

Small power load disaggregation in office buildings based on electrical signature classification

Conference or Workshop Item

Accepted Version

Rodriguez, A., Smith, S. T. ORCID: <https://orcid.org/0000-0002-5053-4639>, Kiff, A. and Potter, B. (2016) Small power load disaggregation in office buildings based on electrical signature classification. In: IEEE International Energy Conference, EnergyCon, 4-8 April, Leuven, Belgium, pp. 1-6. Available at <https://centaur.reading.ac.uk/66197/>

It is advisable to refer to the publisher's version if you intend to cite from the work. See [Guidance on citing](#).

Published version at: <http://dx.doi.org/10.1109/ENERGYCON.2016.7513984>

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the [End User Agreement](#).

www.reading.ac.uk/centaur

Central Archive at the University of Reading

Reading's research outputs online

Small power load disaggregation in office buildings based on electrical signature classification

Ana Rogriguez

The Technologies for Sustainable
Built Environments Centre
University of Reading
Reading, UK

Stefan Thor Smith

School of the Built Environment
University of Reading
Reading, UK

Alan Kiff

AECOM
Scott House
Basingstoke, UK

Ben Potter

School of Systems Engineering
University of Reading
Reading, UK

Abstract—This paper assesses the application of Non-Intrusive Appliance Load Monitoring (NIALM) methods for disaggregating electricity consumption in office buildings. The focus of the research is on small power equipment, which can represent up to 50% of the electricity use in buildings fitted with high efficiency building services. Research in this field has led to numerous algorithms being developed for use with NIALM systems, however, due to the highly variable nature of electrical appliances no suitable common characteristic has been identified for disaggregation. This paper presents an analysis of a set of electrical signatures based on transient and current-voltage phase shift during steady-state conditions for which subsets of the signatures are considered for identifying different small power loads. The ability of this approach to disaggregate appliance loads is demonstrated with the idea of applying disaggregation techniques during energy audits of office buildings.

Index Terms—Disaggregation, electrical signature, decision tree, overall accuracy.

I. INTRODUCTION

As buildings become more energy efficient, small power loads within those buildings become more significant to total energy consumption. These loads also represent a significant source of inaccuracy in energy breakdown audit assessments [1]. The potential of NIALM technology to enhance residential electricity audits has been identified as an efficient opportunity to spot electricity consumption measures in buildings, providing more accurate estimates of the energy consumption of the individual loads in a building at a reasonable cost [2].

NIALM methods were first developed by George Hart [3]. More recent developments in non-intrusive monitoring techniques have focused on the diversity in signatures and the disaggregation methods used for identifying appliances [4], [5]. However, a comparative study of the disaggregation capability of the different signatures has not been demonstrated. This paper develops a number of characteristic electrical signatures for a set of small power loads in office buildings and performs an analysis of the disaggregation obtained with each signature. The implementation of the presented method is intended to yield a proper understanding of the electrical signatures that better characterize small appliance loads in office buildings

and that could be used to create a low-cost monitoring tool to provide detailed energy consumption information.

II. NIALM FRAMEWORK

All currently used NIALM methods have several common principles. Firstly, a hardware installation is used to obtain a signal; then signal processing is used to extract specific signature characteristics; finally, a disaggregation algorithm is used to separate individual appliance loads from the overall signal [6].

A. Hardware considerations

The role of the sensing hardware is to acquire aggregated load measurement at an adequate sample frequency rate to capture the key load patterns and characteristics [7]. The electrical grid in the UK runs at a 50 Hz cycle, therefore, according to Nyquist-Shannon theorem, a minimum of 100 Hz sample rate would be required to capture basic wave shape [8]. The noise captured using frequencies higher than 15 kHz is likely to obscure any gains in signal detection for commercial buildings [9] and so sets a maximum sample rate. The 1 kHz sample rate range, by contrast, is of particular interest (although little work has been done in these frequencies), since it is at this frequency that transient features begin to be captured, but with no excessive high frequency noise. This is reflected by the large availability of commercial hardware capable of getting these sampling frequencies [9]. To coincide with available hardware for potential site install in energy audits, sample rates of 1 kHz are presented in this research.

B. Combination of electrical signatures

Using combinations of electrical features or signatures for load disaggregation, the heterogeneous nature of small appliances has historically made it difficult to standardize the parameters of algorithms used for signature identification [9]. In response to this difficulty, specific transient and steady-state electrical signatures of small power have been identified and their disaggregation capability analyzed in this study.

C. Machine learning techniques for disaggregation

After identifying electrical signatures of appliances a load-disaggregation process can be performed to associate proportions of load across time with different appliances. The

methodology used can vary with the nature of the signature analyzed and whether coded rule or automatic disaggregation by machine learning techniques are applied. For machine learning, supervised learning techniques have achieved more accurate results than unsupervised learning techniques [6]. In order to identifying which class, of a set of predefined classes, the data belongs to an established classification decision tree algorithm can be used, [10]. A database of electrical signatures with associated classifying attribute (appliance type) provides required training set. The decision tree classifiers resulting from decision-tree machine learning are used in this study as they are reported to obtain better accuracy when compared with other classification methods [11].

III. METHODOLOGY

Voltage and current have been monitored using a 1 kHz sample rate for a selection of typical small load appliances. A series of event detection and signature identification algorithms were developed for creating a database of appliance signatures, and a decision tree classification method was used for load disaggregation/appliance detection.

A. Data collection

A medium size office building has been chosen for the experimental part of the project. The monitoring equipment has been installed in two different areas and on two different classes of appliances:

- *Domestic kitchen small appliances:* Heater (H), Kettle (K), Coffee Machine (CM) and microwave (M)
- *Workstation small appliances:* Heater (H), Fan (F), Personal Computer (PC), Mobile Charger (MC) and Incandescent lamp (IL)

For safety reasons, no direct connections were made to the electrical circuits. A modified 15 A extension cord housing a current transformer and voltage probe was utilized. Data were collected first for individual appliance loads and then for aggregated loads.

B. Electrical signatures

The electrical signatures identified by algorithms developed to preprocess raw data have been classified as either transient or steady-state signatures.

Transient signatures: These signatures are derived from transitional properties that occur when an appliance is turned on or off [5].

- *Root Mean Square (RMS) increment:* the change of current magnitude during a transient.
 - The first step is to calculate the RMS current over a number of cycles m , using (1):

$$I_{RMS} = \left(\frac{I}{\sqrt{2}} \right)_{m \text{ cycles}} \quad (1)$$

Where I is the raw current value of the sinusoidal wave in Amperes.

- Secondly, an event $i \in \mathbb{N}$ is defined as followed:

$$\exists i \quad |\Delta I_{RMS}| > Thr_{inc} \quad (2)$$

Where ΔI_{RMS} is the differential between consecutive I_{RMS} , i is increased by 1 each time (2) is fulfilled and Thr_{inc} is a fix positive differential value of I_{RMS} .

- Finally, the *RMS increment* for each event i is defined as followed, for the ΔI_{RMS} accomplishing (2):

$$Increment_i = \Delta I_{RMS} \quad (3)$$

- *Settle time:* the time required for the current to reach a given threshold, Thr_s , of the final steady-state value, after a transient event i occurs.
 - The *Settle Time* I_{RMS} segment for each event i is defined according with (4):

$$\forall n \in \mathbb{N} \quad |I_{RMS}(t_{(n+1),i}) - I_{RMS}(t_{n,i})| > Thr_s \quad (4)$$

Where n is the sampling number, $t_{n,i}$ is the sample time in milliseconds after the event i and Thr_s is a fix positive differential value of I_{RMS} .

- The *Settle Time* time period in milliseconds is defined, as follow:

$$SettleTime_i = (t_{n_f,i} - t_{n_0,i}) \quad (5)$$

Where $n_{0,i}$ is the first and $n_{f,i}$ the last samples of the transient event i accomplishing condition (4).

- *Peak to Trough:* the absolute difference between the consecutive peak-trough couple in the raw current signal immediately following the transient event i , calculated by (6):

$$Peak \text{ to } Trough_i = I_{peak_i} - I_{trough_i} \quad (6)$$

Where I_{peak_i} is the first current peak and I_{trough_i} the first current trough after the event i .

Steady-state signatures: These signatures relate to unique properties of the steady state profile [5].

- *Current-Voltage phase shift:* To detect the current-voltage phase lag difference before and after a transient event i in the raw signals, the algorithm below has been created:
 - First, calculate the Discrete Fast Fourier Transform (DFFT) $X_{I,V}(e^{jw})$ of the different steady-state wave sequences $x_{I,V}[n]_s$, delimited by consecutive transient events $n_i < [n]_s < n_{(i+1)}$, $\forall s \in \mathbb{N}$, for the current and the voltage raw signal using (7):

$$X_{I,V}(e^{jw}) = \sum_{n=n_0}^{n_f} x_{I,V}[n]_s e^{-jw} \quad (7)$$

Where $x_I[n]_s$ and $x_V[n]_s$ represent a current and voltage raw signal sequence s respectively, n_0 is the first and n_f the final samples of sequence s and

$X_{I,V}(e^{jw})$ is a complex function of the angular frequency w of the signal that can be written as:

$$X_{I,V}(e^{jw}) = |X_{I,V}(e^{jw})| e^{j\theta(w)} \quad (8)$$

$$\theta_{I,V}(w) = \arg \{X_{I,V}(e^{jw})\} \quad (9)$$

Where $|X_{I,V}(e^{jw})|$ is called the *magnitude function* and $\theta_{I,V}(w)$ the *phase function*, with both being real functions of w .

- Using a filter, determine the 50 Hz *phase function* values, with 1% error according with the UK frequency control limits regulation [12], for the different current and voltage wave sequences s .
- Current-voltage lag is given by the difference between the current and the voltage 50Hz *phase function* for each sequence s :

$$Lag_s = \theta_I(w)_s - \theta_V(w)_s \quad (10)$$

Where Lag_s is measured in milliseconds.

- And finally, define the current-voltage phase lag difference before and after a transient event i expressed in degrees as follow:

$$\Delta Phase\ shift_i = |Lag_s - Lag_{(s+1)}| \cdot \frac{180}{\pi} \quad (11)$$

C. Disaggregation

A decision tree classification algorithm has been used for disaggregation of the signal into corresponding appliance loads. The algorithm performs classification in two phases and is evaluated in a third stage, in accordance with [13]:

- 1) *The tree building phase*: or growth phase, in which the tree is built by recursively splitting the data into two branches. The value of a split point depends upon how well it separates the different load types. To evaluate the quality of the split, the *Gini* index is used. The advantage of this index is that its calculation requires only the distribution of the load type in each of the partitions. For a data set S containing samples from n appliances, $Gini(S)$ is defined by (12).

$$Gini_{Split}(S) = 1 - \sum p_j^2 \quad (12)$$

Where p_j is the relative frequency of appliance j in S . When a split divides S into two branches S_1 and S_2 , the index of the divided data $Gini_{Split}(S)$, is given by (13)

$$Gini_{Split}(S) = \frac{n_1}{n} Gini(S_1) + \frac{n_2}{n} Gini(S_2) \quad (13)$$

To find the best split point for a node, each of the node's attribute lists is scanned and splits are evaluated based on that signature. The signature containing the split point with the lowest value for the Gini index is then used to split the node.

- 2) *The tree pruning phase*: Once the tree is fully grown, it is pruned by removing dependence on statistical noise

or variation that may be particular only to the training set, selecting the branches with the least estimated error rate. A simple pruning method was used to cut the tree: starting at the leaves, each node is replaced with its most popular class, but only if the prediction accuracy doesn't decrease. This reduced error pruning has the advantage of simplicity and speed.

- 3) *The performance evaluation stage*: Two validation methods have been used during this study to evaluate the performance of the algorithm: The *K-fold validation method* that realizes a partition of the original data set into 10 disjoint subsets, taking one of those 10 folder partitions for testing purposes and using the rest for the learning process. And the *Hold-out-validation method*, in which the original data set is partitioned into two disjoint sets using the Partitioning node, splitting the input table into two separate independent partitions.

D. The code

The disaggregation method has been developed using two Discrete Software tools:

- 1) MATLAB section: Import and processing of raw data; event detection or transient analysis; extraction of electrical signatures.
- 2) KNIME section: Disaggregation and validation using decision tree algorithms (Fig.1)

Fig. 2 below, presents the work flow of the whole disaggregation code.

IV. RESULTS ANALYSIS AND DISCUSSION

This section presents a number of tabled results, with the aim of showing the importance of the different signatures for each appliance in the overall disaggregation process.

The original data set, created by the MATLAB code processing of the raw data, contains ten events samples for each appliance. Table 1 presents the average magnitude for each signature and appliance over the ten samples and their corresponding standard deviation, σ . Signatures such as the *On-Off Current Increments* and the *Peak to Trough*, can clearly be used to distinguish between the appliances, whilst others (as *Phase Shift*) that we term 'impure' signatures due to imprecise identification metrics and large standard deviation, offer little capacity for disaggregation.

To evaluate the overall accuracy of the method, the confusion matrix produced by the *decision tree predictor node* (see fig. 1) is analyzed. The matrix contains true and false predictions, classified according with the matrix terminology given below:

- *True positive*: with hit.
- *True negative*: with correct rejection.
- *False positive*: with false alarm.
- *False negative*: with miss.

And the overall accuracy was calculated using (14).

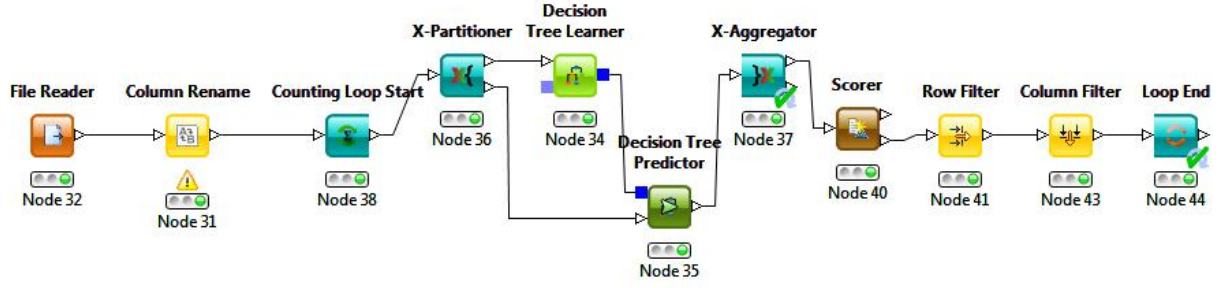


Fig. 1. Decision Tree code in KNIME

TABLE I
SIGNATURES AVERAGES AND STANDARD DEVIATION FOR EACH APPLIANCE

Appliances	ΔI_{On} (A)	σ (A)	ΔI_{Off} (A)	σ (A)	Set. Time (ms)	σ (ms)	Peak (A)	σ (A)	Phase Shift	$\sigma(^{\circ})$
K	11.52	0.06	-11.52	0.06	22.00	6.63	21.40	0.25	—	—
CM	3.91	0.03	-3.91	0.03	0	0	7.20	0.12	—	—
M	6.33	-0.17	-6.34	0.18	3441.10	221.51	16.30	2.79	23.34	0.95
H	3.18	0.06	-3.16	0.07	0	0	6.04	0.12	-1.59	0.08
F	0.10	0.01	-0.10	0.01	556	91.15	0.41	0.04	0	0.01
PC	0.51	0.01	-0.49	0.01	0	0	1.89	0.05	2.66	1.44
MC	0.03	0.01	-0.03	0.01	0	0	0.46	0.02	0.02	0.02
IL	0.14	0.01	-0.15	0.01	0	0	1.18	0.12	0	0.4

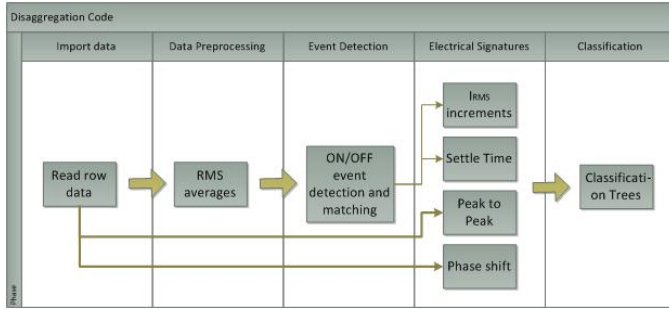


Fig. 2. Code work flow

$$Overall Accuracy = \frac{True_{positive} + True_{negative}}{False_{positive} + False_{negative}} \quad (14)$$

A. Appliance recognition from individual loads

By applying the K-fold cross-validation method over the original data set, the accuracy of the disaggregation method at identifying individual loads was calculated. Table II presents the overall accuracy obtained for appliance recognition, each column corresponds to individual signatures and the *Total* column to the combination of all of them. In the same way, each row corresponds to an individual load and the *All* row to the total loads aggregation.

The *Total* column achieves lower level of accuracy for some appliance than using individual signatures (e.g. coffee machine 90% with ΔI and 80% with *Total*). This is due to the "impurity" of some signatures that affect negatively the

learning process of the algorithm. For example, *Settle Time* is significantly affected by noise and *Phase Shift* by the inductive equipment connected to the phase circuit. However, the *Total* column contains the higher accuracy for *All* aggregated appliance loads, with 96%.

TABLE II
OVERALL ACCURACY FOR INDIVIDUAL LOADS RECOGNITION

Appliance	ΔI	Settle Time	Peak	Phase Shift	Total
K	95%	75 %	90 %	75%	99%
CM	90%	30%	80 %	75%	80%
M	92%	100%	80%	100%	92%
H	95%	75%	70%	75 %	95 %
F	77%	75 %	80 %	75%	97 %
PC	77%	30%	50%	65%	97%
MC	75%	30%	60%	70 %	98 %
IL	72 %	70%	80 %	60%	97%
All	92 %	65%	82%	75%	96%

B. Appliance recognition from aggregated loads

To determine the efficacy of the NIALM model over aggregated load, the *holdout-validation* method is applied over known aggregated consumption profiles. Due to experimental set up limitations, kitchen and workstation appliances have been monitored separately. To demonstrate the disaggregation capability under real-life 'messy' conditions (where the operation of small appliances could limit the information in the signal required for signature detection) a set of scenarios for switching and operational events were designed and tested.

Experimental design for the domestic kitchen appliances aggregated loads collection (fig. 3) :

- Progressive aggregation of all loads.
- Consecutive individual loads.
- Overlapping loads of similar loads.
- Random load aggregation.

Experimental design for the workstation appliances aggregated loads collection (fig. 4) :

- Progressive aggregation of all loads.
- Consecutive individual loads.
- High vs low loads.
- Random load aggregation.

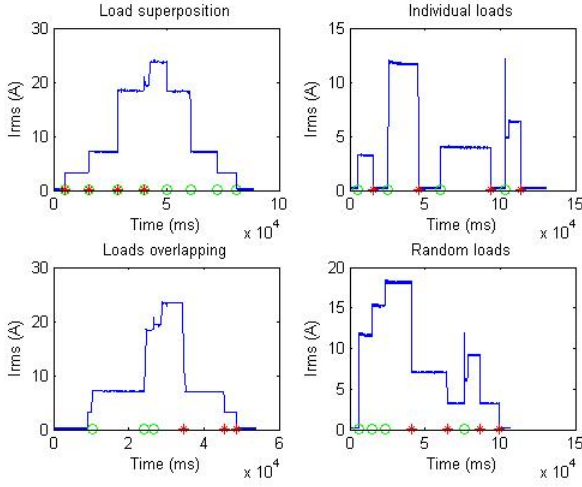


Fig. 3. RMS aggregated MATLAB profiles for kitchen appliances

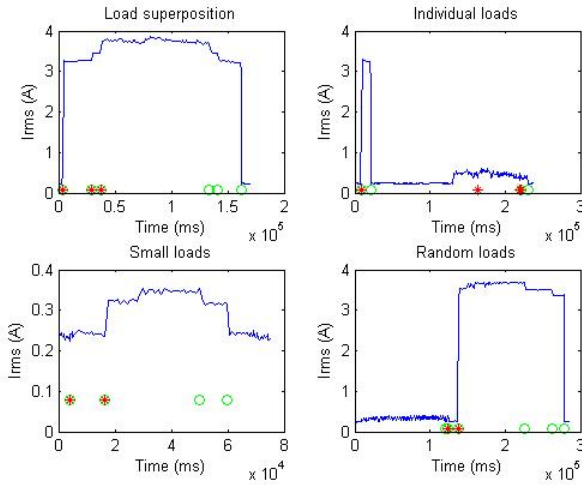


Fig. 4. RMS aggregated MATLAB profiles for workstation appliances

Each scenario was implemented twice, with overall accuracy presented in tables III to VI (kitchen appliances), and VII to X (workstation appliances).

Comparing table II to tables III to X, the ability to accurately disaggregate is reduced when dealing with more than one ap-

TABLE III
OVERALL ACCURACY FOR KITCHEN AGGREGATED LOAD RECOGNITION:
LOAD SUPERPOSITION

Appliance	ΔI	Settle Time	Peak	Phase Shift	Total
K	100%	75 %	50 %	37 %	100 %
CM	87%	25%	100%	25 %	50 %
M	87%	75%	75%	62%	75 %
H	100%	75 %	75%	75%	75 %
All SP	94%	63%	75%	50%	75 %

TABLE IV
OVERALL ACCURACY FOR KITCHEN AGGREGATED LOAD RECOGNITION:
INDIVIDUAL LOADS

Appliance	ΔI	Settle Time	Peak	Phase Shift	Total
K	100%	75 %	25%	75%	100 %
CM	80%	25%	75%	75%	100 %
M	90%	75%	75%	100%	100 %
H	100%	75%	75%	75%	80 %
All SP	90%	63%	63%	81%	95 %

TABLE V
OVERALL ACCURACY FOR KITCHEN AGGREGATED LOAD RECOGNITION:
LOAD OVERLAPPING

Appliance	ΔI	Settle Time	Peak	Phase Shift	Total
K	50%	50 %	75%	25 %	75 %
CM	75%	25%	75%	75%	75 %
M	25%	25%	25%	75%	75 %
H	75%	75%	75%	75%	75 %
All SP	56%	44%	63%	63%	75 %

TABLE VI
OVERALL ACCURACY FOR KITCHEN AGGREGATED LOAD RECOGNITION:
RANDOM LOAD

Appliance	ΔI	Settle Time	Peak	Phase Shift	Total
K	100%	70%	70%	70 %	100 %
CM	80%	30%	70%	80 %	75%
M	80%	80%	100%	100 %	100%
H	90%	80%	80%	70%	75%
All SP	85%	65%	80%	80 %	88%

TABLE VII
OVERALL ACCURACY FOR WORKSTATION AGGREGATED LOAD
RECOGNITION: LOAD SUPERPOSITION

Appliance	ΔI	Peak	Phase Shift	Total
F	60%	65%	60%	75 %
PC	50%	30%	40%	50 %
IL	40%	30%	40%	50 %
H	75%	40%	60%	80%
All SP	55%	40%	50%	65 %

pliance on the circuit. Signatures such as *Settle Time* and *Phase Shift*, that we term 'impure' due to imprecise identification metrics, negatively affect ability of disaggregation when using a combination of signatures.

Regarding the two categories of appliances:

TABLE VIII
OVERALL ACCURACY FOR WORKSTATION AGGREGATED LOAD
RECOGNITION: INDIVIDUAL LOADS

Appliance	ΔI	Peak	Phase Shift	Total
F	40%	80%	60%	80%
PC	80 %	60%	60%	60%
IL	60%	80%	60%	50%
H	10 %	70%	60%	80%
All SP	70%	70%	60%	80%

TABLE IX
OVERALL ACCURACY FOR WORKSTATION AGGREGATED LOAD
RECOGNITION: SMALL LOADS

Appliance	ΔI	Peak	Phase Shift	Total
F	50%	60%	70%	70%
IL	75%	50%	40%	50%
All SP	63%	55%	55%	60%

TABLE X
OVERALL ACCURACY FOR WORKSTATION AGGREGATED LOAD
RECOGNITION: RANDOM LOAD

Appliance	ΔI	Peak	Phase Shift	Total
F	70%	65%	40%	70%
PC	50%	30%	60%	60 %
IL	75 %	40%	40%	65 %
H	60%	30%	60%	85%
All SP	60%	40%	50%	75 %

- *Kitchen appliances:* The higher *Total* accuracy for the *All* row is achieved by the *individual loads* design, with 95% accuracy, and the worst by the *Load overlapping* design, with 75% accuracy. This last low result is due to the electrical nature of appliances: the microwave has the largest transient period, of around 0.6 seconds, and events happening within a 0.6 s window are not identified as such by the code.
- *Workstation appliances:* The higher *Total* accuracy for the *All* row is again achieved by the *individual loads* design, with 80% accuracy. However, workstation appliances have a visibly lower accuracy than kitchen appliances; the main reason for this is the existence of $\Delta I_{RMS} < 0.2A$ and the variable I_{RMS} profile of the PC, not compatible with the thresholds used in conditions (2) and (4). Small powers are affected by noise (the mobile charger has been removed from the tables as noise makes it indistinguishable). The *Small loads* design, with lower accuracy, 60%, corroborate this affirmation. In order to detect the PC profile, the current signal was averaged over 60 cycles instead of 2 cycles (as previously used with kitchen appliances). This new code configuration reduced the disaggregation capability of the *Peak to Trough* signature and makes it very difficult to detect settle times (this last signature has been removed from the tables).

V. CONCLUSION AND FUTURE WORKS

This paper proposes a NIALM method based on a number of distinctive electrical signatures and a decision tree classification method, presenting and analyzing a collection of results based on the overall model accuracy relative to each signature. The negative impact of noise, variable load profiles, loads overlapping and inductive loads in the disaggregation process are identified, and the good performance of the NIALM model for kitchen appliances demonstrated.

In order to improve the proposed disaggregation model, further work needs to be done, improving the performance of "impure" signatures. To reduce noise effects on the *Settle Time* signature, the implementation of filters to smooth load profiles need to be analyzed. The effect of power factor correction, as capacitors, on the *Phase Shift* needs to be considered in order to understand the value added by the current-voltage phase lag to the data set. For more robust signature values and a more comprehensive coverage of appliance loads, a larger database incorporating other classes of appliance is necessary.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the TSBE Centre and AECOM Infrastructure & Environment UK.

REFERENCES

- [1] A. Menezes, A. Cripps, R. Buswell, J. Wright, and D. Bouchlaghem, "Estimating the energy consumption and power demand of small power equipment in office buildings," *Energy Build.*, vol. 75, pp. 199–209, 2014.
- [2] Carbon Trust, "Advanced metering for SMEs Carbon and cost savings," p. 60, 2007.
- [3] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.
- [4] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey," *Sensors (Basel)*, vol. 12, no. 12, pp. 16 838–66, Jan. 2012.
- [5] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the Flick of a Switch : Detecting and Classifying Unique Electrical Events on the Residential Power Line (Nominated for the Best Paper Award)," pp. 271–288, 2007.
- [6] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 76–84, 2011.
- [7] M. E. Berges, "A Framework for Enabling Energy-Aware Facilities Through Minimally-Intrusive Approaches," Ph.D. dissertation, Carnegie Mellon University, 2010.
- [8] H.-H. Chang, "Non-Intrusive Demand Monitoring and Load Identification for Energy Management Systems Based on Transient Feature Analyses," *Energies*, vol. 5, no. 12, pp. 4569–4589, Nov. 2012.
- [9] C. Armel, "Energy Disaggregation What is Disaggregation ?" no. December, 2011.
- [10] R. Agrawal, T. Imielinski, and A. Swami, "Database mining: A performance perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 5, no. 6, pp. 914–925, 1993.
- [11] M. Mehta, R. Agrawal, and J. Rissanen, "SLIQ : A Fast Scalable Classifier for Data Mining," *5th Int. Conf. Extending Database Technol. Avignon, Fr.*, pp. 18–32, 1996.
- [12] B. T. D. Saulles, "The Illustrated Guide to Electrical Building Services By Tom de Saulles," 2001.
- [13] J. Shafer, R. Agrawal, and M. Mehta, "SPRINT: A Scalable Parallel Classifier for Data Mining," *Proc. 22th Int. Conf. Very Large Data Bases*, pp. 544–555, 1996.