

PAPER • OPEN ACCESS

Non-intrusive Load Disaggregation Based on Kernel Density Estimation

To cite this article: Wang Sen *et al* 2017 *IOP Conf. Ser.: Mater. Sci. Eng.* **199** 012078

View the [article online](#) for updates and enhancements.

Related content

- [A Comparison of Generative and Discriminative Appliance Recognition Models for Load Monitoring](#)
Ahmed Zoha, Muhammad Ali Imran, Alexander Gluhak et al.
- [A novel CUSUM-based approach for event detection in smart metering](#)
Zhicheng Zhu, Shuai Zhang, Zhiqiang Wei et al.
- [Kernel density estimation of a multidimensional efficiency profile](#)
A. Poluektov



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Non-intrusive Load Disaggregation Based on Kernel Density Estimation

Wang Sen, Yang Dongsheng, Guo Chuchen and Du Shengxian

College of Information Science and Engineering Northeastern University, Shenyang
China

Sen.Zephyr.Wang@gmail.com

Abstract. Aiming at the problem of high cost and difficult implementation of high frequency non-intrusive load decomposition method, this paper proposes a new method based on kernel density estimation (KDE) for low frequency NILM (Non-intrusive load monitoring). The method establishes power reference model of electricity load in different working conditions and appliance's possible combinations first, then probability distribution is calculated as appliances features by kernel density estimation. After that, target power data is divided by step changes, whose distributions will be compared with reference models, and the most similar reference model will be chosen as the decomposed consequence. The proposed approach was tested with data from the GREEND public data set, it showed better performance in terms of energy disaggregation accuracy compared with many traditional NILM approaches. Our results show good performance which can achieve more than 93% accuracy in simulation.

1. Introduction

Various types of appliances are allowed to connected to grids with the development of smart grid, reasonable monitoring of home grid is of great importance to stable operation of power grids and energy saving. Research has shown that the knowledge of the power consumed by the individual appliances in the house can contribute to energy savings [1]. About 9% to 20% energy savings were observed in studies conducted (see e.g. [1][2]).

Recently, there has been growing interest in home energy management by non-intrusive ways. Non-Intrusive Load Monitoring tries to disaggregate power consumption of individual appliances in normal houses from the total power consumption detected by household meter [3]. One simple way to provide energy data would be to equip sensors to appliances one by one and collect using data. This intrusive monitoring method can predict the operating condition precisely with no doubt [4]. However, the high acquisition, installation [5] and communication costs keep it away from implementation. The same goal may be achieved by using of some intelligent algorithms like what NILM does. This is more convenient and becomes one research focus. Algorithm concepts about NILM can also be expanded to similar environments where detections about appliances are not convenient to implement.

1.1. Related work

Non-intrusive load monitoring was first proposed and developed by G. Hart in 1992[6], and many approaches were proposed after this. Normally, NILM needs to get features of appliances before decomposing the consumption data. High frequency approaches like harmonics [5][7], transients features [8][9] need to get appliances' usage data by high sample frequency (several kHz), and then



can get relatively high decomposing consequences, but using high frequency measurement requires installation of new and costly metering equipment which makes it remain unreasonable [10]. In contrast, low frequency approaches are more practical and meaningful, which just needs a sampling rate of 1Hz or less. Low frequency approaches are also easy to be installed to smart meters with low economic costs.

In recent years, many low frequency decomposing algorithms are proposed with the help of artificial intelligence, like Artificial Neural Networks [11], genetic algorithm [12][13], support vector machine [14][15], Hidden Markov Model(HMM) and its extensions [16][17]. HMM has an inherent benefit to describe NILM problem, the observation sequence in HMMs can represent power consumption, while the hidden states represent the working state of appliances. A new approximate inference method for energy disaggregation based on Factorial Hidden Markov Model (FHMM) was proposed by Kolter and Jaakkola [18]. FHMM combines several HMM chains in net and the observation is a joint function of these chains [19]. FHMM can be solved by Viterbi algorithm with relatively high computation cost. One disadvantage of HMM and its extensions is restrictions of HMM's basic assumptions, which brings some extra error to decomposing algorithm; HMM can't deal with multi-state appliances well not only because of increased states but inaccurate extracted features. HMM or its extensions still need to be developed more. There are some other unsupervised algorithms which uses methods like blind source separation [20], Kalman filtering[21]. These algorithms usually don't need training data and it's convenient to implement. Most of them are usually achieved based on probabilistic analysis methods [18].

There are also some works based on active power and reactive power [22][23]. They can achieve NILM by extracting unique features like slopes and edges of different appliances. The method will provide feature information of each residence [24]and may then be used to distinct different appliances from total consumption. Furthermore, algorithms that depend on real active power can be more practicable. In many of these works, reactive power changes play an important role and some other disaggregation ways are combined for better performance[25][26].

The main problem of low frequency decomposing approaches is that they cannot achieve very high accuracy. In reality, after so many years' research and development, although NILM has gotten great progress, it cannot get very high accuracy since more and more complicated home grid environment and limits of presented NILM ways. Now good low frequency approaches can achieve about 90% or a little higher accuracy [16], but it is still not enough to be applied to reality.

1.2. Main Contributions

In this paper, a new low frequency disaggregated approach using kernel density estimation (KDE) is developed. The algorithm can show good performance with multi-state power appliances at low-frequency sampling rates (1Hz). There are three main contributions in this paper:

- 1) A new way to extract appliance power feature based on KDE is explored. KDE is a useful non-parameter way for probability density estimation, probability density estimation is proved to extract appliances' power fluctuation feature well.
- 2) One useful way to model irregular multi-states appliances is shown by division of sub-states. Some appliances like notebook or television shows irregular power consumption, which adds great difficulty for NILM work. We solve this problem by dividing its irregular working state into several regular sub-states. And they can help algorithm achieve higher accuracy.
- 3) KDE based NILM algorithm shows good data fault tolerance, which means it can be implemented with power data approximately given before implementation and don't need to collect appliances power data in advance. Furthermore, one appliance dataset which has all the main appliances approximate power data can be established and the algorithm can then be used much more convenient.

Reference models will be established first which provides power information about appliances, and then the algorithm will search all the home working state at each period for main appliances' working states if appliances' working data are given or approximately given. Results from simulation tests by

Matlab using the GREEND dataset [27] show that the method is effective for NILM problem with more than high accuracy. Some future work is needed to develop the disaggregating algorithm more.

2. Energy disaggregation algorithm

2.1. Household and appliance models

2.1.1 Home Models. Home working state is defined by one multi-dimension vector, which is the combinations of all appliances' working states:

$$X = \{x_1, x_2, \dots, x_m\} \quad (1)$$

where x_i means the i -th appliances' working state, m is the total number of appliances considered in one house. For single-state appliances, x_i may be 0 when it's closed and 1 if it's turning on; For multi-states appliances, x_i is still 0 when it's closed, but it can be other numbers to express its different working states.

All the possible home working state set is shown below:

$$S = \{X_1, X_2, \dots, X_n\} \quad (2)$$

The house's real state is surely one element of the set above at each time. So the algorithm will search this total home working state set for NILM's solution if only how similar elements in the set and real power data can be estimated. In this paper, similarity is estimated with probability distribution by KDE.

However, we need to make sure search algorithm is executed with one home working state's data but not mixed two or more home working state's data. So target total power consumption data are divided into several groups of data, where only one home working state corresponds to one group of data.

Power fluctuation shows changes of appliances working states in a way. A step change in power or the transition of an appliance's operating state to another state is labelled as an event. Total power consumption can be divided into several working stages by step changes.

In this paper, different home working states are divided roughly according to median changes. A continuous length of points is chosen and their median is seen as the intermediate data's approximate value.

We found that the median of given data keeps relatively stable although home working state's power data fluctuates greatly. So we may divide different home working state stages by proper setting of power threshold after median filtering. The choice of threshold should obey the principle that 'Littler is better than Larger'. If threshold is chosen litter, divided stages may increase but don't influence algorithm's implementation only if the number of point in each stage is not very little. However, if the threshold is chosen larger and it may divide different home working state stages into one stage and then its distribution will not be matched for solution preciously.

Pseudocode of median filter and stages division is shown there:

Algorithm 1 Events Division

```
#Step 1: Median Filter
for i from 1 to length(observation)
  med(i)=median(observation(len_samp-1))
End for

#Step2: Events Detection
k=1;
for i from 1 to (length(med)-1)
  delta=|med(m+1)-med(m)|
  if delta>threshold
```

```

transition_time(k)=I
k=k+1;
End if
End for

```

2.1.2. Appliances Reference Models. Appliances' power data in different working states are needed to implement algorithm, so the algorithm establishes appliances reference models first. When one device is working at one state, its active power usually fluctuates near a stable value. So appliances' working states are divided based on their stable values, which are permitted with some fluctuations. Fluctuations are important because they contain individual information of each appliance.

For single-state appliances with stable consumption, division is easy to implement. For those multi-state appliances with stable consumption in each working state, it's also easy to divide states just according to stable values. It's hard to divide those multi-state or single-state appliances with large fluctuating power. So sub-states are generated for these appliances according to stable values.

Sub-state should meet some requirements:

- Power consumption keeps relatively stable or regularly distributed in each sub-state.
- Appliance's irregular working states can be approximately replaced by sub-state or its combinations.
- Sub-state's working time should be longer enough to provide appliance's working information.

The flowchart in figure 1 shows how to generate sub-states that satisfied requirements above. After dealing with target data by median filter, stages with large variance will be removed, and stages with similar mean will be merged, then the rest stages are what was needed.

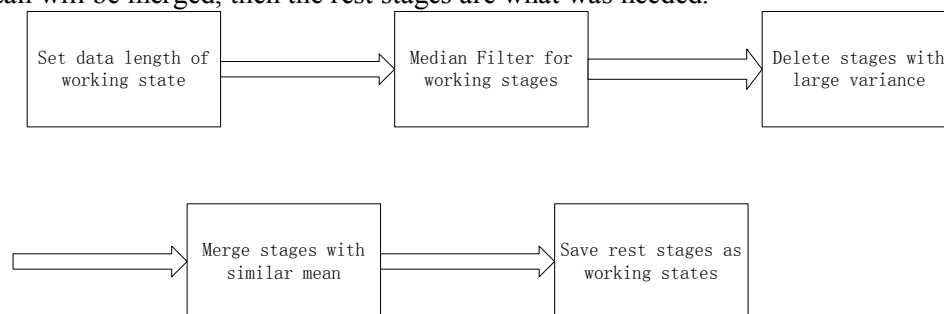


Figure 1. Flowchart about sub-states' generation

Figure 2 shows power consumption and division of sub-states. It can be seen that power consumption fluctuates from 8W to 80W when it's turned on with no obvious regularity. So sub-states are generated for its 'on' working state. After median filter, there are several states that keeps very stable and consumption of notebook in each time period can approximately be replaced with six sub-states as the third part of figure 2 shows. Six sub-states are shown with different colours.

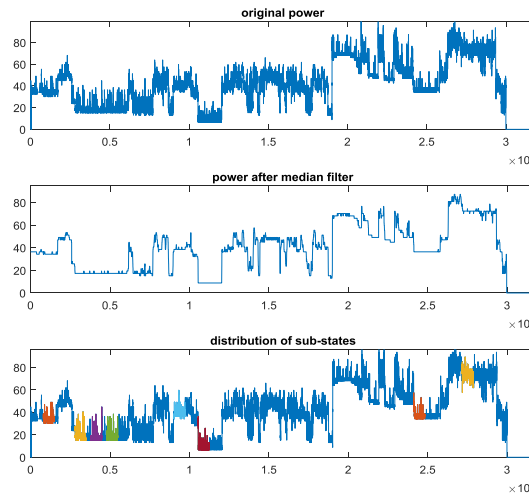


Figure 2. Power Consumption and working state division of notebook

Sub-states division may increase appliances' working states and computation cost, but it includes more information about devices and then help algorithm perform better. Irregular appliances provide more power features by division of sub-states, but these sub-states will all be shown as the original working state in decomposing consequence since there are not many differences between them. Algorithm will distinct appliance's different working states but not sub-states.

2.2. Kernel density estimation

2.2.1. Kernel density estimation (KDE). Kernel density estimation (KDE) is usually used to estimate the probability density function (PDF) of a set of random data. When given a set of one unknown variable's sampling data, we can inference probability distribution with KDE. Assume X_1, X_2, \dots, X_n is a set of observed values when one appliance is working at one state, PDF can be gotten if data size is large enough:

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (3)$$

where $K(\cdot)$ is kernel function—a non-negative function that integrates to one and has zero mean. $h > 0$ is a smoothing parameter called the bandwidth. h usually has a non-negligible influence on smoothing and bias for estimated PDF, so that appropriate selection of bandwidth is usually an important problem of KDE. However, in this algorithm, h does not have an obvious influence on the total accuracy, and a few h is selected to make estimated distribution non-negative.

KDE deals with a group of data not a single data each time, so it can weaken noise's influence, and show the distribution characteristic very well. Any kind of appliances including those with non-regularity distribution or fluctuation appliances can be deal with by KDE. This is also advantages of non-parameter estimation.

Apart from the stable value of one working state, the fluctuation data that appears frequently will also be captured by KDE to provide useful information. So distribution feature includes more information to disaggregated load states compared with many other NILM approaches.

After collecting appliances' individual working state's data, home working state's distribution is needed. Convolution transform is commonly used to combining some distributions. However, convolution transform means large computation cost since there are many appliances in home. So we collect data of each appliance's each working state to simulate a group of data of each needed

combination by addition operation, then KDE can be used to get each combination's distribution. At the same time, calculation time can be saved greatly by this way apparently.

2.2.2. Search for real state. After detecting events and dividing the target data into different groups of data, the algorithm will search for the most possible home working state in the total set. Since each stage of total consumption's home working state must be one element in the total set, we just need to estimate similarity of one group of target data and home working state.

After calculating distributions of target data and each home working state, we can directly compare similarity by numerical method since distribution has been normalized:

$$\text{Similarity} = \sum_{i=1}^N (f_{\text{obs}}(p_i) - f_{\text{home-state}}(p_i))^2 \quad (4)$$

where p_i means power of total consumption, N is number of total points used to generate discrete distribution, $f_{\text{home-state}}(p_i)$ and $f_{\text{obs}}(p_i)$ is values of probability density of home working state and observation data. This kind of comparison can be seen as figure 3 shows. Dotted lines mean discrete points of distribution. When two distributions are different, this comparison way can get the differences between these two distributions. The search algorithm will find the most similar distribution and take its state as observation data's real state. Generally, N can be set as one hundred points. Influence of N is shown in the Results section; it will not make big influence on total accuracy.

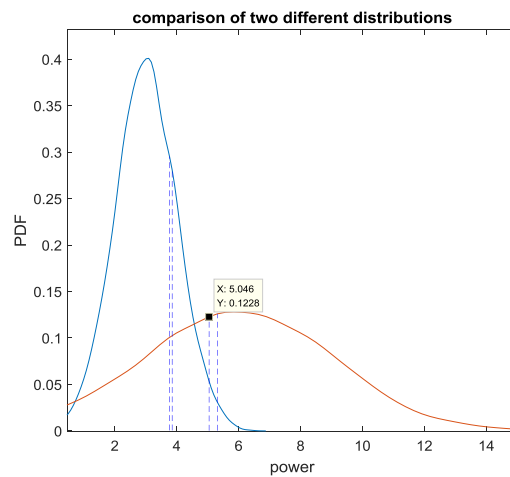


Figure 3. This figure shows differences between two distributions

2.3. KDE disaggregation algorithm

Our energy disaggregation approach can thus be summarized in the following three steps:

Step1: Collect home's main appliance's power data, and divide each appliance's power data into different working states or sub-states. Then we can generate home working state set for convenience of search algorithm.

Step2: For given total active power data of one day or other length of time, median filter is performed and then divide power data into different home working states by step changes.

Step3: Search for the most similar home working state in HWS set by comparison of probability distribution in each stage.

Figure 4 shows how the algorithm searches the most similar HWS for one specific house stage. After dividing target data into different stages, all the search work can be done in the same way.

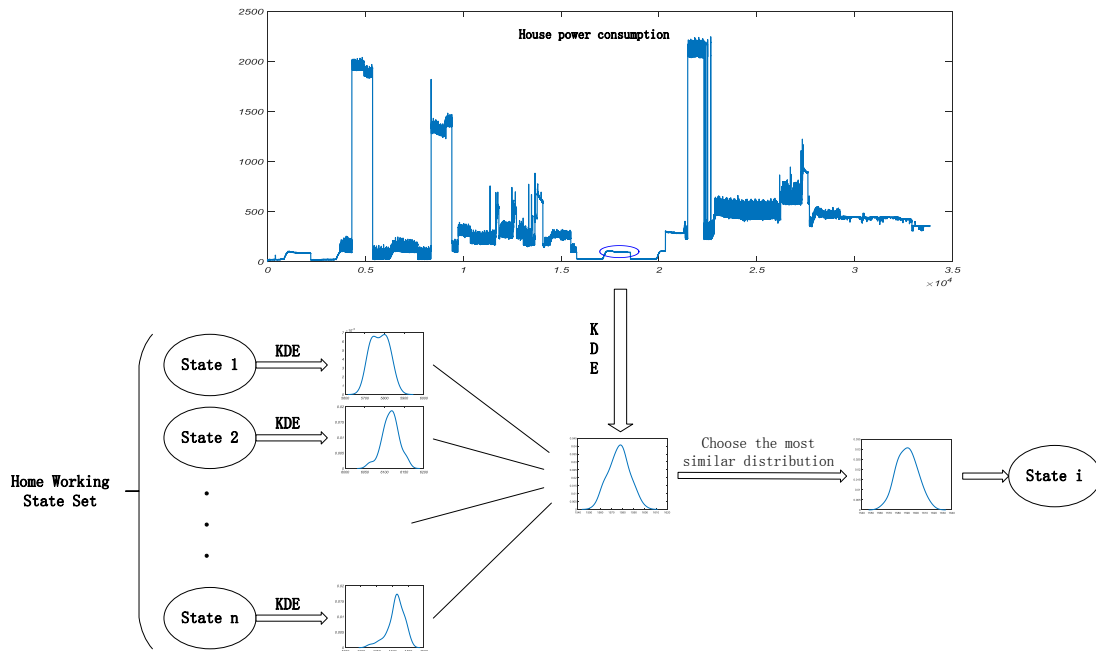


Figure 4. NILM algorithm based KDE

3. Evaluation

After introducing main idea and procedures of decomposing algorithm, some supplementary explanations about our algorithm's implementation and simulation are added in this section.

3.1. Evaluation setting

As the paper shows before, data of appliance's different working states is needed. Appliance's working data are deal to extract appliances' using information. Sub-state is established for those appliances that don't have stable power consumption on some work states. Although computation cost is larger if sub-states are considered in search algorithm, better composition performance can be achieved.

The algorithm can't distinct those low consumption appliances or those working time is very short very well. So those working power that is less than 3W are seen as closing work state. On the other hand, those consequences really contribute not much to energy management in one house.

A little prior knowledge is added to algorithm simulation since we usually can get some prior knowledge in one house in real world. Prior knowledge has been introduced before. One appliance or two with sub-states is set to be on all the time in House0 and House 2. This prior setting will also improve algorithm performance.

Algorithm tests were implemented with GREEND dataset. We chose House0 and House2 for our test's implementation with 1Hz sampling rate. House0 is a detached house owned by a retired couple who spends most of time at home. House2's residents are a mature couple (1 housewife and 1 employed) and an employed adult son. These two houses is just the representation of two typical kind of residents.

3.2. Evaluation about algorithm accuracy

In this paper, algorithm accuracy is evaluated based on identification between appliances' real working states and decomposed working states:

$$error_app = \frac{1}{T \cdot M} \sum_{t=1}^T \sum_{k=1}^M \left(\hat{s}(t,k) - s(t,k) \right) \quad (5)$$

where M is the number of appliances in house, and T is the length of target power data. Figure 5 shows differences between real working state and decomposed working state. When decomposed working states are different from real working states, it will be added into error states.

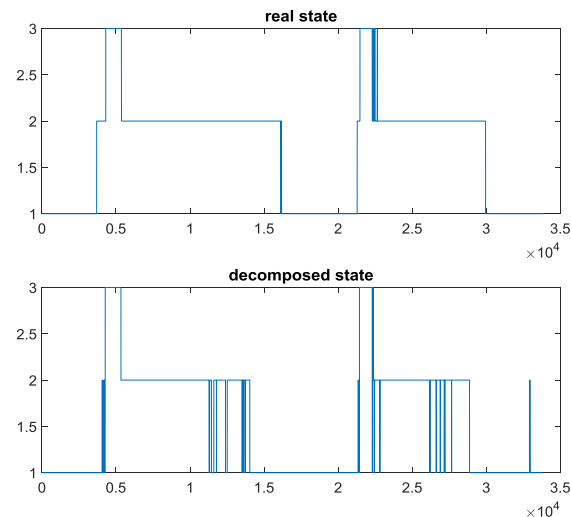


Figure 5. Comparison between the real working state and the decomposed working state of freezer in House0.

4. Results

To test the proposed approach on real data, the GREEND dataset was chosen to be detected, which has many compositions of appliances detected. All the appliances in dataset had joined the test. The used appliances and using time of the former family was less than the latter one, and consequently it showed higher accuracy. Table 1 shows the total accuracy while Table 2 shows different appliances' individual accuracy.

Table 1. Total accuracy of House0 and House2 in GREEND dataset

House0	House2
96.95%	91.09%

Table 2. Accuracy in appliance level of House0

Appliance	Accuracy
Washing machine	87.03%
Radio	100%
Water kettle	100%
Fridge	95.02%
Dishwasher	100%
Kitchen lamp	98.82%
Television	95.57%
Vacuum cleaner	99.79%

Table 3. Influence of different number of points in House0

points	Total Accuracy	Time(s)
50	93.17%	717

100	96.65%	760
200	96.67%	770
300	95.03%	894
400	96.83%	1010
500	94.94%	1198
800	96.95%	1463

Experiment consequences are tested with data of House0's 2013-12-07 data and House2's 2014-02-15 data. It can be seen that the total accuracy decreases as number of appliances in home increases. Although total accuracy did not increase strictly as number of points increases as expecting, differences are not very obvious. Computation cost is usually one of the major concerns when designing algorithms. It can be seen that our algorithm does not cost very much time to deal with thirty-three thousand data (9.5 hours' sampling data with 1Hz sampling rate) in two minutes.

5. Discussion

In the previous section, experiment of GREEND dataset was presented. Table 1 and Table 2 show that our algorithm can perform with good accuracy with real world data. KDE can extract more appliances' working information apart from stable value but fluctuation characteristic, so it shows more accuracy in the tests. There are many parameters influencing the performance of the algorithm, like the bandwidth of KDE, the number of points used for regeneration of home working state, the sampling rate and so on. We did not make tests about all of their influence because most of them will not influence algorithm accuracy very much, which convinces the algorithm's validity. However, if many parameters are not set appropriately, algorithm will not perform very well. Besides, carefully selected appliance's working data surely performs better than those selected carelessly do, which is not shown in Results section.

Table 3 shows the computation time with different number of points used to regenerate home working states. As the number increases from 50 to 500, (100 points are defaulted), the accuracy didn't change a lot, but time consumed by algorithm increases about two times. Generally, 100 points are enough to implement in reality.

The algorithm cannot distinguish those two different home working states both have very similar distribution. This is also the main problem of NILM algorithms. In these situations, accurate sampling of appliance's original working data becomes very important. If working environment has changed and make the distribution similar to another home working state, the algorithm still falls to get the real home working state. Further features such as reactive power measurements or some other variables can be added which may help the algorithm perform better, but more tests and considerations are needed in the future work.

6. Conclusion

In this work, a method based at kernel density estimation was presented. KDE can extract power data's fluctuation feature and distribution regularity very well. It can also weaken noise's influence on the data and NILM system, and achieve higher accuracy with complex home grid system. This algorithm does not need training procedure, neither power consumption data of appliances in the house to be very accurate, which means it is possible to evaluate a set of appliance's different working state data before, and our algorithm can be implemented without testing appliances' usage data. This makes the algorithm more convenient to implement, and more development will be achieved in the near future.

Reference

- [1] Parson O, Ghosh S, Weal M J, et al. An unsupervised training method for non-intrusive appliance load monitoring[J]. Artificial Intelligence, 2014,217(C):1-19.
- [2] Kim H S. Unsupervised disaggregation of low frequency power measurements[C]// Eleventh Siam International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona,

- Usa. 2011:747-758.
- [3] Basu K, Debusschere V, Douzal-Chouakria A, et al. Time series distance-based methods for non-intrusive load monitoring in residential buildings[J]. *Energy & Buildings*, 2015, 96:109-117.
 - [4] Jiang X, Dawson-Haggerty S, Dutta P, et al. Design and implementation of a high-fidelity AC metering network[C]// *International Conference on Information Processing in Sensor Networks, IPSN 2009*, April 13-16, 2009, San Francisco, California, Usa. DBLP, 2009:253-264.
 - [5] Inagaki S, Egami T, Suzuki T, et al. Nonintrusive appliance load monitoring based on integer programming [J]. *Ieej Transactions on Power & Energy*, 2008, 128(11):2742-2747.
 - [6] Hart G W. Nonintrusive appliance load monitoring [J]. *Proceedings of the IEEE*, 1993, 80(12):1870-1891.
 - [7] Jung D, Nguyen H H, Yau D K Y. Tracking appliance usage information using harmonic signature sensing[C]// *IEEE International Conference on Smart Grid Communications*. IEEE, 2015:459-465.
 - [8] Lin Y H, Tsai M S. Development of an Improved Time-Frequency Analysis-Based Nonintrusive Load Monitor for Load Demand Identification [J]. *IEEE Transactions on Instrumentation & Measurement*, 2014, 63(63):1470-1483.
 - [9] Thiruvanan T, Phung T, Ambikairajah E. Automatic identification of electric loads using switching transient current signals[C]// *Tencon Spring Conference*. IEEE, 2013:252-256.
 - [10] Aiad M, Peng H L. Non-intrusive load disaggregation with adaptive estimations of devices main power effects and two-way interactions[J]. *Energy & Buildings*, 2016, 130:131-139.
 - [11] Biansoongnern S, Plangklang B. Nonintrusive load monitoring (NILM) using an Artificial Neural Network in embedded system with low sampling rate[C]// *International Conference on Electrical Engineering/electronics, Computer, Telecommunications and Information Technology*. 2016:1-4.
 - [12] Chang H H, Chien P C, Lin L S, et al. Feature Extraction of Non-intrusive Load-Monitoring System Using Genetic Algorithm in Smart Meters[C]// *IEEE, International Conference on E-Business Engineering*, Icebe 2011, Beijing, China, October. DBLP, 2011:299-304.
 - [13] [Lin Y H, Tsai M S, Chen C S. Applications of fuzzy classification with fuzzy c-means clustering and optimization strategies for load identification in NILM systems[C]// *IEEE International Conference on Fuzzy Systems*. 2011:859-866.
 - [14] Figueiredo M, Almeida A D, Ribeiro B. Home electrical signal disaggregation for non-intrusive load monitoring (NILM) systems[J]. *Neurocomputing*, 2012, 96(3):66-73.
 - [15] Lin G Y, Lee S C, Hsu Y J, et al. Applying power meters for appliance recognition on the electric panel[C]// *Industrial Electronics and Applications*. IEEE Xplore, 2010:2254-2259.
 - [16] Egarter D, Bhuvana V P, Elmenreich W. PALDi: Online Load Disaggregation via Particle Filtering [J]. *IEEE Transactions on Instrumentation & Measurement*, 2015, 64(2):467-477.
 - [17] Mauch L, Yang B. A novel DNN-HMM-based approach for extracting single loads from aggregate power signals[C]// *IEEE International Conference on Acoustics, Speech and Signal Processing*. 2016.
 - [18] Kolter J Z, Jaakkola T. Approximate inference in additive factorial HMMs with application to energy disaggregation[C]// 2012.
 - [19] Ghahramani Z, Jordan M I. Factorial Hidden Markov Models[J]. *Machine Learning*, 1997, 29(2):245-273.
 - [20] Gonçalves H, Ocneanu A, Bergés M. Unsupervised disaggregation of appliances using aggregated consumption data[J]. 2011.
 - [21] Shaw S R, Laughman C R. A Kalman-Filter Spectral Envelope Preprocessor[J]. *IEEE Transactions on Instrumentation & Measurement*, 2007, 56(5):2010-2017.
 - [22] Laughman C, Lee K, Cox R, et al. Power signature analysis[J]. *IEEE Power & Energy Magazine*, 2003, 1(2):56-63.
 - [23] Belley C, Gaboury S, Bouchard B, et al. An efficient and inexpensive method for activity recognition within a smart home based on load signatures of appliances[J]. *Pervasive & Mobile*

- Computing, 2014, 12(3):58-78.
- [24] Esa N F, Abdullah M P, Hassan M Y. A review disaggregation method in Non-intrusive Appliance Load Monitoring [J]. Renewable & Sustainable Energy Reviews, 2016, 66:163-173.
 - [25] Farinaccio L, Zmeureanu R. Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the majorend-uses[J]. Energy & Buildings, 1999, 30(3):245-259.
 - [26] Baranski M, Voss J. Nonintrusive appliance load monitoring based on an optical sensor[C]// Power Tech Conference Proceedings, 2003 IEEE Bologna. IEEE, 2003:8 pp. Vol.4.
 - [27] Makonin S, Popowich F, Bajić I V, et al. Exploiting HMM Sparsity to Perform Online Real Time Nonintrusive Load Monitoring[J]. IEEE Transactions on Smart Grid, 2015, 7:1-11. 4.