# Estimating Power Loads from Partial Appliance States

Nicolas Roux*†, Baptiste Vrigneau*, and Olivier Sentieys*†
*Univ Rennes, CNRS, IRISA, 6 rue de Kerampont, 22300 LANNION, France
†Inria
Email: {firstname.lastname}@irisa.fr

***Abstract*—Knowing the plug-level power consumption of each appliance in a building can lead to drastic savings in energy consumption. Non-Intrusive Load Monitoring (NILM) is a method for disaggregating power loads in a building to the single appliance level, without using direct sensors or electric meters. This paper addresses the issues of NILM inaccuracy in the context of commercial and industrial buildings, by adding to the problem data from a low-cost, non-dedicated, smart sensor network. The SmartSense platform gathers environmental data and allows us to make an approximate guess on the states of some monitored appliances. The considered problem is the power estimation of each device states, subject to partial knowledge of the device states. The problem, formulated using linear algebra, is solved to estimate the power load values of these steady states on sliding windows of data with varying size. In this paper we show the principle and interest of the approach, its limited complexity, and its applicability to real datasets.**

***Index Terms*—Smart Buildings, Non-Intrusive Load Monitoring, Sensor Network, Convex Optimization.**

## I. Introduction

Developing smarter/greener electric grids has been an expanding field of research for the past decades. One of the essential requirements for energy utilities is the knowledge of power consumption patterns at single-appliance level. To do so without using an individual power meter for each appliance, Non-Intrusive Load Monitoring (NILM) consists in disaggregating electrical loads by examining the appliance specific power consumption signature within the aggregated load single measurement. Therefore, the method is considered non-intrusive since the data is collected from a single electrical panel outside of the monitored building. Thus, NILM has been a very active field of research for the past two decades with renewed interest in the last years.

A lot of research has been made on residential power disaggregation, due to the growth of home smart meters, and likely uniqueness of most relevant appliances. Nevertheless, industrial or commercial buildings present many difficulties for NILM techniques, such as the multiplicity of users and appliances or specific devices. Some companies already offer solutions to evaluate and reduce consumption of buildings and, recently, for personal residence. However, a key-point of NILM algorithms is the power estimation of each device states. Of course, a dispersion exists for a given type of load and it is obvious that a fridge will be different on each home. Moreover, it permits one to detect an abnormal behaviour and alert on a possible breakdown. In this work, we propose to apply numerical solver using linear algebra, formulated with partial knowledge of the device states. Information to solve the problem comes not only from the total power consumption, but also from low-cost wireless sensors. Indeed, the algorithm will be applied on our platform called SmartSense which associates NILM with various sensors.

The paper is organized as follows. Section II briefly introduces the state of the art. Section III formulates the problem, while Section IV presents the used numerical solving methods. Results are described in Section V before to present some methods to reduce complexity. Finally, Section VI draws some conclusion.

Please note that the work around this dataset is still in progress, and the results shown in this paper are yet incomplete. Further results will be provided at the time of the workshop and the complete dataset will be released soon.

## II. Related Work

Previous work on Non-Intrusive Load Monitoring have been initiated by G.W. Hart [5] more than two decades ago when he introduced the concept of measuring individual appliance consumption without relying on direct meters. Since, researchers strived to find ways to improve NILM accuracy and solve its limitations. Various approaches have been considered with interesting results. Zeifman [13] used a probabilistic approach to tackle this challenge. Pattern recognition approach has been studied in [4] for major end-uses in residential environments. Recently, machine learning methods, especially deep neural networks, have shown significant improvements in classification problems over the last few years and was applied to improve NILM in [3], [6]. Furthermore, disaggregation motives have been discussed in [1] with remarkable results on the potential annual energy savings due to an individual appliance feedback (more than 12%). This confirms both environmental and economical purposes of NILM.

On the other hand, the authors in [14] claimed that environmental sensing and additional heterogeneous information can be exploited to address some of the prevailing challenges faced by the current NILM techniques, however at the cost of increased complexity. Assuming that, load monitoring with extra information, other than the power consumption of the whole building, seems like the way to go. Moreover, with the increase in the presence of smart sensors in buildings for multiple purpose, collecting this kind of data without additional installation has become more and more plausible. This can explain the recent interest in this kind of approach. For example, in [2], the authors suggested incorporating time-of-day usage patterns into disaggregation algorithms, to improve accuracy and reduce computational complexity. We can observe that while [14] predicted that complexity would be increased with extra information, whereas some relevant information can directly reduce NILM algorithm execution time.

Furthermore, in [10], the author proposed several methods to improve NILM algorithm performance with the use of data extracted from sensors networks. Using these environmental sensors, and testing several algorithms, the authors concluded that monitoring a few appliances could drastically improve NILM performance. Detecting the state of an appliance with the adequate sensors can be a low-complexity task. For example, the operation of a workplace printer may be readily recognized with an audio sensor while lights can be easily monitored by sensors as well.

Assuming the state of an appliance is known, an interesting task is to estimate the characteristics of steady-state power of an appliance, on a length-significant power trace. In their work around ViridiScope [8], the authors implemented a power monitoring system by indirect sensing with self-learning automatic calibration of each sensor. Our objective is to propose a multi-purpose, low-cost sensor network to improve the accuracy of NILM algorithms with information about the environment rather than dedicated sensing.

## III. Disaggregating Appliance Power Loads

### A. Problem Statement

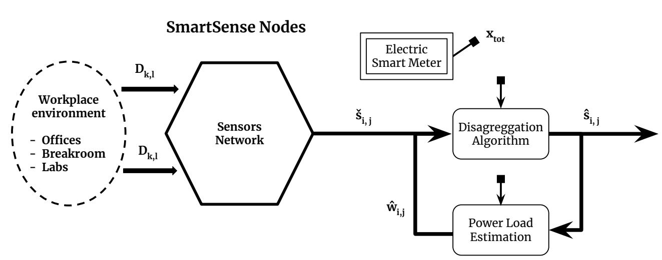


Fig. 1. Representation of the role of power load estimation in the full scope of the SmartSense usage for NILM

A basic problem of NILM is disaggregating the main power load of a building. We consider a low-rate sampling (typically 1Hz) and our approach is optimization-based. It consists in minimizing the difference between the main power meter output and the sum of disaggregated reconstructed appliances. We use the term disaggregated for an appliance whose steady-state temporal matrix is known, and steady-state power values in Watts (W) need to be estimated. The problem can then be written as

$$\min_{w_{i,j}} \ \|x_{tot}(t) - \sum_{i=1}^{N}\sum_{j=1}^{M_i} w_{i,j} \times s_{i,j}(t)\|_d \tag{1}$$

with

- $x_{tot}$ the main power of the building,
- $N$ the number of appliances,
- $M_i$ the number of steady states per appliance $i$,
- $w_{i,j}$ power consumption of appliance $i$ in steady-state $j$,
- $s_{i,j}(t) \in \{0,1\}$ a boolean equal to 1 when appliance $i$ is in state $j$, and
- $\|.\|_d$ a given norm. In this work we consider $l1$-norm or absolute value.

Note that the optimization-based NILM approach often tries to solve the mirror problem of (1)

$$\min_{s_{i,j}(t)} \ \|x_{tot}(t) - \sum_{i=1}^{N}\sum_{j=1}^{M_i} w_{i,j} \times s_{i,j}(t)\|_d \tag{2}$$

to estimate the states of the devices, knowing each individual power load.

To sum up, knowing partial information on the states of the appliance, we aim to estimate the power values of appliance $w_{i,j}$.

### B. Environment Description

To estimate the state matrix of an appliance, we use SmartSense, a fine grained, multi-modal sensor network platform. SmartSense is research platform based on a sensor network designed for various data acquisition. Each network node is composed of fifteen sensors to retrieve a wide range of information, including:

- Video sensors (Video-Graphic-Array (VGA) and Infra-Red (IR) cameras) and audio sensors.
- Radio sensors (2.4 GHz, Sub-GHz, and Ultra-Wide-Band) to sense the radio-frequency band occupancy and to estimate positions.
- Air quality sensors (temperature, humidity, carbone dioxyde concentration, air pressure, etc.).
- Light sensors (Ultraviolet (UV) and Red-Green-Blue-White (RGBW)).
- Distance sensor (telemeter).

SmartSense is composed of more than 150 nodes disseminated in rooms ranging from 9 to 40 square meters and corridors. The mesh of the sensor network is quite dense, with from two to four nodes in each room, according to the room surface. The frequency of the data acquisition will vary according to the data nature. For example, air quality sensor data do not need to be retrieved as frequently as audio or light sensor data.

Moreover, since the whole raw data cannot realistically be retrieved from the sensor nodes to the main system, some pre-processing is performed by the network node itself. Therefore, the pre-processing has to be chosen carefully, according to the data final use. Not all of the above sensors are relevant for NILM. The purpose of the SmartSense platform is to provide scientists with a wide diversity of data for research around the topic of Smart Buildings. Since SmartSense data is not yet available (the platform should be deployed in 2018), our algorithms have first been tested on the Reference Energy Disaggregation Data Set (REDD) [9] and we hope to extend

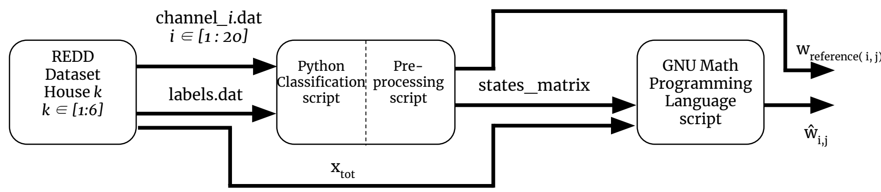


Fig. 2. The three steps of our experimental protocol to estimate power loads from main power trace and validating the results with individual power loads references

to other datasets such as the UK-DALE dataset [7] and the AMPds [12].

To sum up, a first estimation of the states matrix is obtained via an environmental sensors network which, after retrieving the data $D_{k,l}$ with $k$ the room identifying number, and $l$ the node sensor number in the room, gives us a first rough estimate of the monitored appliances states (see Figure 1).

## IV. Solving the Disaggregation Problem

In the case of NILM, SmartSense raw data will be used such as described schematically in Figure 1. The total power of the building $x_{tot}$ is read from a smart power meter sampled at roughly 1 Hz. The raw environmental data (audio, luminosity, etc.) $D_{k,l}$ is acquired by SmartSense nodes. For network load and privacy issues, this raw data is not retrieved from the sensor by the main system. Instead, each node processes individually a rough estimation of the state $\check{s}$ of the monitored appliances. Given $\check{s}$ and $x_{tot}$, a classic NILM disaggregation algorithm is run to determine a better estimation of the partial steady states $\hat{s}$. Given $\hat{s}$ and $x_{tot}$, the individual power load estimation of steady states determines $\widehat{w}$ for each monitored appliance, which improves the accuracy of another iteration of the previous step. A few steps can be made for each sample of $x_{tot}$, thus increasing accuracy at the cost of computation complexity.

As mentioned in Section III, our problem is based on studying the steady states of various appliances. To evaluate the accuracy of our algorithm, we use the experimental protocol depicted in Figure 2 and detailed below.

- First, the raw data are extracted from REDD. The *channel_i.dat* files contain time-stamped power metering information for single appliances or small circuit of same type appliances, for example a small lighting circuit or kitchen outlets.
- To define a reference for appliance steady states, we then process each appliance individually with a python script. To perform this, we use a custom 1-D edge detection algorithm for detecting steady states locations. Then, the whole individual power trace is clustered into a list of known steady states, to simulate the *a priori* knowledge of appliances states. Results of this experiment are collected and used as a reference for our further experiments, due to the fact that they are obtained from the individual power traces of REDD appliances, and thus are more accurate than disaggregated power characteristics. These results show that, although they are similar from one appliance (of the same function) to the other, identifying appliances solely on these steady-state values is not a trivial task due to their strong variations.
- Knowing the state matrix of studied appliances and main power, we use the GNU Linear Programming Kit (GLPK) [11] to execute an optimization script. To solve the optimization problem, we use the simplex algorithm, which performs best when upper and lower bounds on the optimization solution are known.

## V. Reducing Optimization Complexity

Since linear optimization can be a computing-intensive task and it is desirable to run the actions described in Figure 2 in real time, the complexity of the problem must be reduced as much as possible without affecting the accuracy of the power estimation or the disaggregation. Two methods are introduced to reduce complexity: windowing the power trace, and pre-processing the data according to the state matrix to remove redundant information.

### A. Windowing

To reduce the size of the optimization problem, and therefore to allow for the application to run in real-time, data are processed inside a time window. The first issue to address when using windowed data instead of the whole available dataset is how to take account of information from the previous windows when processing the optimization problem of the next window. To achieve this, in the GMPL optimization file, we set lower and upper boundaries based on the previous optimized results, and an arbitrary tolerance, which is typically set around $\pm 10\%$.

The next important step is how to choose the size of the window. Through our experiments, we noticed that the first window should be significantly longer, e.g. one day, while the next windows could be significantly lower, e.g. half an hour, without losing accuracy. This also allows for a dynamic tracking of characteristic power loads, which can be interesting for detecting quickly and pinpointing temporally a faulty appliance. Different window lengths and their corresponding execution times are detailed in Figure 3.

While shorter windows are useful for a dynamic monitoring of our appliances, a too short window can make it harder to observe all the states of an appliance and obtain the characteristic steady states values. Figure 4 collects the number of observed states, relatively to the window length.

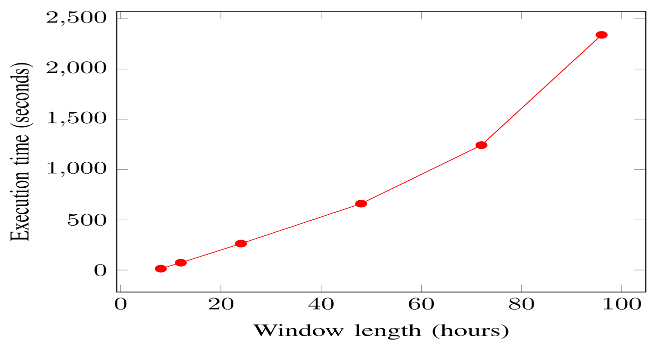


Fig. 3. Optimization script execution time depending on the window length run on an Intel(R) Core(TM) i7-6600U CPU 2.60GHz with approximately three seconds between each sample.

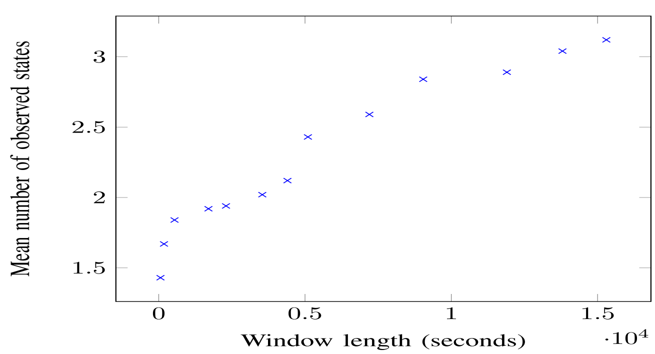


Fig. 4. Evolution of the number of states observed relatively to the window observation length for 100 random samples for various bathroom appliances. All appliances have a total of four different states. This hints that even a long window length cannot guarantee a complete observation of all the states of the appliances.

### *B. Data Pre-Processing*

Even when using small windows of data, computation can be sped-up by pre-processing the data. To perform this task, we scan the data window looking for duplicate samples, i.e. time samples with the very same state row, and a $x_{tot}$ similar power value with a very low tolerance, e.g., 1W. If a duplicate of a row is found, we remove the duplicate and increment a duplicate counter of the state/power value combination. This is a sort of compression of the states matrix, with a loose tolerance (1 W). Pre-processing is done after the computation of the states matrix (see Figure 2), and tends to reduce the size of this matrix with the intent to reduce both linear optimization computation time and the storage requirement of NILM. This pre-processing is the most time consuming task of our program, but the execution time is drastically decreased by this technique which is no more than a trivial duplicate search.

## VI. Conclusion

In this paper, we present ongoing work around Non-Intrusive Load Monitoring using a hybrid reverse approach *via* the SmartSense platform. Environmental information is used to infer the steady states of monitored appliances, and an optimization-based method is used to estimate the characteristic power loads of the steady states. While our method has been used on labeled datasets for the moment, future work will include the uncertainty around steady states, and processing of environmental SmartSense data.

## Acknowledgment

This publication is supported by the European Union through the European Regional Development Fund (ERDF), and by Ministry of Higher Education and Research, Brittany and Rennes Metropole, through the CPER Project SOPHIE - STIC & Ondes. The authors warmly thank Tomofumi Yuki for his help with the GNU Math Programming Language.

## References

[1] Carrie K. Armel, Abhay Gupta, Gireesh Shrimali, and Adrian Albert. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy*, 52:213–234, 2013.

[2] Chinthaka Dinesh, Stephen Makonin, and Ivan V Bajic. Incorporating time-of-day usage patterns into non-intrusive load monitoring. In *5th IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE SigPort, 2017.

[3] Pedro Paulo Marques do Nascimento. *Applications of Deep Learning Techniques on NILM*. PhD thesis, Universidade Federal do Rio de Janeiro, 2016.

[4] Linda Farinaccio and Radu Zmeureanu. Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses. *Energy and Buildings*, 30(3):245 – 259, 1999.

[5] G.W. Hart. Non-intrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, Dec. 1992.

[6] Jack Kelly and William Knottenbelt. Neural NILM: Deep Neural Networks Applied to Energy Disaggregation. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, BuildSys'15, pages 55–64, New York, NY, USA, 2015.

[7] Jack Kelly and William Knottenbelt. The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes. *Scientific data*, 2:150007, 2015.

[8] Younghun Kim, Thomas Schmid, Zainul M. Charbiwala, and Mani B. Srivastava. Viridiscope: Design and implementation of a fine grained power monitoring system for homes. In *Proc. Ubicomp09*, pages 245–254, 2009.

[9] J. Zico Kolter and Matthew J. Johnson. REDD: A public data set for energy disaggregation research. In *SustKDD*, 2011.

[10] X-C Le. *Improving Performance of Non-Intrusive Load Monitoring with Low-Cost Sensor Networks*. PhD thesis, University of Rennes, 2017.

[11] A Makhoren. Gnu linear programming kit, modeling language gnu math prog. *Department for Applied Informatics, Moscow Aviation Institute*, 2005.

[12] Stephen Makonin, Fred Popowich, Lyn Bartram, Bob Gill, and Ivan V Bajic. Ampds: A public dataset for load disaggregation and eco-feedback research. In *IEEE Electrical Power & Energy Conference (EPEC)*, pages 1–6. IEEE, 2013.

[13] Michael Zeifman. Disaggregation of home energy display data using probabilistic approach. *IEEE Transactions on Consumer Electronics*, 58(1):23–31, 2012.

[14] Ahmed Zoha, Muhammad Ali Imran, Alexander Gluhak, and Michele Nati. A comparison of generative and discriminative appliance recognition models for load monitoring. In *IOP Conference Series: Materials Science and Engineering*, volume 51:1. IOP Publishing, 2013.