A Framework for Non Intrusive Load Monitoring using Bayesian Inference

Krishnan Srinivasarengan, Goutam Y G, M Girish Chandra
Innovation Labs, Tata Consultancy Services,
Bangalore, 560066, India
{krishnan.srinivasarengan, goutam.yg, m.gchandra}@tcs.com

Swanand Kadhe
ECE Department, Texas A&M University
College Station, TX 77843, US
kswanand1@tamu.edu

***Abstract*—Non-Intrusive Load Monitoring (NILM) refers to the disaggregation of electric appliances from a single point measurement. The problem is gaining a lot of attention recently, primary due to the promising energy savings as well as potential business prospects such a solution brings. However, in a large scale deployment, the digital meter is unlikely to have multiple electrical parameters which most existing NILM research rely on. In this paper, we report the results of using a Bayesian approach to obtain the disaggregation of the loads where only active power measurements are available at a sampling rate of a few seconds. The proposed method requires the prior availability of appliance information (i.e., the prior probability and appliance ratings). To obtain the appliance information for the disaggregation algorithm, we adopt an unsupervised learning approach. Further, we present the results of these algorithms on a simulated and an open household electric consumption data set.**

***Index Terms*—Electric load disaggregation, Non-Intrusive Load Monitoring, Unsupervised Learning, Bayesian Inference**

I. Introduction

In the Internet of Things (IoT) paradigm, it is envisioned to connect all the facilities and devices into the Internet, so that monitoring and control of them shall be done effectively [1]. In the energy sphere, an increasing thrust is being given to migrate from the traditional grid to a smart grid, which through greater intelligence can enforce policies, monitor and self-heal whenever necessary. This requires an integration of a multitude of grid components. The increasing deployment of smart meters and their connectivity through the Home Area Network (HAN) to a central server is one such example. Many services shall be offered by processing this data, the central to it being the Electric Load Disaggregation or Non-Intrusive Load Monitoring (NILM).

Electric load disaggregation can be viewed as an instance of the broad class of source separation problem. The motivations are multi-fold with different stake holders: energy cost savings for a household, value addition for utility companies' service offerings, a positive step for a green campaigner in the fight against climate change and so on. A number of recent works ([2], [3]) have pointed out the potential impact of appliance level consumption in reinforcing energy conservation among users. However, load disaggregation is an ill-posed inverse problem and hence the attempts have been to obtain approximate solutions. The human element, the geographic diversity and other factors necessitate that the solution frameworks will depend on the problem scenario. The NILM problem has been actively pursued ever since Hart's seminal work [4]. However, many of the solution frameworks assume the availability of a number different electrical parameters to identify the appliances [5], [6], [7] [8].

Further, many of the evolving smart digital meter standards across the world limit the availability of electrical parameter to the active power only (see for instance, [9]). Hence any solution framework to be useful in a mass market would likely to be constrained by the availability of only active power. In this paper, we discuss some preliminary results of a solution framework being developed for NILM, where only the active power is available. The preliminary results of the framework focuses on using the power ratings of the appliance as a motif to identify an appliance. We propose a Bayesian disaggregation algorithm assuming that the information about the household electric appliances is available. Since the assumption is not realistic in many scenarios, we wrap this algorithm with a learning algorithm to identify the appliance information from the history of power measurement of the household.

The paper is organized as follows. In Sec II, we discuss some of the relevant literature. In the Sec. III, we discuss the problem statement and the proposed Bayesian disaggregation algorithm. We also discuss the learning algorithm which is proposed to overcome the problem of unavailability of household appliance ratings. In Sec. IV, we discuss the results of the proposed approach on a simulated and an open dataset. We also discuss the learnings from this implementation. Finally, we make concluding remarks in Sec. V and point out to the directions of improvements and generalization.

II. Relevant Works

In the following, we briefly review the recent NILM literature that focus on using only the active power data as the input. One of the popular approach is to use generative models for appliances and their interconnections by means of probabilistic graphical models and using statistical inferences algorithms for disaggregation. Variants of Hidden Markov Models are prominent in this regard. In [10], the authors build a generic prior variant of a Hidden Markov model (HMM) based on appliance characteristics and train the model using the aggregate power data and then use an extended Viterbi algorithm to identify the loads present. In [11], the authors introduce a Conditional Factorial Hidden Semi Markov Model (CFSHMM), an extension to Factorial HMMs to capture the ON-time duration distribution and correlations to the conditions like time of day, etc. In this work, the authors restrict the appliances with comparable, but distinct power ratings since the focus is to emphasize on the gains by incorporating correlations of conditions into the model. In [12], the authors use an additive Factorial HMM to model the appliances and propose an Additive Factorial Approximate MAP algorithm for inference. In [13], an approach to detect appliances which can be modeled as a Finite State machines (FSM) is proposed. The events are detected and clustered and the clusters are used to train the FSM through Genetic Algorithm. A dynamic programming approach is then used to optimize the FSM.

Another popular approach is to use discriminative methods to classify the appliances based on the aggregate power data. In [14], the authors describe a context-aware automated Demand Response system, *Watzzup*, where disaggregation is a part. They use a discriminative approach for disaggregation where they discretize the appliance states and use Nearest Neighbour classifiers to obtain the disaggregation. A real-time disaggregation approach is proposed in [15] based on threshold for event detection and a *k*-Nearest Neighbour algorithm to classify the identified events as different appliances. An unsupervised disaggregation approach without learning is proposed in [16]. This is done by first clustering the steady state changes and then using a matching pursuit algorithm for reconstructing the power signal. A post processing in the form of coincidence detection on the results of clustering is done to detect two or more coincident events. In [17], a temporal motif mining approach is illustrated for unsupervised disaggregation by characterizing the stable power consumption events, instead of transients.

III. Algorithm

In this section, we discuss the methodology being proposed and their components. We first outline the problem statement, followed by discussing the Bayesian Algorithm used for the disaggregation. The Bayesian algorithm assumes the knowledge of the electric appliances, their ratings and parameters of stochastic models. We handle the realistic scenario of not having the knowledge of appliance ratings by proposing a learning algorithm to obtain an approximate value of appliances ratings. Also, to mitigate the need for prior probability of an appliance we propose an approach to dynamically estimate the prior probability of each appliance.

A. Problem Statement

We develop a disaggregation algorithm with the following problem description: Given the appliance knowledge in the form of ratings and probability of them being ON, obtain the most probable loads that are ON. Only the active power measurements will be available with a sampling rate between 1 to $1/10$ Hz. The learning algorithm is developed with the problem that given a history of power meter data in a household, identify the ratings of different appliance present.

B. Data Pre-conditioning

The disaggregation algorithm is sensitive to the accuracy of the power data that is fed in as the input. Hence we need to ensure that stray spikes and momentary power variations do not influence the algorithm results. The data pre-conditioning performs this by identifying the stable states and events in the raw active power measurement. The steps involved in identifying stable states and events shall be summarized as below:

- Identify stable states through a recursive use of *k*-means clustering. The recursion is to make up for lack of knowledge about the number of clusters as well as to ensure that the stable states identified are at least a minimum power threshold apart from each other. The latter is to guarantee that small variation in power data doesn't translate to different stable states.
- The differences between the ordered stable states correspond to events.

These stable states are used as the power input to the disaggregation algorithm. The stable events are used by the *a priori probability computation* module to obtain the prior probability of an appliance. The above steps are performed over data split into windows of a few hundred samples (depending upon the sampling frequency).

C. Bayesian Disaggregation Algorithm

We propose a Bayesian inference algorithm that are popular in different applications [18] for load disaggregation. The algorithm requires that the prior probabilities of individual loads, i.e. $P(L_1)$, .. $P(L_N)$ and the power ratings of the load (W_{L_x}) are available. The broad steps in the algorithm can be described as,

1) Identifying and eliminating impossible load combinations, i.e. $\bar{L}_a$, for $1 \leq a < 2^N$, where $\bar{L}_a$ refers to the load combination vector.
2) Computing the likelihood of the observation for each possible load combination i.e. $P(W = w|\bar{L}_a)$.
3) Obtaining the Maximum a Posteriori probability for each load combination i.e. $P(\bar{L}_a|W = w)$
4) Obtaining the Maximum a Posteriori probability for individual loads by marginalization i.e. $P(L_k = ON|W = w)$, for $k = 1, ..N$

1) Eliminating Load Combinations: For N loads, the total number of load combinations is 2^N (considering only two-states for appliances). Analyzing the likelihood of all the load combination will be complex and the computational requirement rises exponentially. However, since the appliance ratings are available, we can considerably reduce the possible load combinations, even after accounting for errors in power measurements and accuracy of appliance knowledge. We define lower and upper thresholds at every disaggregation step based as a function of observed power, $Th_{low} = f_l(w)$ and

$Th_{up} = f_u(w)$. A simple linear instance of the thresholds could be,

$$f_l(w) = 0.5w \;\; \& \;\; f_u(w) = 1.5w \tag{1}$$

For cases where load ratings differ by factors of hundreds, complex threshold functions would be required. Such a function will depend on the observed power rating, confidence on the power measurement, confidence on the knowledge of appliance ratings, etc. However, the results discussed in this paper use only a simple threshold. To identify the set of possible load combinations, we sort the appliance ratings such that the load combinations will have their aggregate power in an increasing order (without actually computing them). We then use a search algorithm to find the load combination corresponding to the upper and lower thresholds. Since this step is primarily to reduce computational complexity of the algorithm, we don't discuss the effect of this step in the results.

2) Likelihood Computation: For a given power measurement (i.e. w_{ob}), and the set of possible load combinations from the step above, we compute the likelihood of each load combination as the inverse of the 1-norm of the difference between observed power and the load combination power as,

$$P(\bar{L}_a | W = w_{ob}) = \frac{1}{|w_{ob} - W_{\bar{L}_a}|} \tag{2}$$

where, $W_{\bar{L}_a}$ is the total power consumed by the load combination $\bar{L}_a$. The resultant likelihood vector for possible load combinations are then normalized.

3) MAP of Load Combination: We compute the Maximum a Posteriori (MAP) probability of a given load combination as,

$$P(\bar{L}_a | W = w_{ob}) = P(W = w_{ob} | \bar{L}_a) \times P(\bar{L}_a) \tag{3}$$

and then normalize the MAP across load combinations. Here, $P(\bar{L_a}$ is the probability of load combination $\bar{L_a}$, which is obtained from,

$$P(\bar{L_a}) = \prod_i P(L_i)$$

by considering the appliance operations to be independent of each other. Here L_is constitute the load combination vector $\bar{L_a}$.

4) MAP of Individual Loads: By marginalization over the MAP of load combinations, we can then obtain the Maximum a Posteriori (MAP) estimate of the individual loads as,

$$P(L_x | W = w_{ob}) = \sum_{\sim\{L_x\}} P(\bar{L}_a | W = w_{ob}) P(L_x) \tag{4}$$

where L_x is the load to be marginalized and $\sim \{L_x\}$ refers to all the loads other than L_x.

D. Unsupervised Learning of Electric Appliances

Here, we propose a learning algorithm which would use the history of power measurement in a household to identify the possible ratings of the electric appliances. The components of the preliminary version of the learning algorithm involves the following steps:

1) Identifying states (stable power levels) and events (power transitions) in the power data, same as that in the preconditioning block.
2) The above steps are evaluated for a small window of samples and repeated on through the entire data on non-overlapping windows.
3) The appliance ratings identified over different windows are consolidated to merge closely placed ratings.
4) The consolidated events are checked for their positive and negative transitions. An approximately equal number of $+$ve and $-$ve events is taken as the sign of the rating corresponding to a possible appliance.

This will give the list of appliance ratings which change state in the training data. These appliance ratings are used as input to the disaggregation algorithm as the *known* appliance ratings.

E. A priori probability Computation

We had assumed in the disaggregation algorithm that the prior probability of an appliance being ON is known. We remove this assumption by a simple progressive learning. Initially, the disaggregation algorithm starts with each appliance given an equally likely probability. As the disaggregation progresses, the *a priori* probability of an appliance is considered as a weighted sum of confidence measures obtained in the previous few disaggregation steps as:

$$P(L_x) = \sum_i w_i [P_{conf}(L_x)]_{-i} \tag{5}$$

where, w_i are such that $\forall\ j, k \in \{i\}$ and $j < k \Rightarrow w_j > w_k$ with $\sum_i w_i = 1$. $[P_{conf}(L_x)]_{-i}$ is the confidence measure of L_x obtained during the ith disaggregation prior to the present instance. A typical case shall be with $i = \{1, .., 5\}$ and $w_i = 0.5^i$. In the results discussed in this paper, we use the confidence measure of only the last disaggregation instance (i.e. $w_1 = 1$). To make use of the event information, we modify the prior probability of those appliances whose ratings match close to the event detected, as:

$$[P(L_x)]_{-1} = P_+, \;\; \text{if } ||W_{L_x} - W_{EV}||_2 < f_1(W_{EV}) \tag{6}$$

for a positive event ($W_{EV} > 0$). For a negative event ($W_{EV} < 0$),

$$[P(L_x)]_{-1} = 1 - P_+, \;\; \text{if } ||W_{L_x} - W_{EV}||_2 < f_1(W_{EV}) \tag{7}$$

Where, P_+ (> 0.5) is a probability measure to boost the confidence measure of the appliance identified through the event. The function f_1 shall be similar to that of f_l and f_u is (1), but with a tighter bound. In the preliminary results discussed, $f_1(W_{EV}) = 0.1W_{EV}$ is used. However, as discussed for the case of f_l and f_u, a complex function will be required to handle different problem scenarios.

The overall framework in which the disaggregation algorithm is put to work is illustrated in the Fig. 1. In the results discussed the learning algorithm shall be used when the appliance ratings are not available.

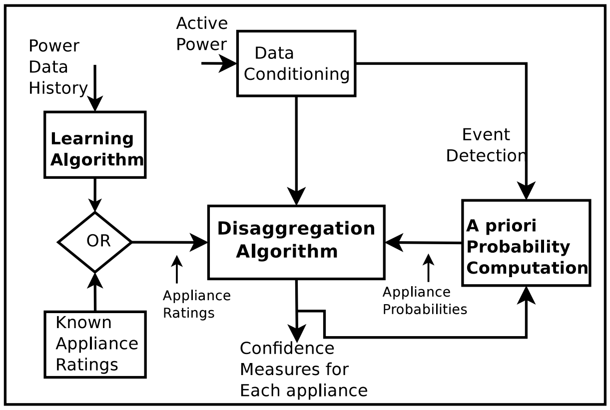

Fig. 1. Proposed framework for NILM

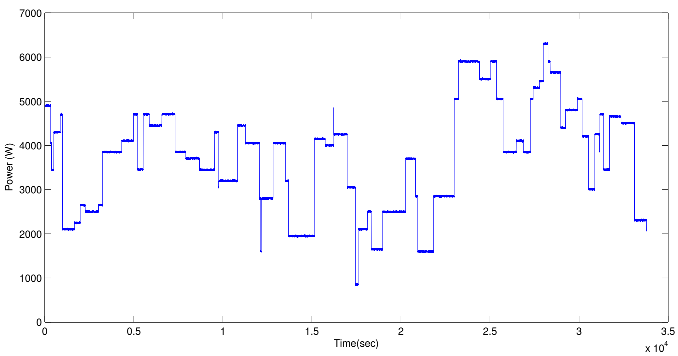

Fig. 2. A Sample of the Simulated Data

IV. Results

We illustrate the results of the disaggregation algorithm in this section on two datasets. One is from a simulated dataset and the other is the open Reference Energy Disaggregation Dataset (REDD) [19]. The simulated data was generated by choosing a set of appliance ratings and randomizing their transitions from one state to another, allowing at most one change at a time. A typical simulated data will look as in Fig. 2. A typical REDD dataset for a 1 day period is given in the Fig. 3. While REDD dataset has annotation for a number of appliances, it doesn't annotate all appliance and also a number of appliances have erratic appliance characteristics. The requisite test data is derived from the REDD dataset as the sum of annotated individual appliances for a period of 1 day. This ensures that only those appliances whose ratings are consistent are used. The learning algorithm uses these two datasets. Additionally, it also uses power mains measurement of different houses from REDD to illustrate the results.

A. Disaggregation with appliance knowledge

We discuss the results of the disaggregation algorithm here with the assumption that the appliance ratings are available. We benchmark the algorithm on the following parameters:

- TPR: True Positive Rate, the ratio of the number of positive identifications to the number of positives.
- FPR: False Positive Rate, refers to the cases when an OFF state appliance is identified incorrectly as ON. This is obtained as 1-TNR, where TNR refers to True Negative Rate, the ratio between the number of negative identifications to the number of actual negatives.

TABLE I
Disaggregation Results on Simulated data

Appliance	**Rating**	Sim-Case1 TPR	Sim-Case1 FPR	Sim-Case2 TPR	Sim-Case2 FPR
Appl 1	150W	1	0	0.97	0
Appl 2	255W	1	0	1	0
Appl 3	400W	0.92	0.17	-NA-	-NA-
Appl 4	600W	0.91	0	1	0
Appl 5	850W	1	0.15	1	0
Appl 6	1200W	0.95	0	1	0
Appl 7	1250W	0.83	0	-NA-	-NA-
Appl 8	2200W	1	0.07	1	0

TABLE II
Disaggregation Results on REDD dataset

Appliances	REDD-1 Rating (W)	REDD-1 TPR	REDD-1 FPR	REDD-2 Rating (W)	REDD-2 TPR	REDD-2 FPR
AC	-NA-	-NA-	-NA-	1000	0.64	0.09
Bathroom GFI	-NA-	-NA-	-NA-	1280	0.36	0.01
Heater	800	0.82	0.026	800	0.71	0.09
Kitchen Outlets	1050	1	0.04	1050	1	0.06
Refrigerator	190	0.91	0.08	190	0.97	0.17
Stove	400	1	0.05	400	1	0.08
Washer/ Dryer	2670	0.92	0.113	2670	0.89	0.06

The results of the disaggregation algorithm for the simulated data is given in Table I. The datasets have been chosen such that the effectiveness and limitations of the algorithm in its present form are illustrated. In the simulated data, the Case 2 had near perfect results because of distinct appliance ratings. In Case1, however, the close ratings in the form of 1250W and 1200W has brought down the TPR for both the appliances whose ratings are closely placed. Also the FPR of 850 and 400 went up because they together contribute to 1250 and hence are wrongly interpreted at some instances.

The disaggregation results on the REDD datasets is given in Table II. Similar to simulation case, when the REDD-1 had appliances with distinct ratings, near perfect results were obtained. However, REDD-2 dataset illustrates the case when appliance ratings are comparable. For instance, the bathroom GFI (Ground Fault Indicator) has a poor TPR because it occurs for a very short period during the day and is misunderstood to be some other appliance.

B. Learning Algorithm

The learning algorithm was run on different REDD datasets. The first two data sets are same as used for disaggregation (1-day each). This preliminary version of the learning algorithm is benchmarked on the following parameters:

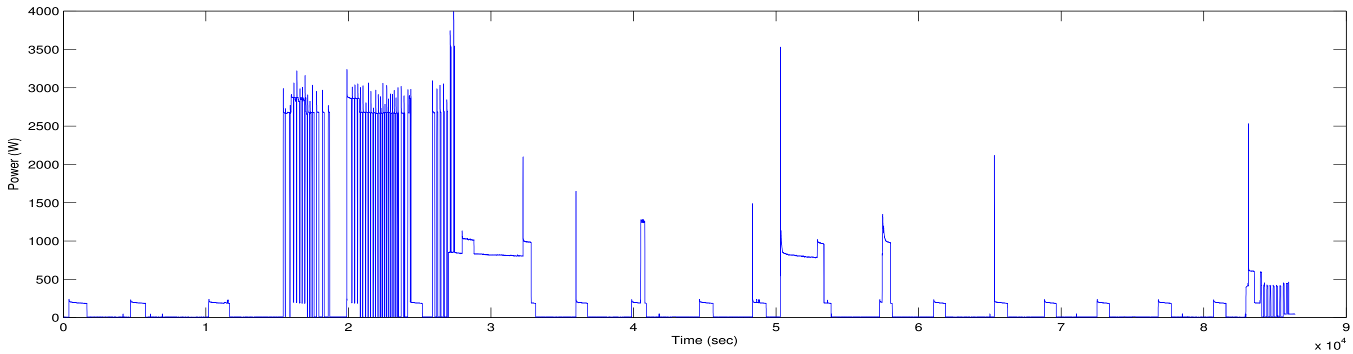

Fig. 3. A Sample of mains power from the REDD dataset for 1 day

TABLE III
RESULTS ON LEARNING ALGORITHM

Dataset Type ↓		No. of Appl.	TP	FN	FP
Simulated	Case 1	8	7	1	1
	Case 2	6	6	0	1
REDD-1 (1-day)		5	4	1	1
REDD-2 (1-day)		7	4	3	2
REDD-3 (13-days)		6	4	2	10
REDD-4 (18 days)		11	5	6	11

- TP (True Positive): Appliances identified within 10% of ratings and also escaped the rejection criterion.
- FN (False Negative): Appliances not identified within the 10% ratings or was rejected.
- FP (False Positive): Appliances Identified as being present, but are not present.

The results of the learning algorithm are illustrated in Table III for both the simulated data (for two different cases) and different REDD datasets. Almost all ratings were identified by this learning algorithm, however, not all of them passed the criterion for confirmation as an appliance (i.e. an FN). The reasons were multi-fold. A number of appliances didn't have matching positive and negative transients and hence these ratings were not identified as a single appliance. Also many appliances with closely placed ratings were not identified as individual appliances (in Case-1 of simulated data and in REDD-4).

C. Discussion

The algorithm results are comparable to that discussed in [12] i.e. the appliances are disaggregated with a greater accuracy when the appliance ratings are distinct and those appliances with closely placed ratings have a poorer results. Also the possibility of some load combination power matching another appliance also affects the disaggregation results. While this can be partly overcome by improving the prior probability of appliances, future improvements in the algorithm should incorporate temporal characteristics of the appliance power consumption. This will also enable the multiple states of an appliance to be identified as a single appliance instead of multiple appliances. It was also noted that the power variation in high rating appliances (say 2kW) can subsume the events in low power rating ones (say 0.2kW).

The learning algorithm is effective in identifying all events, but the False negatives and False positives were very high. Some appliances have different positive and negative event amplitudes due to their ON and OFF characteristics. These were identified as two different appliances. Also closely placed appliance ratings were invariably subsumed into one rating. Hence while there is a need to modify the appliance ratings learning, a better appliance identification approach is required to accept/reject a rating. As illustrated through the results, false positives grow in number as the amount of data over which the algorithm is run increases. Also the problem of mapping an appliance based only on the ratings is not addressed completely. To mitigate these problems, the use of temporal characteristics of the appliances' power consumption will be of considerable help and the work is underway in that direction.

V. CONCLUSION AND FUTURE WORKS

We have proposed a framework for load disaggregation using a Bayesian Inference algorithm wrapped with a learning module (which is pivoted around clustering). We have illustrated the results of the methodology and also discussed the limitations of the algorithms which indicate the directions in which we plan to take this work forward. One of the key improvements in the algorithm for future will be to identify appliances with complex waveform by incorporating the temporal characteristics of the appliance for learning and inference.

REFERENCES

[1] I. Smith (Ed). *The Internet of Things 2012: New Horizons*. Helifax, United Kingdom, 2012.

[2] J. Froehlich. Promoting energy efficient behaviors in the home through feedback: The role of human-computer interaction. In *Proc. HCIC Workshop*, volume 9, 2009.

[3] J. Froehlich, L. Findlater, and J. Landay. The design of eco-feedback technology. In *Proceedings of the 28th international conference on Human factors in computing systems*, pages 1999–2008. ACM, 2010.

[4] G.W. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[5] H.S. Matthews, L. Soibelman, M. Berges, and E. Goldman. Automatically disaggregating the total electrical load in residential buildings: a profile of the required solution. *Intelligent Computing in Engineering (ICE08) Proceedings*, pages 381–389, 2008.

[6] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, and S. Patel. Disaggregated end-use energy sensing for the smart grid. *Pervasive Computing, IEEE*, 10(1):28–39, 2011.

[7] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong. Power signature analysis. *Power and Energy Magazine, IEEE*, 1(2):56–63, 2003.

[8] A. Zoha, A. Gluhak, M.A. Imran, and S. Rajasegarar. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors*, 12(12):16838–16866, 2012.

[9] KEMA Consulting (for Enbin). Dutch smart meter requirements v2.2, 2008.

[10] O. Parson, S. Ghosh, M. Weal, and A. Rogers. Nonintrusive load monitoring using prior models of general appliance types. In *26th AAAI Conference on Artificial Intelligence*, 2012.

[11] H.S. Kim. *Unsupervised disaggregation of low frequency power measurements*. PhD thesis, University of Illinois, 2012.

[12] J.Z. Kolter and T. Jaakkola. Approximate inference in additive factorial hmms. In *15th International Conference on Artificial Intelligence and Statistics*, 2012.

[13] M. Baranski and J. Voss. Detecting patterns of appliances from total load data using a dynamic programming approach. In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, pages 327–330. IEEE, 2004.

[14] D. Rahayu, B. Narayanaswamy, S. Krishnaswamy, C. Labbé, and D.P. Seetharam. Learning to be energy-wise: discriminative methods for load disaggregation. In *Future Energy Systems: Where Energy, Computing and Communication Meet (e-Energy), 2012 Third International Conference on*, pages 1–4. IEEE, 2012.

[15] T. Alasalmi, J. Suutala, and J. Röning. Real-time non-intrusive appliance load monitor. In *Smart Grids and Green IT Systems. SMART-GREENS'2012. 1st International Conference on*, 2012.

[16] Hugo Goncalves, Adrian Ocneanu, and Mario Berges. Unsupervised disaggregation of appliances using aggregated consumption data. In *1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, August 2011.

[17] H. Shao, M. Marwah, and N. Ramakrishnan. A temporal motif mining approach to unsupervised energy disaggregation. In *Proceedings of 1st International Non-Intrusive Load Monitoring Workshop*, 2012.

[18] H. Wymeersch. *Iterative receiver design*, volume 234. Cambridge University Press New York, NY, 2007.

[19] J.Z. Kolter and M.J. Johnson. REDD: A public data set for energy disaggregation research. In *Workshop on Data Mining Applications in Sustainability (SIGKDD), San Diego, CA*, 2011.