HIGH ACCURACY EVENT DETECTION FOR NON-INTRUSIVE LOAD MONITORING

Mohamed Nait Meziane, Philippe Ravier*, Guy Lamarque*,*
Jean-Charles Le Bunetel† and Yves Raingeaud†

* PRISME Lab. - University of Orléans, 12 rue de Blois, 45067 Orléans, France
† GREMAN Lab. UMR 7347 CNRS-University of Tours, 20 avenue Monge, 37200 Tours, France

ABSTRACT

This paper proposes a new event detection algorithm for the use in Non-Intrusive Load Monitoring (NILM). This latter is a field where the main concern is to break down, in a non-intrusive manner, the global electrical energy consumption into individual appliances consumption. Detecting events is thus of importance for appliance clustering in event-based NILM systems. A simple and fast algorithm that detects the variations of the signal's envelope is proposed in this paper. Its main advantage is the high localization accuracy of the start times of events. Its performance is evaluated using simulated and real data and is compared to one of the recently proposed algorithms in the field. Simulations show that the proposed detection algorithm gives 100 % precision and 97.13 % recall at a Signal-to-Noise Ratio (SNR) of 50 dB.

***Index Terms*—** Event detection, energy disaggregation, event-based NILM, Non-Intrusive Load Monitoring (NILM), unsupervised NILM

1. INTRODUCTION

Non-Intrusive Load Monitoring (NILM), or energy disaggregation, is a field that appeared in the late 1980s with the work of Hart [1]. Nevertheless, it was not until recently that it started to know a rapidly growing interest. The main objective of a NILM system is to break down the total energy consumption into its different end-use parts (e.g. individual appliance consumption).

There are different approaches to solve the NILM problem and they can be classified in different ways. We have, for example, supervised vs. unsupervised approaches. Supervised approaches need a training or learning so that the system can learn the appliances it is supposed to measure before deployment for real use which adds more complexity to them. Unsupervised methods, on the contrary, are free from this constraint but are more prone to errors related to environment variability. A review for NILM approaches, in general, can be found in [2, 3, 4] and unsupervised approaches, in particular, in [5].

From the sampling frequency point of view, NILM approaches can be classified in Low Sampling Frequency (LSF) (1 Hz or less) or High Sampling Frequency (HSF) (hundreds of Hz to MHz) approaches. Due to the fact that data and datasets are more easily available in LSF, LSF-NILM is more widespread. Another dichotomy of the field is event-based vs. non event-based NILM depending on whether they rely on detecting and classifiying transitions or not [6, 7].

In event-based approaches the following stages are usually found: event detection, feature extraction, appliance clustering and energy consumption estimation. The event detection stage is particularly important since it is situated in the upstream of the NILM system. Most of the works on event-based approaches use the real and reactive power as input to the detection stage [8, 9]. But few others use other inputs as voltage distorsion [10] or current signal [11].

Our approach is an event-based unsupervised one that uses the HSF current signal as input to the event detection stage. In this paper, we only focus on the latter and discuss its performance in comparison to another recently proposed detector in the field. The intuition behind the proposed event detector is to improve the detection accuracy by avoiding the use of any kind of averaging (e.g. power computation), filtering, or transformation (e.g. FFT), etc. that may cause a loss of precious time-related information.

The novelty of the proposed algorithm with respect to the state of the art in the field is, mainly, its detection *accuracy*. To the best of our knowledge, our algorithm is the first one to provide such accuracy for a NILM event detection and moreover with *low complexity* and *fast computation*.

This paper is organized as follows: section 2 presents the proposed event detection algorithm. Section 3 presents and discusses the obtained evaluation results for the detector and section 4 gives concluding remarks and few perspectives.

2. EVENT DETECTION ALGORITHM

In this section we present the proposed algorithm, called hereafter High Accuracy NILM Detector (HAND), for the event

We would like to thank the support of the Région Centre-Val de Loire (France) for the project MDE-MAC3 (Contract n° 2012 00073640) under which this study was conducted.

detection. An event, or *active section*, corresponds to the part of the signal that deviates from the previous steady state and lasts as long as no steady state has been reached [11]. The proposed algorithm is simple and fast. It tracks the variation of the standard deviation (*std*) $\sigma_d(t)$ of the detected envelope $e_d(t)$ of the current signal using a moving window. Then, a threshold separates the events characterized by high amplitude variation (high $\sigma_d(t)$ values) from the steady states characterized by low amplitude variation (very low $\sigma_d(t)$ values). The algorithm outputs the starting and stopping times of the detected events. The algorithm is as follows (with some steps detailed after):

1. Detect current signal envelope $e_d(t)$.
2. Fix the moving window size L (for our simulations we chose $L \equiv 4$ time-cycles [samples]).
3. Initialize $\sigma_d(t_k)$, for $k = 1, \ldots, L$, with the standard deviation of $e_d(t), t = t_1, \ldots, t_L$.
4. Compute *iteratively* the mean $\mu_d(t_k)$ and the unbiased standard deviation $\sigma_d(t_k)$ of $e_d(t)$, $k = L+1, \ldots, N$, using:

$$\begin{aligned} \mu_d(t_k) = &\quad \mu_d(t_{k-1}) + \tfrac{1}{L}\left[e_d(t_k) - e_d(t_{k-L})\right] \\ \sigma_d^2(t_k) = &\quad \sigma_d^2(t_{k-1}) + \tfrac{1}{L-1}\left[e_d^2(t_k) - e_d^2(t_{k-L})\right] \\ &+ \tfrac{L}{L-1}\left[\mu_d^2(t_{k-1}) - \mu_d^2(t_k)\right] \end{aligned} \quad . \tag{1}$$

5. Choose detection threshold value γ (for our simulations we chose $\gamma = 0.1$) and find the starting and stopping times for each event such that:
   - start time t_s is defined as the first point of an event where $\sigma_d(t) > \gamma$ and $\dfrac{d\sigma_d(t)}{dt} > 0$.
   - end time t_e is defined as the last point of an event where $\sigma_d(t) > \gamma$ and $\dfrac{d\sigma_d(t)}{dt} < 0$.
6. Post-processing:
   - event approval: only if $\sum_t \sigma_d(t) > \delta, t \in [t_s, t_e]$.
   - window delay correction: $t_e = t_e - L$.

To detect the current signal envelope (step 1) we use an ad-hoc method. We detect the peak value on each time-cycle (20 ms for a 50 Hz power line frequency) of the current signal and interpolate between all the detected peaks (cubic interpolation). This interpolation causes the smoothing of current signal abrupt changes at the beginning of an event (see $e_{d1}(t)$ in Fig. 1). To overcome this, we propose to test the median of the absolute value of the current signal $x_{p_k,p_{k+1}}(t)$ situated between each consecutively detected peaks p_k and p_{k+1} with a threshold α (empirically fixed). If median$(|x_{p_k,p_{k+1}}(t)|) < \alpha$ then for this part $e_d(t) = x_{p_k,p_{k+1}}(t)$, otherwise the previously detected envelope is kept. This additional test allows the envelope to better follow the variations of the current signal, especially when the previous steady state time-cycle has low amplitude or contains a portion of the starting event (see $e_{d2}(t)$ in Fig. 1).

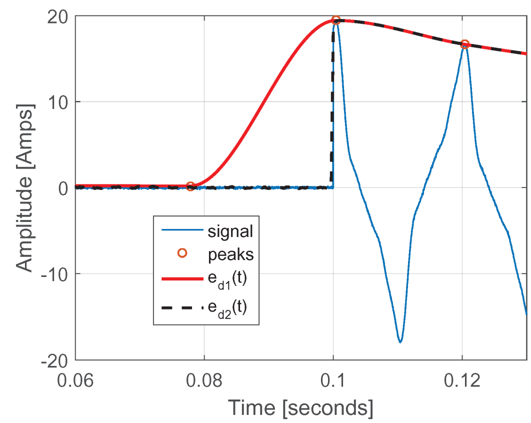

Fig. 1. Improvement of the envelope detection on a current signal turn-on event: $e_{d1}(t)$ is detected without median thresholding and $e_{d2}(t)$ with median thresholding.

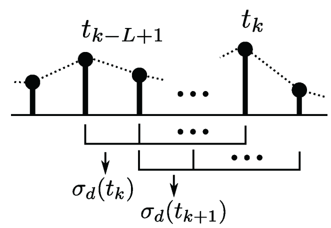

Fig. 2. Illustration of the computation "principle" of $\sigma_d(t)$ using a moving window of size L. Note that concretely this is done equivalently in an iterative manner.

After detecting $e_d(t)$, we compute its moving *std* $\sigma_d(t)$, i.e. the detector's feature (steps 2 to 4) (see Fig. 2). This is done *iteratively* to improve the computation speed. After that, we fix a threshold γ (step 5) to define the start time t_s and end time t_e for each event. Since our feature is the *std* of the current signal's envelope, fixing γ is guided by the *std* value that corresponds to noise variance (for example if the noise *std* = 80 mAmps, $\gamma = 0.08$ corresponds to considering all signal sections with variance above noise variance as events). Hence, this simplifies the parameter selection for the proposed algorithm.

The post-processing (step 6) corrects, mainly, for two things: (1) signal sections barely exceeding threshold γ (event approval) and (2) the time-delay (window delay correction) that appears in the stop time, t_e, detection due to the moving window using the previous $L-1$ data points to compute the present point *std* (see Fig. 2). Note here that the threshold δ used for event approval is fixed empirically and that no correction is needed for start time t_s.

3. EXPERIMENTAL RESULTS

To test the HAND, we propose to use simulated and real data and to compare the results with the ones obtained using the recently proposed Kernel Fisher Discriminant Analysis (KFDA)-based algorithm [11] (called loosely KFDA hereafter). This algorithm uses current harmonics I_k as features, estimated using IQ-demodulation technique, and the kernelized Fisher linear discriminant function [12] as a classifier to distinguish between the "event" and the "steady state" classes. It computes a test statistic and compares it to an adaptive threshold.

For reliable results, the used data for the test has to be provided with *accurate ground truth* time instants for the events' occurrences. It is important to keep in mind that the turn-on event of an appliance is considered as one event that has a start and end times and the turn-off event of an appliance is considered as another event having also its start and end times. This means that it is the start times of the events we can know with precision since they correspond to their triggering (for example, turning on or off an appliance). The stop time of the turn-on event of an appliance is not precisely defined! Hence, for the test, we will be comparing ground truth start times of events with the detected ones.

3.1. Simulated data tests

For simulations we use the current signal model proposed in [13], with stationary parameters, i.e. $s(t) = (A_0 e^{\mathbf{p}^\mathbf{T}\mathbf{t}} + 1)\sum_{i=1}^{d} A_i \cos(2\pi f_i t + \phi_i)$ where A_0 is a scalar and $\mathbf{p} = [p_1, \ldots, p_n]^T$ is a vector of n polynomial coefficients such that $\mathbf{p}^\mathbf{T}\mathbf{t}$ is a n^{th} degree polynomial, with time vector $\mathbf{t} = [t, \ldots, t^n]^T$. A_i, f_i and ϕ_i are, respectively, the amplitudes, frequencies and phases of d sinusoids. The simulation signal $x(t)$ is the sum of K different delayed versions of $s(t)$ plus noise: $x(t) = \sum_1^K s_k(t, \tau_k, D_k) + w(t), t = 0, \ldots, D$. The noise is white Gaussian with zero-mean and variance σ_w^2. τ_k and D_k are, respectively, time delays and durations. The versions of $s(t)$ are randomly generated and the chosen parameters for the simulation are as follows ($\mathcal{U}(a, b)$: is the uniform distribution on interval (a, b)): $A0 = 1, n = 1, d = 5, f = [50, 150, 250, 350, 450]^T, \tau_k = \mathcal{U}(0, D), D_k = \mathcal{U}(1, 10), p_1 = \mathcal{U}(-10, 0), A_i = \mathcal{U}(0, 5), \phi_i = \mathcal{U}(0, \pi)$. The sampling frequency F_s =10 kHz and $K = 4000$, but after removing versions with end time exceeding total signal duration D = 2000 minutes, we ended up having less (around 3800 versions $\equiv$ 7600 events for each simulation).

As evaluation criteria we use the following measures: True Positives (TP), False Negatives (FN), False Positives (FP), precision and recall as suggested by different works in the field [14, 15]. The precision and recall are defined as: precision = TP/(TP+FP), recall = TP/(TP+FN).

Along with these measures, we also define a *delay tolerance* Δt such that for a detected event with start time t_s to be considered as a true positive, the condition $|t_s - t_t| \le \Delta t$, where t_t is the ground truth time instant, must be verified. A false positive is, then, defined as a detected event that violates this condition. This way, this tolerance delay is a quantity that defines the required accuracy of the detector to perform well.

Table 1. TP, FN, FP, precision and recall of HAND vs. KFDA, SNR = 50 dB, Δt = 200 ms

	TP	FN	FP	precision	recall
HAND	7417	253	0	100.00%	96.70%
KFDA	1517	6153	5269	22.35%	19.78%

In the following we compare the two algorithms for accuracy (varying Δt) and sensitivity to noise (varying SNR).

3.1.1. Accuracy comparison

To test the performance of the compared algorithms with respect to a required accuracy, we fix the SNR at 50 dB and consider two Δt values: 200 ms and 500 ms. Table 1 and the first row (50 dB) of Table 2 give the results. These indicate that the HAND satisfies the required accuracies and gives remarkably perfect precision (100%). KFDA, on the other hand, performs well (precision/recall = 98.19/86.20%) for Δt = 500 ms but has very poor performance for the required accuracy of Δt = 200 ms. The best achievable accuracy depends on the SNR and the parameter choice for each of the algorithms and its study is out of the scope of this paper but we expect that the HAND is able to satisfy at least Δt = 50 ms at SNR = 50 dB.

The approach of KFDA is attractive but what makes it loose in accuracy is mainly its reliance on a low-pass filter in the IQ-demodulation stage that adds filter artifacts around abrupt changes and an averaging, that deteriorates the accuracy, is then necessary to solve for that.

3.1.2. Sensitivity to noise

Following the results of the previous sub-section, we fix Δt = 500 ms to not penalize KFDA and we vary the SNR (see Table 2). The results show that KFDA is particularly robust against noise (precision and recall have around the same value for a decrease of 20 dB (from 50 to 30 dB) in the SNR, around an average of 98.23 % and 86.83 %, respectively, for precision and recall) whereas the precision of the HAND is sensitive to noise (decreases from 100 % to almost 95 % for a SNR decrease of 20 dB), which is to be expected. Since the detector's feature is the standard deviation of the envelope, as the SNR gets lower more FP might be present. On the other hand, the recall of the HAND is not affected (stays at around 97.35 %) and is high which is an advantage for the HAND compared to KFDA. A good recall means that misses do not occur often and this is desirable in a NILM system since, if possible, we do not wish to miss any true event. Having FP is not desirable

Table 2. TP, FN, FP, precision and recall of HAND vs. KFDA, Δt = 500 ms, different SNRs

SNR	Algo.	TP	FN	FP	prec.(%)	rec.(%)
50 dB	HAND	7399	219	0	100.00	97.13
	KFDA	6567	1051	121	98.19	86.20
40 dB	HAND	7432	202	15	99.80	97.35
	KFDA	6695	939	123	98.20	87.70
30 dB	HAND	7490	186	399	94.94	97.58
	KFDA	6801	875	117	98.31	86.60

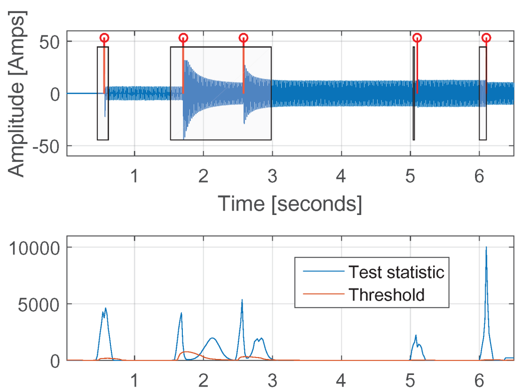

Fig. 3. KFDA results on real data. Top: Red peaks indicate ground truth start times and detected events are framed in boxes.

either but is less disadvantageous than having misses because these FP can be dealt with in the clustering stage, for example, by affecting them to an "unknown" class instead of an "appliance" class.

Finally, note that HAND is at least three times faster than KFDA due to both its low complexity and fast computation.

3.2. Real data test

Here we show a detection example on real data obtained using the measurement system presented in [16]. This system allows the creation of consumption scenarios with the advantage of providing a high precision control over the turn-on and off of appliances. The measured signal is sampled at 100 kHz and contains five events (4 turn-on and 1 turn-off: (1) turn-on of the electronic part of a vacuum cleaner, (2) turn-on of the motor of the same vacuum cleaner, (3) turn-on of a drill, (4) turn-on of a halogen lamp, (5) turn-off of the drill). The detection results are shown in Fig. 3 and 4 and the start time detection errors are given in Table 3.

The average detection error over these five events is 9.94 ms (around half a time-cycle) for HAND and 87.25 ms for KFDA. Also KFDA missed the 3^{rd} event merged with the 2^{nd}. This shows on a simple example how accurate is HAND compared to KFDA.

It is important to accurately detect events since this will

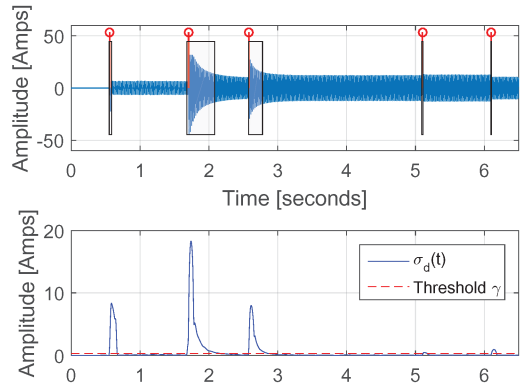

Fig. 4. HAND results on real data. Top: Red peaks indicate ground truth start times and detected events are framed in boxes.

Table 3. Error in start time detection for the five events showed in Fig. 3 and 4.

	HAND	KFDA
Event 1	6.7 ms	99 ms
Event 2	24 ms	180 ms
Event 3	4.8 ms	merged with event 2
Event 4	11.1 ms	60 ms
Event 5	3.1 ms	10 ms
Average	9.94 ms	87.25 ms

automatically affect the energy consumption estimation (estimated using consumption part duration, itself estimated from event times). Moreover, from a classification system perspective, an event detection stage with low accuracy may affect the result of the parameter estimation and hence the feature extraction stage that aims to extract relevant and unique features for use in appliance clustering.

4. CONCLUSION

A new event detector for event-based NILM systems was proposed. It is simple due to the use of the standard deviation of the current signal's detected envelope as feature. The algorithm is also fast since it computes its features iteratively. Moreover, the choice of the detection threshold value is guided by the noise level in the measured signal which is a big help. The results show that the proposed algorithm is more accurate than the KFDA-based one [11] for a delay tolerance $\leq$ 200 ms (at 50 dB SNR) and has a higher recall but it's precision is sensitive to noise whereas the KFDA-based algorithm is very robust against noise.

For future work, we are considering the adaptation of the thresholding to the SNR in order to alleviate the noise sensitivity of HAND. We also plan to test the detector on more real data after creating a dataset of controlled scenarios using the measurement system presented in [16].

5. REFERENCES

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

[2] Hala Najmeddine, K El Khamlichi Drissi, Christophe Pasquier, Claire Faure, Kamal Kerroum, Alioune Diop, Thierry Jouannet, and Michel Michou, "State of art on load monitoring methods," in *Power and Energy Conference, 2008. PECon 2008. IEEE 2nd International*. IEEE, 2008, pp. 1256–1258.

[3] Yi Du, Liang Du, Bin Lu, Ronald Harley, and Thomas Habetler, "A review of identification and monitoring methods for electric loads in commercial and residential buildings," in *Energy Conversion Congress and Exposition (ECCE), 2010 IEEE*. IEEE, 2010, pp. 4527–4533.

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

[5] R. Bonfigli, S. Squartini, M. Fagiani, and F. Piazza, "Unsupervised algorithms for non-intrusive load monitoring: An up-to-date overview," in *Environment and Electrical Engineering (EEEIC), 2015 IEEE 15th International Conference on*, June 2015, pp. 1175–1180.

[6] Kyle D Anderson, Mario E Bergés, Adrian Ocneanu, Diego Benitez, and José MP Moura, "Event detection for non intrusive load monitoring," in *IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 2012, pp. 3312–3317.

[7] B. Zhao, L. Stankovic, and V. Stankovic, "On a training-less solution for non-intrusive appliance load monitoring using graph signal processing," *IEEE Access*, vol. 4, pp. 1784–1799, 2016.

[8] Leen De Baets, Joeri Ruyssinck, Dirk Deschrijver, and Tom Dhaene, "Event detection in nilm using cepstrum smoothing," in *3rd International Workshop on Non-Intrusive Load Monitoring*, 2016, pp. 1–4.

[9] Awet Abraha Girmay and Christian Camarda, "Simple event detection and disaggregation approach for residential energy estimation," in *Workshop on Non-Intrusive Load Monitoring (NILM), 2016 Proceedings of the 3rd International*, 2016.

[10] R. Cox, S. B. Leeb, S. R. Shaw, and L. K. Norford, "Transient event detection for nonintrusive load monitoring and demand side management using voltage distortion," in *Applied Power Electronics Conference and Exposition, 2006. APEC '06. Twenty-First Annual IEEE*, March 2006.

[11] B. Wild, K. S. Barsim, and B. Yang, "A new unsupervised event detector for non-intrusive load monitoring," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Dec 2015, pp. 73–77.

[12] S. Mika, AJ. Smola, and B. Schölkopf, "An improved training algorithm for kernel fisher discriminants," San Francisco, CA, Jan. 2001, Max-Planck-Gesellschaft, pp. 98–104, Morgan Kaufman.

[13] Mohamed Nait Meziane, Philippe Ravier, Guy Lamarque, Karim Abed-Meraim, Jean-Charles Le Bunetel, and Yves Raingeaud, "Modeling and estimation of transient current signals," in *Signal Processing Conference (EUSIPCO), 2015 Proceedings of the 23rd European*, 2015, pp. 2005–2009.

[14] Philipp Klein, Jean Merckle, Dirk Benyoucef, and Thomas Bier, "Test bench and quality measures for non-intrusive load monitoring algorithms," in *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*. IEEE, 2013, pp. 5006–5011.

[15] Stephen Makonin and Fred Popowich, "Nonintrusive load monitoring (nilm) performance evaluation," *Energy Efficiency*, pp. 1–6, 2014.

[16] Mohamed Nait Meziane, Thomas Picon, Philippe Ravier, Guy Lamarque, Jean-Charles Le Bunetel, and Yves Raingeaud, "A measurement system for creating datasets of on/off-controlled electrical loads," in *Conference on Environment and Electrical Engineering (EEEIC), 2016 Proceedings of the 16th IEEE International*, 2016, pp. 2579–2583.