An Improved Event Detection Algorithm for Non-Intrusive Load Monitoring System for Low Frequency Smart Meters

Abdullah Al Imran
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Minhaz Ahmed Syrus
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Hafiz Abdur Rahman
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

***Abstract*— The traditional event detection algorithm used in Non-Intrusive Load Monitoring (NILM) for low frequency smart meters, is aimed to detect high power consuming appliances in particular. As the level of noise generated by the high capacity loads, detecting events caused by low power consuming devices is a major challenge. In this paper, we have looked into various issues of low frequency smart meters for the steady-state method of NILM algorithm. Also, we have proposed an improved event detection algorithm to extract more consistent and exclusive event for individual loads. Later, we have used the proposed event detection algorithm on practical data collected from a low frequency smart mete deployed in a residential building. The evaluation shows that, our algorithm detects events more precisely and accurately than the traditional algorithm. Also, it reduces the event space of individual appliance and extract an additional feature which helps to differentiate similar events of different appliances in real and reactive power region.**

Keywords—Non-Intrusive Load Monitoring; Low Frequency; Edge Detection; Additional Load Feature

I. Introduction

The basic idea of NILM proposed by Hart [1] was to disaggregate electrical loads by exploring the appliance specific power consumption signatures within the combined load data. The data is captured from the main electrical distribution point of a consumer and later analyzed with a computer system. Since, no additional device is needed to be installed in consumers' premise, the system is considered to be non-intrusive. However, this system required one-time intrusion for training purpose.

Since the conception of NILM in early '80s, many researchers have proposed different models of NILM [2]. However, in almost every cases, the fundamental framework proposed by Hart is unchanged. So, the disaggregation problem can be expressed as follows: If the total power consumption data or the aggregated meter readings is P(t), then the mathematical expression is defined as:

$$P(t) = \sum_{i=1}^{N} P_i \tag{1}$$

where P_i is the total consumed power of an individual appliance which adds to the cumulative measurements and N is the total number of active appliances within the time period t. In steady-state method of NILM for low frequency meter, raw temporal power consumption data is captured and processed to extract "load signatures" and detect events. This is the pre-processing part. Here the raw data is first normalized, then filtered and after that, an edge detection algorithm is used to find out any significant step change in the temporal power consumption data due to the appliances' state changing activities. These significant step changes in real and reactive power data, are termed as the "events". So, the major part of event detection algorithm is edge detection. After, detecting events, events are clustered and identified. Then a database events is created which consists individual appliance's states and corresponding event information generated from training data. Finally, a match finding algorithm is used to determine the appliances' activity information from the collected meter data set for any particular time period.

Primarily, it is assumed that the event spaces of individual appliances do not overlap and only a single event occur within a reasonable time interval. However, in practical, these assumptions are not always true. The reason is that event detection algorithm is designed to detect the high power appliances, so low power appliances are overlooked. Furthermore, these assumptions are ideal only for binary states devices and some Finite State Machines (FSM). For variable state devices and FSMs with 20 or more states, the assumptions are not true. Moreover, due to various limitation of inexpensive low frequency meters, edges are often offset or undetected. For this reason, event spaces get larger which increases the probability of overlapping of different event spaces. Also, the missed events jeopardize the disaggregation process for corresponding appliances.

In this paper, we propose an improved event detection algorithm. Our algorithm tackles the issues related to capturing data from a low frequency smart meter. While we were performing real-time training on different appliances, we observed that, for visually identical steps generated by the same device, clustered values were not identical. In fact, those were dispersed by a significant margin. This phenomenon caught our interest and after further study we discovered that, due to the constant sized windowing method used in this type of meters for calculating rms quantities, edge offset occurs. First, to counter the edge offset, we have used two different amplitude threshold values for consecutive meter readings, whereas traditional algorithm uses just one amplitude threshold. Second, we have used different threshold values for real and reactive power. This removes any false resistive event detection. Finally, unlike the conventional edge detection, we have separately detected the devices' turn-on impulses caused by transients and used it as an additional feature for the event classification and identification. This additional feature comes in handy when event space overlap occurs for two or more devices.

For evaluation, we have used a utility grade smart meter and captured 1 data set per second. Each data set contains timestamp, true rms voltage, true rms current, rms reverse current, frequency, real power, reactive power and power factor. Since, it provides true rms values, the collected data is more accurate than regular low frequency meters which offers the approximate rms quantities. Also, we have utilized the power factor to determine the type of reactive load. So, in cluster analysis, we have three regions: resistive, capacitive and inductive; whereas regular meter can only provide resistive and reactive regions. Then we have trained three binary state appliances and two finite state machines to with both traditional and proposed event detection algorithm. Later we compared both algorithm and found that, the proposed algorithm detects events accurately whereas traditional event often misses some. Also, the event space size is more reduced in the improved algorithm compared to the conventional algorithm. Furthermore, we successfully differentiated two event space overlap by using the additional transient edge information.

The results show that, using the improved algorithm the accuracy and precision of the event detection process can be significantly improved. Also, transient edge feature promises to be a reliable parameter for consideration.

II. Related Work

Since the inception of NILM in early '80s by George Hart [1], various approaches have been tried out. Most of these approaches mainly explored different types of machine learning and match finding algorithms [2]. However, very few attempts were made in the field of event detection and pre-processing of raw data for low frequency meters. On contrary, some effort was made to bypass pre-processing, but they failed to show any evident advantage [3]. Even today, most researches of NILM concentrates on post-processing of detected events and very few researchers are working on event detection of high frequency metering data. For this reason, the general event detection algorithm demonstrated by Hart [1] is still exclusively prevalent in low frequency metering arrangements. A brief description of different segments of the traditional event detection algorithm is given below.

A. Normaliztion

As the appliances' power consumption varies with the fluctuation of line voltage and frequency, a normalization scheme is necessary before edge detection. Usually, a ±10% variation of the rated voltage of the system is set as the standard. However, this ±10% variation in system voltage results in up to ±20% fluctuation for the real and reactive power. Moreover, different types of appliances show dissimilar response to the voltage fluctuation [4]. So, to offset or minimize the effect of voltage fluctuation, real power P and reactive power Q are normalized to a rated voltage U_{ref} by using the following generalized formulas [1]:

$$P_{norm}(t) = \left(\frac{U_{ref}}{U(t)}\right)^{\alpha} \times P(t) \qquad (2)$$

$$Q_{norm}(t) = \left(\frac{U_{ref}}{U(t)}\right)^{\beta} \times Q(t) \qquad (3)$$

Although, the values of the exponents α and β differ for each appliance, for simplicity α=β=2 is assumed.

B. Filtering

Another key problem that needs to be addressed before the edge detection process is the switching transient or the switching impulse. Since, the amplitude of the impulse generated by switching transient is very large and fluctuating, the chances of false event detection are also very high; moreover, the actual event is not detected at all. This problem can be solved by using nonlinear median filter [5]. Alternatively, mean filter or Gaussian kernel filter or different combinations of these filters can be used to offset switching transients [6]. However, performance of these filters varies on the nature of the data set, so selecting appropriate filtering technique is usually experimental [6]. Another key consideration for filter selection is the window size of it. Usually, larger window size means more signal smoothing, but it can also filter out small event signatures. Also, large window size results in more delay for the online real-time detection.

C. Conventional edge detecction algorithm

The conventional edge detection algorithm proposed by Hart [1] is a simple and fast algorithm which uses absolute differencing for edge detection. However, detection of the "continuously variable states" are not possible with this method. For such cases, image processing algorithms like Canny Edge detection algorithm [7] can be used to locate significant edges in large data set with high noise, but it cannot detect the magnitude of the edges.

In Hart's algorithm, the normalized and transient-filtered data is used to fist determine period of the steady-states which is defined to be a certain minimum length, usually three samples; and the period of change. Then, a tolerance level for both real and reactive power is set and the samples in each steady-states are separately averaged to minimize the effect of noise. Later, the differences of the averages across each period of change is taken as the step-size. Finally, the step-sizes are saved with the corresponding time-stamps as events. As the algorithm is aimed to detect the high power consuming devices (greater than 200W), small power consuming devices are disregarded for any detection or evaluation.

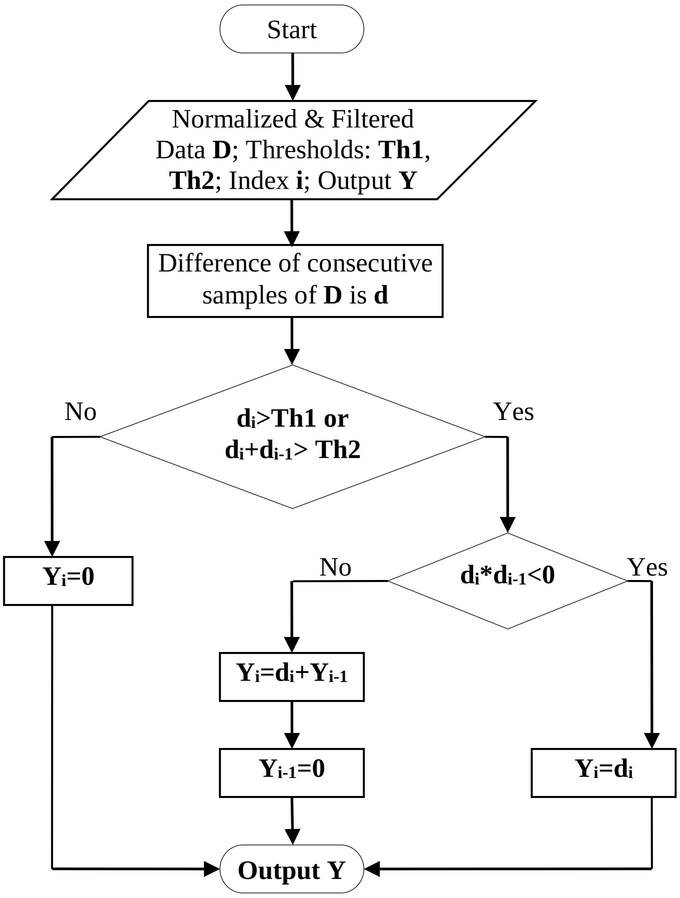

Fig. 1. Flow chart of the improved algorithm. Here diff means the difference of consecutive samples and i represents the index of time samples.

III. Approach

The goal of our improved event detection algorithm is to resolve the issues of event missing and false event related to using practical low frequency smart meter data. The key intuition behind our approach is that, when the variables of the system are same, any event generated by a specific activity should be consistent all the time up to a reasonable level. So, we investigated the reason behind discrepancies and proposed an improved algorithm to offset the effects and minimize the errors. Also, we have used the transient edge as a feature which is generally offset in traditional event detection algorithm. However, we have used the traditional approach discussed before for normalization and filtering.

A. Improved edge detection algorithm

The normalized and filtered raw data is used for the edge detection. Here, we have proposed a two threshold edge detection algorithm. The first threshold th1 is used for one-unit time interval and the second threshold t2 is used for the two consecutive time interval. So, if any portion of an edge is less than th1, it is not offset, but detected by th2. For simplicity, first we generate the difference data, which is the amplitude difference of consecutive samples. Then the differences are used to find the one-dimensional step-sizes of each edge. The improved algorithm is demonstrated in Fig. 1. Here, th1 and th2 can be set suitably to detect steps of slow rise or fall time.

In contrast to Hart's [1] edge detection algorithm, our proposed algorithm does not detect edges from the difference of averaged steady states, rather, it evaluates consecutive difference of samples to detect true edges. Also, it requires few samples to detect edges, so it can be used to implement real-time NILM system.

B. Edge matching

Due to the processing power limitation of smart meters, often the real power edge ΔP and reactive power edge ΔQ for a particular event are offset from each other by a range of ±3 samples. If these two components are accepted separately in the event plane, then two different event is detected instead of the original single event. So, to offset this problem, the developed an edge matching algorithm as depicted in Fig. 2.

C. Edge pruning

Since, only real power edge represents any true event occurrence, it is considered as the principal edge. However, after edge matching, some reactive edges are found without any corresponding real power edge. These edges do not convey any important information for event detection, so these edges

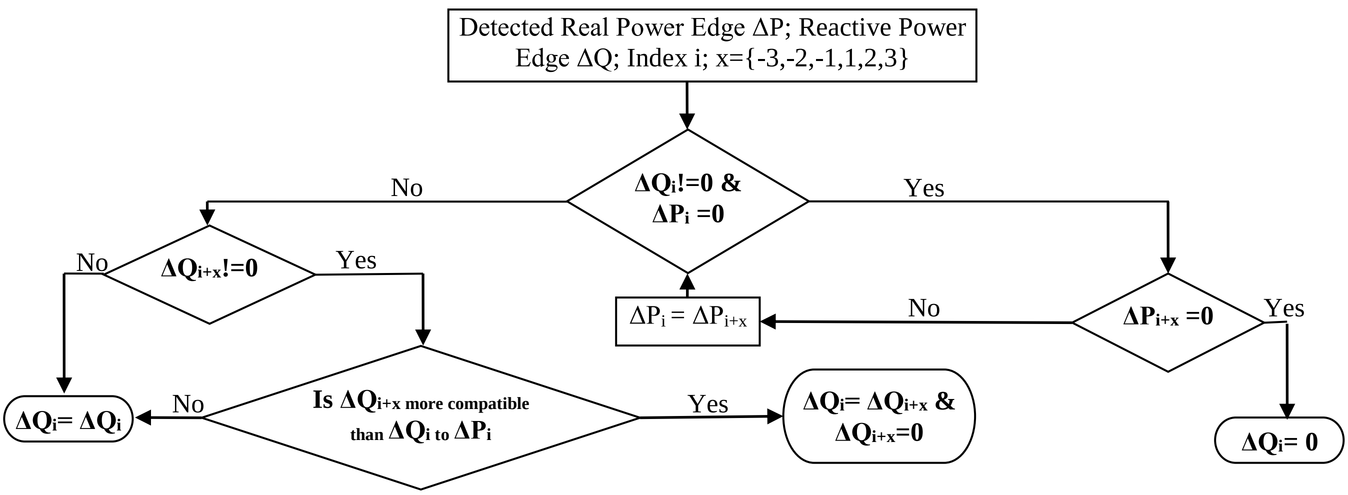

Fig. 2. Edge matching algorithm

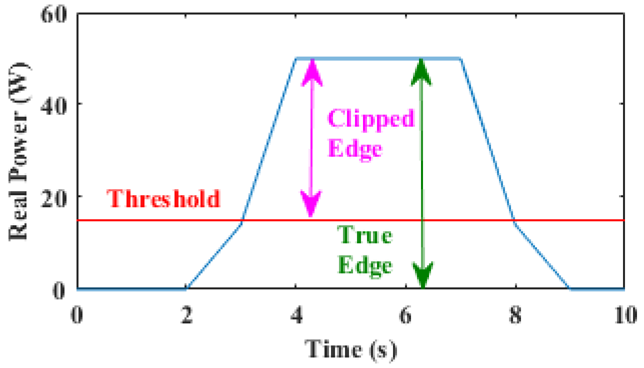

Fig. 3. Edge offseting for traditional event detection algorithm

are pruned from the edge list. Finally, we get a list of real power edges with their corresponding reactive edges and timestamps.

D. Transient impulse detection

We detected the turn-on transient impulses by subtracting normalized and filtered data from the raw data. Then, we have used our two threshold edge detection algorithm to detect the turn on impulse value. However, the threshold values were set experimentally for the whole data set to avoid any false detection.

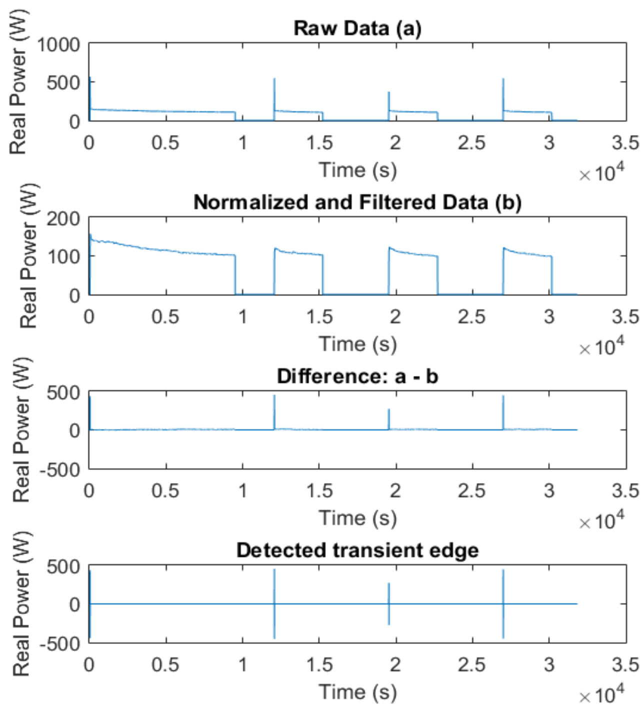

Fig. 4. Transient Edge detection

IV. Evaluation

We have collected data from a utility grade smart meter (EKM omnimeter v.4) deployed at a residential apartment. Then, we selected a ceiling fan, a television set, a blower, an air-fryer and a fridge. We have separately trained each appliance in real time and saved both raw and processed data in a database.

At first we have used the originally proposed algorithm [1] to detect the edges, but we noticed one problem. Initially, the tolerance for steady-state was set 15W per consecutive temporal point difference. Using this setting, we measured the edges for the two-state 80W fan for several times and we observed that, the amplitude of the edges was fluctuating between 65W to 80W, but, a 15W fluctuating range is not acceptable for a low power device. Initially, it appeared that voltage fluctuation might be the reason but the test data showed otherwise. After some more experimentation, we were able to discover the cause; and it was the 1Hz sampling window of the meter. Though the rise time to turn 'ON' the fan was less than a second, due to the sampling window, the turn on power was divided into two consecutive time intervals as shown in Fig. 3. As the threshold was set to 15W, so maximum 15W can fluctuate for this reason. Later, we used our proposed algorithm and found the measurements more consistent. As a result, the cluster size or event space for a device reduced significantly.

After creating the load profiles using our improved event detection algorithm, we observed that the event spaces of the fan and the television overlaps. However, their turn-on impulse clusters do not overlap. So, the impulse feature is used as an additional differentiating feature.

We also tried to test our algorithm with different available public dataset but only iAWE [8] dataset provides 1 Hz sampled voltage, real power and reactive power data. However, it does not provide any ground truth to evaluate quantitively.

V. Conclusion

The traditional event detection algorithm of NILM does not address the issue of event offset for low frequency meter data. In this paper, we presented an improved algorithm which counters the issue and provides more accurate and consistent event detection. Our improved algorithm, reduces the event space size to minimize the chance of overlaps. Also, it uses the transient impulse as a valid feature and adds another dimension to disaggregation.

References

[1] G. Hart, "Nonintrusive appliance load monitoring," Proceedings of the IEEE, vol. 80, no. 12, pp. 1870–1891, 1992.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[2] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey.," Sensors (Basel, Switzerland), vol. 12, no. 12, pp. 16838–66, Jan. 2012.

[3] R. Streubel and B. Yang, "Identification of electrical appliances via analysis of power consumption," Power Engineering Conference (UPEC), pp. 1–6, Sep. 2012.

[4] G. Hood, “The effects of voltage variation on the power consumption and running cost of domestic appliances,” Australasian Universities Power Engineering …, no. September, pp. 26–29, 2004.

[5] L. K. Norford and S. B. Leeb, “Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms,” Energy and Buildings, vol. 24, no. 1, pp. 51–64, Jan. 1996.

[6] M. Weiss, A. Helfenstein, F. Mattern, and T. Staake, “Leveraging smart meter data to recognize home appliances,” 2012 IEEE International Conference on Pervasive Computing and Communications, pp. 190–197, Mar. 2012.

[7] J. Canny, “A computational approach to edge detection.,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, pp. 679–698, 1986.

[8] N. Batra, M. Gulati, A. Singh, and M. Srivastava, “It’s Different: Insights into home energy consumption in India,” Proc. 5th ACM Work. Embed. Syst. Energy-Efficient Build., no. August, pp. 1–8, 2013.