Iterative Signal Separation Assisted Energy Disaggregation

Nilavra Pathak*, Nirmalya Roy* and Animikh Biswas†
*Department of Information Systems, University of Maryland, Baltimore County
† Department of Mathematics, University of Maryland, Baltimore County
nilavra1@umbc.edu, nroy@umbc.edu, abiswas@umbc.edu

***Abstract*—Providing itemized energy consumption in a utility bill is becoming a priority, and perhaps a business practice in the near term. In recent times, a multitude of systems have been developed such as smart plugs, smart circuit breakers etc., for non-intrusive load monitoring (NILM). They are integrated either with the smart meters or at the plug-levels to footprint appliance-level energy consumption patterns in an entire home environment. While deploying the existing technologies in a single home is feasible, scaling these technological advancements across thousands of homes in a region is not realized yet. This is primarily due to the cost, deployment complexity, and intrusive nature associated with these types of real deployment. Motivated by these shortcomings, in this paper we investigate the first step to address scalable disaggregation by proposing a disaggregation mechanism that works on a large dataset to accurately deconstruct the cumulative signals. We propose an iterative noise separation based approach to perform energy disaggregation using sparse coding based methodologies which work at the single ingress point of a home, i.e., at the meter level. We performed a ranked iterative signal removal methodology that effectively isolates appliances' individual signal waveform as noise on an aggregate energy datasets with moderate granularity (1 min). We performed experiments on real dataset and obtained approximately 94% energy disaggregation, i.e., disaggregated appliance-wise signal estimation accuracy.**

Keywords*—Green Building, Energy Disaggregation, Sparse Coding.

I. Introduction

Energy analytic feedback has vital effect on the specific usage of different appliances. Nowadays, the utility companies only provide the customers with an end of the month utility bill. It has been shown in [1] that an itemized consumption will give better understanding of energy usage and help save energy expenses in residential and commercial sectors. Itemized energy consumption can be monitored by the use of smart plugs, where one smart plug per device is required, which is of course an expensive solution and strategically infeasible to deploy at large scale. Alternatively, algorithmic approaches can construct disaggregated output of building's energy consumption that requires some prior knowledge about the characteristics of the appliances present in a home. In USA, smart meters have been installed approximately in 33 million homes. This instrument grants the utility providers with access to a large scale data repository. It helps perform not only energy disaggregation task but also different types of energy analytics, like comparisons within a community's fine-grained individual appliance power consumption to detect any malfunctioning appliance or abnormal usage pattern.

While disaggregation has been widely studied since its inception [5], only a few approaches have investigated the feasibility of deploying it at a large scale. The energy disaggregation method proposed in [9] used variants of Factorial Hidden Markov Model (FHMM) to learn energy disaggregation, but needed additional features like time of day, inter-appliance correlation etc., which makes it impractical for learning in case of large scale data, where such information may not be obtained or readily available, without excessive ground truth data collection. The probabilistic graphical based approach proposed in [9] works on a comparatively higher granularity (second-level) while we plan to look into a moderate granularity sizes, such as at the minute-level or higher, to make the disaggregation approach scalable across a variety of datasets differing in their veracity, volume and sparsity. A discriminative sparse coding based methodology has been proposed in [2] for low sample rate energy signal disaggregation, and applied structured prediction to create a discriminatively learned dictionary to minimize the disaggregation error. The performance of this discriminative sparse coding energy disaggregation approach degrades on data with a different granularity such as minute-wise.

In this paper, we look specifically at the task of disaggregating appliance energy consumption for data of a comparatively moderate granularity such as at *one minute* level. There are many appliances in regular households and commercial environments which are used not continuously but intermittently for a shorter duration of time. While high load appliances (refrigerator, HVAC, water heaters etc.) are generally used for a longer duration, a variety of low to medium load appliances (microwave, coffee-maker, oven, vacuum cleaner etc.) are typically used for a shorter duration. While the disaggregation algorithms designed with an hour (even 30 or 15 mins) level data granularity help capture the signatures and individual power consumption patterns of many continuously 'ON' appliances, but they fail to capture the signals from appliances with shorter usage duration. Non-detection of those appliances even if ignored, tend to add transient variations in the aggregate signal which could mislead the entire disaggregation process. Motivated by this shortcoming, we ask two basic questions:

- *Is there a middle ground in the data sparsity which may help capture a spectrum of appliances with both shorter and longer usage variations without incurring the installation of any special metering equipment?*
- *Can the overhead of building dictionary for discriminative sparse coding based approach be reduced to fully exploit its scalability capabilities?*

Our research attempts to build an effective energy disaggregation model which handles the data sparsity at the minute level and advocates an inclusive approach to generalize and promote the feasibility of energy disaggregation approaches at scale. More specifically, we propose an iterative energy signal separation methodology, which is motivated from two prior works - i) A wind noise reduction method using sparse coding [20], and ii) A hierarchical recursive discriminative sparse coding method for water disaggregation [17]. We propose an iterative appliance signal isolation method using sparse coding based approach to perform energy disaggregation on data with minute wise granularity. We first investigate the hierarchical recursive structure [17] to develop a heuristic for the structural signal decomposition and employ the wind noise separation approach to isolate each appliance's signal in every step. We learn appliance energy models using sparse coding based approach and perform efficient disaggregation on aggregated different appliances datasets potentially consisting of different feature vectors length reflecting the time series shapelets of their power consumption. We learn and selectively choose the smaller feature vectors depicting at least one cycle of power consumption differences among the appliances while being in 'ON' mode. Our method reduces the exhaustive sparse coding based learning of a large disaggregation dictionary which is time consuming and tedious when the set of appliances present are large in number. The contributions in this paper are listed as follows.

- **Sparse coding based approach for high frequency energy disaggregation:** We design this method particularly to disaggregate energy signals for higher frequency data. Earlier methods were mostly designed to work for low frequency signals [2].
- **Construction of the iterative decomposition structure:** We propose an iterative disaggregation algorithm that separates the signals incrementally and helps select which one to separate first depending on depending on our noise-separation assisted heuristic.
- **Efficient optimization techniques:** We employ efficient optimization techniques i.e L1 regularized non-negative sparse coding and non-negative matrix factorization based dictionary learning for fast disaggregation solution [18], [20].
- Finally, we perform experiments on our collected smart home and real dataset AmpDs [13] and show that our iterative dictionary aided discriminative sparse coding algorithm performs better than traditional sparse coding techniques.

The rest of the paper is organized as follows. Section II describes the preliminary concepts and elaborate the different approaches. Section III presents an overview of our iterative signal extraction based energy disaggregation framework and detailed description of our proposed methodologies focusing on each building block of the framework. Section V presents the related works. The simulation details and results are explained in Section IV and finally we conclude in Section VI.

II. PRELIMINARY CONCEPTS

We focus on the energy disaggregation via discriminative sparse coding based approach [2]. First we describe some of the notations and formulations before delving into the details of our energy disaggregation approach.

A. Notations

We begin with defining the disaggregation problem [2]. Given K different appliances or appliance classes, for every $i = 1, 2, \ldots, K$, we consider a matrix $Y_i \in R^{m \times n}$ where each column of Y_i contains a week of energy usage (measured every minute) for a particular house with a specific type of appliance. The aggregate signal is represented as $Y = \sum Y_i$. Each column of Y_i represents the power consumption for a period of time where each tuple contains an unit of measurement (unit depends on the data granularity or sparsity as in our case it is 1 min).

B. Sparse Coding Background

To apply the sparse coding approach for signal separation [20] we train separate models for each individual class Y_i and use these trained models to separate an aggregate signal. Sparse coding helps model the i^{th} data matrix using the approximation $Y_i \approx A_i X_i$ [1] where the columns of $A_i \in \mathbf{R}^{m \times k}$ contain a set of k basis functions, called the dictionary, and the columns of $X_i \in \mathbf{R}^{k \times n}$ contain the activation of these basis functions [18]. The sparsity condition when imposed on Non-negative matrix factorization (NMF) [3], it becomes the problem of Non-negative Sparse Coding i.e. where $A_i \geq 0$ and $X_i \geq 0$. The idea here is that the activation matrix should have to be sparse or in other words has to have a large number of zeros. Different sparse coding approaches have been proposed like the Fast Sparse Coding technique [18] and the Non-negative matrix factorization [3] where multiplicative updating rules have been used. Thus the problem can be defined as follows. If Y be the original data consisting of a mixture of K-components or signals such that $Y = \sum_{i=1}^{K} Y_i$, where Y_i are the individual components. Hence there are K dictionaries to be learned for the K different appliances.

Finally, the objective of the problem is to reconstruct the individual components. Now Y here can be represented by the product of two matrices A and X where A and X denote the dictionaries and codes respectively. A further restriction on X is that X should be sparse. Hence the problem reduces to the following sets of equations.

$$\arg\min_{A,X} ||Y_i - A_i X_i||_F + \lambda \|f(X_i)\|_1 \tag{1}$$

where $||_F$ is the Frobenius norm and $\|\|_1$ is the first norm.

The first part of the Equation 1 represents the reconstruction part and the second part captures the sparsity condition on the optimization problem. To solve this problem we need to construct the dictionary matrix A and train the sparse codes X. We briefly articulate an approach to solve this problem. Since A and X jointly are not convex, direct optimization is not possible and therefore we solve this optimization problem using alternating update [2].

[1] All matrices are in caps and vectors are in small letters. If A and B are two matrices then a usual matrix multiplication is denoted as AB and the entry-wise multiplication(i.e. Hadamard product)is denoted by $\odot$.

If $\mathbf{Y}$ be a dataset of dimension $m \times n$ i.e., $\mathbf{Y} \in \mathbf{R}^{m \times n}$, where m is the dimension of each element and n is the number of data samples, we attempt to find a representation of the form $\mathbf{AX}$ such that $Y = AX$ or $\|Y - AX\| < \epsilon$. This transformation makes the equations jointly non-convex, so we apply is alternative updates to solve the following two equations independently.

$$\arg\min_{X} \frac{1}{2}\|Y - AX\|_F^2 + \beta|X|_1 \quad (2)$$

$$\arg\min_{X} \|Y - AX\|^2 \quad (3)$$

$$\text{ST:} \quad a_{ij}^2 \leq 1 \;\forall\, j \text{ and } A \geq 0,\;\; X \geq 0$$

where A is the dictionary matrix which consists of the set of bases. The Equation 2 tries to reduce the difference between the reconstructed and the original matrix. The added $|X|_1$ is a penalizing factor which adds a penalty on high values of X thus maintaining the sparsity constraint. Each of the basis vectors in Equation 3 are enforced to have a unit norm.

C. Energy Disaggregation via Discriminative Sparse Coding

Energy disaggregation is the method for finding out the individual electrical signals from a cumulative whole house consumption. In other words we need to find Y_i such that $Y = \sum_{i=1}^{k} Y_i$. Energy disaggregation can be done by sparse coding with the assumption that A and X both are ≥ 0. The first part involves training the appliances individually which is done by sparse coding as described before, and then disaggregation is performed by using the following formulations.

$$\hat{X}_{1:K} = \arg\min_{X_{1:K} \geq 0} \|\bar{Y} - [A_1 A_2 \ldots A_K] \begin{bmatrix} X_1 \\ \cdot \\ \cdot \\ X_K \end{bmatrix} \| + \lambda \|X\|_1 \quad (4)$$

subject to the minimization of disaggregation error as calculated using Equation 5.

$$E = \sum_{i=1}^{k} \frac{1}{2} \|Y_i - A_i \hat{X}_i\|^2 \quad (5)$$

Now since the aggregate of all the individual signals is same as reconstructing all the bases, it can be replaced by the following Equation.

$$\hat{X}_{1:K} = \sum_{i=1}^{k} \arg\min_{X_i \geq 0} \|Y_i - A_i X_i\| + \lambda f(|X_i|) \quad (6)$$

The dictionary obtained is trained on the individual appliances, however they are not trained to perform disaggregation. The method employed to learn the dictionary for disaggregation is a structured prediction approach where the discriminative dictionary is updated using the following rules.

$$\tilde{A} \leftarrow \tilde{A} - \alpha((\bar{Y} - \tilde{A}\hat{X})\hat{X} - (\bar{Y} - \tilde{A}X^*)X^*) \quad (7)$$

where $A^* \leftarrow [A_1 A_2 \ldots A_K]$ and $X^* \leftarrow [X_1 X_2 \ldots X_K]$. The problem with this method is learning the discriminative dictionary as it becomes difficult if a large amount of dataset is not available for training.

D. Noise Separation

A non-negative sparse coding based approach to reduce the noise in the speech signal has been proposed in [20]. A unique feature of this approach is that it requires a source model for the wind noise but not for the speech. The objective was to remove the wind noise, which distorted the speech signal. $Y = Y_s + Y_n \in [A_s\ A_n][X_s\ X_n\ ]^T$, where Y_s and Y_n denote the source and noise models respectively. Initially the dictionary for wind noise is learned separately and then a modified non-negative sparse coding is used to remove the wind noise from the corrupted signal. We propose to use the similar methodology to model the individual appliances' signal instead of using arbitrary activation vectors based on sparse coding. We build a dictionary of different appliances power consumption patterns using this noise separation technique and store them in the dictionary to model the energy consumption patterns of each individual appliance. We employ those learned dictionaries to guide the energy disaggregation task instead of arbitrary learning the activation vectors in each new settings.

III. Iterative Signal Extraction Based Disaggregation Framework

We briefly describe our proposed methodology of iterative signal separation based energy disaggregation, in Fig1. The following Sub-Sections describe each aspect of the method. The disaggregation approach consists of two phases. First is the Appliance Model Phase where we learn the dictionaries of the different appliances by using non-negative sparse coding. The dictionaries are then fed into the Disaggregation Learning Phase. The first step of the Disaggregation Learning Phase is rank generation which is the order in which appliance signals are chosen to be extracted from the cumulative consumption. Appliances are then iteratively removed following that order and residual dictionaries are being learned. For any particular signal, once the removal has been done, the residual signal becomes the new cumulative signal and disaggregation is being performed for the next signal following the ranked order list.

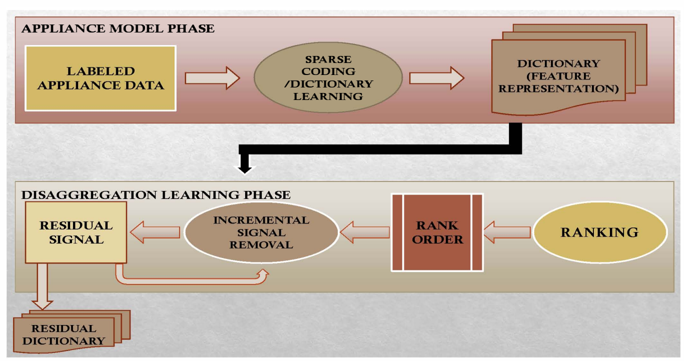

Fig. 1. Functional Overview of Our Framework

A. Sparse Coding For Learning Appliance Model

We segment the cumulative time series into time-segments of length being multiples of 60 instances (60 min). All appliances are not recognized well with a feature vector of size 60 as it might chunk an entire 'ON' signal which results in an incorrectly learned dictionary. For appliances having signals more than 60 or more than a multiple of 60, we take the next multiple of 60 as a feature vector length. For example, if an appliance has an 'ON' state of 75 min, we consider the feature vector length to be 120, i.e., m = 120 where m is $Y_i \in \mathbf{R}^{m \times n}$ and $A_i \in \mathbf{R}^{m \times k}$. The first stage of the sparse coding method described in Equation 2 and 3 used a modified version of the active sets method [18] called the Non-negative lasso algorithm which is similar to our proposed non-negative least squares Algorithm 1, but a L1 penalty is added to ensure sparsity in the activation signals. The second stage, the dictionary learning phase, is done by using the Multiplicative update method [3], which is given by:

$$A_j = A_j \odot \frac{\sum_i X_j^i [Y_i + (R_i^T A_j) A_j]}{\sum_i X_j^i [R_i + (Y_i^T A_j) A_j]} \tag{8}$$

where the dictionaries are normalized to have a column-wise unit norm and $R = AX$.

Algorithm 1 Active Sets method for Non-Negative Lasso

procedure ACTIVE-SET NNLS (Input Y_i $\forall$ i=1,2,...,K; K number of appliance)
  A is dictionary and X activations such that Y $\approx$AX
  Activated: $P \leftarrow \phi$
  Elements: $R \leftarrow \{1, 2, \ldots, K\}$ // the dimension of A
  X $\in$ x_i $\leftarrow$ zero-vector of length n
  w = A^T (y - Ax) + λa //Partial derivative across the dimensions
  while R $\neq \phi$ and max(w) $>$ tolerance **do**
    j $\leftarrow$ index of max(w) in w
    Add j to P
    Remove j from R
    $s^p = ((A^P)^T A^P)^{-1}((A^P)^T y - \lambda)$
    while min(s^P) $\leq$ 0 **do**
      α = min(x_i / (s_i - x_i)) $\forall$ i in P, s_i $\leq$ 0
      a = a + α (s - x) // α obtained by Line Search or pre-determined value
      Move to R all indexes j in P such that a_j = 0
      $s^p = ((A^P)^T A^P)^{-1}((A^P)^T y - \lambda)$
      s^R = 0
    end while
    x = s
    w = $A^T(y - Ax) + \lambda$
  end while
end procedure

B. Iterative Disaggregation order determination

One of the major problems of using sparse coding for time-series data is that blocking effect creates arbitrary alignments with underlying signal structure which may unnecessary increases the rank of a basis. Let us consider an appliance's with three 'ON' cycles as shown in Fig 2. Now if the signal is segmented in fixed lengths then there might be cases where the 'ON' cycle is segmented somewhere in the middle which would increase the rank of the basis. In this work we consider that the signals are segmented into features of length which are multiples of 60 (units) and thus cover the most informativeness part of each cycle and help for better basis learning. Convolutional sparse coding [21] relaxes this constraint and allows shiftable basis functions to discover a lower rank structure. However, the method itself is computationally expensive and thus we select most informative signal segments and employ selection-set so that most of the 'ON' characteristics of appliances remain intact within the signals. We propose to use general sparse coding which proves to be sufficient to perform disaggregation in this case.

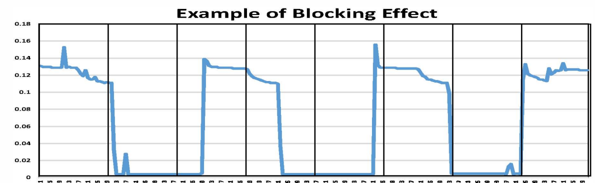

Fig. 2. Example of blocking results in a signal structure, this increases the rank of the dictionary matrix required to reconstruct the signal.

We determine the selection order by using appliance 'ON' state distribution as a ranking parameter. Let the set consist of K appliances $S = \{S_1, S_2, \ldots, S_K\}$ and each appliance has an 'ON' state distribution, i.e., $S_i \sim p_i(ON)$ with a mean μ_i. We perform an approximate subset-sum as follows. i) Form the list of appliances S ranked by the means, ii) Sort the list by descending order of probability of usage, iii) Select the highest mean less than Y and then subtract the mean from Y and mark the appliance as included, iv) Continue this process until all the appliances are included. This provides the ranked list of appliances to be separated. The selection strategy has been described in Algorithm 2.

Algorithm 2 Iterative Disaggregation order determination

procedure *Rank_Appliance_Selection* (Input $Y_i \in K$, K number of appliance, Y the cumulative signal)
  $S = \{S_1, S_2, \ldots, S_K\}$ and each appliance has an 'ON' state distribution, i.e., $S \sim p(ON)$ with a mean μ.
  Using the means perform an Approximate-Subset-Sum (Y, S, μ)
  Elements: $R \leftarrow \{1, 2, \ldots, K\}$ the dimension of A.
  Obtain the order for $S_{sub} = \{S'_1, S'_2, \ldots, S'_K\}$ in descending order of μ.
end procedure

C. Noise Removal based Disaggregation

Once we have obtained the sequence of appliances which are to be disaggregated we perform iterative signal removal. Given a sequence of appliances $S_{seq} = \{S_1, S_2, \ldots, S_K\}$our first objective of designing a scalable disaggregation algorithm is to identify the individual signals. It can be considered as a subset of the main sequence set S where $\forall S_i \in S \notin S_{seq}$ can be represented by zeroes in S. This assists in simplifying the signal removal sequence; without these steps dictionary learning would have been much more computationally complex.

We perform the disaggregation via signal removal as follows. Y is the cumulative signal and the signal to be removed Y_a and the remaining signal is Y_r, i.e., $Y = Y_r + Y_a$ where $Y = [A_r A_a][X_r X_a]^T$. The dictionary learning method is given as follows.

$$X_a = X_a \odot \frac{A_a^T Y}{A_a A X + l_a} \tag{9}$$

$$X_r = X_r \odot \frac{A_r^T Y}{A_r A X + l_r} \tag{10}$$

$$A_r = A_r \odot \frac{Y X_r^T + A_r \odot (1^{m \times m}(A X X_r^T \odot A_r))}{A X X_r^T + A_r \odot (1^{m \times m}(Y X_r^T \odot A_r))} \tag{11}$$

where X_a, X_r represent the activations and A_a, A_r represent the dictionaries respectively for the appliances and the rest of the appliances. The $\odot$ represents element-wise matrix multiplication. The sparsity factors for the appliance and rest of signal is represented by l_a and l_r respectively. This is performed iteratively to learn the reconstruction signal for appliances. Algorithm 3 describes the iterative disaggregation training pseudo code.

Algorithm 3 Iterative Signal Separation

procedure ITERATIVE SIGNAL SEPARATION (Input $Y_i \in K$, K number of appliance, S Set of appliances)
  Set S = $Rank\_Appliance\_Selection$(S)
  for i = 1 : K-1 **do**
    Randomly Initialise A_r
    $A_a = A_1$ where A_1 is the dictionary for S_1
    $S = S - \{S_1\}$
    while do$\|Y - AX\| > \epsilon$
      Update X_a using Eqn. 9.
      Update X_r using Eqn. 10.
      Update A_r using Eqn. 11.
      Columnwise normalize Ar to have a unit norm. i.e., $\|a_r\|_2 = 1$ where a_r is a column of A_r.
    end while
    Correct reconstructed signal $Y_a = A_a X_a$.
    Y = Y - Y_r
  end for
end procedure

The Algorithm 3 performs disaggregation training using the dictionaries for the appliances with cumulative consumption and the ranked structure as inputs. The appliances in the list is selected sequentially in sorted order for removal. Once the training is done a correction measure is performed on the extracted signal by removing all the erroneous values that fall within 20% of the estimated average of the appliance 'ON' state to the 'OFF' state mean consumption. This error is caused as the sparse coding-based disaggregation overestimates the significance of noise in the reconstruction phase, but this simple correction measure helps reduce the error to a large extent. Once one appliance's signal has been learned the next appliance's signal in the list is chosen and the residual signal becomes the cumulative signal in the next phase.

D. Testing Phase

During testing phase we used both the appliance dictionaries and reconstruction dictionaries along with the reconstruction structure. While testing, if we find that an appliance not likely to be on the list then we skip estimating that appliance power consumption. The pseudo code for testing is given in Algorithm 4.

IV. EXPERIMENTAL RESULTS

We conducted energy disaggregation on a wide range of datasets which are discussed as follows.

A. Data Set

We used the AMPds [22] which has a single home's data for about 2 years of time. We chose the most commonly used appliances for disaggregation and took the sum of the appliances to get the aggregate. For AMPds we performed the

Algorithm 4 Appliance Energy Estimation: Test Phase

procedure ITERATIVE SIGNAL SEPARATION (Input $Y_i \in K$, K number of appliance, S Ordered Set of appliances, A_R set of residual signal dictionaries, A_{App} set of appliance model dictionaries)
  Set Stest = $EstimateApp$(S) // Predict the set of active appliances
  for i = 1 : k-1 **do**
    if S **then**test(i) $\in$ S
      $A_a = A_1$ where A_1 is the dictionary for S_1
      $S_{test} = S_{test} - \{S_{test_1}\}$
      A = $[A_r \; A_{app}]$
      X = $[X_a \; X_r]^T$
      Estimate X_a using Eqn. 9.
      Estimate X_r using Eqn. 10.
      Correct reconstructed signal $Y_r = A_r X_r$.
      Y = Y - Y_r
    end if
  end for
end procedure

experiments with current component of the power signal only, as the fluctuation in case of current is much less when compared to real power; since real power is measured by measuring two factors - voltage and current and thus is more susceptible to fluctuations than current alone. We first show a distribution of the energy consumption of the different appliances. We avoid some of the very infrequently-used appliances as their contributions to the overall power consumption are minimal. The pie chart as shown in Figure 3 represents the energy consumption distributions of various appliance while Table I presents the name and abbreviation of all those appliances used in our experiments.

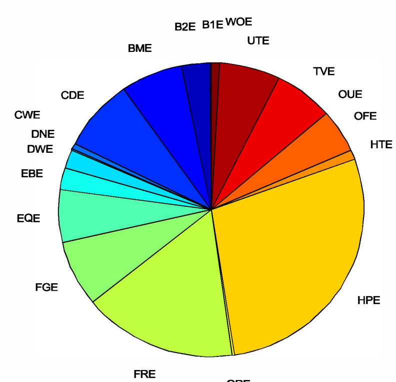

Fig. 3. Pie Chart of AMPds Data Set Energy Consumption

TABLE I. APPLIANCES IN THE DATASET

Dataset	Appliances Chosen
AMPds	Basement Plug Lights (BME), Dryer (CDE), Washer (CWE),Dining Plugs (DNE), Dish washer (DWE), Fridge (FGE) , Air Furnace (FRE), Heat Pump (HPE), Instant Hot Water Unit (HTE), Entertainment (TVE), Wall Oven (WOE)

B. Benchmark

We compare our results with the Combinatorial Optimization (CO) and Factorial Hidden Markov Model (FHMM) results presented in [24]. In [24] the simulations have been

TABLE II. APPLIANCE-WISE ENERGY DISAGGREGATION ACCURACY

Algorithm	CDE	CWE	DWE	FGE	FRE	HPE	WOE	BME	TVE	UNE
Iterative Signal Extraction	98.44%	93.83%	95.49%	87.30%	99.87%	98.82%	93.18%	57%	72.9%	87.33%
SMDA	97.9%	97.4	97.3%	55%	33.8%	84.7%	99.5%	77.0%	57%	11.3%

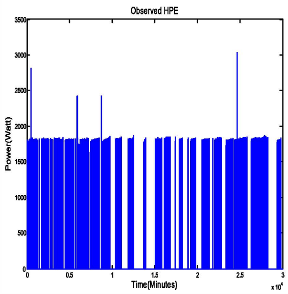

Fig. 4. Heat Pump Observed

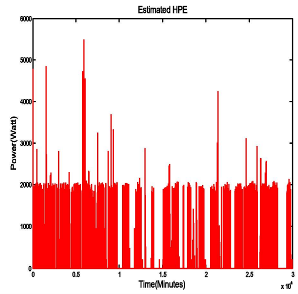

Fig. 5. Heat Pump Reconstruction

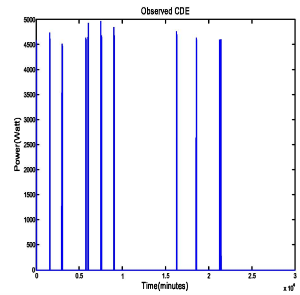

Fig. 6. Dryer Observed

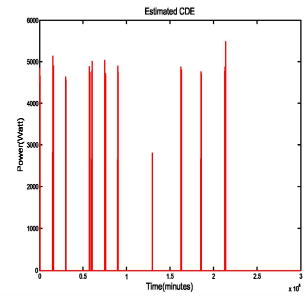

Fig. 7. Dryer Reconstruction

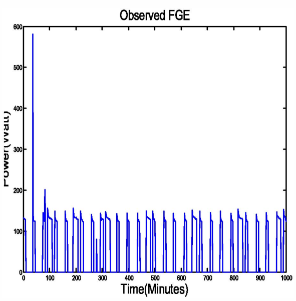

Fig. 8. Refrigerator Observed

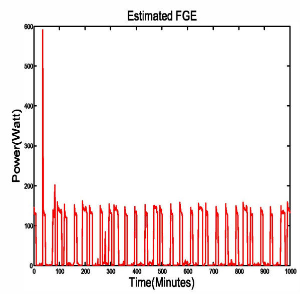

Fig. 9. Refrigerator Reconstruction

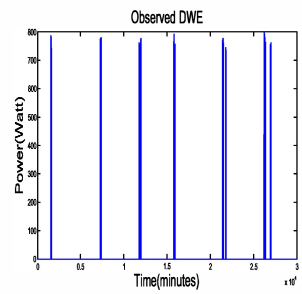

Fig. 10. Dishwasher Observed

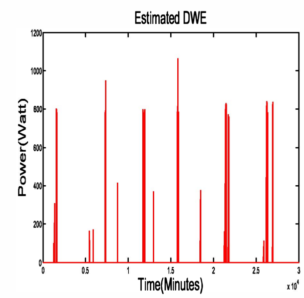

Fig. 11. Dishwasher Reconstruction

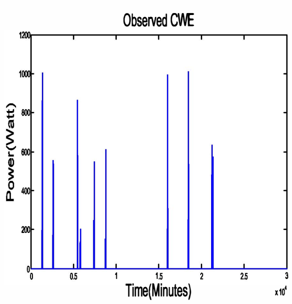

Fig. 12. Clothes Washer Observed

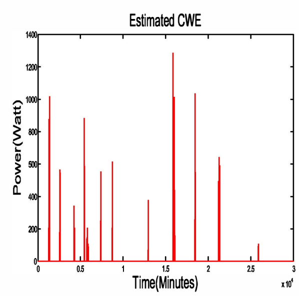

Fig. 13. Clothes Washer Reconstruction

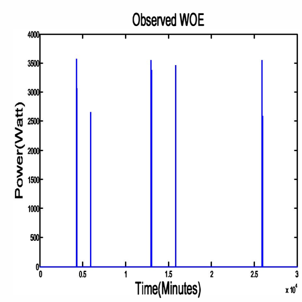

Fig. 14. Wall Oven Observed

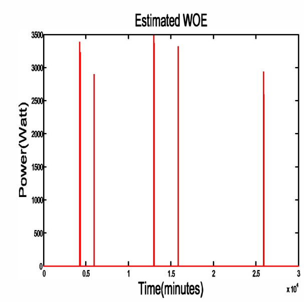

Fig. 15. Wall Oven Reconstruction

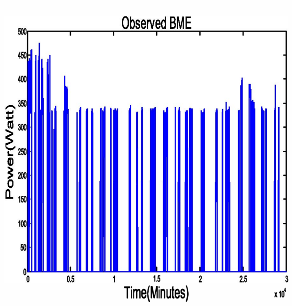

Fig. 16. Basement Plugs Observed

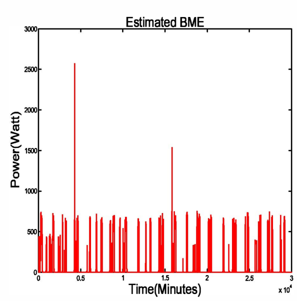

Fig. 17. Basement Plugs Washer Reconstruction

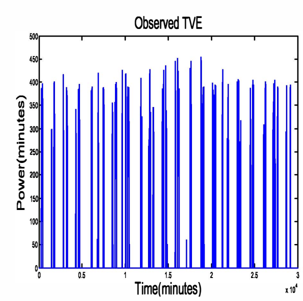

Fig. 18. TV Entertainment Observed

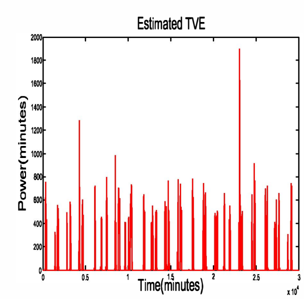

Fig. 19. TV Entertainment Reconstruction

done with a 1 minute granularity and for the top 95% energy consuming appliances. We also compare our simulation results with the single-measurement disaggregation algorithm (SMDA) proposed in [22].

C. Metrics Used

We used the disaggregation accuracy metric [12] - DisaggregationAccuracy $= 1 - \frac{\sum_{t=1}^{T}\sum_{m=1}^{M}|\hat{y}_t^{(i)} - y_t^{(i)}|}{2\sum_{t=1}^{T}\bar{y}_t}$. For appliance-wise accuracy calculation we used a modified form of disaggregation accuracy metric for the m^{th} appliance given by AccuracyAppliance(m) $= 1 - \frac{\sum_{t=1}^{T}|\hat{y}_t^{(m)} - y_t^{(m)}|}{2\sum_{t=1}^{T}\bar{y}_t^{(m)}}$. For comparative purposes we also used other metrics such as normalised error in assigned power (NEP), fraction of total energy assigned correctly (FTE) etc.[24]. Fig. 4 and Fig 5 show the observed and disaggregation signal of a heat pump respectively. Similarly Fig. 6, Fig. 7, Fig. 8, Fig. 9, Fig. 10, Fig. 11, Fig. 12, Fig. 13, Fig. 14, Fig. 15, Fig. 16, Fig. 17, Fig. 18, Fig. 19 depict the observed and reconstruction signals of dryer, refrigerator, dish washer, clothes washer, wall oven, basement plugs and TV Entertainment respectively. The sparse coding based approach not only tries to reconstruct the signals' contour but also manages to approximate the appliance level power consumption with very low error rate as shown in the above Figures.

D. Implementation Details

We implemented the algorithms in MATLAB2014 in a I7 CPU with 8GB RAM. We omitted the time of execution here as our aim was not to generate a faster solution but obtained an average training time of 30 min. However our optimizations runs in comparatively much lesser time than implementing in CVX toolbox. We generated the dictionaries for all the appliances a priori and then learn the best order for disaggregation. This order ascertains about the selection of appliance signal whose removal will result in the least reconstruction error. We determined the presence of an appliance by comparing the mean of the sample signal with the mean 'ON' time consumption of the appliance. We learned all the appliances using sparse coding and tested for the parameters by experimentation. Note that to avoid a blocking effect we learned some of the basis using time signal of length equal to multiples of 60 which helps capture the entire cycle of consumption patterns. During signal separation we adjusted the learning and disaggregation dictionaries accordingly.

E. Disaggregation performance

We show the comparative performance of disaggregation with the SMDA algorithm for appliance-wise accuracy for similar problem settings. Where disaggregation has been performed on the set of 7 appliances (CDE, CWE, DWE, FGE, FRE, HPE, WOE) and another with (BME, TVE, UNE) as abbreviated in Table I. Table II shows that appliance-wise accuracy is better in case of our proposed approach compared to single-measurement disaggregation algorithm (SMDA) [22]. We obtain the sequence pattern by using the Algorithm 2 as: HPE $\rightarrow$ CDE $\rightarrow$ FGE $\rightarrow$ DWE $\rightarrow$ CWE $\rightarrow$ WOE. Although FRE comes second in the list, we omit it and try to learn the other six, as the dictionary learning was easier to perform on them. Similarly the other list is BME $\rightarrow$ TVE $\rightarrow$ UNE. The accuracy metric we use incorporates the factors such as (a) the load being correctly identified, and (b) *ampere* amount in ground truth signal as in [22]. However our accuracy metric addresses the accuracy in terms of signal reconstruction values.

We assessed the results compared with the FHMM and the single-measurement disaggregation algorithm and our method proved to be more efficient in Table II. We compared the disaggregation accuracy between FHMM, CO and Iterative Signal Extraction approach on a specific set of appliances. We consider the medium to high load appliances as named in Table II which contribute approximately 95% of the total home energy consumption. In Table III, the comparative overall performances have been compared with the benchmark algorithms and we compare using the NEP and FTE metrics. A low NEP and a high FTE score is preferable in terms of performance and the results in Table III show better results in case of Iterative Signal Extraction. We also calculated the overall Disaggregation Accuracy of Iterative Signal Extraction is 94%.

TABLE III. COMPARATIVE RESULTS OF OVERALL PERFORMANCE FOR DIFFERENT ALGORITHMS

Algorithms	NEP	FTE
CO	2.23	0.44
FHMM	0.96	0.84
Iterative Signal Extraction	0.93	0.99

F. Qualitative Evaluation

Figures 4 to 19 show snapshots of the different appliances' signals ground truth and their reconstruction phase. Due to the huge size of the data plotting the entire time-lines compresses the graphs making it difficult to get better qualitative evaluation and understand visualization differences. Thus we chose to show some of the snapshots for smaller segments of time. We presented an observed and estimated signal construction phase for each appliance. Fig. 4 shows the observed original signal for the heat pump and Fig. 5 shows the reconstructed signal. In some cases there are spikes which arise as a result of the error in reconstruction which quantitatively have been found not to affect the accuracy by a large margin.

V. RELATED WORKS

The seminal work by Hart [5] introduced the field of NIALM. Hart provided his insights and goals into this field and also provided 'Total Load Model' which is the combinatorial optimization model and the 'Appliance Model' which is the basis for the extension of HMM-based models. In [26], the authors discussed the performance of steady-state and transient-state disaggregation methods. Transient loads have been used in many cases as a feature for disaggregation which help tackle the problem by using power harmonic and transient analysis in addition to steady-state disaggregation methods [7]. While the feature engineering gives one aspect of solving the problems, most of the methods rely on data of very high granularity which requires expensive instrumentation. For example in [28] acoustic sensors in unison with smart meters were used to perform context based energy disaggregation and in [29] where activities of daily living were considered to improve disaggregation performance. The other aspect of disaggregation is learning. A HMM based energy disaggregation has been proposed in [27]. An extended version of HMM based model using FHMM and its variants was proposed in [9] which used many other factors like time of day and inter-appliance dependence to improve energy disaggregation performance. The sparse coding based disaggregation approach [2] proposed a method to generate features in an unsupervised manner to perform disaggregation which alleviates the task of feature generation and can pave the way to realize a large scale disaggregation. Inherently this problem can be defined as a blind time-series signal separation method where the time-series values of individual wattage consumption needs to be extracted. However the appliance consumption has many factors like active and reactive power and hence better solutions can be found using multidimensional time series analysis. Also power being a product of voltage and current has more fluctuations, and hence randomness than current itself.

VI. Conclusion

In this paper we proposed an iterative sparse coding based signal extraction methodology for scalable energy disaggregation. Our approach works with a level of data granularity and sparsity which is inclusive across small, medium and high load appliances making it a feasible solution for smart energy disaggregation applications. In each step each appliance is treated like a noise signal and iteratively removed from the cumulative sum. We proposed a ranked structure heuristic which provides the order of signal removal and helps employ a noise removal technique to perform energy disaggregation.

We plan to extend our initial study and test it across various other datasets. While a sparse coding-based approach is more accurate when performing disaggregation on data which encapsulates specific patterns, our objective is to improve the performance on more sporadic appliance signals and in presence of arbitrary combinations of them in different households and commercial environments. While sparse coding based methods help to identify presence of appliances which are easier to learn and have more prominent patterns, our approach helps scale energy disaggregation at a lower granularity and computational cost. Finally, the ranked structure has been done using a heuristic method which we plan to formalize using Learning to Rank [25] techniques.

Acknowledgement

This work is supported by the NSF grants #1344990, #1544687, and Constellation E^2: Energy to Educate Grant.

References

[1] Carrie Armel, K.; Gupta, A.; Shrimali, G.; Albert, A. *Is disaggregation the holy grail of energy efficiency? The case of electricity.* Energy Policy 2013, 52, 213234.

[2] J. Z. Kolter, S. Batra, and A. Y. Ng *Energy Disaggregation via Discriminative Sparse Coding*, NIPS 2010.

[3] Julian Eggert and Edgar Krner, *Sparse coding and NMF* in Neural Networks, IEEE International Conference on, 2004, vol. 4, pp. 25292533.

[4] D. D. Lee and H. S. Seung. *Algorithms for non-negative matrix factorization.* In Todd K. ken. Thomas G. Diettench, and Volker Tresp, editors. Advances in Neural Informtion Processing System 13, pages 556562. MIT press, 2001

[5] G.W. Hart, *Nonintrusive appliance load monitoring* IEEE Proc. 1992, 80, 18701891.

[6] F. Sultanem. *Using appliance signatures for monitoring residential loads at meter panel level.* IEEETransaction on Power Delivery, 6(4), 1991.

[7] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong. *Power signature analysis.* IEEE Power & Energy Magazine, 2003.

[8] A Zoha, A Gluhak, MA Imran, S Rajasegarar. *Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey.* Sensors 2013.

[9] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, *Unsupervised Disaggregation of Low Frequency Power Measurements*. SIAM 2011.

[10] J. Z. Kolter and T. Jaakkola, *Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation*, AISTATS 2012.

[11] Rasmus Bro1 and Sijmen De Jong, *A fast non-negativity-constrained least squares algorithm*, Journal of Chemometrics Vol 11 Issue 5 pg 393 - 401. 1997.

[12] J. Z. Kolter, and M. J. Johnson, *REDD: A public data set for energy disaggregation research.* SusKDD 2011.

[13] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I. V. Bajic (2013). *AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research.* In Proceedings of the 2013 IEEE Electrical Power and Energy Conference (EPEC).

[14] Oliver Parson, Siddhartha Ghosh, Mark Weal, Alex Rogers. *Non-intrusive Load Monitoring using Prior Models of General Appliance Types.* In: 26th AAAI Conference on Artificial Intelligence. Toronto, Canada. 2012.

[15] Oliver Parson, Mark Weal, Alex Rogers. *A Scalable Non-intrusive Load Monitoring System for Fridge-Freezer Energy Efficiency Estimation.* In: 2nd International Workshop on Non-Intrusive Load Monitoring. Austin, TX, USA. 2014.

[16] Schmidt, M.N., Larsen, J., Fu-Tien Hsiao *Wind Noise Reduction using Non-Negative Sparse Coding*, Machine Learning for Signal Processing, 2007 IEEE Workshop.

[17] Haili Dong, Bingsheng Wang, Chang-Tien Lu *Deep Sparse Coding Based Recursive Disaggregation Model for Water Conservation*, Proceedings of the 13th International Joint Conference on Artificial Intelligence,2013.

[18] Lee, Honglak; Battle, Alexis; Raina, Rajat; Ng, Andrew Y. (2006). *Efficient sparse coding algorithms.* Advances in Neural Information Processing Systems.

[19] Rasmus Bro1 and Sijmen De Jong, *A fast non-negativity-constrained least squares algorithm*, Journal of Chemometrics Vol 11 Issue 5 pg 393 - 401. 1997.

[20] Schmidt, M.N., Larsen, J., Fu-Tien Hsiao *Wind Noise Reduction using Non-Negative Sparse Coding*, Machine Learning for Signal Processing, 2007 IEEE Workshop.

[21] Hilton Bristow, Anders Eriksson, and Simon Lucey. *Fast Convolutional Sparse Coding.* Computer Vision and Pattern Recognition (CVPR), 1(2), 2013.

[22] S. Makonin, F. Popowich, L. Bartram, B. Gill, and . V. Bajic. *AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research.* In IEEE Electrical Power and Energy Conference, Halifax, NS, Canada, 2013.

[23] C. Holcomb.*Pecan Street Inc.: A Test-bed for NILM.*In International Workshop on Non-Intrusive Load Monitoring, Pittsburgh, PA, USA, 2012.

[24] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh, Mani Srivastava, *NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring*, Fifth International Conference on Future Energy Systems (ACM e-Energy), Cambridge, UK. 2014.

[25] Hang LI ,*A Short Introduction to Learning to Rank*, IEICE Transaction of Information & System, Vol.E94D, No.10 October 2011.

[26] Norford LK, Leeb SB.*Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms.* Energy and Buildings. 1996;24(1):51-64.

[27] Oliver Parson, Siddhartha Ghosh, Mark Weal, Alex Rogers. *Using Hidden Markov Models for Iterative Non-intrusive Appliance Monitoring.* In: Neural Information Processing Systems, workshop on Machine Learning for Sustainability. Sierra Nevada, Spain, 2011.

[28] Nilavra Pathak, Md. Abdullah Al Hafiz Khan, and Nirmalya Roy. *Acoustic based appliance state identifications for fine grained energy analytics*, in Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom), March 2015.

[29] Nirmalya Roy, Nilavra Pathak, and Archan Misra. *AARPA: Combining Mobile and Power-line Sensing for Fine-grained Appliance Usage and Energy Monitoring*, in Proceedings of the IEEE International Conference on Mobile Data Management (MDM), June 2015.