An Algorithm based on Finite State Machines with Fuzzy Transitions for Non-Intrusive Load Disaggregation

Pietro Ducange, Francesco Marcelloni, Dario Marinari
Dipartimento di Ingegneria dell'Informazione: Elettronica, Informatica, Telecomunicazioni
University of Pisa
Pisa, Italy
{pietro.ducange, francesco.marcelloni}@iet.unipi.it

***Abstract*— Despite the raising awareness in general public on environmental changes and high energy costs, a significant part of energy consumption is still due to an improper use of electrical appliances. Thus, there is a growing interest in developing systems for profiling the use of electrical appliances and suggesting adequate policies for energy saving. In this context, we propose a novel approach to extract the power consumption of a set of appliances from aggregate measurements collected from a smart meter. Our approach employs finite state machines based on fuzzy transitions (FSMFT) and a novel disaggregation algorithm. The FSMFTs are used to coarsely model how each type of appliance works. The disaggregation algorithm exploits a database of FSMFTs for, at each meaningful variation of real and reactive aggregate powers, hypothesizing possible configurations of active appliances. This set of configurations is concurrently managed by the algorithm which, whenever requested, outputs the configuration with the highest confidence with respect to the sequence of detected events. We have successfully tested our approach in an experimental environment in which five appliances have been monitored for 30 minutes.**

Keywords- non-intrusive load disaggregation; finite state machines; fuzzy transitions; smart meters; electricity consumptions

I. INTRODUCTION

Recent studies have highlighted that a significant part of the electrical energy consumption in residential and business buildings is due to an improper use of the electrical appliances [1]. Electricity monitoring systems can help to make the users aware of possible energy wastes, as shown by initiatives like Google Power Meter [2] and Microsoft Hohm [3]. However, these systems generally provide only an analysis of the electrical consumption in a building based on the aggregate measure collected by a smart meter, without studying how single appliances are used. Even though both Google and Microsoft claim that they gave an important contribution to the field of intelligent energy metering, their services have been retired. On the other hand, the electricity smart meter market is still growing and several low cost models are available for monitoring domestic device consumptions. Further, also the national electric utilities have increased their efforts in deploying smart meters in their supply networks [4].

Following the trend of exploiting the data provided by smart meters and aiming to develop a low cost and intelligent energy monitoring system, in this paper we introduce a novel technique to extract the individual power consumption of a set of appliances from aggregate measures collected by a smart meter. We aim to identify the appliances that contribute to the energy consumption in order to highlight inefficiencies and wastes, and to provide recommendations on how power consumption can be reduced.

The proposed approach models the appliances by using finite state machines based on fuzzy transitions (FSMFTs). The events that trigger a state transition are described in terms of linguistic labels, such as low, medium and high, whose meaning is expressed by fuzzy sets defined on the universes of the linguistic variables that characterize the behavior of the appliances. We recall that a fuzzy set A defined on a universe of discourse U is characterized by a membership function $A(x)$ which associates with each element x of U a number in the interval [0,1] representing the grade of membership of x in A [5]. The collection of fuzzy sets used in defining the linguistic labels composes a fuzzy partition of U. In our models of appliances, the linguistic variables are defined on the variations of both the real (P) and reactive (Q) powers. The use of fuzzy sets allows us to deal with the tolerance of the smart meters and the noise which affects the measures. Further, the linguistic terms permit to coarsely describe the events, thus enabling the modeling of appliance types rather than single appliances.

Once defined a data base (DB) of FSMFTs, we employ a novel load disaggregation algorithm. Whenever meaningful variations of P and Q measured by the smart meter are detected, the algorithm verifies whether these variations are compatible with a possible state transition of one or more FSMFTs, and for each possible transition creates a corresponding configuration of active appliances. Thus, at each instant, several different configurations are possible. By exploiting an appropriate strategy based on the analysis of the compatibility of the sequence of the events with the sequence of the possible states of each appliance, the algorithm reduces dynamically the possible candidate configurations whenever a new event is detected.

We tested our approach by using five appliances plugged in a power strip and measuring the power consumption at the root of the strip for thirty minutes. At the end of the experiment, for each detected event, the disaggregation algorithm outputs a unique correct configuration of the appliances that have been contributing to the total load.

978-3-901882-46-3

II. RELATED WORKS

Two main approaches have been proposed for recognizing and profiling the use of appliances in buildings, namely intrusive load monitoring (ILM) and non-intrusive load monitoring (NILM) approaches. ILM is based on using sensor networks to extract measures directly on the appliances or on some specific position of the load tree. Some ILM prototypes have been recently presented, together with interesting preliminary results, in [1], [6] and [7].

In an NILM system, individual appliance power consumption is disaggregated from single-point measurements provided, for instance, by a sensor or a smart meter. In general, the different appliances are recognized by extracting their "signatures" from the aggregate measure. These signatures are obtained by modeling how the appliances operate. The pioneer work in the NILM contest is based on the analysis of the active and reactive powers and on the detection of relevant variations of these powers [8]. The appliances are described by using standard finite state machines. Similar to several works proposed in the last years [9], the FSMFT-based approach proposed in this paper takes inspiration directly from the NILM scheme introduced in [8]. Unlike these works, however, FTFSM allows: i) handling more effectively the noise which typically affects the power signal provided by the smart meters, ii) defining in linguistic terms the events that characterize the behavior of each appliance, and iii) at each instant, identifying the working states and the corresponding consumptions of the monitored appliances. Finally, at each instant, our disaggregation algorithm can manage multiple configurations of candidate active appliances so as to ensure a high level of robustness and accuracy.

III. THE PROPOSED APPROACH

Two main elements compose the proposed NILM system, namely the DB, which contains the FSMFTs of the appliances, and a novel load disaggregation algorithm. In the following, first we introduce the main concepts related to FSMFTs and, then, we present our strategy for disaggregating the aggregate power signals.

A. *Modelling Appliances by Finite State Machine based on Fuzzy Transitions*

In NILM pioneer work [8], Hart models how each appliance works by using Finite State Machines (FSMs). In particular, the event that describes a state transition is expressed in terms of a crisp variation of the specific electrical features considered in the model. Also, each working state is associated with a crisp power value representing the consumption of the appliance in the working state. Actually, some problems can arise when using standard FSMs for appliance modeling. Indeed, defining an FSM for a specific appliance requires to specify the exact consumption in each state and the exact values of the variations for each state transition. However, it is well known that the actual power consumption, measured for example using a smart meter, can be quite different from the nominal consumption and that this value can vary depending on the other appliances connected to the circuit. Indeed, in power circuits, load dependent power drops can occur, for example, in reaction to a switching event of an appliance.

In order to overcome the aforementioned problems, in this paper we introduce FSMFTs for modeling the appliances. The behavior of each appliance is described by considering both *P* and *Q* components of the power.

We suppose that the working states of each appliance are known but we do not associate them with a specific power consumption value. The states can be extracted by analyzing how the appliance itself or appliances of the same type work. A methodology for identifying the appliance working states is discussed in [10] and is based on the use of bi-dimensional histograms and clustering algorithms.

Let $S=\{S_0,\dots,S_i,\dots,S_N\}$ be the set of the working states of a generic appliance. In order to manage the uncertainty, which characterizes the measurements collected by the smart meters, the transitions from a state to another state are not expressed as crisp numerical variations of *P* and *Q*, but rather through linguistic values defined on the universes $U_{\Delta P}$ and $U_{\Delta Q}$ of these variations. To this aim, we define two fuzzy partitions $FP_{\Delta P}=\left\{A_{\Delta P,1},\dots,A_{\Delta P,T_{\Delta P}}\right\}$ and $FP_{\Delta Q}=\left\{A_{\Delta Q,1},\dots,A_{\Delta Q,T_{\Delta Q}}\right\}$ on the two universes $U_{\Delta P}$ and $U_{\Delta Q}$, respectively, and model a generic fuzzy transition $T(S_i \to S_j)$ from S_i to S_j as a rule:

$$R_{i,j}: \textbf{IF } \textit{State} \textbf{ is } S_i \textbf{ AND } \Delta P \textbf{ is } A_{\Delta P,k_{i,j}} \textbf{ AND } \Delta Q \textbf{ is } A_{\Delta Q,h_{i,j}} \textbf{ THEN } \textit{State} \textbf{ is } S_j \quad (1)$$

where ΔP and ΔQ are linguistic variables defined on the two universes $U_{\Delta P}$ and $U_{\Delta Q}$, respectively, $k_{i,j}\in[1,T_{\Delta P}]$ and $h_{i,j}\in[1,T_{\Delta Q}]$ identify the indices of the fuzzy sets which have been selected for ΔP and ΔQ, respectively, in rule $R_{i,j}$. A complete FSMFT can be completely described by a rule base composed by all fuzzy rules that define the state transitions of the appliance.

An appliance is considered active if it is not switched OFF, i.e. if it is not in the S_0 state. We denote as OFF_ON rules, those rules which describe the transition from the state S_0 to another state S_k, where $k\in\{1,\dots,N\}$. At least one OFF_ON rule must be contained in the rule base associated with each appliance.

Whenever a new couple $(\Delta p_t,\Delta q_t)$ of variations of *P* and *Q*, where $\Delta p_t = P(t)-P(t-1)$ and $\Delta q_t = Q(t)-Q(t-1)$, is measured at instant *t*, all the rules $R_{i,j}$, where S_i is the current state, are fired. The firing strength $w_{i,j}(\Delta p_t,\Delta q_t)$ associated with each rule $R_{i,j}$ is computed as:

$$w_{i,j}(\Delta p_t,\Delta q_t)=T\left(A_{\Delta P,k_{\Delta P}}(\Delta p_t),A_{\Delta Q,k_{\Delta q}}(\Delta q_t)\right) \quad (2)$$

where T is a t-norm [11]. In our experiments, we used the product as t-norm. If $w_{i,j}(\Delta p_t, \Delta q_t) > \gamma$, where γ is a prefixed threshold, then the transition $T(S_i \rightarrow S_j)$ is labeled as *candidate transition*.

More than one candidate transition can be identified for each couple $(\Delta p_t, \Delta q_t)$ of power variations at instant *t*. Further, when dealing with an appliance DB containing several FSMFTs, the number of candidate transitions could rapidly increase. In the following, we will describe an algorithm which is able to handle several candidate state transitions in order to extract the consumption of each appliance from an aggregate measure.

B. The Load Disaggregation Algorithm

The proposed algorithm is based on the analysis of the candidate transitions of the FSMFTs included in the appliance DB. At each instant *t*, a new couple ($\Delta p_t, \Delta q_t$) of power variations is analyzed by the load disaggregation algorithm to verify whether one or more FSMFTs may change their current working state. In practice, the algorithm uses the FSMFTs in the DB as a set of appliance simulators.

We can realistically assume that in a domestic environment, at each instant, just a unique appliance can change its working state, i.e. we do not consider concurrent activations or state changes of the monitored appliances.

Since at each instant *t* more than one candidate transition can be activated, different possible configurations of active appliances have to be concurrently managed by the algorithm. By analyzing the behavior of the different configurations along the time and exploiting the firing strength, the algorithm can make hypotheses on the actual configuration and therefore on the power consumption of each appliance.

The main data structures handled by the load disaggregation algorithm are:

- the configuration of active appliances at instant *t*, that is, a collection of triplets ($S_{i,a}(t)$, $P_{i,a}(t)$, $Q_{i,a}(t)$), where $S_{i,a}(t)$ is the candidate state of the active appliance *a*, and $P_{i,a}(t)$ and $Q_{i,a}(t)$ are, respectively, the estimated real and reactive powers consumed in state $S_{i,a}(t)$ by the appliance *a*;
- the collection *CC*(*t*) of active configurations at instant *t*;
- the list *LC* of collections *CC*(*t*), *CC*(*t*-1), ..., *CC*(1).

When the algorithm analyzes a new couple ($\Delta p_t, \Delta q_t$) at instant *t*, a list of candidate transitions is created for each configuration in collection *CC*(*t*-1). The list is generated by considering:

a) for the appliances in the DB, the OFF_ON rules whose firing strengths are higher than a prefixed threshold, and
b) for each active appliance in the configuration, all the possible transitions from the current state to another state allowed by the corresponding FSMFT.

Let us assume that *CC*(*t*) is empty at the beginning. For each configuration, the following cases can occur:

1. The list of the candidate transitions is empty, because no new appliance can become active or no appliance can change its working state. The configuration is added to *CC*(*t*).
2. The list contains only one candidate transition. If the transition corresponds to the activation of a new appliance, a new triplet is added to the configuration: the state in the triplet is specified in the consequent part of the fired rule and the estimated real and reactive powers are equal to Δp_t and Δq_t, respectively. Otherwise, if the transition is associated with a change of working state, the state of the active appliance in the configuration is updated accordingly and the estimated values of real and reactive powers are updated by adding, respectively, the measured values of Δp_t and Δq_t to the previous values. The updated configuration is added to *CC*(*t*).
3. The list contains more than one transition. For each transition in the list, first a new configuration is generated by duplicating the current configuration, then the new configuration is updated by following the procedure detailed in case 2.; finally, the updated configuration is added to *CC*(*t*).

Once all the configurations of *CC*(*t*-1) have been analyzed, if at least one configuration has been modified for each unmodified configuration added to *CC*(*t*) (case 1 in the previous list of items), the configuration is removed from *CC*(*t*) and from *CC*(*t*-1). Indeed, this configuration is not compatible with the measured event. Further, if the configuration in *CC*(*t*-1) was the unique configuration generated from a configuration in *CC*(*t*-2), then this configuration is removed from *CC*(*t*-2). In general, if all the configurations in *CC*(*z*), with $z<t$, originated from a configuration *c* in *CC*(*z*-1), are removed, then also the configuration *c* is removed from *CC*(*z*-1). Indeed, all these configurations are considered as "inconsistent", because with a high probability they were erroneously generated. In this way, we dynamically reduce the size of the collections *CC(z)*. When the user requests to output the use of the appliances and the corresponding power consumptions, the algorithm shows, for each collection, the survived configurations. In the experiment discussed in the next section, a unique configuration for each collection is survived at the end of the monitoring interval. Thus, the algorithm has returned a unique sequence of appliance use. In the case of multiple survived configurations, currently we show all the alternatives, but we are studying a strategy based on the firing strengths for selecting a unique configuration for each collection.

IV. EXPERIMENTAL RESULTS

We developed a software prototype which includes three main modules: the DB of the appliances, the load disaggregation algorithm and the GUI. The main functions supplied by the GUI are: i) to describe the appliances in terms of FSMFTs, ii) to start the load monitoring, iii) to analyze the different configurations generated by the disaggregation algorithm at a specific time instant.

We collected the aggregate measures of *P* and *Q* by using the Plogg Ext CT 100 smart meter equipped with ZigBee wireless communication and an external split core 100A current transducer [12].

A. Experimental Setup

Five appliances were considered in our experiment. We connected the smart meter, which collects the aggregated power measures, to a power strip where the five appliances were plugged in. The smart meter had a wireless communication with a laptop where our software prototype was running. The sampling frequency was set to 2Hz.

The possible states of each appliance can be derived from the experience in using the appliances and from the use manual of each appliance. We recall that our approach does not need to identify the exact real and reactive powers for each state, but rather needs to coarsely know the variations of real and reactive powers which can generate a state transition for each appliance. Indeed, the identification of the power consumption of each single appliance from the aggregate measure is not performed by comparing the measured values of the two powers with the exact values which characterize each working state, but rather by analyzing the compatibility of the sequence of the measured variations with the sequence of the possible states of each appliance. This allows us to describe how each appliance works without exploiting a specific training phase. Further, the use of FSMFTs as modeling tool permits to make the description of how an appliance works quite general. This description can therefore be associated with appliances of the same type which differ from each other only for some value of the *P* and *Q* consumed in the working states.

Table I summarizes the working states and their descriptions for the five appliances used in the experiment. Table II shows the transition rules identified for each appliance. The labels used in the rules for the linguistic variables ΔP and ΔQ are defined by the partitions shown in Figure 1. As regards ΔQ, we used 14 uniformly distributed fuzzy sets (labelled as FS1,…, FS14) in the interval [-200 Var, 200 Var]. As regards ΔP, since the real power consumptions of the different devices is distributed along different ranges, we decided to use a non uniform partition with 16 fuzzy sets (labelled as FS1,…, FS16) in the interval [-2000 W, 2000 W]. We defined fuzzy sets with a narrower support for low power ranges and with a wider support for high power ranges. As an example, if we consider the fluorescent lamp, the hair dryer and the food cutter, which have moderate real power consumptions, their transition rules are defined using the narrowest fuzzy set of the partition. In contrast, if we analyze the oven and the toaster, their rules are defined using wider fuzzy sets, because their state changes involve high real power transitions.

We would like to observe that only for commodity we labeled the fuzzy sets in the partitions with indexes in ascending order. Actually, these labels could be easily replaced by meaningful linguistic terms, thus making the rules easily interpretable. The interpretability of the linguistic values is the main reason why we adopted a Ruspini partition [13], that is, a partition where for each value of the universe the sum of the membership grades to all the fuzzy sets is always equal to 1. Actually, only some of the fuzzy sets are used in the rules shown in Table II.

TABLE I
DESCRIPTION OF THE WORKING STATES OF THE FIVE APPLIANCES

Appliance	State	Description
Food Cutter	F_0	Switched Off
	F_1	Switched On
Hair Dryer	H_0	Switched off
	H_1	Speed 1
	H_2	Speed 2
Flourescent Lamp	L_0	Switched off
	L_1	Very low Intensity
	L_2	Low Intensity
	L_3	Medium Intensity
	L_4	High Intensity
Oven	O_0	Switched Off
	O_1	Switched On
Toaster	T_0	Switched Off
	T_1	Switched On

TABLE II
TRANSITION RULES USED IN THE FSMTFS OF THE FIVE APPLIANCES

Appliance	IF state is	and ΔP is	and ΔQ is	THEN state is
Food Cutter	F_0	FS10	FS13	F_1
	F_1	FS7	FS2	F_0
Oven	O_0	FS14	FS6	O_1
	O_1	FS3	FS9	O_0
Hair Dryer	H_0	FS9	FS8	H_1
	H_1	FS9	FS8	H_2
	H_2	FS8	FS7	H_1
	H_1	FS8	FS7	H_0
Fluorescent Lamp	L_0	FS9	FS9	L_1
	L_1	FS8	FS6	L_0
	L_1	FS9	FS8	L_2
	L_2	FS8	FS7	L_1
	L_2	FS9	FS6	L_3
	L_3	FS8	FS9	L_2
	L_3	FS9	FS7	L_4
	L_4	FS8	FS8	L_3
	L_4	FS7	FS8	L_0
Toaster	T_0	FS12	FS7	T_1
	T_1	FS5	FS8	T_0

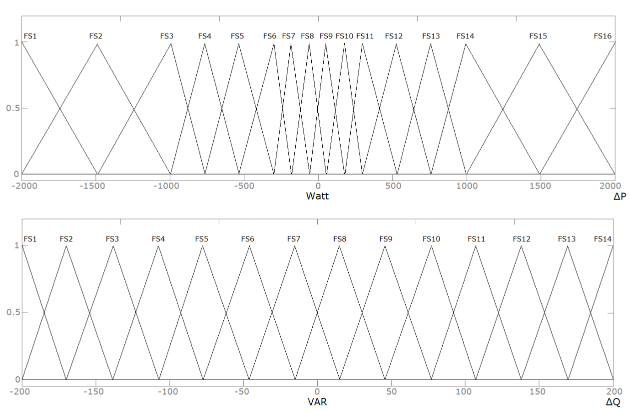

Figure 1. Definition of the linguistic variables ΔP and ΔQ

B. *Evaluation of the Load Disaggregation Algorithm*

In order to evaluate the effectiveness of our algorithm in identifying the correct configuration of active appliances and determining the power consumption of each appliance, the following experiment was carried out. We monitored the five appliances for 30 minutes. In this time interval, thirty-three changes of working states were performed. In Figure 2, we show the aggregate real power measured directly by the Plogg smart meter from the power strip. We tagged each state transition with the corresponding state label of the involved appliance. In this way, the reader can identify which appliance has changed its state. We can note from Figure 2 that during the experiment, up to three appliances were concurrently active. At the end of the experiment, just one configuration was returned by the disaggregation algorithm. In Figure 3, we show the final output provided by the algorithm. We observe that at each monitoring instant, the algorithm managed to correctly disaggregate the power signal and identify the single appliances that were contributing to the aggregate power consumptions.

For the sake of brevity, we have shown only the values of the real power consumed by each appliance during the experiment. Actually, for each appliance and at each time instant, we know also the working state and the value of the reactive power of each appliance.

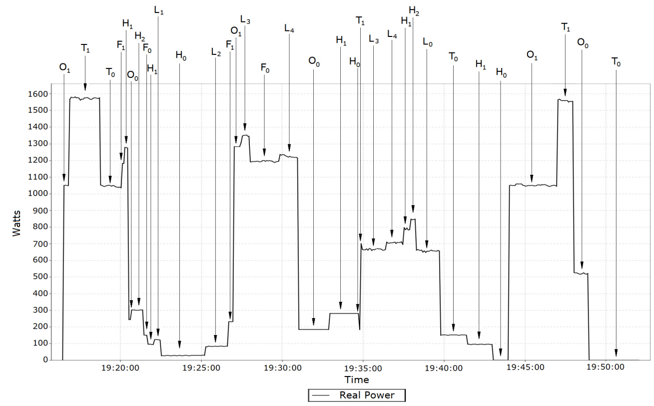

Figure 2. The aggregated real power measured by the smart meter

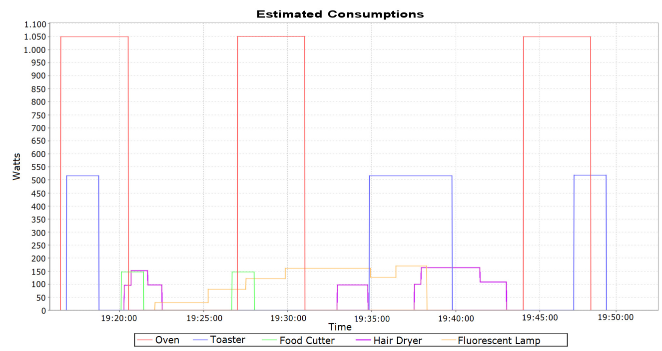

Figure 3. The output of the disaggregation algorithm

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have introduced a novel approach for non-intrusive load monitoring of electrical appliances deployed in domestic environments. The approach exploits a database of appropriately-defined appliance models and an ad-hoc load disaggregation algorithm. The appliances are modeled by using finite state machines based on fuzzy transitions (FSMFT). The disaggregation algorithm exploits the database of FSMFTs for, at each meaningful variation of real and reactive aggregate powers, hypothesizing possible configurations of active appliances.

We have developed a prototype that implements the proposed approach and have tested it on an experimental scenario in which five appliances have been deployed and monitored for thirty minutes. We have shown that at the end of the experiment, our prototype is able to disaggregate the power signal measured by the smart meter extracting the correct power consumption of each single appliance.

In the future, we plan to perform a more intense experimentation by monitoring a larger number of appliances and different appliances of the same type for a longer time interval (at least one day). Further, we are going to improve some aspects of the approach. In particular, we are studying a strategy based on the firing strength associated with each rule and consequently with each working state for selecting a configuration in case of multiple possible configurations recognized by the disaggregation algorithm.

ACKNOWLEDGEMENTS

This work was supported by the Fondazione Cassa di Risparmio di Lucca under the project "**E-tutor**: Un sistema per responsabilizzare l'utente nell'uso dell'energia elettrica".

REFERENCES

[1] G. Anastasi, F. Corucci, and F. Marcelloni, "An Intelligent System for Electrical Energy Management in Buildings," Proc. ISDA 2011, Cordoba, Spain, pp. 702–707, 2011.

[2] "Google PowerMeter," http://www.google.com/powermeter.

[3] "Microsoft Hohm," http://www.microsoft-hohm/.

[4] S. Rogai, "Telegestore Project Progresses and Results," Proc. IEEE International Symposium on Power Line Communications and Its Applications, ISPLC, Pisa, Italy, 2007.

[5] L. A. Zadeh, "Fuzzy Sets," Information and Control. vol. 8, no.3, pp. 338–353, 1965.

[6] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler, "Design and Implementation of a High-Fidelity AC Metering Network," Proc. ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN 2009), San Francisco, USA, pp. 253-264, 2009.

[7] Z. C. Younghun Kim, T. Schmid, and M. B. Srivastava, "Viridiscope: Design and Implementation of a Fine Grained Power Monitoring System for Homes," Proc. UbiComp 2009, Orlando, Florida, pp. 245–254, 2009.

[8] G. Hart. "Nonintrusive Appliance Load Monitoring," Proceedings of the IEEE, vol. 12, pp. 1870–1891, 1992.

[9] M. Zeifman and K. Roth, "Nonintrusive Appliance Load Monitoring: Review and Outlook," IEEE Transactions on Consumer Electronics, vol. 57, pp. 76-84, 2011.

[10] A. Marchiori, D. Hakkarinen, Q. Han and L. Earle, "Circuit-level load monitoring for household energy management," IEEE Pervasive Computing, vol.10, no.1, pp. 40-48, 2011.

[11] E. Klement, R. Mesiar, and E. Pap, Triangular Norms. Dordrecht, The Netherlands: Kluwer, 2000.

[12] Plogg http://www.plogginternational.com/

[13] E. Ruspini, "A new approach to clustering," Inform. Control, vol. 15, pp. 22–32, 1969.