Back to Basics: Simplifying Non-Intrusive Appliance Load Monitoring Using Combinatorial Optimization

Author1
Indraprastha Institute of Information Technology
India

Author2
Twentieth Century Fox
Springfield, USA
Email: homer@thesimpsons.com

James Kirk
and Montgomery Scott
Starfleet Academy
San Francisco, California 96678-2391
Telephone: (800) 555–1212
Fax: (888) 555–1212

***Abstract*—Non-Intrusive appliance load monitoring (NIALM) is the process of disaggregating the overall electricity usage into constituent appliances. In this paper we extend the Combinatorial Optimization (CO) approach for disaggregation, which was originally proposed in the seminal work on NIALM, in following two ways: 1) Breaking the problem into subproblems and reducing the state space; 2) Applying additional constraints backed by sound domain expertise. We evaluate our approach using REDD dataset and show practical problems which need to be solved while dealing with the dataset. We also propose a metric for evaluating NILM, which we believe overcomes many shortcomings of commonly used metrics.**

I. Introduction

- Motivate the importance of energy consumption in building
- Motivate that appliance level information is crucial - detailed feedback and optimized decision making [1]
- Challenges with getting appliance level information - introduce NIALM [2]
- introduce your proposed approach
- Enumerate the contributions

Primary contributions of our work are:

- Fill it up with 2-3 crisp points

Open source implementation of the proposed work is released for comparative analysis with other NIALM approaches as an IPython notebook[1]. We believe this is the first extensive release of a generic NIALM

II. Related work

NIALM has been well studied in the recent past and survey papers [3], [4], [5] present its classification across various dimensions. Following are three important classification dimensions:

- **Frequency of data collection**: Approaches such as harmonic analysis require data to be sampled at more than a thousand samples a second. Whereas approaches
- **Supervised/Unsupervised**:

[1] http://www.ipython.org

When you do the comparison, bring up how is your work different rather than just saying X did A and Y did B.

- Classification of different NIALM approaches - High/Low frequency, Time/Frequency domain analysis, supervised/unsupervised [3], [4], [5].
  For a more detailed overview the reader is referred to the above mentioned survey papers.
- Discuss the modeling approaches that are used
  - Additive Factorial HMM
  - Difference HMM [6]
- Datasets used: Recent datasets have spurred this field
  - REDD [7]
  - Blued [8]
  - Smart* [9]

III. NIALM

Discuss in brief the NIALM problem

A. Terminologies/ Notations

Borrow the notation used by Parson and Hart.

- Time slice: $t \in 1, ..T$
- Appliance: $n \in 1, ..N$
- Input: Aggregate power sequence: $x = \{x_1, .., x_T\}$
- Input: Ground truth power sequence for each appliance: $\theta^n = \{\theta_1^n, ..., \theta_T^n\}$
- Infer: Power draw by constituent appliance: $y^n = \{y_1^n, ..y_T^n\}$
- Each appliance has K (from k=1 to K)states and consumes μ_k power corresponding state
- Appliance state: $z^n = \{z_1^n, ..z_T^n\}$ where $z_i^n \in [1, ..K]$

B. NIALM using combinatorial optimization

This approach resembles subset sum problem and tries to minimize the difference of total observed power from the sum of various possible subsets coming from various combinations of appliances in different states. For each appliance we assume

K states and at a given time, an appliance can only be in a single state. This is given as: $z^n_{t,k} \in \{0, 1\}$ and,

$$\sum_{k=1}^{k=K} z^n_{t,k} = 1$$

The power drawn by n^{th} appliance in k^{th} state is given by:

$$\mu^n = \{\mu^n_1, ..\mu^n_K\}$$

Thus, CO can be formulated as:

$$z_t = argmin_{z_t} |x_t - \sum_{n=1}^{N} \sum_{k=1}^{K} z^n_{t,k} \mu^n_k|$$

Correspondingly the power draw by n^{th} appliance is given by: $y^n = \{\mu^n_{z^n_1}, ..\mu^n_{z^n_T}\}$

- Statespace is K^N
- We assign different loads to different mains, N_i loads to $Mains_i$, $\sum_1^p N_i = N$. Now different state spaces are K^{N_1}.... We can define the overall state space as $\max K^{N_i}$
  As a practical example, two mains, 20 appliance, state space before = 2^{20}. After = 2^{10}. Exponential reduction in state space.

Highlight what is the simplification you are bringing forth.

IV. Divide and Conquer NIALM (DaCo-NIALM)

In this section we explain the various steps involved in DaCo-NIALM which is shown in Figure 1.

1) **Downsample and align raw data**: While performing Combinatorial Optimization it is desired that transients and fluctuations in the power signal are filtered. The transients occur due to the high starting current of the appliance, whereas the fluctuations are a consequence of minor voltage fluctuations and oscillatory nature of loads. Figure 2a and Figure 2b show how starting current and voltage fluctuations can be filtered by downsampling. Further realignment amongst the appliance level data and mains level data is needed owing to different frequency of data collection and missing data.
2) **Assigning Loads to Mains**: This is the most important step of the algorithm and aims to identify the mapping between appliances and mains. Based on domain expertise we label the appliances in a home into background (loads which run independently throughout the day without user interference) such as refrigerator, and foreground (loads which are highly correlated with human usage) such as stove. Background loads are easier to detect since they are ON even during periods of low human activity such as night time. Thus, we first aim to assign background loads to different mains. Loads with higher mean power consumption are easier to identify and thus we sort background loads based on mean power in descending order. Starting from the head of this list (appliance having highest mean power consumption)

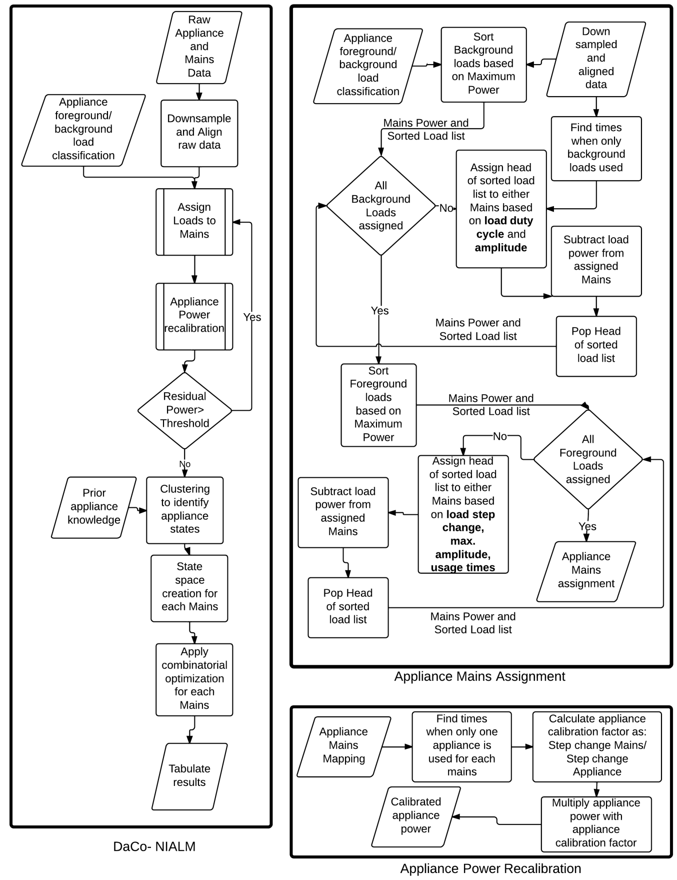
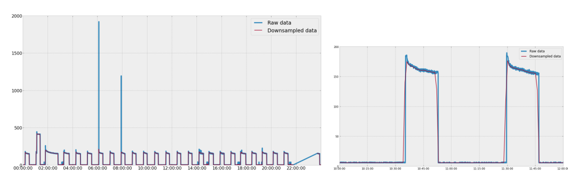


Fig. 1: Divide and Conquer NIALM

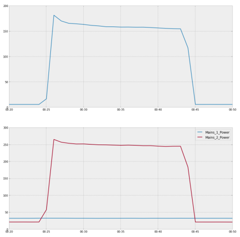


(a) Filtering startup transients (b) Filtering voltage fluctuations and oscillations

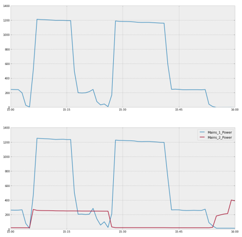
Fig. 2: Effect of downsampling appliance data

we iteratively attempt to do its assignment and once assign subtract this load from assigned mains to make further analysis easier. As a first check we see if the mean power of the appliance is greater than mean power of any mains for all time instances. If so, we can safely assign the appliance to the other mains. If this step is unable to provide conclusive evidence we look at the periodicity associated with such background loads during periods of low or no human activity (such as night time). Figure 3a shows how based on refrigerator duty cycle it is mapped to Mains 2. On similar lines assignment of foreground loads can be done. Figure 3b shows assignment of dishwasher to Mains 1, which is easy to do, since during this time window, mean dishwasher power is greater than mean Mains 2 power.

3) **Appliance Power Recalibration**: Since different



(a) Assigning based on duty cycle



(b) Assigning based on power threshold and step change times

Fig. 3: Assigning Refrigerator and Dishwasher to Mains 2 and Mains 1 respectively

hardware is used for measuring appliance and mains data there may be a need to calibrate the two. Since mains data is usually collected using better precision hardware, we keep mains data as a reference and calibrate appliance data against it. In practice we found appliance level monitors to usually provide only real power whereas the mains monitors can provide much more like reactive and active power. Like the previous step, time instances when an appliance in a particular mains is single used are identified. The ratio of mains and appliance power step changes occurring this window serve as the calibration factor for that appliance. Further each appliance power is corrected with the corresponding calibration factor.

4) **Clustering to identify appliance states**:
   a) Step changes occurring in Mains vs Appliances
   b) Isolating single appliance usage We use [10] to run our clustering
5) Appliance states identification using clustering (Possibly talk about unbalanced data, but leave it for future work)
6) State space creation
7) Applying CO for different mains
8) Find energy distribution by appliance and assign weights (To be used in results)

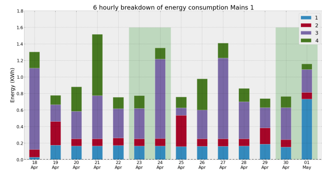
A. *Load assignment*

Draws inspiration from work by Parson et. al [6]. From prior knowledge we divide the loads into two different categories: Periodic such as refrigerator and non periodic such as Television.

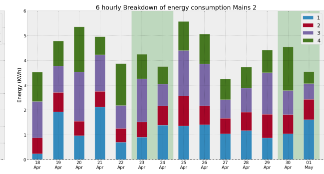
V. Evaluation

A. *About Dataset*

We use REDD dataset [7] for validating our algorithms. This dataset contains power and voltage data for mains (2 phases) as well as appliances from 6 homes in Boston area collected in the summer of 2011. The data is made available as raw, high frequency (sampled at 15 KHz) and low frequency (Mains at 1 Hz, appliances at .3 Hz). Considering the practical implications of residential smart meter installation, we believe that low frequency data represents the most realistic scenario and thus we use this data for analysis. Figure 4 shows 6 hourly



(a) Mains 1



(b) Mains 2

Fig. 4: 6 hourly energy usage breakdown Home 2

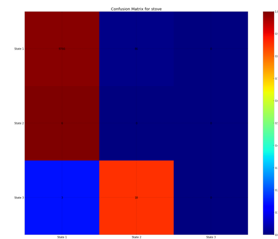


Fig. 5: Confusion Matrix showing predicted state accuracy for Stove

breakdown of energy consumption across the different mains in Home 2 .

B. *Evaluation Metric*

Commonly used metrics such as accuracy, sensitivity and specificity can be misleading when applied to NIALM. It can be seen from Figure 5 that since stove is mostly in state 0 (Off), accuracy will be largely decided by accuracy for this state, which is misleading, since it is easy to predict off states of appliances irrespective of the approach. Armel et. al [3] discuss the lack of a common metric while comparing NIALM approaches. We use the following metrics which have been used in the past work [6], [7] and were also suggested by Hart [2]:

- **Mean Normalized Error (MNE %)**: Normalized error in the energy assigned to an appliance over the test period, given by

$$\frac{|\sum_{t=1}^{T}\theta_t^n - \sum_{t=1}^{T}y_t^n|}{\sum_{t=1}^{T}\theta_t^n}$$

- **RMS Error (RE Watts)**: RMS error per time slice given by

$$\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\theta_t^n - y_t^n)^2}$$

- Image of confusion matrix
- Show confusion matrix and argue that accuracy can be misleading, state 0 dominates in all appliances and will most probably always be predicted correctly
- Hart pointed towards residual power as an evaluation metric, REDD paper talks about % of energy recovered

TABLE I: Calibration Factors, Mains Assignment and States

Appliance	Mains	States Power (W) Pre calibration	States Power (W) Post calibration
Refrigerator	2	7,162,423	9,210,423
Microwave	2	210,832,1730	10,832,1730
Lighting	2	9,96,156	10,110,178
Dishwasher	1	0,256, 1195	0,256, 1195
Stove	1	0,374	0,374
Kitchen	1	5,727	5,727
Kitchen 2	1	1,204,1032	1,204,1032

TABLE II: Mean Normalized Error and RMS error with and without DiCaCo NIALM

	Without Recalibration				With Recalibration			
	Without Load Division		With Load Division		Without Load Division		With Load Division	
Appliance	R.E. Watts	M.N.E. %	R.E. Watts	M.N.E %	R.E. Watts	M.N.E. %	R.E. Watts	M.N.E %
Refrigerator	136	109	71	32	130	95	59	21
Microwave	102	98	97	110	104	97	96	109
Lighting	51	164	48	195	44	83	38	60
Dishwasher	406	2947	63	100	377	2517	63	100
Stove	77	1191	36	281	75	1118	36	281
Kitchen	64	182	58	168	69	196	58	168
Kitchen 2	95	267	91	117	92	230	91	117
Overall	478	187	161	58	450	157	168	39

- METRIC 1: Given an appliance having m states, we propose power weighted appliance accuracy, as follows $ApplianceAccuracy = \frac{\sum_{i=1}^{m} Accuracy(i)*Power(i)}{Power(i)}$
- METRIC 2: In this we also take into consideration the amount of time an appliance is in a particular state
- Overall Accuracy : Based on energy weights of different appliance, for instance .4* Fridge +.2*Light+... This is important since it shows the relative importance of larger loads
- Switch continuity

C. Empirical Analysis

We analyze data from Home 2 of the REDD dataset and believe that the same analysis can be easily repeated across multiple homes.

- Train on first 7, test on last 7 days
- State assignment, Mains assignment in Table I
- Overall results in Table II, first column NILM without dividing into mains and without recalibration, last column with DiCaCo NIALM. Vast reduction in R.E. and M.N.E , especially for most appliance contributing most like refrigerator and lighting

VI. Conclusion

The conclusion goes here. We also provide mains load assignment of all 6 homes from REDD to further the research in this direction.

VII. Future Work

- Applying model on noisy datasets
- 2 D CO (when Real and Reactive Power are known)
- Factoring in Time of Day etc.
- Factoring in Appliance Correlation
- Factor in switch continuity, essentially leads to Factorial HMM
- Distributed NILM
- Adaptive Learning

Acknowledgment

The authors would like to thank TCS Research and Development for supporting the first author through PhD. fellowship. We would also like to thank NSF- DEITy for funding the project.

References

[1] S. Darby, "The effectiveness of feedback on energy consumption," *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, vol. 486, 2006.

[2] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[3] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, 2012.

[4] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012.

[5] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *Consumer Electronics, IEEE Transactions on*, vol. 57, no. 1, pp. 76–84, 2011.

[6] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in *26th AAAI Conference on Artificial Intelligence*, 2012.

[7] J. Z. Kolter and M. J. Johnson, "Redd: A public data set for energy disaggregation research," in *proceedings of the SustKDD workshop on Data Mining Applications in Sustainability*, 2011, pp. 1–6.

[8] A. Filip, "Blued: A fully labeled public dataset for event-based non-intrusive load monitoring research."

[9] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht, "Smart*: An open data set and tools for enabling research in sustainable homes," in *The 1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[10] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.