would like to minimize the objective function

$$\begin{aligned} \rho_{2a}(\boldsymbol{c}(t)) &\triangleq \frac{1}{2} \sum_{(i,j) \in \mathbb{A}} (\boldsymbol{e}_i^\top \boldsymbol{c}(t) - \boldsymbol{e}_j^\top \boldsymbol{c}(t))^2 \\ &= \frac{1}{2} \boldsymbol{c}(t)^\top ( \sum_{(i,j) \in \mathbb{A}} (\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j))^\top \boldsymbol{c}(t), \end{aligned} \tag{16}$$

where $\boldsymbol{e}_i \in \mathbb{R}^N$ denotes a vector whose entries corresponding to powerlets from device $i$ are one and the rest of its entries are zero. The set $\mathbb{A}$ indicates the set of all pairs of devices that work simultaneously.

**Case B:** if device $i$ is on, then device $j$ is off. In this case, we do not want the two devices to be working at the same time. We can cast this as minimizing the objective function

$$\begin{aligned} \rho_{2b}(\boldsymbol{c}(t)) &\triangleq \sum_{(i,j) \in \mathbb{B}} (\boldsymbol{e}_i^\top \boldsymbol{c}(t))(\boldsymbol{e}_j^\top \boldsymbol{c}(t)) \\ &= \frac{1}{2} \boldsymbol{c}(t)^\top ( \sum_{(i,j) \in \mathbb{B}} \boldsymbol{e}_i^\top \boldsymbol{e}_j + \boldsymbol{e}_j^\top \boldsymbol{e}_i ) \boldsymbol{c}(t), \end{aligned} \tag{17}$$

where $\mathbb{B}$ denotes the set of all pairs of devices that belong to this case.

### Temporal Smoothness Prior

Given the fact that for each device $i$, $|\mathbf{1}^\top \boldsymbol{c}_i(t) - \mathbf{1}^\top \boldsymbol{c}_i(t+1)|$ is zero except at times when it turns on or off and the fact that such switching does happen at a small number of time instants compared to the entire time period, we would like to minimize the objective function

$$\rho_3(\{\boldsymbol{c}(t)\}_{t=1}^T) = \sum_{i=1}^{L} \sum_{t=1}^{T-1} |\mathbf{1}^\top \boldsymbol{c}_i(t) - \mathbf{1}^\top \boldsymbol{c}_i(t+1)|. \quad (18)$$

Notice that the above cost function allows to change the active powerlet within a dictionary of each device without paying penalty, only counting if the device gets on or off.

### Disaggregation Optimization

Considering all the constraints and priors studied so far, we propose to solve the optimization program

$$\begin{aligned} &\min \lambda\, \rho(\{\boldsymbol{c}(t)\}_{t=1}^T) + \sum_{t=1}^{T} \ell(\bar{\boldsymbol{y}}(t) - \boldsymbol{B}\boldsymbol{c}(t)) \\ &\text{s. t.} \;\; \mathbf{1}^\top \boldsymbol{c}_i(t) \leq 1, \;\; \boldsymbol{c}_i(t) \in \{0,1\}^{N_i}, \; \forall t, i. \end{aligned} \quad (19)$$

where the composite prior on the coefficients is defined by

$$\begin{aligned} \rho(\{\boldsymbol{c}(t)\}_{t=1}^T) \triangleq \sum_{t=1}^{T} \rho_1(\boldsymbol{c}(t) + \eta \sum_{t=1}^{T} (\rho_{2a}(\boldsymbol{c}(t)) + \rho_{2b}(\boldsymbol{c}(t))) \\ + \eta' \rho_3(\{\boldsymbol{c}(t)\}_{t=1}^T), \end{aligned} \quad (20)$$

where $\eta, \eta' > 0$ are regularization parameters and $\ell(\cdot)$ denotes the loss function. Typically, we choose $\ell(\cdot) = \|\cdot\|_1$, which provides robustness to robustness to errors, transients and unmodeled dynamics. Once we solve the optimization program (19) and obtain $\boldsymbol{c}^*(t)$, we estimate the energy consumption of each device at time $t$ by $\hat{\boldsymbol{y}}_i(t) = \boldsymbol{B}_i \boldsymbol{c}_i^*(t)$.

## Experiments

In this section, we evaluate our proposed energy disaggregation framework on the real-world REDD dataset (Kolter and Johnson 2011), a large publicly available dataset for electricity disaggregation. The dataset consists of power consumption signals from six different houses, where for each house, the whole electricity consumption as well as electricity consumptions of about twenty different devices are recorded. The signals from each house are collected over a period of two weeks with a high frequency sampling rate of 15kHz. We exclude House 5 data from our experiments, since, for the majority of devices, it contains very few or no events in the entire recording period. We refer to our method as Powerlet-based Energy Disaggregation (PED) algorithm.

### Experimental Settings

Since our framework can work with both high-frequency and low-frequency signals and given the fact that low-frequency sampling is more practical, less costly and more challenging, we use the low-frequency sampling rate of 1Hz. To learn the dictionary of powerlets for each device in a house, we use the first week of recorded electricity signals for training and use the rest of data for testing. We choose a window size of $w = 15$. It is important to note that a small window size results in lower performance, due to overemphasizing transients when building the dictionary, while a large window size results in having different dynamics or operation modes of a device within a given window, hence obtaining lower performance. With a sampling rate of 1Hz, typically, $w \in [10, 50]$ corresponds to temporal windows that capture transients and contain electricity from a single operation mode. We model each device as a mixture of dynamical systems of the form (8), where we set the model order to be $m = 3$ (the experimental results are robust to other choices of $m$ around this value). We use our subset selection scheme in (7) to extract about 20 powerlets for each device. Thus, we obtain a compact dictionary $\boldsymbol{B}$ in our method, which typically consists of a few hundred powerlets. In order to perform disaggregation using the optimization program (19), we set $\lambda = 30$ and $\eta = \eta' = 1$ and use the prior that kitchen appliances typically work together (without this prior we obtain between 3% and 5% lower performance). To implement (19), we use the standard integer programming solver of MOSEK. With $w = 15$ and 400 powerlets and without the temporal smoothness prior, it takes about 12 seconds to perform disaggregation on a temporal window of 15 seconds. As a result, our method can perform disaggregation in real-time. However, it is important to mention that we can make use of the standard convex relaxation by using the constraint $\boldsymbol{c}_{ij} \in [0,1]^{N_i}$ instead of binary constraints on $\boldsymbol{c}_i$ elements and solve the resulting problem using faster algorithms such as ADMM or Proximal methods. However, since the integer programming formulation is our desired formulation that, given the compact size of our dictionaries, can be solved efficiently, we choose not to use convex relaxations in this paper. We leave investigating convex relaxations of our desegregation optimization for future work.

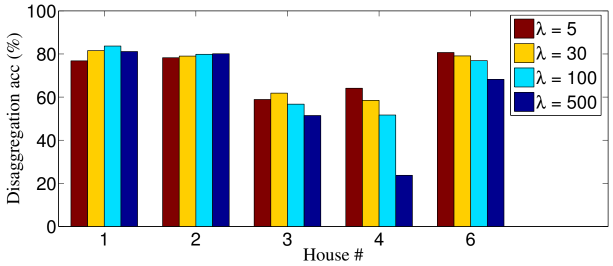

Figure 2: Effect of the regularization parameter $\lambda$ on the disaggregation accuracy of our proposed algorithm for houses in the REDD dataset.

Once we solve the disaggregation optimization for the test aggregate electricity signal in a house, we compute the disaggregation accuracy, similar to (Kolter and Johnson 2011), by

$$\text{disaggregation acc} = 1 - \frac{\sum_{t \in \mathbb{W}} \sum_{i=1}^{M} \|\hat{\boldsymbol{y}}_i(t) - \boldsymbol{y}_i(t)\|_1}{2 \sum_{t \in \mathbb{W}} \|\bar{\boldsymbol{y}}(t)\|_1}, \quad (21)$$

where $\mathbb{W} \triangleq \{1, w+1, 2w+1, \dots\}$ and the 2 factor in the

Table 1: Energy disaggregation accuracies (%) of different algorithm over the six different houses in the REDD dataset.

| | House 1 | House 2 | House 3 | House 4 | House 6 | Total |
|---|---|---|---|---|---|---|
| Simple Mean | 41.4% | 39.0% | 46.7% | 52.7% | 33.7% | 42.7% |
| FHMM | 71.5% | 59.6% | 59.6% | **69.0%** | 62.9% | 64.5% |
| PED (proposed) | **81.6%** | **79.0%** | **61.8%** | 58.5% | **79.1%** | **72.0%** |

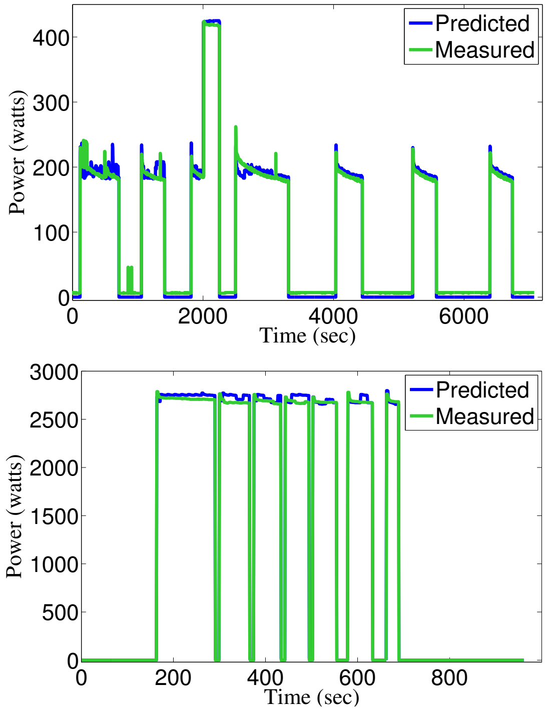

Figure 3: Actual and estimated electricity consumption of refrigerator (top) and washer-dryer (bottom) for House 1 using our proposed framework.

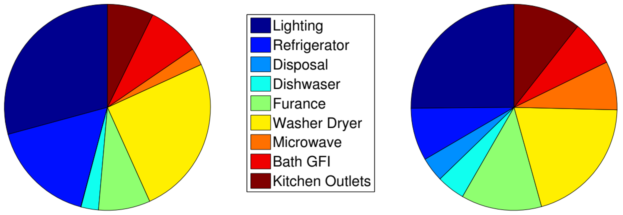

Figure 4: Pie charts showing actual energy consumption (left) and predicted energy consumption by our framework (right) for different devices in House 3 during one week.

denominator comes from the that that the absolute value results in "double counting" errors. We compare our method with the FHMM algorithm (in its supervised setting) (Kolter and Johnson 2011) and a Simple Mean prediction algorithm, which estimates the total consumption percentage of each device and predicts that the whole electricity signal breaks down according to this percentage at all times.

## Experimental Results

Figure 2 shows the disaggregation results of our algorithm for different houses as a function of the regularization parameter $\lambda$ in the proposed optimization program (19). Notice that for a large range of the regularization parameter, $\lambda \in [5, 100]$, our algorithm performs well. However, for large values of $\lambda$, the performance can decrease, as the accuracy results for House 4 shows. This comes from the fact that by putting higher emphasis on the device-sparsity, i.e., having a smaller number of operating devices in the disaggregation results, the optimization prefers to set the contribution of devices with small consumption to zero.

Table 1 shows the disaggregation results for all the six houses in the REDD dataset, with $\lambda = 30$ in our method. Notice that our algorithm performs significantly better than FHMM and the naive Simple Mean on the dataset, achieving about $7.5\%$ higher accuracy overall. This comes from the fact that we have modeled and separated the two steps of learning a dictionary and performing disaggregation, focusing separately on a convex method for learning the desired dictionary of powerlets and on the desired disaggregation formulation. On the other hand, FHMM learns a dictionary using EM, which is prone to local optima, and uses approximate inference for disaggregation.

Figure 3 shows the actual and estimated energy consumption, obtained by our method, for two devices in the House 1. Notice that our method perfectly captures transients and different steady states in each device, thanks to our effective approach for building the powerlets dictionary, and achieves a close prediction to the actual energy consumption. Finally, Figure 4 shows the Pie charts corresponding to the actual and the predicted energy consumption by our algorithm for the House 3 over one week. Notice that the predicted consumption of our method is close to the actual consumption, achieving $81.8\%$ accuracy. In fact, this accuracy is much higher than the one reported in Table 1 for House 3. This comes from the fact that while predicted consumption can be different from actual consumption at each time instant, e.g., due to prediction lag, if we aggregate the predictions over a longer time period, such errors decrease and we often obtain much higher accuracy.

## Conclusions

In this paper, we proposed a new algorithm for energy disaggregation which consists of the two steps of learning a dictionary of power consumption signatures and a disaggregation optimization. To address the first step, we modeled each device as a mixture of dynamical systems, computed dissimilarities between energy snippets of each device using the learned models, and generalized state-of-the-art convex subset selection schemes to deal with sequential data in order to find signature power consumptions for each device. Collecting powerlets from all devices in a dictionary, we proposed an optimization program for disaggregating a whole energy signal by incorporating several priors such as device-sparsity, knowledge about devices that do or do not work together, and temporal smoothness. Finally, by experiments on a real energy dataset, we showed that our framework provides promising results for energy disaggregation.

Investigating convex relaxations of our proposed integer program for desegregation and their efficient implementation are the subject of our ongoing research. In addition, investigating conditions on the powerlets, under which our proposed optimization recovers the true disaggregation of a given aggregate signal, is the subject of our current theoretical analysis.

## Acknowledgment

E. Elhamifar and S. Sastry are supported by grants ONR N000141310341, ONR N000141210609, ONR N000140910230, DARPA FA8650-11-1-7153. This work is supported by NSF CPS:Large:ActionWebs award number 0931843, TRUST (Team for Research in Ubiquitous Secure Technology) which receives support from NSF (award number CCF-0424422), and FORCES (Foundations Of Resilient CybEr-physical Systems) which receives support from NSF (award number CNS-1239166).

## References

Aharon, M.; Elad, M.; and Bruckstein, A. M. 2006. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing* 54(11):4311–4322.

Berges, M.; Goldman, E.; Matthews, H. S.; and Soibelman, L. 2010. Enhancing electricity audits in residential buildings with non-intrusive load monitoring. *Journal of Industrial Ecology: Special Issue on Environmental Applications of Information and Communications Technology* 14(5).

Darby, S. 2006. The effectiveness of feedback on energy consumption. *Technical report, Environmental Change Institute, University of Oxford*.

Dempster, A.; Laird, N.; and Rubin, D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39(1).

Elhamifar, E.; Sapiro, G.; and Sastry, S. S. 2014. Dissimilarity-based sparse subset selection. *under review in IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Elhamifar, E.; Sapiro, G.; and Vidal, R. 2012. Finding exemplars from pairwise dissimilarities via simultaneous sparse recovery. *Neural Information Processing Systems*.

Froehlich, J.; Larson, E.; Gupta, S.; Cohn, G.; Reynolds, M.; and Patel, S. 2011. Disaggregated end-use energy sensing for the smart grid. *IEEE Pervasive Computing* 10(1).

Ghahramani, Z., and Jordan, M. I. 1997. Factorial hidden markov models. *Machine Learning* 29(2-3).

Gupta, S.; Reynolds, S.; and Patel, S. N. 2010. Electrisense: Single-point sensing using emi for electrical event detection and classification in the home. *Conference on Ubiquitous Computing*.

Hart, G. 1992. Nonintrusive appliance load monitoring. *Proceedings of the IEEE* 80(12).

Kim, H.; Marwah, M.; Arlitt, M.; Lyon, G.; and Han, J. 2011. Unsupervised disaggregation of low frequency power measurements. *In Proceedings of theSIAM Conference on Data Mining*.

Kolter, J. Z., and Jaakkola, T. 2012. Approximate inference in additive factorial hmms with application to energy disaggregation. *International Conference on Artificial Intelligence and Statistics*.

Kolter, J. Z., and Johnson, M. J. 2011. Redd: A public data set for energy disaggregation research. *SustKDD Workshop on Data Mining Applications in Sustainability*.

Kolter, J. Z.; Batra, S.; and Ng, A. Y. 2010. Energy disaggregation via discriminative sparse coding. In *Advances in Neural Information Processing Systems*.

Laughman, C., and Leeb, S. 2003. Advanced non-intrusive monitoring of electric loads. *IEEE Power and Energy*.

Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G.; and Zisserman, A. 2008. Supervised dictionary learning. *NIPS*.

Neenan, B., and Robinson, J. 2009. Residential electricity use feedback: A research synthesis and economic framework. *Technical report, Electric Power Research Institute*.

Wytock, M., and Kolter, J. Z. 2014. Contextually supervised source separation with application to energy disaggregation. In *AAAI Conference on Artificial Intelligence*.

Ziefman, M., and Roth, K. 2011. Nonintrusive appliance load monitoring: Review and outlook. *IEEE Transactions on Consumer Electronics* 57(1).