



Department of Electronics and Telecommunications
Master of Science in Information and Communication Technologies
for Smart Societies

Master Thesis

Semi-supervised Event Pairing Method for Non-Intrusive Load Monitoring

Designing an event-based algorithm to separate the power profile of electricity-powered devices from an aggregated power signal in different application environments.

Supervisors

Prof. Michela MEO

M.Sc. Piero MACALUSO

M.Sc. Hamidreza MIRTAHERI

Candidate

Juan Gabriel PIESCHACÓN VARGAS
s273856

ACADEMIC YEAR 2020-2021

This work is subject to the Creative Commons Licence

Abstract

The increasing deployment of energy management systems (EMS) is assisting end-users to become more aware of their energy consumption, intending to mitigate energy waste. In this context, non-intrusive load monitoring (NILM) has emerged as a promising energy management technique to conserve energy. This method aims at distinguishing the individual load consumption from the aggregate power signal measured at a single point. The majority of NILM methods have been applied to residential settings. Low data availability in the industrial domain makes it difficult to study solutions to disaggregate appliances in this kind of environment.

In this work, the household and industrial fields are analyzed and compared. A semi-supervised event pairing method was applied to both scenarios. The algorithm consists of (i) a cluster-based event detection; (ii) the extraction of specific features from each of the events obtained from a small appliance-specific training set; (iii) labels for each detected event of the aggregated power measurement; and (iv) an estimation of the individual load consumption for the present electric devices in the establishment.

The REDD residential and the IMDELD industrial datasets are the considered case studies to evaluate the performance of the proposed approach. For both, residential and industrial applications, it was found that for frequent event appliances, the algorithm accurately detects and classifies the events. Despite this significant result, there is a performance gap for devices that lack frequent events. Thus, an analysis of the strengths and weaknesses of this approach was carried out to define further research ideas or next steps.

Acknowledgements

The amount of support I have received not only during the completion of this thesis but throughout my academic career has been immense.

I would like to thank all those who day by day encouraged me, pushed me, and drove me to achieve my goals. This victory is not only mine but also yours.

I start by thanking my supervisor Prof. Michela Meo, thank you for being always available and for the valuable suggestions. Thanks to Links Foundation for believing in me and giving me the opportunity to work on such an interesting topic. To Hamid and Piero, I think I couldn't ask for better co-supervisors, your selfless contribution to my work gave it an added value that I find it hard to believe that anyone else could have done it, it was a pleasure to work with you.

I would also like to thank my lifelong friends, who no matter the distance always brought a smile to my face. I especially want to thank, my brother from another mother, *mi pana*, Santiago Quiroga, the incessant company you gave me was always a motivator to move forward, thank you very much for being you, I appreciate it very much.

To my fellow students, the dream team, and my travel companions, thanks, Sebastián, Lina, and Felipe. The mutual support we gave each other was key to our academic and personal success, thank you for all the beautiful experiences along the way.

Finally, I would like to give an honorable mention to my parents. The love you have given me from the beginning has been the fuel. I dedicate this work to you as a symbol of what you have achieved with me and what is yet to come. Thank you very much.

Juan Gabriel Pieschacón Vargas, Torino, December 2021

Contents

Contents	II
List of Tables	IV
List of Figures	V
List of Algorithms	VII
1 Introduction	1
1.1 Motivation	1
1.2 Terminology and Problem Statement	2
1.2.1 Terminology	2
1.2.2 Event-based NILM Problem	4
1.3 Organization	4
1.4 Github Repository	5
2 Non-Intrusive Load Monitoring	6
2.1 NILM General Framework	7
2.1.1 Data Acquisition	8
2.1.2 Appliance Feature Extraction	9
2.1.3 Appliance Classification	11
2.1.4 Energy Disaggregation	11
2.2 Implementation Scenarios	11
2.3 Application Environment	12
2.4 Related Work	14
3 NILM Disaggregation Approach Design	16
3.1 Event Detection	17
3.1.1 Event Detector Robustness	20
3.2 Feature Extraction	21
3.2.1 Feature Selection	22
3.2.2 Feature Engineering	23
3.2.3 States of Operation Retrieval	23

3.3	Classification	26
3.4	Energy Disaggregation	32
4	Experimental Results	34
4.1	Dataset Selection	34
4.1.1	NILMTK	35
4.2	Evaluation Metrics	36
4.2.1	Metrics Definition	36
4.2.2	True Positive, False Positive, and False Negative Computation	37
4.3	Residential Use Case	37
4.3.1	Event Detection	39
4.3.2	Feature Extraction	40
4.3.3	Classification	42
4.3.4	Energy Disaggregation	45
4.4	Industrial Use Case	46
4.4.1	Event Detection	47
4.4.2	Feature Extraction	48
4.4.3	Classification	49
4.4.4	Energy Disaggregation	52
4.5	Real-time evaluation	53
5	Conclusions and Future Work	54
5.1	Future Work	55
	Bibliography	56

List of Tables

4.1	Overall event detection performance without discrimination.	40
4.2	Fridge transition intervals before applying the distance-based merging policy.	41
4.3	Appliances' mode transitions after the distance-based merging policy.	41
4.4	Appliance-level event detection performance.	43
4.5	Number of false positives that meet the criteria of not belonging to any ground truth interval.	45
4.6	True vs. predicted energy consumption residential use case.	45
4.7	Number of shared simultaneous events between machines of the same kind.	48
4.8	Machines' mode transitions.	48
4.9	Machine-level event detection performance.	50
4.10	Event detection performance only considering one machine of each. .	52
4.11	True vs. predicted energy consumption industrial use case.	52

List of Figures

1.1	Active cycle and ground state examples.	3
2.1	NILM disaggregation task.	6
2.2	NILM general framework.	7
2.3	Finite state machine model, refrigerator with defrost state example.	8
2.4	Appliance's load signature categories.	8
2.5	V-I trajectory example.	10
3.1	Disaggregation pipeline.	16
3.2	Event detection flowchart.	19
3.3	Event detection example.	20
3.4	Event detector Robustness Conditions. On the left, noise samples clustered as a mode of operation labeled with C2. On the right, two modes of operation (C1 and C2) clustered on a long transient segment.	21
3.5	Time of the day feature.	23
3.6	Power interval definition.	25
3.7	Classification procedure.	26
3.8	Active cycle identification.	27
3.9	Event labeling example.	28
3.10	Refrigerator transition power intervals KDE.	28
3.11	Tiebreak procedure.	29
3.12	First (left) and second (right) compatibility conditions, which are governed by Equations 3.10 and 3.11, respectively.	31
3.13	Matching of remaining unmatched events.	32
4.1	True positive, false positive, and false negative computation.	38
4.2	REDD's house 1 disaggregated and aggregated power signal comparison.	38
4.3	Event detection on a segment of house 1 REDD's aggregated power signal.	39
4.4	False positive and false negative occurrences.	40
4.5	Feature extraction for microwave.	42
4.6	Confusion matrix predicted vs. true label. On the left, the number of predicted events. On the right, percentage of the total of predicted events.	43
4.7	Refrigerator and dishwasher feature resemblance.	44
4.8	IMDELD's disaggregated and aggregated power signal comparison.	46

4.9	Event detection on a segment of IMDELD’s aggregated power signal.	47
4.10	Feature extraction of reactive power transition intervals.	49
4.11	Confusion matrix predicted vs. true label for the industrial use case. On the left, the number of predicted events. On the right, percentage of the total of predicted events.	50
4.12	Correlation matrix between appliances. On the left, residential use case. On the right, industrial use case.	51
4.13	Active cycle duration. On the top, residential duration. On the bottom, industrial duration.	53

List of Algorithms

3.1 Energy disaggregation procedure.	33
--	----

Chapter 1

Introduction

Over the years, concern about energy expenditure has grown to such an extent that consensuses such as the Paris Agreement [35] have been established globally. The agreement sets tight yearly global warming limits, aiming at adapting towards a downward climate change that can only be achieved with the participation of all energy sectors.

Reducing CO₂ emissions is essential to slow global warming, and the active inclusion of renewable energies production is a key factor to reach this goal. The smart grid initiative brings significant actors in the prosumers (consumers and producers) to not only include another renewable energy source but also incorporate consumers into efficient use of resources. It is required a feedback technique that provides knowledge about energy single-device consumption to manage successful end-user participation. In-depth energy expenditure awareness alongside demand side management programs guide consumers into more sustainable energy utilization.

This thesis presents an all-inclusive Non-intrusive Load Monitoring (NILM) algorithm design that intends in covering the challenges of a residential and industrial environment. Even though there have been plenty of NILM approaches contributions, most of them are in the household domain. The ongoing development with Industry 4.0 and Smart Factories emphasizes the importance of employing technology like NILM, enabling a well-informed decision-making process. Therefore, this work seeks to contribute by giving a comprehensive overview of the considerations to apply a NILM solution to a relatively new field like the industrial.

1.1 Motivation

The progressive transition from fossil fuels to renewable energy sources has led to impressive innovations to try to reduce the human ecological footprint. The smart grid paradigm is an initiative that has been forging a new energy usage pattern. One of the emerging features of smart grids is the shift from centralized to distributed decision making. It consists of a multitude of self-interested players that interact

with a network through several information and communication technologies.

Smart meters are important actors for smart grid deployment. They measure the aggregate energy consumption in a household or building and allow a bi-directional communication and power transfer [10]. The gradual inclusion of smart meters answering the smart grid initiative has created a growing interest in assessing how overall energy consumption can be reduced. In this distributed network, end-users play a significant role in effectively saving energy. Demand-side management (DSM) programs aim at involving consumers in the loop, trying to modify their consumption behaviour in function of the power supply.

Final users need feedback to change their consumption habits and hopefully decrease power consumption. As stated in the study [23], a saving of around 4.5 % can be achieved with appropriate consumption feedback. However, the aggregate power profile does not provide sufficient information to take action. Making available information on the individual consumption of each element in an establishment, the consumers can understand which are the most power-hungry appliances and can define strategies for load management.

Aggregate energy consumption can be monitored in two different ways: multi-point and single-point installations [15]. Multi-point energy monitoring consists of dedicated sensors for each of the electricity-powered devices. Measuring power consumption can accurately determine appliance-level utilization with this method. However, the initial investment and future maintenance are reasons that discourage this type of installation.

On the other hand, to acquire the overall electricity expenditure, single-point energy monitoring is installed at a unique location, the electricity main. Research has been pushing to develop new techniques to separate from the aggregated power signal, the individual appliance consumption. Non-intrusive load monitoring has emerged to address this problem.

1.2 Terminology and Problem Statement

This section is devoted to introducing the terminology used in this thesis to avoid ambiguities from inconsistent definitions found in the literature. Furthermore, it is presented the event-based NILM classification problem considered in this research.

1.2.1 Terminology

- **Electric device:** a device that is powered by electricity.
- **Power:** refers to active power; unless indicated differently.
- **Active cycle:** is the interval of time where an appliance is withdrawing energy from the main power source (see Figure 1.1).

- **Ground state:** is the interval of time where no detectable electric device is operating (see Figure 1.1).
- **Operation modes:** is a set of modes, including the inactive mode (OFF mode), in which an appliance can operate.
- **State of operation:** is the amount of power withdrawal in one of the operation modes of an electric device.
- **Steady-state:** in a power signal is when the power draw achieves a stable change and is no longer affected by the transient effects. Specifically, a state of operation is considered stable if the power withdrawal change is equal to or less than ε . ε is a parameter used in the cluster-based event detection, further explained in Section 3.1.
- **Transient-state:** in a power signal is when the power draw has not yet reached a steady state. Meaning that the power consumption behavior is unstable and greater than ε .
- **Aggregated power:** or **aggregated power signal**, is the sum of all contributions by the active appliances in a given time.
- **Event:** is where the aggregated power signal has abrupt changes or a change of an operation mode of active or inactive appliances.

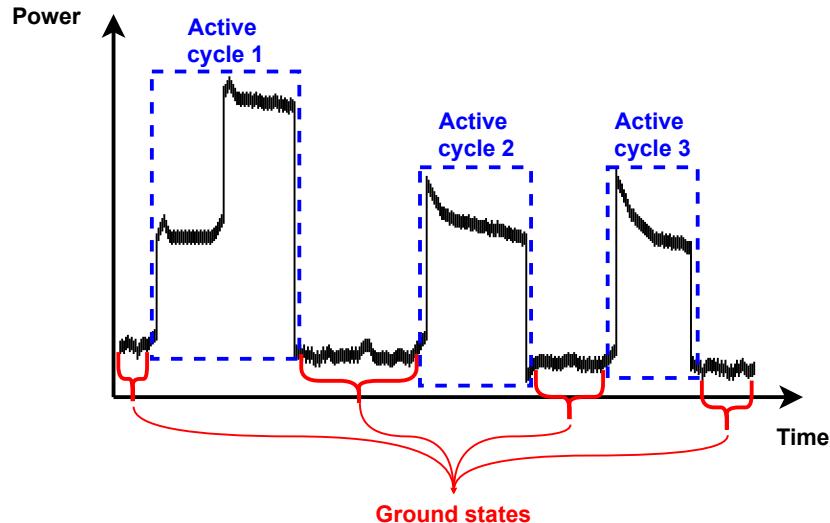


Figure 1.1. Active cycle and ground state examples.

1.2.2 Event-based NILM Problem

The event-based NILM classification is characterized by first recognizing events from the aggregated power signal. For the detected events, assign appropriate labels to each of them. The labelling process is done for each of the active cycles in the aggregated power signal. The switching events within the active cycle are paired and then designated to a specific appliance along with its estimated power consumption. Therefore, this work concentrates on the following research objectives/challenges.

- **Challenge 1:** Accurately detect switching (ON/OFF) events while recording their important features in an environment that may include many simultaneous events and overlapping appliances.
- **Challenge 2:** Pairing and classifying events within an active cycle that may include several appliances and operation modes. The electric devices may be of the same kind, which brings another level of complexity to the problem since they may have a resemblance in their electric signature.
- **Challenge 3:** Defining the physical variables from which the features are derived to obtain the fingerprint of the electric devices.

1.3 Organization

This section aims at presenting the structure of the thesis.

Chapter 1 - Introduction

Apart from presenting the thesis' structure, the current chapter explains the motivation behind disaggregating power profiles from each of the active electricity-powered devices in an establishment. Moreover, it is introduced the terminology, and problem statement underlying this work.

Chapter 2 - Non-Intrusive Load Monitoring

This chapter presents the background knowledge necessary to understand the principal concepts to address the energy load disaggregation problem. In addition, the implications of applying energy monitoring in different environments are explored.

Finally, the chapter ends with a related work review to examine how the energy load disaggregation has been treated.

Chapter 3 - NILM Disaggregation Approach Design

This chapter is devoted to presenting the design of the NILM disaggregation approach adopted in this work. Each of the steps of the NILM architecture will be covered, explaining the intuition behind the design choices.

Chapter 4 - Experimental Results

The effectiveness and accuracy of the NILM solution proposed in Chapter 3 are evaluated in this part of the work. The chapter initiates with the reasons for selecting the residential Reference Energy Disaggregation Dataset (REDD) [26] and the Industrial Machines Dataset for Electrical Load Disaggregation (IMDELD) [2] homologous to test the solution. Then, the chapter follows with the experimental results applied to both use case scenarios, highlighting the differences in employing the solution to contrasting environments.

Chapter 5 - Conclusions and Future Work

This chapter summarizes the most important findings in this thesis work together with a critical analysis. The chapter starts with a short recap of the design of the event-based solution. Subsequently, an interpretation is given to the main outcomes of this work. Finally, the thesis closes with possible future improvements pointing at the current challenges still to be faced.

1.4 Github Repository

The implemented ideas of this research work are publicly available on Github at <https://github.com/links-nilm-thesis-21/load-disaggregation>. People interested on the field can examine the open work and contribute in further improvements even after the conclusion of the thesis.

Chapter 2

Non-Intrusive Load Monitoring

Non-Intrusive Load Monitoring (NILM) is a research field that explores how to separate the energy profile of each of the working appliances in an establishment from the aggregate power signal, as shown in Figure 2.1. Contrary to the Intrusive Load Monitoring (ILM), NILM only uses measurements from a single-point energy monitor installed at the electricity main. NILM has gained a lot of attention due to the cost-efficient installation and reduced maintenance involving the hardware.

As presented in Figure 2.2, NILM consists of four modules [15, 34], namely Data Acquisition, Appliance Feature Extraction, Appliance Classification, and Energy Disaggregation. A detailed explanation of each of them is given in Section 2.1. The remaining of this chapter will focus on explaining NILM general framework, the features used to find a unique signature for each appliance, how the problem can be treated depending on the application domain, and finally the related work in the field that illustrates the main recent ideas tackling NILM. This will be the basis that will provide the necessary tools to understand the results presented in this work.

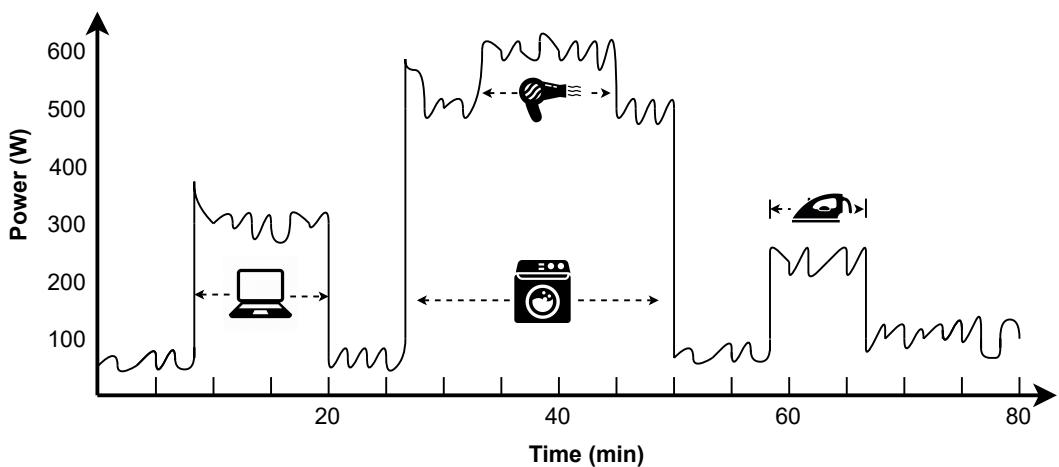


Figure 2.1. NILM disaggregation task.

2.1 NILM General Framework

NILM can be formulated as the break down of the aggregated power signal P_t of a building or residence into individual appliance loads [43]; it can be expressed mathematically as in Equation 2.1.

$$P_t = \sum_{n=1}^N p_t^n \quad (2.1)$$

where p_t^n is the separated appliance consumption that contributes to the overall energy expenditure at time t ; N is the number of appliances (active or inactive) present in the establishment. Therefore, the task of NILM is to distinguish which power elements are in use, at a given time t , to accomplish complete energy disaggregated monitoring.

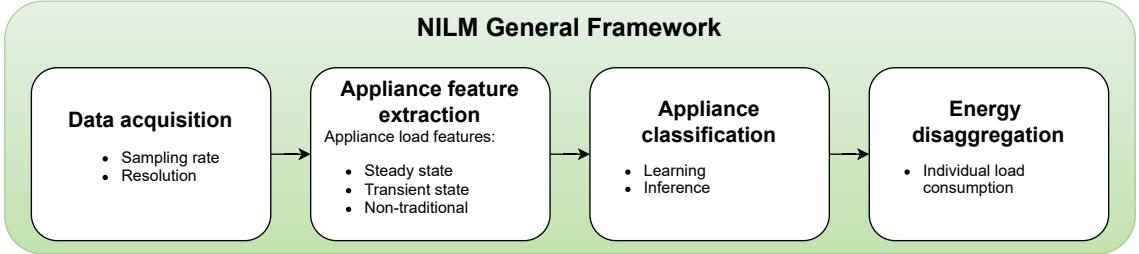


Figure 2.2. NILM general framework.

When working, the specific appliance power waveform could influence energy disaggregation [43]. Given an active cycle for a certain electric device and its modes of operation, appliances can be categorized as proposed by [16] as follows:

- **Type 1:** These appliances only have ON and OFF operation modes. Where the ON state of a specific appliance will maintain a single power range throughout the whole operation.
- **Type 2:** These are appliances with a fixed set of modes of operation (more than two, to distinguish between types 1 and 2). In literature, these appliances are also referred to as Finite State Machines (FSM). As presented in Figure 2.3, each mode of operation is identified by a circle with a name and power range. Allowed transition states are represented by arcs with either positive or negative power withdrawal.
- **Type 3:** Appliances belonging to this category are an extension of type 2 appliances, denominated as continuously variable appliances, which have an infinite number of power states.

- **Type 4:** Additionally, [42, 3] introduce another category. *Permanent consumer devices* as they call them, are appliances that are constantly consuming the same amount of energy and remain active the whole time. In this work, this kind of appliance is not considered for the disaggregation task.

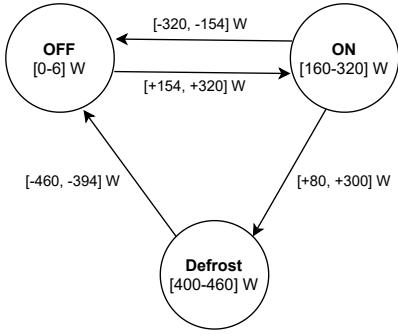


Figure 2.3. Finite state machine model, refrigerator with defrost state example.

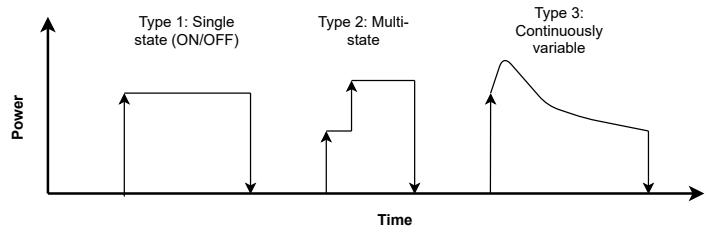


Figure 2.4. Appliance's load signature categories.

Each one of the appliance's load signature categories is depicted in Figure 2.4. The kind of devices present in a determined environment will highly influence the method for solving NILM as will be highlighted later in Chapter 3.

Now in Sections 2.1.1 through 2.1.4 will be explained the combination of the main modules (see Figure 2.2) that compose the NILM architecture.

2.1.1 Data Acquisition

The data acquisition module is in control of sensing the aggregated power consumption in a house or building. Data must be collected at an adequate rate to distinguish unique patterns in the aggregated load, which is motivated by the type of disaggregation algorithm and the area of application [15].

The data acquisition phase is in charge of the granularity and resolution of data. The utilized hardware for the acquisition will determine which features can be extracted in the next step. The energy monitor's resolution is determined by the number of bits with which the analog to digital converter (ADC) is equipped. The minimum change in the signal level becomes smaller with more bits available. Thus, better accuracy of the measurement can be achieved. A high resolution reduces the likelihood of events occurring at the same time in the aggregated signal.

On the other hand, the sampling frequency defines how often a measurement is collected. Data can be acquired at a low frequency (1 Hz or less [33, 11, 8]) or a high frequency (hundreds of Hz to MHz [33]). According to authors in [7], at higher

frequencies, more numbers and types of appliances can be distinguished. Moreover, fine-grained sensing enables the possibility to construct an energy signature with high-frequency features (e.g., transient state features).

2.1.2 Appliance Feature Extraction

Capturing features from the aggregated load is the following step after acquiring the overall energy measurements from the electricity main. First, raw data such as voltage and current waveforms are pre-processed to derive the power metrics, e.g., active and reactive power. This phase may be skipped depending on the power meter's capabilities in internally computing the derived parameters from the fundamental quantities.

Steady-state, transient-state, and non-traditional features are the groups of characteristics to build a unique identification. Steady-state features are available in a specific operation mode of an appliance (excluding the OFF mode). Steady-state feature examples are summarized by [43]:

- **Active (P) and reactive (Q) power change:** These two are widely used for the disaggregation solutions, usually the step-change (ΔP and ΔQ) is calculated to detect the change of modes of operation, which can be achieved with low sampling frequency.
- **Time and frequency domain characteristics of VI waveforms:** Voltage (V) and current (I) waveforms are processed to obtain time and frequency domain features such as the peak and Root Mean Square (RMS) values as well as current harmonics to characterize non-linear loads (i.e., not pure sinusoidal current draw). A high rate acquisition is necessary for these attributes since the waveforms constantly vary in short interval periods.
- **V-I trajectory:** In this category, voltage and current waveforms are also employed. It consists of drawing the *trajectory* of the instantaneous current and voltage normalized values within an active cycle of an appliance (see Figure 2.5). Wave-shape features are obtained from the V-I trajectory, such as the asymmetry, looping direction, area, number of self-intersections, and others.

Steady-state features can require a high sampling rate, whereas for transient-state features, is indispensable this sensing frequency. The transient characteristics are captured from the transitions of modes of operation, and hence they are less overlapping than their steady-state counterpart. A list of examples are presented by [43]:

- **Transient power:** From the active and/or reactive power the repeatability in a transient power profile is used to separate appliances.

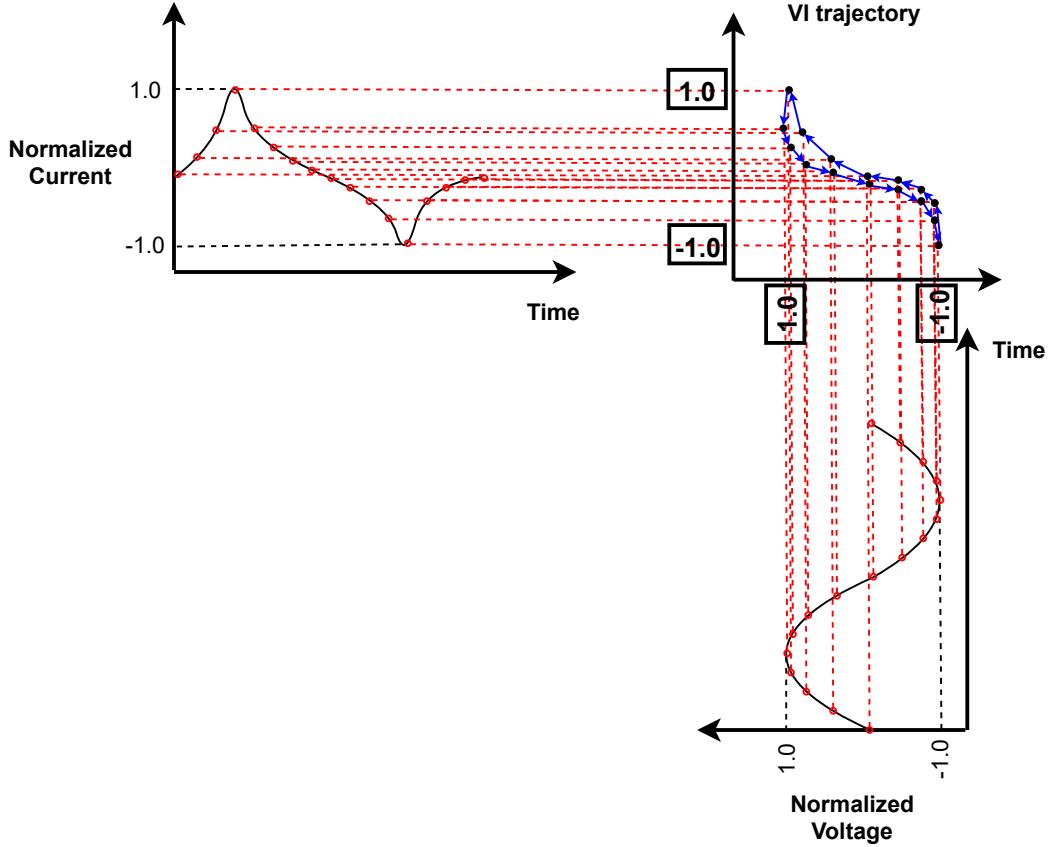


Figure 2.5. V-I trajectory example.

- **Start-up/turn off current transients:** Every time a device is turned on or off there are usually power spikes or overshoots that are distinctive for each appliance. Current spikes, duration, the shape of switching transients are just a few of the characteristics included in this category.

Finally, research [24, 41] has also been including non-traditional features in the disaggregation task. Some of which are not related to electrical variables such as the time of the day, frequency of appliance usage, the correlation among devices' operation. These can be combined with the traditional ones to obtain a better performance.

As reported by authors in [40], NILM solutions can be grouped into event-based and non-event-based approaches. Event-based approaches detect the time instants where the aggregate power signal has abrupt changes, an event. In contrast, non-event-based approaches consider the whole signal. With every power sample, non-event-based methods try to recognize the state of appliances.

The feature extraction in event-based methods occurs every time an event is

identified. The relevant features are recorded for future individual appliance associations. Instead, signal characteristics in the non-event-based techniques have to be continuously registered and processed. Event-based solutions are considered more computationally efficient [40] since appliances are related only to the samples taken from the events and not the whole power signal.

2.1.3 Appliance Classification

Once the features are extracted, now they can be used to classify the individual electrical devices. An appliance is characterized by a unique signature that is obtained by selecting the appropriate features. To recognize the distinctive signature, load identification research work is divided into supervised, semi-supervised, and unsupervised machine learning techniques [43, 39].

The supervised methods rely on labels to train the classifier. In most cases, individual appliance consumption must be acquired, increasing the cost of this solution by adding sub-meters for each of the appliances. On the other hand, unsupervised learning does not require any training and minimal or no previous information. These two characteristics, along with the low setup cost, make desirable unsupervised approaches. However, the shortcomings of applying a supervised learning strategy can be relieved with the often-greater accuracy compared with the unsupervised counterparts [39]. A compromise between the aforementioned approaches can be found on the semi-supervised procedure, which consists of a small amount of annotated data to improve performance on much larger unseen observations.

2.1.4 Energy Disaggregation

Finally, after the process of learning and inferring appliance-level power profiles, it is necessary to assign to each the contribution on the total power consumption. From this information, the end-user can have individual consumption feedback, which enables DSM mechanisms to modify the consumer's demand profile.

Event-based or non-event-based classification methods should propose a solution to add appliance-specific power information to accomplish energy disaggregation.

2.2 Implementation Scenarios

NILM can be exploited in several advantageous ways:

- **Demand response:** Is a class of demand-side management mechanisms that aims at shifting the users' load patterns motivated by power utility's incentive or electricity price programs. This is one of the techniques that can be propelled by the feedback NILM could give, thus the end-user changes their consumption behavior accordingly.

- **Preventive maintenance:** From the learning process in the NILM framework, unique signatures from each of the appliances were extracted. An unknown signature is marked as a suspect for an appliance that needs closer examination.
- **Detection of new appliances:** Linked with the previous point, an unknown isolated load pattern could also indicate the presence of a new device.
- **Fraud detection:** Finally, an unrecognized electricity behavior, can also be associated with possible energy thefts.
- **Occupancy detection:** The activity from the electric devices could suggest the presence of people in an establishment at a given time. This is a sensitive issue since it involves an intrusion into the privacy of users of the electric network.
- **Real time load monitoring:** The individual consumption from the electric devices are displayed to the consumer in real-time. Information such as the energy cost related to each appliance could be connected to demand response initiatives to obtain energy savings while reducing overloads in distribution systems and preventing emergency conditions.

2.3 Application Environment

Electricity monitors have been spreading due to the energy transition occurring globally. Housing units (residential sector), commercial premises (commercial sector), and factories (industrial sector) comprise the establishments of the three categories where a power meter can be installed [20]. In this work, residential and industrial domains are studied for the disaggregation task. Considering the residential datasets [26, 12, 22, 29] and the industrial counterpart [6, 2], an overview of the most significant differences between these two sectors is made.

Dataset availability

Most of the research has been concentrating on household contexts attributable to the extensive efforts in measuring campaigns such as: [26, 12, 22, 29]. Therefore, creating a convenient set-up to develop research in this sector.

Conversely, industrial datasets are scarce. This may be due to companies preferring not to share their data, making progress in this field more difficult.

Appliance temporal dependency

In residential settings, not necessarily do the appliances have a working dependency. Usually, the electric devices run separately with less correlation.

On the other hand, the industrial sector that consists of manufacturing units with fixed machinery has temporal dependencies from industrial processes. The common production lines are an example that machine usage in this environment depends on each other.

Appliance working overlap and event simultaneity

Associated with the previous point, as appliances generally operate independently, the overlap of working devices is sparse and simultaneous events are rare in a household.

In contrast, in the industry, many electric devices are working at the same time. The related interaction between them increases the likelihood of events occurring at the same time.

Appliance activity

On a residential unit, appliances can work at any given time. Provided that inhabitants can utilize powered devices at will, with no time restriction. Power withdrawal from the main power source can occur whenever. Instead, active appliances in an industrial environment are bounded by working days and hours. Hence, on non-working days, no power activity should be detected.

Appliance usage scheduling

Stochastic behavior is found for household appliances with a wide variety of operational programs or adjustable settings such as temperature or intensity. Moreover, device operation is dependent on the usage by the house occupants (duration of appliance usage). Contrariwise, workdays and assignments with machinery are repetitive each day/week. So, appliance predictability becomes easier due to the stabler behavior.

Appliance diversification

In a residence is difficult to find two or more equivalent appliances that satisfy the same purpose. Normally, one electric device of a kind is sufficient for every tenant. Moreover, there is a high variety of devices that fulfill the many daily tasks that an occupant may have. Whereas in industrial settings, the number of assignments is reduced. Although, for the same assignment, several machines are required. Therefore, in this scenario, there is low appliance diversity and high appliance density.

2.4 Related Work

NILM was first defined by Hart in 1992 [16]. Since then, several solutions have been proposed to solve the NILM problem. As mentioned earlier in Section 2.1.2, NILM solutions are broadly divided into event-based and non-event-based. Examples of non-event-based algorithms are the widely used Hidden Markov Model (HMM) [30, 27] and its alternative forms as the additive factorial HMMs [25] or factorial HMMs [24]. The latter was also used for industrial domain purposes [9]. They identified that machines with a large share of the total energy consumed are better disaggregated. A second version of the research was released later [31]. The implemented approach this time was a deep learning generative model, the Wave-NILM. Although a better result with the neural network was achieved in most of the metrics for all appliances, the conclusion was the same. These approaches, however, have the disadvantage of requiring elevated amounts of training data to learn the model. Also, they do not handle well the addition of new unseen devices as the computational complexities increase exponentially [17]. In comparison, event-based methods relieve some processing power as they only allocate resources for the detected events in the disaggregation task [40, 17].

Event-based NILM considers three research categories: supervised, unsupervised, and semi-supervised techniques [43, 39]. When no preliminary information is available, unsupervised algorithms are in charge of clustering the events, hence separating in each group a specific appliance. Subtractive clustering was proposed by [18], where authors demonstrated that type 1 appliances with low sampling rate data were accurately detected without prior knowledge of the number of appliances or the number of different type 1 power profiles. Another approach using agglomerative clustering and genetic k-means was employed by [13] using steady-state active and reactive power change (ΔP , ΔQ) features. However, a common drawback from these unsupervised procedures is the inability of distinguishing type 2 (multi-state) appliances as they form several clusters from the multiple states of operation, which results in creating numerous clusters for the same appliance.

On the other hand, supervised NILM classification algorithms require a training dataset with appliances' consumption profiles data. These preliminary inputs contain information about multiple appliance-specific features that help to construct the structure and parameters of the recognition algorithm. This set of solutions often consider the mode of appliances as class labels. The methods using these parameters range from K-Nearest-Neighbours [36], multi-label classification [38] to deep learning [37]. Unlike the unsupervised techniques, the supervised analogs present more accurate results, even with type 2 appliances. Yet, the prevalent difficulty is the generation of accurate appliance-specific models with small quantities of training data [1].

Acquiring insightful information from small amounts of data is encouraged to overcome the already mentioned problems from the supervised and unsupervised

methods. A semi-supervised procedure implemented by [1] utilizes the modes of operation of each electric device as class labels with low training data volume achieving state-of-the-art results. Therefore, a semi-supervised scheme reduces the effort of acquiring extensive quantities of training data and still produces accurate prediction labels.

Chapter 3

NILM Disaggregation Approach Design

This chapter presents the design of a semi-supervised event pairing method for energy load disaggregation. The solution will cover the appliance feature extraction, appliance classification, and energy disaggregation modules from the NILM general framework presented in Section 2.1. The data acquisition module is not considered as the algorithm performance is verified on already existing data. Nevertheless, the decision of selecting the datasets will be further explained in Section 4.1, analyzing the data parameters that also motivate the construction of the recognition algorithm.

The building blocks that describe the disaggregation pipeline, Figure 3.1, start with the data extraction, followed by the cluster-based event detection module, then the feature extraction is performed based on the detected events. These first three components are common for the learning and inference procedures. The modules that concern only the inference side are event pairing (where appliance classification occurs) and energy disaggregation. Additional clarification on the design of each of the blocks will be given from Sections 3.1 through 3.4.

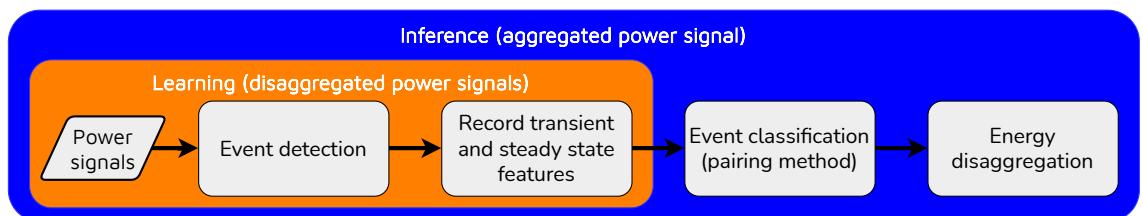


Figure 3.1. Disaggregation pipeline.

3.1 Event Detection

The event detection part is the core of this solution. The training and test phase depends on the event detection. In training, it is used for appliance-specific feature extraction, while in testing, the detected events are divided into the available electricity-powered devices given their features. In any event-based solution, incurring false detected events or not distinguishing them will affect the rest of the blocks. Inevitably, it is fundamental to build a good event detector that will provide the basis for the disaggregation task.

A great part of detectors identifies events based on the difference of two or more consecutive samples [14, 32, 28], and significant value change may not be sufficient for identifying type 3 appliances' events, where the change is slow and characteristic in an industrial environment [21]. Moreover, many rely on pre-processing (e.g., filtering) the signal to increase the detection accuracy [1, 32]. The proposed event detection method intends to skip the filtering phase and accurately define the transition time intervals instead of employing the typical change-point detection. Additionally, the detector can handle active and reactive power time-series data, considering that accuracy also depends on the power features. Reactive power can give another dimension to distinguish events from appliances that have similar active power curves [32]. However, if only active power is available, the algorithm can still work.

The event detector consists of a clustering algorithm applied recurrently to the active and the corresponding reactive power samples. The solution efficiency can be reduced as the signal's frequency increases. To reduce the processing power, the active (P) and reactive (Q) power signals are transformed to logarithmic power signals (P_l and Q_l) based on Equation 3.1. From this procedure, the power signal is enclosed in a narrower range, which results in reduced computation time [4].

$$X_l = \begin{cases} \ln(X), & X > 0 \\ 0, & X = 0 \\ -\ln(-X), & X < 0 \end{cases} \quad (3.1)$$

where $X \in \{P, Q\}$

The desire for real-time aggregate signal separation implicates that the clustering algorithm must be computationally efficient so that it can be recurrently utilized in consecutive time intervals of the power feature signals. Density-based spatial clustering of applications with noise (DBSCAN) is the selected algorithm for this purpose. The cluster-based solution aims at grouping two consecutive steady states, i.e., two different operation modes in a given time interval. If two operation modes are recognized within the time interval, it means that an event occurred. DBSCAN is constructed to distinguish outliers or noise. Therefore, given the time interval where two consecutive steady states were detected, the outliers (if any) represent the transient state samples.

The time interval must be selected, so *exactly* two steady states are contained.

In this context, an expanding window with increasing width is used. The initial window size depends on one parameter of DBSCAN, $minPts$. $minPts$ defines the minimum number of data points that should be included in a group to be considered a cluster and is defined as the product of the minimum time duration of a steady-state ΔT_{min} and the sampling frequency of the data f_s (see Equation 3.2). Then, the initial window size is two times $minPts$ plus one sample, which is the least number of data points to build up to two clusters. From there, the window size increases by one sample every iteration while applying the clustering algorithm in each incremental step until two clusters are found. In the next iteration, the window size returns to its initial size after detecting both modes of operation.

$$minPts = \Delta T_{min} \cdot f_s \quad (3.2)$$

The other parameter that defines DBSCAN is ε , which determines the maximum distance between two samples for one to be considered as the neighborhood of the other. In Equation 3.3, this parameter is defined as the average of the euclidean distances between the consecutive samples inside the expanding window plus two times the standard deviation of the euclidean distances between the consecutive samples inside the expanding window. The standard deviation part is a safety margin to consider inside steady-state data points affected by noise.

$$\varepsilon = mean \left(\sqrt{(l_i - m_i)^2} \right) + 2 \left(std \left(\sqrt{(l_i - m_i)^2} \right) \right) \quad i = 1, 2, 3, \dots, n \quad (3.3)$$

where $l, m \in \{P, Q\}$; m is the previous data point of l . Consequently, l is the succeeding sample of m . i can take any integer value from 1 to n , and n is the total number of samples inside the expanding window.

In this application, the DBSCAN parameters have a physical meaning. ε becomes an adaptive distance depending on the samples bounded in the expanding window. This dynamic distance adapts to the average changes in the steady-states, so when an uncommon abrupt change is observed is labeled as a transient sample (an outlier) or noise. On the other hand, $minPts$ considers clusters according to the sampling frequency of the power signal. Both ε and $minPts$ make the event detector noise resistant, which avoids performing initial filtering to the signal.

Additional remarks on the cluster-based event detector:

- To keep the computational complexity low, the expanding window must not exceed 1350 samples. The number of samples was found to be sufficient to avoid overloading the computational capabilities. If the number of samples is surpassed, the window size returns to its initial size.
- Not sufficiently long-lasting active cycles require special treatment as they can be mistaken for noise. Once two consecutive steady-states are identified, in the next iteration, the expanding window will shift back to the first sample

of the second steady-state instead of continuing to scan the time series from that point (see Figure 3.3). As a result, correct detection of the falling event. Occasionally, the $minPts$ parameter will not be enough to detect a steady state if the search is initiated at the same point where the second steady-state of the last iteration ended.

The flowchart of the cluster-based algorithm is illustrated in Figure 3.2. Notice that the blue and red process blocks are not related to the event detection. The feature extraction block is common for learning and inference, while appliance classification and energy disaggregation are only considered for the inference part.

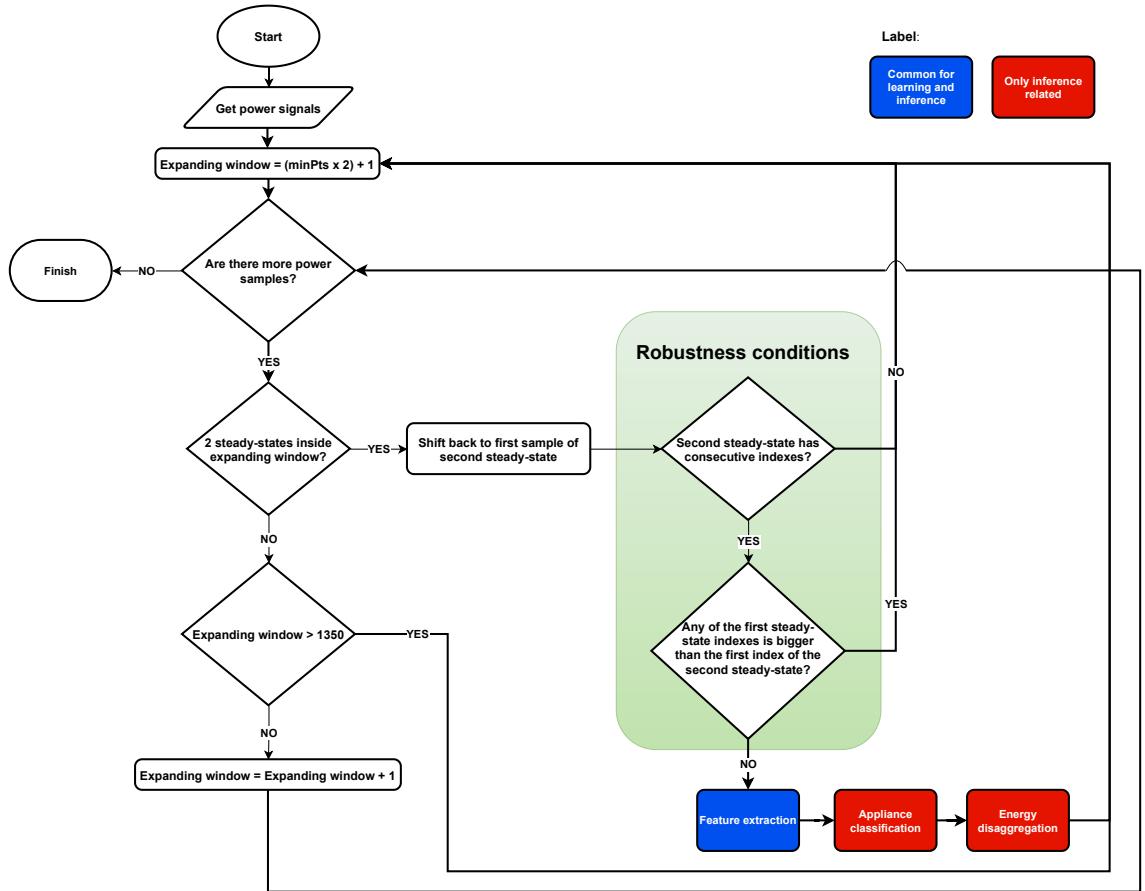


Figure 3.2. Event detection flowchart.

To provide a better explanation of how the algorithm works, Figure 3.3 presents an illustrative example of an active power signal which explains how the events are detected. In this example, the sampling frequency is 1 Hz. The Figure show when the first two steady-states were found and a rising event was identified (upper part of the illustration). In this case, the expanding window initial size was 11 samples

and grew up to 14, which is when both modes of operation were detected. The red data points represent the transient rising state as they do not belong to either of the steady-states contained in ε . Then, the window size returns to its initial size, and the clustering process starts from the first sample of the second steady-state. The sequential window increment goes up to 16 samples where the next two operation modes are found (lower part of the illustration).

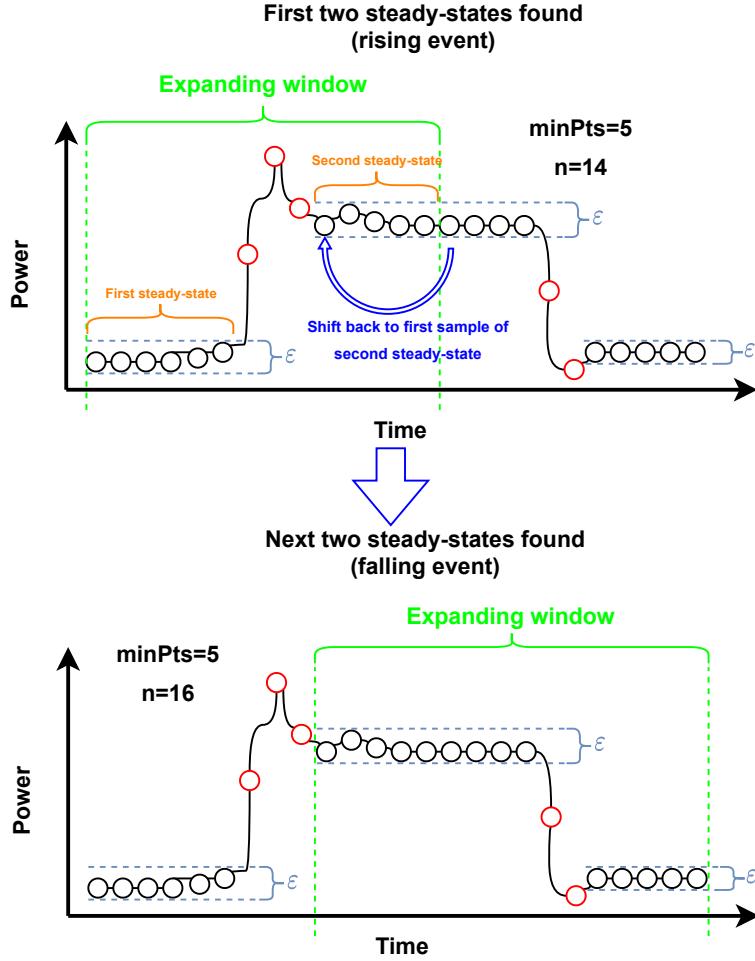


Figure 3.3. Event detection example.

3.1.1 Event Detector Robustness

Two further refinements were employed to achieve a more robust event detection. Not sufficiently separated abrupt changes can be mistaken as a steady-state as shown in the left-most part of Figure 3.4. The non-sequential samples are confused with a mode of operation. However, in this case, the data points are noisy intervals. The

algorithm evaluates if the indexes of the steady-states are consecutive to prevent this from happening; that is the first *robustness condition* surrounded by the green rectangle in Figure 3.2.

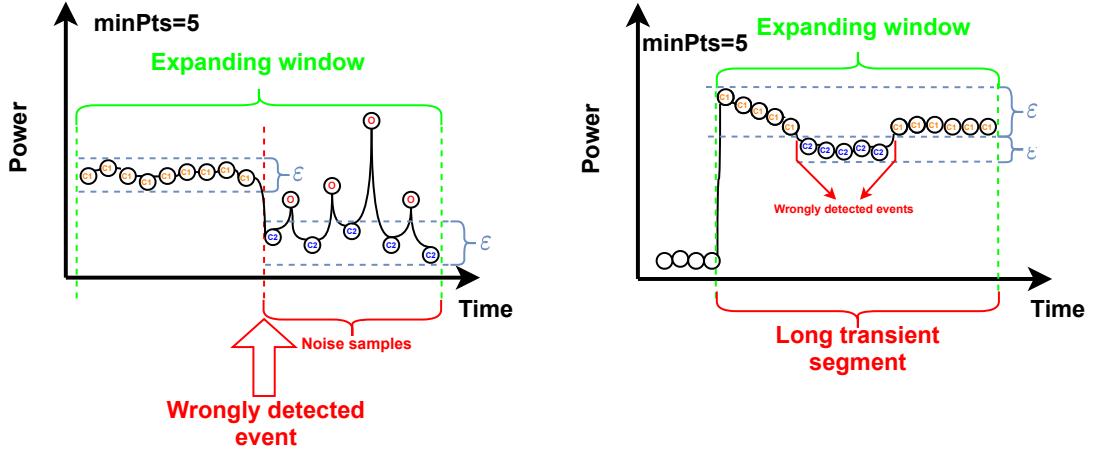


Figure 3.4. Event detector Robustness Conditions. On the left, noise samples clustered as a mode of operation labeled with C2. On the right, two modes of operation (C1 and C2) clustered on a long transient segment.

The second enhancement consists of dealing with type 3 appliances. Type 3 devices typically have long transients. All first steady-state samples should chronologically happen before the second steady-state data points to mitigate wrongly detected events generated by a long transient (a false positive). Therefore, as shown in the right-most part of Figure 3.4, if any of the indexes in the first steady-state samples is higher than the first index sample of the second steady-state, then the event is a probable false-positive (second *robustness condition* surrounded by the green rectangle in Figure 3.2).

3.2 Feature Extraction

The feature selection and collection are crucial for appliance inference. From an appropriate choice of features, a unique signature is obtained. Thus, recognizing distinctions is easier among electricity-powered devices and can respond to part of challenge 1 stated back in Section 1.2.2. States of operation are a generally utilized characteristic in the NILM task, that is, the power interval corresponding to an operation mode [1]. However, power overlap among the electric devices may occur. A more effective disaggregation comes from acquiring extra features, hence providing additional dimensions for appliance differentiation [43].

As mentioned earlier, feature extraction is a common procedure for training and testing. Regardless, in both cases, the extraction is treated differently. In training, a small dataset is used to collect all the features and learn from that appliance-specific models. Conversely, for testing, the power attributes are obtained to label the electric device in the classification procedure (Section 3.3). Notice from Figure 3.1 that for learning, the single consumption contribution from each device (the disaggregated power) is utilized, whereas for inference, only the aggregated signal.

3.2.1 Feature Selection

The approach considers the three feature categories, i.e., steady and transient state and non-traditional attributes. Specifically, the selected steady-state property is the power interval $\Delta\Pi$ corresponding to an operation mode. Apart from gathering the states of operation, other significant attributes are collected from the transient section. Selected features are the transient state duration ΔT_Ψ , transient spike $\delta\Psi$, transient power change $\Delta\Psi$. With this transient set, the intention is to find fewer overlapping characteristics as compared to the steady-state complement. The features are computed in the active P and reactive Q (if available) power signals as in Equations 3.4 through 3.7. Finally, the chosen non-traditional feature is the time of the day usage. It was chosen because it can prove significant to *well-behaved* appliances, such as machines in an industrial environment where tasks are repetitive each week. Moreover, it was shown in [24] that non-power features have a positive impact in discriminating electric devices with similar load characteristics.

$$\Delta\Pi^X = \text{mean}(\Pi_2^X) - \text{mean}(\Pi_1^X) \quad (3.4)$$

$$\Delta\Psi^X = \max\Psi^X - \text{mean}(\Pi_1^X) \quad (3.5)$$

$$\delta\Psi^X = \max\Psi^X - \text{mean}(\Pi_2^X) \quad (3.6)$$

$$\Delta T_\Psi = T_{\Pi_2}(0) - T_{\Pi_1}(N_{\Pi_1}) \quad (3.7)$$

In Equations 3.4 through 3.7, $X \in \{P, Q\}$, Π_2 and Π_1 are the first and second steady-state segments, respectively. Ψ is the transient section, T_Π is the steady-state timestamp, so $T_{\Pi_2}(0)$ is the second steady state timestamp evaluated in the first sample. Finally, N_{Π_1} is the total number of samples in the first steady-state, meaning that $T_{\Pi_1}(N_{\Pi_1})$ is the first steady state timestamp evaluated in the last sample.

Every time an event is detected, the features are distinguished between a rising or falling characteristic. The active power interval $\Delta\Pi^P$ is employed for this purpose. If $\Delta\Pi^P > 0$, then the features are rising event attributes. Otherwise, they are a falling event characteristic. After the binary categorization, while training, the features

are stored for each device. On the other hand, in testing, the features from the aggregated power signal are temporally stored and examined for further appliance designation.

3.2.2 Feature Engineering

As stated in Section 3.2.1, the time of the day usage could be a helpful feature to exploit a possible periodic trend for the time-series data. If any periodic behavior is present, the assumed meaningful frequency is a daily periodicity. Even though the patterns in a household are more stochastic than in the industrial scenario, it might still have a similar behavior from one day to another. From the industrial side, the assignments are repeated on a daily basis (see Section 2.3).

To extract the daily frequency, first, the index column (i.e., the `DateTime` index) is converted into a timestamp in seconds. However, the timestamp in seconds is not useful for the model because it is just a value that increases over time. The approach adopted to make the time a suitable feature was to map the daily periodicity to a sine and cosine function, Figure 3.5. As a result, another two useful features (signals) were added to the model: “Day sin” and “Day cos”.

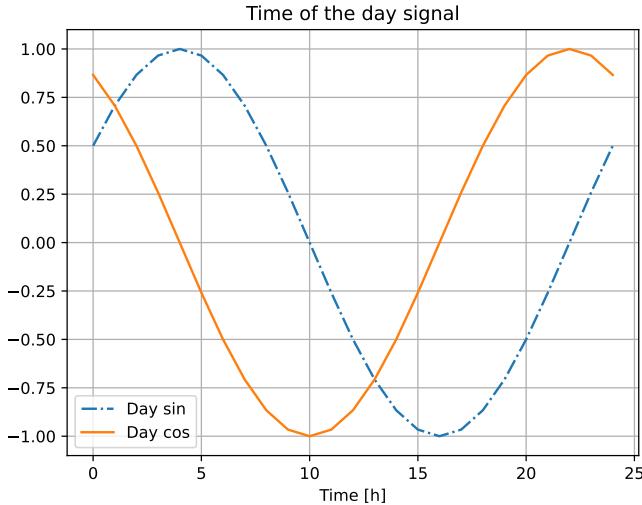


Figure 3.5. Time of the day feature.

3.2.3 States of Operation Retrieval

Similar to authors in [1], to train this solution, the states of operation of an appliance are extracted automatically; instead of using handcrafted methods (e.g., visual power inspection or using the datasheet of the appliance). The learning employs a

cluster-based approach to separate the states of operation using an agglomerative clustering procedure. In particular, for grouping, Ward's linkage (WL) method [19] was executed.

The grouping method states that the sum of squares will increase depending on the cost (see Equation 3.8) of merging two clusters, A and B , separated at a given distance.

$$\begin{aligned}\Delta(A, B) &= \sum_{p \in A \cup B} \|p - m_{A \cup B}\|^2 - \sum_{p \in A} \|p - m_A\|^2 - \sum_{p \in B} \|p - m_B\|^2 \\ &= \frac{n_A n_B}{n_A + n_B} \|m_A - m_B\|^2\end{aligned}\quad (3.8)$$

where m_A , m_B , and $m_{A \cup B}$ are centroids of clusters A , B , and $A \cup B$, respectively, and n_A and n_B are the size of clusters A and B .

A new cluster is generated by agglomerating group points A and B at minimum cost $\Delta(A, B)$ by evaluating any two clusters. The process is repeated until just one object is left or stops in the desired number of data point sets. The non-negative cost $\Delta(A, B)$ will increase as the groups are merged together. One of the often-used stopping conditions to determine the optimal number of groups is the so-called *elbow method*. Particularly, for this application, the optimal number of clusters is achieved when the increasing objective function becomes “too costly” compared with the previous merging stage.

The main drawback with the elbow grouping procedure is the susceptibility to unbalanced data [1]. Considering the cost function in (3.8), the combination of clusters is not interrupted where small-numbered clusters are present. The cost is not too high to merge them, even if their centroids are far apart. In power signals, there may be modes that do not occur frequently, which may lead to combining with another cluster that is considerably separated. Therefore, the consolidation of the states of operation should prevent the union of distant power intervals.

Following the aforementioned state of operation grouping guidelines, the WL method is applied to the disaggregated power data considering $K = 10$ clusters. $K = 10$ was chosen under the assumption that appliances cannot have more than ten modes of operation. Additionally, after clustering the ten operation modes, a *distance-based* merging policy as proposed by [1] is adopted. The technique consists of grouping the sufficiently close power intervals. First, the ten centroids are computed and sorted in descending order with the obtained clusters from the WL method. Second, the technique groups together the power intervals whose centroids distance is less than 15% of the highest centroid. If the condition is met, then that power interval is merged with its corresponding highest centroid. The merging is performed until there are no more centroids to evaluate.

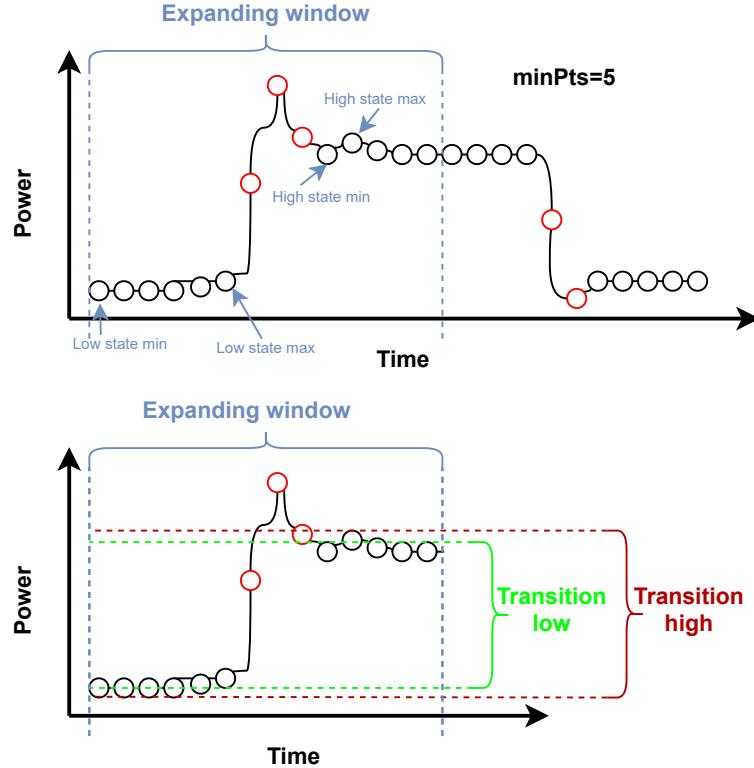


Figure 3.6. Power interval definition.

Transition intervals of appliances

As presented in Figure 3.6, the power interval that characterizes the change from one mode of operation to another can range from the minimum power data point of the lowest steady-state LS_{min} to the maximum power point of the highest steady-state HS_{max} , namely the “Transition High Interval”. The interval can also be included in the range from the maximum power point of the lowest steady-state LS_{max} to the minimum power sample of the highest steady-state HS_{min} , i.e., the “Transition Low Interval”. The transition high and low intervals are recorded from the detected events to extract a wider power change. The agglomerative clustering procedure plus the distance-based merging policy is applied for the stored transition high and low intervals, so the mode transitions for each electricity-powered device are calculated.

A mode transition is defined as the power range from a high state of operation $\text{HS} = [\text{HS}_{\text{min}}, \text{HS}_{\text{max}}]$ to a low state of operation $\text{LS} = [\text{LS}_{\text{min}}, \text{LS}_{\text{max}}]$. Then, the power interval of this transition is defined as Equation 3.9.

$$[\text{MT}_L, \text{MT}_H] = [\text{HS}_{\text{min}} - \text{LS}_{\text{max}}, \text{HS}_{\text{max}} - \text{LS}_{\text{min}}] \quad (3.9)$$

3.3 Classification

This section is dedicated to explaining the proposed event pairing method to classify electricity-powered devices present in an establishment. The classification procedure is made up of several steps illustrated in Figure 3.7.

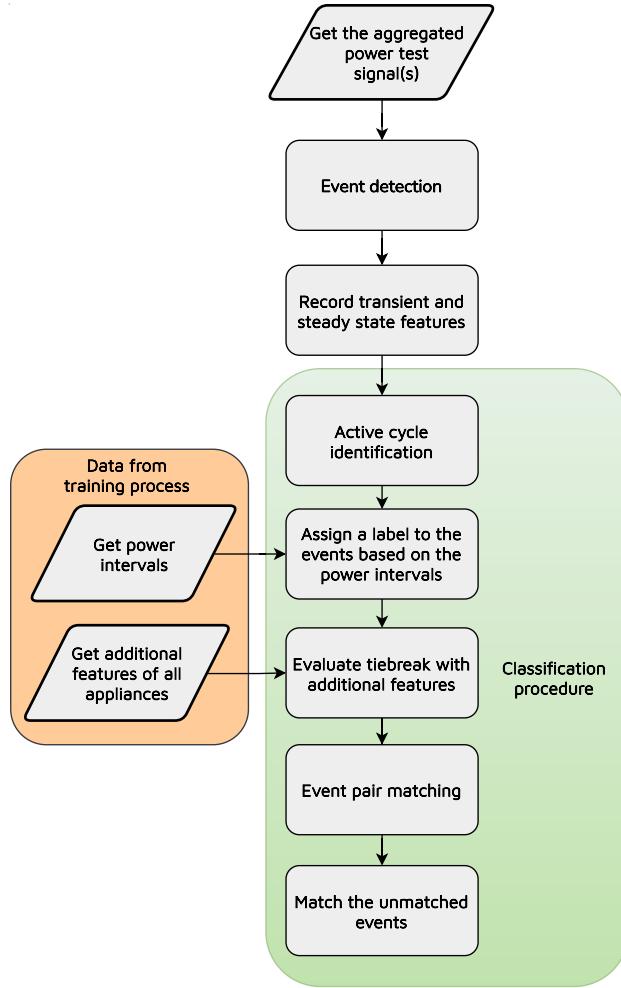


Figure 3.7. Classification procedure.

After recording all the relevant features of an event, the classification procedure starts. Real-time disaggregation monitoring is a requisite to give effective feedback to the end-user. For that reason, instead of waiting until the end of the day to evaluate the stored events' features, the method begins with the **evaluation of events inside an active cycle**. The search space is significantly reduced under the assumption that an active cycle lasts for the activity of an operating appliance. Moreover, the ground states are not considered, which diminishes the computational overhead.

The start of an active cycle is determined by the detection of a rising event, followed by saving the average power consumption of the first steady-state, i.e., the ground state. Next, if a falling event is detected, check if the power withdrawal of the second steady-state (candidate ground state) is lower or equal to the previous ground state (see an example of active cycle identification in Figure 3.8).

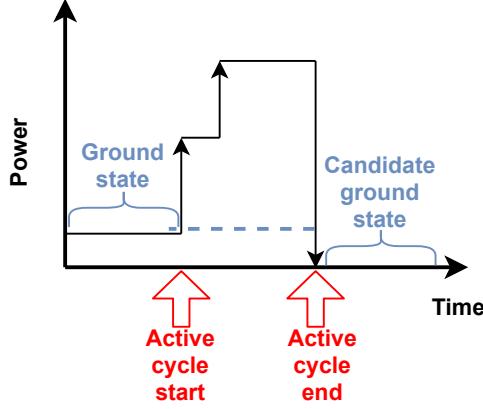


Figure 3.8. Active cycle identification.

After an active cycle has ended, the **label assignment** initiates based on the power intervals retrieved from the training process (see Section 3.2.3). Each event within the active cycle is subject to labeling. The power intervals are collected and associated with the respective event. The event's power interval feature $\Delta\Pi^P$ obtained from the aggregate power signal is confronted with every mode of operation transition $[MT_L, MT_H]$ to relate the mode of operation with the event. The event is assigned to the power interval if $\Delta\Pi^P$ lies in $[MT_L, MT_H]$.

The approach defines a binary-valued matrix M_a of dimensions $N_{MO} \times N_e$, where N_{MO} is the rows containing the modes of operation gathered in training, and N_e is the columns with the events enclosed in the active cycle. A value of 1 in M_a represents that a power interval was designated for a corresponding event.

As discussed in Section 3.2, the power interval is a feature that overlaps among devices. Therefore, one event (column) in M_a could have several associated power intervals (rows). Still, some events may not be included in any of the intervals. In this case, the event is associated with the closest power interval. Figure 3.9 presents an example where four events were detected in an active cycle. Each of the events was labeled with its corresponding power interval. However, for the first event, multiple labels were assigned.

A **tiebreak strategy** was employed using the additional transient and non-traditional features to deal with the power overlap. In this respect, the power characteristics are subjected to a non-parametric probability density estimation.

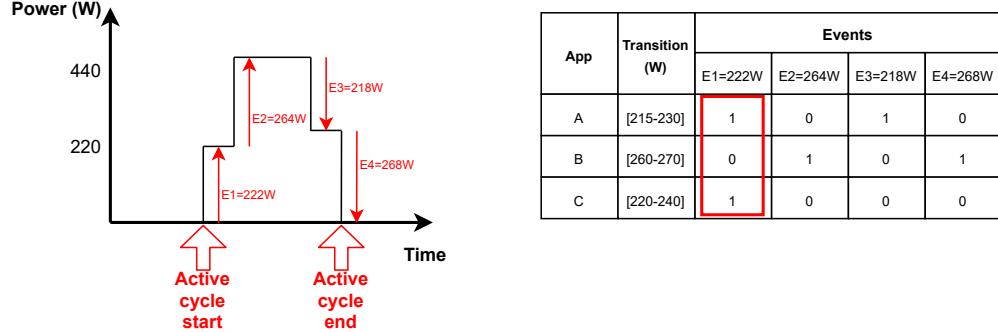


Figure 3.9. Event labeling example.

The probability density is the relationship between observations and their probability. The density of a group of samples forms a shape denominated as a probability distribution, and for an unseen measurement, the probability is calculated with a probability density function (PDF). The PDF will estimate how likely a given observation is depending on a related sample of data.

In this context, an event's feature has a related probability distribution (formed by the stored samples from the training process). However, the stored features do not resemble a common probability distribution, such as the case when the data has two (bimodal distribution) or more (multimodal distribution) peaks, e.g., power intervals of a refrigerator (see Figure 3.10), which leads to a non-parametric density estimation to compute the related PDF. Specifically, kernel density estimation (KDE) is used as a non-parametric approach.

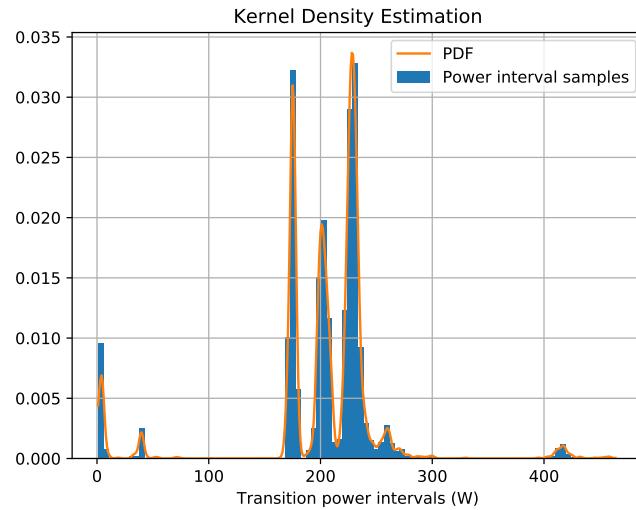


Figure 3.10. Refrigerator transition power intervals KDE.

The tiebreak strategy is as follows:

1. Discriminate the events that conflict with two or more power intervals.
 - (a) Obtain all the power features related to that event.
 - (b) Retrieve the stored power samples (from training) of all the features related to the power intervals with conflict.
2. For each of the feature samples extracted from training, estimate a PDF through KDE.
3. For each of the measured features, calculate the logarithmic probability (to prevent computational underflow) with the PDF from step 2.
4. Sum the logarithmic probability of each of the features.
5. Choose among the overlapping power intervals the appliance whose probability score is higher, i.e., assign a zero in M_a to the corresponding column and row to the power intervals that scored a lower value than the maximum probability score.

As a final remark, the tiebreak procedure evaluates the joint probability of the features to determine which of the power intervals corresponding to an appliance is more likely to be associated with a given event.

Figure 3.11 shows an example of the tiebreak procedure following the event labeling case in Figure 3.9. In the example, after evaluating the sum of logarithmic probabilities for appliances A and C, appliance A obtained a greater score. Thus, the 222 W event was designated to power interval [215 – 230], which corresponds to appliance A.

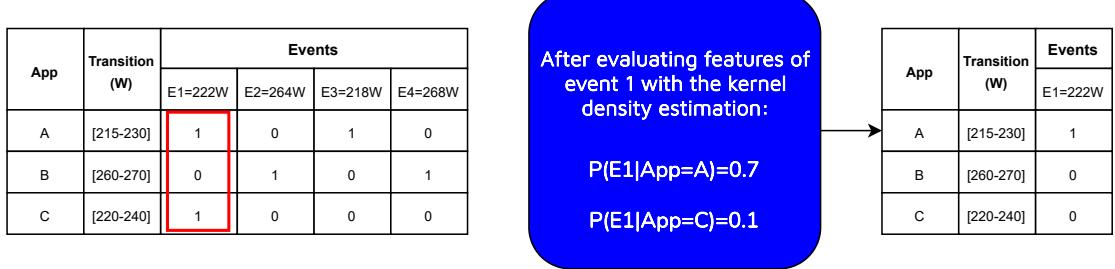


Figure 3.11. Tiebreak procedure.

Unfortunately, the earlier tiebreak method is not enough for low-frequency datasets. In a low-frequency time series, the likelihood of finding transient data points is reduced, especially for falling events. So, in the expanding window presented before,

occasionally, there will be only steady-state samples, removing the possibility of calculating transient state characteristics. For this reason, an **event paring approach** is proposed to mitigate the negative effects of the shortage of transient samples.

The method consists of matching rising (ON) with falling (OFF) events of the same power appliance within an active cycle. When an active cycle ends, it is necessary to check the compatibility between events. Three conditions were formulated to assess this compatibility.

The first condition is related to assessing the absolute power change, where the average steady-state power after a falling event P_{mean}^- is bounded in the steady-state power before a rising event as defined in Equation 3.10 and illustrated in the left-most part of Figure 3.12.

The second compatibility condition treats the relative power changes. This condition manages when two or more appliances are operative in the same active cycle. The events may not occur in order, so the relative changes cannot be associated chronologically. However, an event of the same appliance has equivalent relative rising and falling power change. The relative falling power change ΔP^- must fall in the relative power change of a rising event ΔP^+ for the events to be considered a match (see Equation 3.11 and the right-most part of Figure 3.12).

$$P_{Low}^+ - (P_{Low}^+ \times 0.02) \leq P_{mean}^- \leq P_{High}^+ + (P_{High}^+ \times 0.02) \quad (3.10)$$

In Equation 3.10, P_{Low}^+ is the minimum power sample in the steady-state before the rising event, whereas P_{High}^+ is the maximum power data point in the steady-state before the rising event.

$$\Delta P_{Low}^+ - (\Delta P_{Low}^+ \times 0.02) \leq \Delta P^- \leq \Delta P_{High}^+ + (\Delta P_{High}^+ \times 0.02) \quad (3.11)$$

In Equation 3.11, ΔP_{Low}^+ is the relative power change range between the minimum power sample in the steady-state posterior to a rising event and the maximum power data point of the preceding steady-state of the rising event. ΔP_{High}^+ is the relative power change range bounded by the maximum power sample in the steady-state following a rising event and the minimum power data point of the previous steady-state of the rising event.

The third and last matching condition evaluates if the event pair corresponds to the same appliance. This means that the rising and corresponding falling event power intervals should belong to the same electric device. Following the four-event scenario (Figures 3.12 and 3.9), rising event 1 can be paired with falling event 3 because both power intervals belong to appliance A. Likewise, the power interval of appliance B allows rising event 2 to be matched with falling event 4.

Overall, to pair a rising and falling event, at least one of the first two conditions must be met, and the third condition must be met.

The whole event pairing procedure is defined as follows:

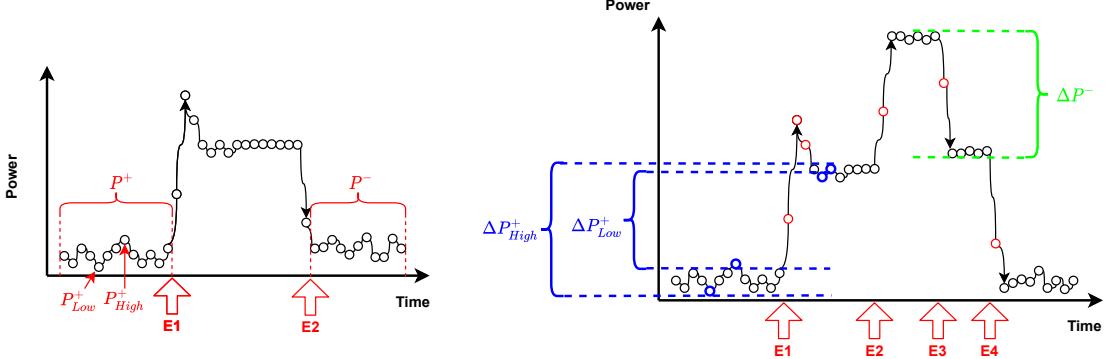


Figure 3.12. First (left) and second (right) compatibility conditions, which are governed by Equations 3.10 and 3.11, respectively.

1. Pair the opening event (rising event) with the closing event (falling event). Both are considered a match only if one of their power intervals coincides with the respective appliance. Otherwise, they are not matched, and the procedure starts from step 2.
2. The unmatched rising and falling events are stored in two different sets.
3. Extract in chronological order the corresponding rising event and check the compatibility conditions with the immediate succeeding falling counterpart. Note that a rising event cannot be associated with falling events that occurred first. If a match was found, delete the pair of events from the rising and falling sets, label the events with the corresponding appliance, and continue with the next rising event. Otherwise, evaluate the matching conditions with the remaining falling events until there is no left. If after evaluating all the falling events, a match was not found, then continue with the next rising event. This step is repeated until there are no more rising events.
4. Store in the unmatched set, the rising and falling events for which a pair was not found.

Notice that from the event pairing, there might still unmatched events. The unmatched events mostly occur from appliances that have multiple operation modes, and within an active cycle, the number of events is odd. One last matching attempt is proposed to **pair the remaining events**. The simple approach is to check if the matchless event's power intervals coincide with the already paired events appliance. Additionally, evaluate if the unmatched event is chronologically in between the matched events. If this is the case, the peerless event is associated with the power interval corresponding to the appliance of the already combined events. An example can be found in Figure 3.13, which exhibits how the unmatched event 2

is related to appliance A since it is in between the matched events 1 and 3 and all correspond to the same appliance.

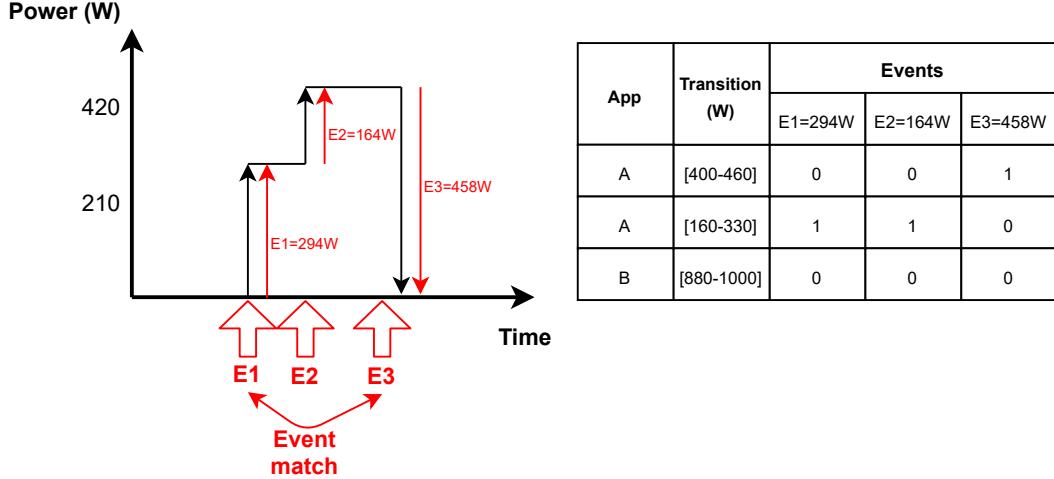


Figure 3.13. Matching of remaining unmatched events.

3.4 Energy Disaggregation

Finally, the energy disaggregation completes the NILM cycle. This NILM stage gives meaning to the DSM programs, which relate consumers to greener power consumption habits by giving them the appropriate single-device expenditure feedback.

The event pairing method in Section 3.3 permits the formulation of an appliance-specific energy consumption calculation. From the event matching, the amount of time that a device was used and the approximate power withdrawal consumed in that interval can be obtained. These two variables are sufficient for an energy consumption estimation.

In the same way, as in event classification, energy disaggregation is restrained to be calculated in every active cycle aiming for a real-time solution. The approach considers the power interval feature $\Delta\Pi^P$ and the timestamps (in seconds) of rising and falling events forming rectangles resembling the typical power withdrawal steps. The events inside the active cycle are separated according to the classification procedure, then for each appliance, sort the events in chronological order and retrieve with the timestamps the duration intervals (in hours) between one event and the next. The rectangle step power construction is simply multiplying the duration intervals by the power interval of the first occurring event in the respective period, which gives Watt-hour energy consumption. Ultimately, the total energy expenditure is the sum of every duration interval's energy, which is then stored for

energy estimations outside the active cycle window. In addition, a pseudocode of this approach is presented in Algorithm 3.1.

Algorithm 3.1: Energy disaggregation procedure.

Input: Power interval features $\Delta\Pi^P$ and timestamps (in seconds) of labeled rising and falling events.

```

1 foreach appliance do
2   | Sort in chronological order the segregated events
3   | foreach timestamp do
4     |   | Calculate the duration interval in hours between the current event and
        |   | the next:
5     |   | Get current and next timestamp event
6     |   | duration interval = (next timestamp - current time stamp) / 3600
7   | end
8   | foreach  $\Delta\Pi^P$  and duration interval do
9     |   | energy =  $\Delta\Pi^P \times$  duration interval
10    |   | total active cycle energy += energy
11  | end
12 end
```

Output: total active cycle energy

Chapter 4

Experimental Results

The proposed NILM algorithm of Chapter 3 is evaluated in this part of the work. First, the reasoning behind the dataset selection is introduced. Then, the event detection effectiveness and event classification accuracy is presented, followed by the actual and estimated energy consumption by each of the electric devices in the establishment. The assessment of the mentioned criteria is performed on the selected residential REDD and industrial IMDELD datasets.

4.1 Dataset Selection

The available measurements, either directly acquired with a sensor or derived from them, would determine which type of solution (algorithm) is best suited to address the NILM problem. For example, VI-trajectory approaches need the voltage and current waveforms to build the appliance signature (see Section 2.1.2). Therefore, a solution with this requirement cannot be implemented with datasets without this kind of data. The broadly used physical variables are the active and reactive power for extracting the fingerprint of an electricity-powered device [43]. The NILM approach explained in Chapter 3 was designed considering these two traditional features that are also included in several public datasets [2, 6, 12, 22, 26, 29]. Another significant consideration for choosing the data is the time granularity. As the sampling frequency increase, more data storage and larger processing power are needed. However, higher accuracy can be achieved as more features can be derived, and more numbers and types of appliances can be distinguished in a complex environment. A middle ground between the intent of higher accuracy and low processing power must be found. Data sampling between 3 samples per second and 10 Hz should be sufficient for the proposed algorithm. For the solution, it is necessary the disaggregated power signal for building appliance-specific models in training. Finally, when contemplating dataset selection, one should focus on the ease of data manipulation to perform different experiments. Thus, further research development becomes simpler.

Bearing in mind the above-mentioned considerations, the Reference Energy Disaggregation Dataset (REDD) [26] for the residential use case and the Industrial Machines Dataset for Electrical Load Disaggregation (IMDELD) [2] for the industrial use case were the selected datasets. Datasets like Building-Level fULLy-labeled dataset for Electricity Disaggregation (BLUED) [12] was discarded even though it met all the requirements. The data is not available at the time of writing this thesis work.

4.1.1 NILMTK

The Non-intrusive Load Monitoring Toolkit (NILMTK) was introduced by [5] to facilitate the comparison of benchmark energy disaggregation algorithms. This open-source toolkit enables the reproducible implementation of NILM solutions in different datasets.

The open source software includes a complete documentation¹ in which it provides:

- Installation commands to equip a dedicated environment with the NILMTK package.
- Conversion of power related data into the NILMTK format. Both the data and the metadata is stored in a HDF5 binary file format.
- Statistical and diagnostic functions which provide in-depth analysis of each dataset.
- Example execution on the recently launched rapid experimentation API which favors NILMTK users to “*focus on which experiments to run rather than on the code required to run such experiments.*”
- A set of example implementations to execute the benchmark disaggregation algorithms on one or more time series.
- A collection of accuracy metrics to validate the NILM solutions on the same basis, so that, direct comparison is possible with the same metric definition.
- A complete development guide so the NILM community can contribute on bugs report, or implement its own disaggregation algorithm and metrics.

In this work, NILMTK is used as the parser of the compatible REDD and IMDELD time series. Once obtained the data, the event pairing classification method can be executed. Moreover, this open source toolkit provides easy data

¹<https://github.com/nilmtk/nilmtk/tree/master/docs/manual>

administration for the appliance-level energy consumption ground-truth. Energy consumption can be easily extracted for further comparison with the results obtained in the following experiments. Finally, the temporal dependency between appliances is studied with the time correlation (see Section 4.4.3), computation which is straightforward with the toolkit.

4.2 Evaluation Metrics

The recurrent problem with the accuracy comparison between the different NILM approaches is the lack of a “standard” evaluation metric. The set of metrics used here is what we believe is the most employed in event-based research.

4.2.1 Metrics Definition

Before introducing the evaluation metrics and how they are computed, the definitions for ground truth event comparison along with its notation are presented as follows,

- A **True Positive** is a correct claim that a given event occurred and was associated with the actual appliance. The number of true positives is denoted with TP .
- A **False Positive** is an incorrect claim that a given event occurred or that was associated with an appliance different from the true one. The number of false positives is denoted with FP .
- A **False Negative** represents that an event that truly occurred, i.e., was labeled in the ground truth, was not detected by the algorithm. The number of false negatives is denoted with FN .
- For an event-based algorithm, there is not much meaning in defining a **True Negative**. However, if a definition had to be given, it would be when a correct claim that an event did not occur.

Recall

Recall represented as RC is the measure of the ratio between how many events were correctly detected and the total number of events, namely $TP + FN$ (see Equation 4.1). If RC approaches one, it means that a high percentage of the correctly detected events are actually present in the aggregated power signal.

$$RC = \frac{TP}{TP + FN} \tag{4.1}$$

Precision

Precision expressed with PR quantifies the event detection success rate, which is the ratio of the correctly detected events and the total number of predicted events, that is, $TP + FP$ (see Equation 4.2). If PR approaches one, that means that the algorithm is assigning correct labels to most of the detections.

$$PR = \frac{TP}{TP + FP} \quad (4.2)$$

F-score

F-score measures the combined performance of the precision PR and recall RC . It punishes low precision or recall results by giving them the same weight (see Equation 4.3). Even if an $RC = 1$ was achieved, PR must also be close to 1 for an overall good algorithm performance.

$$F\text{-score} = 2 \frac{RC \cdot PR}{RC + PR} \quad (4.3)$$

4.2.2 True Positive, False Positive, and False Negative Computation

The events' ground truth for both REDD and IMDELD datasets was manually labeled with the provided disaggregated signals. Considering that the manual labeling could have some time shift in the actual happening of the event and that the disaggregated power signal was sampled at a lower rate than the correspondent aggregated signal (in the case of REDD). It was decided that a true positive tag is given to the algorithm's assigned event timestamp that is enclosed in a ground truth's event time margin.

Therefore, a predicted event is considered a **True Positive** if its time of occurrence is between plus-minus 20 seconds of one truth event, as shown in Figure 4.1. In the case that a predicted event is inside the ground-truth event boundaries but is labeled with a different appliance than the actual one, then it is considered a **False Positive**. A **False Positive** can also mean that an electric device was marked outside every ground-truth limit. A **False Negative** is calculated when no predicted event was inside the ground-truth event time margin.

4.3 Residential Use Case

The residential use case exploits house 1 of the REDD dataset for the algorithm validation. The time series considers 23 days of data. The dataset provides the separated power signal sampled every 3 seconds of an oven (OV), a microwave (MW), kitchen outlets (KO), bathroom GFI (BGFI), a washer/dryer (W/D), a

Experimental Results

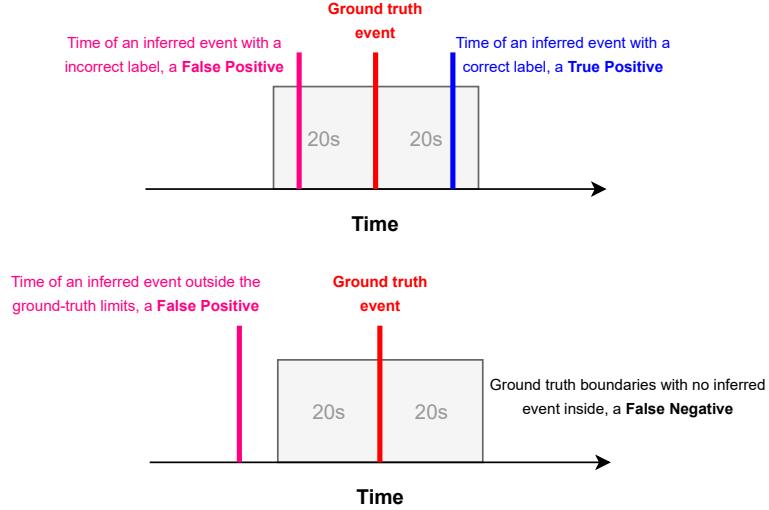


Figure 4.1. True positive, false positive, and false negative computation.

refrigerator (RFG), and a dishwasher (DW), all illustrated in Figure 4.2. These appliances are then used for training in the interval from 2011-04-27 to 2011-05-12, which corresponds to 10 days (the data has some time gaps in-between days). The test sample goes from 2011-04-18 through 2011-04-25, which is extracted from the 1 Hz aggregated power signal.

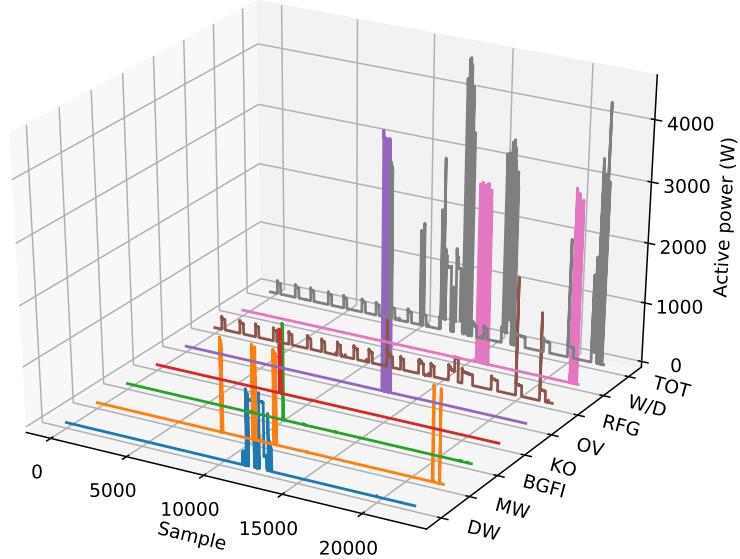


Figure 4.2. REDD's house 1 disaggregated and aggregated power signal comparison.

4.3.1 Event Detection

For event detection, DBSCAN is employed to detect the events of the circuit-level and aggregated power signal. In this case, a $minPts = 15$ was chosen, so at least fifteen samples should build a cluster (i.e., a steady-state) inside the expanding window. Figure 4.3 displays an example of how the detection is performed in a segment of the aggregated power signal. The noise-resistant nature of DBSCAN ignores the sudden peaks in-between states of operation thanks to the parameter $minPts$, which “filtrates” the not sufficiently long steady-states, i.e., noise. Moreover, from the short time series segment, the method correctly detects the rising and falling events (marked with a red X).

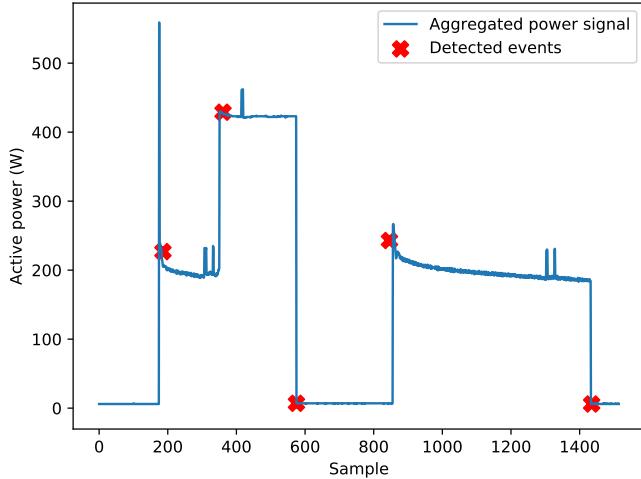


Figure 4.3. Event detection on a segment of house 1 REDD’s aggregated power signal.

Table 4.1 presents the event detection performance for all detected events without discrimination. In this case, a true positive and a false positive have slightly different definitions than the presented in Section 4.2.2. A true positive is considered an inferred event within a ground-truth time margin without considering the label. Instead, a false positive is a predicted event outside the ground-truth limits.

From the detected events in the aggregated power signal portion acquired for the test, 561 were accurately identified from a total of 683 ground-truth labels. A total of 714 predicted events were recognized, from which 153 did not correspond to any ground truth event. So there is room for improvement to accurately detect all the actual events so that the false-negative count approaches zero and mitigates the false positives. An example of wrongly identified or not recognized events is illustrated in Figure 4.4.

Figure 4.4 displays 14 steady-state segments marked with Π and 14 transient states labeled with Ψ . In this time series slice, 11 events out of a total of 14 were

TP	FP	FN	Precision	Recall	F-score
561	153	122	0.79	0.82	0.80

Table 4.1. Overall event detection performance without discrimination.

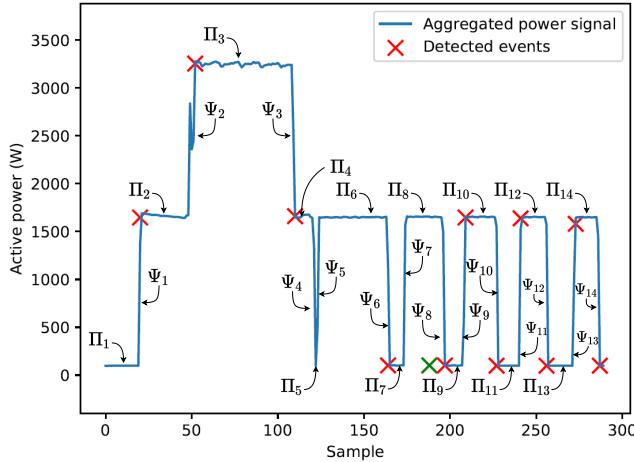


Figure 4.4. False positive and false negative occurrences.

accurately detected (red Xs). The events corresponding to edges Ψ_4 and Ψ_5 are not identified, causing false negatives. The algorithm detects Π_5 as noise samples because $minPts$ does not allow such a small sample to be considered a cluster, thus incurring false negatives. Probably in a higher sampling data, Π_5 would have been considered another cluster. The silver lining is that examples like this, in their majority, are indeed noise, so the algorithm in these cases will avoid false positives.

A false positive is found in the interval that comprehends Ψ_7 and Ψ_8 (i.e., the green X). The nature of this false positive may be related to the false negative corresponding to the rising edge of Ψ_7 . Alternatively, the false positive in this case is an isolated event, where no reasonable explanation of its origin can be given. There is no apparent noise to which this event can be related, which would be the main reason for the wrong detection.

4.3.2 Feature Extraction

For training, the features are extracted on the individual appliances' signal, as explained in Section 3.2. The states of operation of an appliance are retrieved considering that each device has ten modes of operation using the WL method underlined in Section 3.2.3. After the process, all the detected transition intervals are grouped into ten sets, as illustrated in Table 4.2. The table presents the transition interval centroids with their corresponding upper and lower limits for the refrigerator in

house 1 of the REDD dataset.

	MIN	MAX	Centroid
Active power (W)	395	465	421
	277	328	291
	250	275	260
	229	248	233
	217	228	225
	205	214	207
	187	204	200
	165	182	175
	32	73	41
	1	17	4

Table 4.2. Fridge transition intervals before applying the distance-based merging policy.

The ten-cluster WL procedure demonstrates that there are power transitions that can still be united as a single one due to their closeness. Therefore, the transitions are combined by adhering to the distance-based merging policy. Finally, the probable mode transitions are derived following Equation 3.9 with the resultant power intervals from the distance-based merging policy. Table 4.3 displays the mode transition outcome for all appliances, excluding the OFF operation mode. Low power transitions are probably noise, so intervals below 40 W are filtered. In the case of the refrigerator, intervals [1-17] W and [32-73] W are eliminated. The dishwasher, microwave, and refrigerator are type 2 appliances, i.e., multi-state. The rest of them are type 1, ON and OFF devices.

Appliance	Mode transitions
DW	[116, 202], [207, 265], [380, 397], [432, 547], [601, 670], [825, 1416]
MW	[885, 1009], [1176, 1632]
RFG	[165, 329], [395, 466]
BGFI	[1558, 1685]
KO	[1012, 1126]
OV	[3284, 4294]
W/D	[2666, 3221]

Table 4.3. Appliances' mode transitions after the distance-based merging policy.

Regarding the extraction of the features to mitigate the negative effects of the overlapping power mode transitions, Figure 4.5 illustrates the example of how they

are acquired for the rising events of the microwave. From the transient spike, power change, and the transition interval, the probability distribution forms several peaks (multi-modal distribution), reinforcing that a kernel density estimation is necessary to compute a PDF. Notice that some outliers could affect the PDF estimation, as in the case of the transient power change, where a small portion of samples around 1500 W is far from the prevalent sub-groups below 500 W.

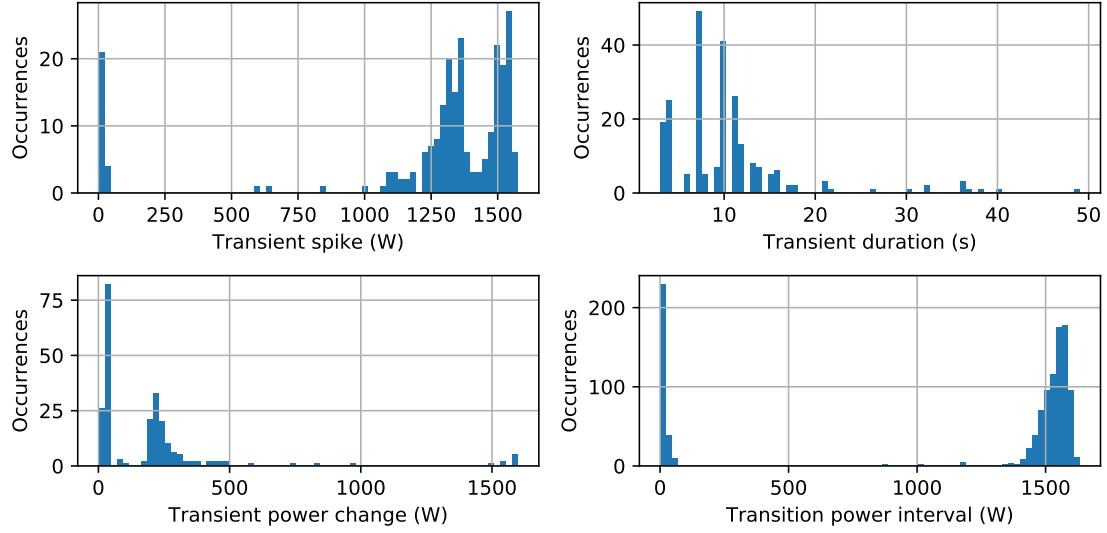


Figure 4.5. Feature extraction for microwave.

4.3.3 Classification

Appliance inference is applied to the detected events in the aggregated power signal. Multiple-labeled events after the interval association phase are the subject of the event pairing method. The results of the event association are summarized in Table 4.4.

The performance of the algorithm up to the classification part was in general:

- Poor for the dishwasher, there are a massive amount of false positives related either to another electric device (see Figure 4.6) or not associated with any ground truth event (see Table 4.5). The several operation modes make it prone to more mistakes as its power intervals are shared with multiple electricity-powered devices.
- The two appliances that present the worse performance, bathroom GFI and kitchen outlets, are the less event-frequent devices. The metrics are susceptible

Appliance	TP	FP	FN	Precision	Recall	F-score
DW	82	157	29	0.34	0.74	0.47
MW	100	40	40	0.71	0.71	0.71
RFG	189	42	81	0.82	0.70	0.75
BGFI	4	0	18	1.00	0.18	0.31
KO	2	4	14	0.33	0.13	0.18
OV	34	10	3	0.77	0.92	0.84
W/D	40	0	23	1.00	0.63	0.78
Average				0.71	0.57	0.58

Table 4.4. Appliance-level event detection performance.

to a small quantity of predicted and actual events. Notice that just four FP from the kitchen outlets drops the precision to 0.33. In the same way, only four TP represents a perfect precision score for the bathroom GFI.

- The precision metric is better than the recall, regarding the total amount of predicted events. The correctly detected ones appreciably tend to be more, on average. On the other hand, among the actual events, on average, around half of them will not be recognized. The combination of these two parameters gives an overall low F-score compared with the standards in the literature.

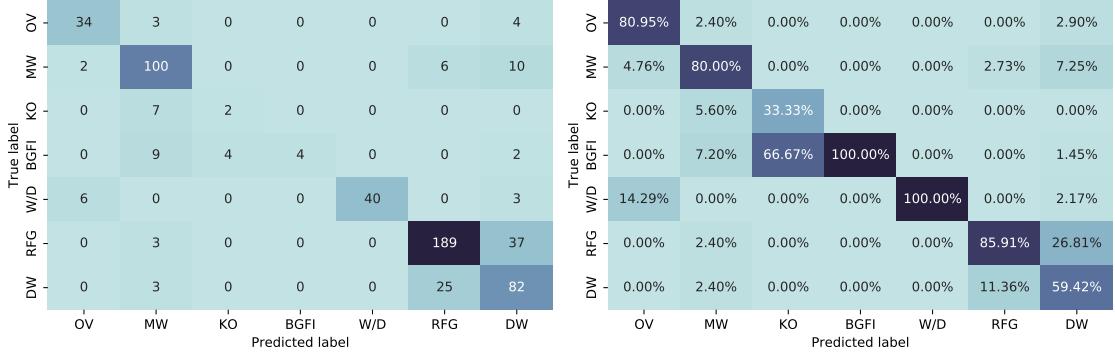


Figure 4.6. Confusion matrix predicted vs. true label. On the left, the number of predicted events. On the right, percentage of the total of predicted events.

In addition to Table 4.4, Figure 4.6 presents the confusion matrix of the inferred events' labels to provide more insights into the wrongly marked events, namely false positives. The matrix shows the number of predicted events that are inside the limits of a ground-truth label. Each column contains the total of labels associated with

that appliance. For example, element (2, 3) corresponds to the number of kitchen outlets predictions, but in reality, these events should be related to the microwave. So, the off-diagonal elements represent erroneous designations.

There is a strong connection between the refrigerator and the dishwasher. The predicted tags correspond to 11 % and 27 % of the ground-truth-related events, respectively. One of the possible reasons is the similar characteristics from the other features, that indeed, as illustrated in Figure 4.7, shows a resemblance between them.

The predicted labels associated with the dishwasher are misclassified with five out of six other appliances, which means that the widespread mode transitions implicate more possible false connections. Moreover, in the tiebreaker procedure, there is an evident edge for the dishwasher as 56 FP results from it. As the dishwasher has multiple problems with other appliances, proposals like acquiring a non-traditional feature as the modes that should occur during an active cycle for a multi-state appliance increase the algorithm’s performance [1]. From visual inspection, the authors claim that all operation modes should happen if the dishwasher is ON, so if the algorithm detects a non-overlapping interval like [601-670] W, it means that the appliance is working. However, non-automated feature extraction like this detracts from the meaning of the purpose of NILM. The solution is hardly scalable. If another device is added, the algorithm’s maintenance becomes hard and time-consuming. Therefore, automated feature extraction like the proposed here is desired to avoid the just-mentioned problems.

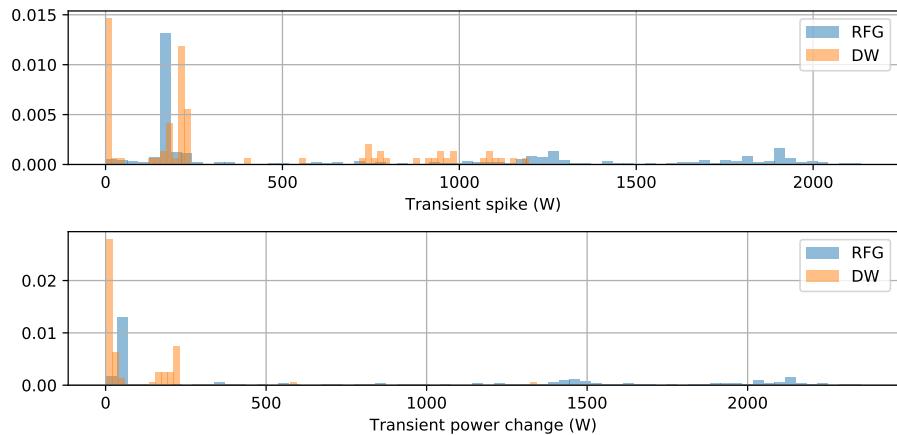


Figure 4.7. Refrigerator and dishwasher feature resemblance.

The confusion matrix in Figure 4.6 includes the FP that was designated an incorrect label compared to the true one. Table 4.5 presents the other set of FP that was labeled but did not belong to any ground-truth interval. The dishwasher designations, among all, is the most inaccurate, with 42 % of its recognized events

outside any ground-truth limit. This FP count is attributed to the initial event detection procedure. The error is propagated to this stage causing a lower precision for most of the appliances.

	OV	MW	KO	BGFI	W/D	RFG	DW
FP	2	15	0	0	0	11	101

Table 4.5. Number of false positives that meet the criteria of not belonging to any ground truth interval.

4.3.4 Energy Disaggregation

The proposed energy disaggregation algorithm explained in Section 3.4 is applied to the events of each active cycle in the aggregated time series.

Surprisingly, the refrigerator that has the second-largest share of detected events is the third less consuming appliance (see Table 4.6). The oven, microwave, washer-dryer, and dishwasher are electric devices whose consumption is overestimated. This may be to a time-distant event paring. If two far events are incorrectly matched, the computed consumption will be higher than the real one. For example, the oven has a low portion of inferred events, but as it is the most power-hungry device, distant paired events considerably elevate the estimated consumption.

Overestimation goes hand in hand with devices that obtained an under energy calculation, as in the case of the kitchen outlets and the bathroom GFI. The overwhelming consumption of other devices reduces the true percentage of energy expenditure. In addition, these electricity-powered devices only have a total of six related events, which does not add much to the final sum. However, they are the actual less power-consuming equipment.

Contribution	OV	MW	KO	W/D
True	0.077	0.194	0.023	0.184
Estimated	0.232	0.322	0.001	0.232
	RFG	DW	BGFI	
True	0.345	0.131	0.047	
Estimated	0.060	0.154	0.001	

Table 4.6. True vs. predicted energy consumption residential use case.

4.4 Industrial Use Case

The industrial use case adopts heavy-machinery data from a poultry feed factory located in Brazil. The factory operation is described as the creation of pellets of ration for poultry from corn or soybeans and added nutrients [2]. The factory operates from Mondays through Fridays with some exceptions on Saturdays (in case the factory production is below the monthly target) during 22h to 17h. The factory is closed from 17h to 22h due to the higher electricity price rate during that interval.

The dataset includes 1 Hz data points of: RMS voltage, RMS current, active power, reactive power, and apparent power for eleven energy meters in the period from 2017-12-11 to 2018-04-03. Three meters measure the distribution circuit variables, one of which is the site meter that provides the aggregated signals. The remaining eight meters produce the segregated machine samples. The considered machines are two pelletizers (PI and PII), two double-pole contactors (DPCI and DPCII), two exhaust fans (EFI and EFII), and two milling machines (MI and MII). The latest ones are only measured for the last 12 days of the data collection campaign. Figure 4.8 presents the aggregated power signal compared with the machine-level consumption for one working day.

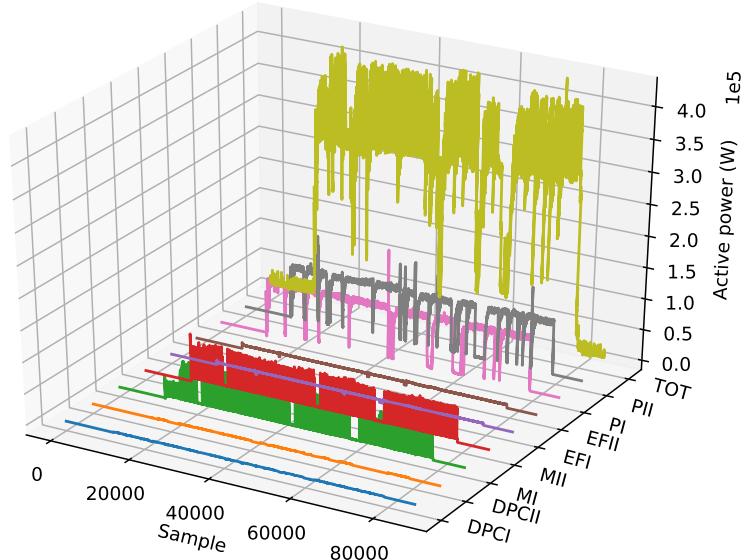


Figure 4.8. IMDELD's disaggregated and aggregated power signal comparison.

The learning and inference processes were divided as follows:

- Training (on disaggregated samples)

- From 2018-02-19 to 2018-02-28 for the milling machines, which roughly corresponds to 9 days (the data has time gaps).
- From 2017-12-11 to 2017-12-21, for the remaining machines, which corresponds to 10 days.
- Testing (on aggregated power signal)
 - From 2018-03-26 to the end of the time series, which corresponds to the last 10 days of the whole time series.

4.4.1 Event Detection

In the industrial use case, the selected *minPts* parameter was 10 samples. The number is lower than the residential use case as the disaggregated power signals are faster sampled, thus reducing the likelihood of detecting transient or noise segments as a steady-state.

IMDELD provides reactive power as an additional feature that can be exploited to detect and classify events. Figure 4.9 presents an example of how the active and reactive aggregated power signals are employed to distinguish the events (marked with a red X). The active power signal, at first glance (Figures 4.9 and 4.8), suggest that the power withdrawal is massive compared to the residential counterpart. The event detection should not be affected since the signals are transformed to the logarithmic domain (see Section 3.1).

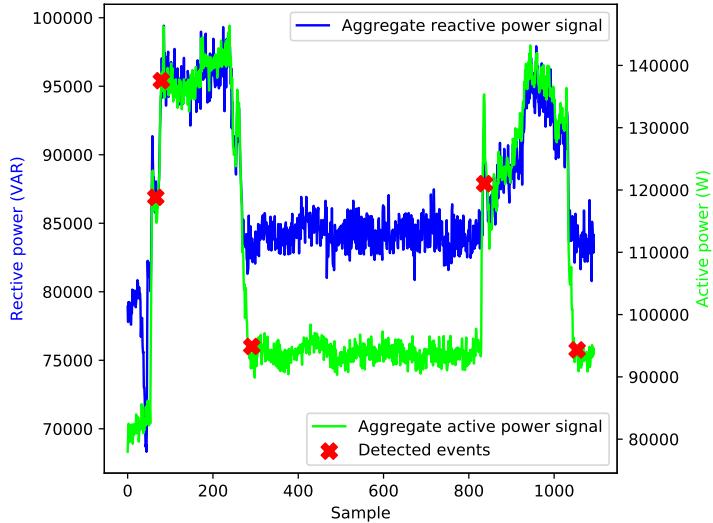


Figure 4.9. Event detection on a segment of IMDELD’s aggregated power signal.

As anticipated in Section 2.3, it was found from exploratory data analysis that all appliances had simultaneous events. Just considering the pair of machines of the

same kind, 2434 events are common, as listed in Table 4.7. The ground-truth reports 7559 events, most of which are repeated due to the appliances' time simultaneity (ground-truth was extracted for each machine individually). Meanwhile, from the aggregated power signal, the predicted events are 3244, which combines the time matching occurrences with the independent ones.

Machine pair	MI-MII	PI-PII	EFI-EFII	DPCI-DPCII
# of events	2245	16	31	142

Table 4.7. Number of shared simultaneous events between machines of the same kind.

4.4.2 Feature Extraction

The proposed states of operation extraction method in Section 3.2.3 is employed on the disaggregated reactive and active power signals. The mode transitions are retrieved after the distance-merging policy and are computed with Equation 3.9, as reported in Table 4.8. Opposed to the residence studied here, all machines in the Brazilian industry are type 2 appliances but also present type 3 traits as illustrated in the slow changes of Figure 4.9.

Appliance	Mode transitions
PI	[6967, 28023], [28277, 46049], [46421, 98332], [100670, 134542], [160232, 174166]
PII	[6804, 32221], [32662, 60165], [60388, 93820], [95238, 112228], [142098, 174022]
EFI	[1003, 2519], [2690, 4049], [4162, 6339]
EFII	[2139, 3678], [4128, 11221], [11578, 13234]
MI	[4388, 21518], [21548, 36755], [36811, 68150]
MII	[3265, 20821], [20851, 26509], [26564, 58938]
DPCI	[289, 396], [397, 1025], [1030, 1412], [21539, 23048]
DPCII	[221, 349], [350, 903], [907, 1110], [1117, 1497]

Table 4.8. Machines' mode transitions.

The reactive power transition interval feature can bring another quality to decouple the recurrent active power intersection. This might be the case for common sections of the double pole contactor and exhaust fan, as the reactive intervals for both kinds of machines are separated from each other, Figure 4.10. However, the milling and pelletizer machines have a widespread reactive interval spectrum that covers even the double pole contactor and exhaust fan. So, it is more challenging to distinguish appliances from solely the reactive intervals of overlapping segments, including the milling and/or pelletizer machines.

On the other hand, the only machine of the same kind that differentiates by its reactive power interval is the exhaust fan. The others possess resembling distributions, which brings another complexity to the disaggregation task.

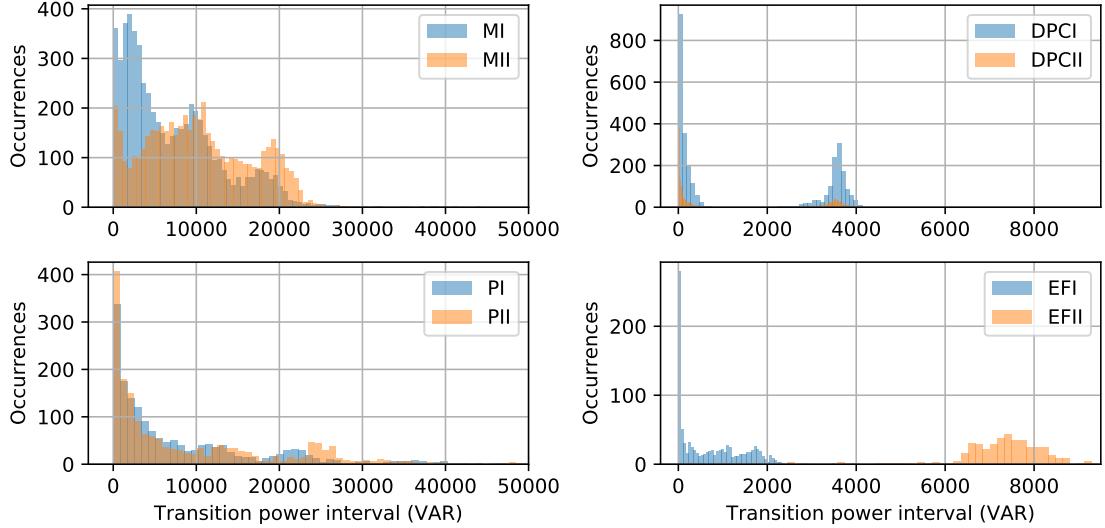


Figure 4.10. Feature extraction of reactive power transition intervals.

4.4.3 Classification

The classification of the aggregated signal events takes place after the event pairing method. Table 4.9 lists the results of the event-appliance matching for the industrial case.

Even designing an algorithm aware of multi-mode appliance intersection, event feature separation, and event pair matching considering the most probable appliance. The solution fails in distinguishing appliances in industrial settings (see Table 4.9), where there is a continuous operation, and temporal dependency between them is highly correlated. However, the most frequent-event appliances (MI and MII) show that the FP count is reduced compared to the total number of detections.

The confusion matrix in Figure 4.11 demonstrates that the true labels of MI are being confused with a wide variety of other machines. In particular, the pelletizers are wrongly associated with MI. The pelletizers are the most power-hungry appliance (see Figure 4.8 and Table 4.8), and as there are a considerable amount of simultaneous events between MI and MII (see Table 4.7), there is a high chance that the pelletizers are associated with the power sum of events. The power contribution of coincident events of MI and MII does not match any of the power intervals. This results in the association of that event with the closest power interval, i.e., with

Experimental Results

Appliance	TP	FP	FN	Precision	Recall	F-score
MI	170	64	2772	0.73	0.06	0.11
MII	91	30	3025	0.75	0.03	0.06
PI	31	972	249	0.03	0.11	0.05
PII	36	525	284	0.06	0.11	0.08
EFI	2	20	92	0.09	0.02	0.03
EFII	0	0	54	0.00	0.00	0.00
DPCI	9	527	383	0.02	0.02	0.02
DPCII	11	756	350	0.01	0.03	0.02
Average			0.21	0.05	0.04	

Table 4.9. Machine-level event detection performance.

either of the pelletizers. In general, a frequent-event appliance like MI overwhelms the time series. The events' power intervals range from around 4300 W to 68000 W, which covers most of the power intervals of EFI, EFII, DPCI, DPCII, thus the elevated confusion rate.

		MI	MII	PI	PII	EFI	EFII	DPCI	DPCII		MI	MII	PI	PII	EFI	EFII	DPCI	DPCII	
True label	Predicted label	MI	170	6	814	489	1	0	24	24	MI	82.93%	5.88%	92.19%	90.06%	33.33%	0.00%	55.81%	55.81%
		MII	26	91	18	7	0	0	1	1	MII	12.68%	89.22%	2.04%	1.29%	0.00%	0.00%	2.33%	2.33%
True label	Predicted label	PI	3	2	31	8	0	0	7	6	PI	1.46%	1.96%	3.51%	1.47%	0.00%	0.00%	16.28%	13.95%
		PII	6	2	18	36	0	0	1	1	PII	2.93%	1.96%	2.04%	6.63%	0.00%	0.00%	2.33%	2.33%
True label	Predicted label	EFI	0	1	2	2	2	0	0	0	EFI	0.00%	0.98%	0.23%	0.37%	66.67%	0.00%	0.00%	0.00%
		EFII	0	0	0	0	0	0	0	0	EFII	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
True label	Predicted label	DPCI	0	0	0	1	0	0	9	0	DPCI	0.00%	0.00%	0.00%	0.18%	0.00%	0.00%	20.93%	0.00%
		DPCII	0	0	0	0	0	0	1	11	DPCII	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	2.33%	25.58%

Figure 4.11. Confusion matrix predicted vs. true label for the industrial use case. On the left, the number of predicted events. On the right, percentage of the total of predicted events.

The time coincident events show that machines are somehow related in their processes. A correlation matrix between appliances was developed through NILMTK and depicted in Figure 4.12 to confirm this. The rightmost part of the figure reports the correlation between all possible pairs of machines in the factory. Equipment of the same kind has a correlation above 0.89, which is not surprising because they should have the same task and are aligned in time. Moreover, not only does the equal pair of appliances have a high correlation, but also different electric devices

show a significant relationship. Pelletizers, exhaust fans, and double pole contactors display this quality, which suggests that their assignments are related and operate in parallel. On the other hand, the milling machines' correlation with the other electricity-powered devices indicates that their job is different and should be involved in another factory process.

On the other side, the residential use case presents an insignificant correlation between appliances (leftmost part of Figure 4.12). This reinforces not only the fact that simultaneous events are rare in this kind of environment but also suggests high stochasticity in the powered electricity elements' usage.

OV	1.000	0.080	-0.002	-0.002	0.036	0.040	-0.005	MI	1.0000	0.9609	0.1561	0.1525	0.1593	0.1592	0.1419	0.1428
MW	0.080	1.000	0.037	0.022	0.016	0.034	-0.016	MII	0.9609	1.0000	0.1624	0.1582	0.1659	0.1655	0.1478	0.1479
KO	-0.002	0.037	1.000	0.012	0.008	0.028	-0.005	PI	0.1561	0.1624	1.0000	0.8991	0.9179	0.9097	0.8999	0.8531
BGFI	-0.002	0.022	0.012	1.000	0.003	-0.010	-0.010	PII	0.1525	0.1582	0.8991	1.0000	0.8600	0.8912	0.9759	0.9275
W/D	0.036	0.016	0.008	0.003	1.000	0.031	0.064	EFI	0.1593	0.1659	0.9179	0.8600	1.0000	0.9328	0.8732	0.8289
RFG	0.040	0.034	0.028	-0.010	0.031	1.000	0.022	EFII	0.1592	0.1655	0.9097	0.8912	0.9328	1.0000	0.8989	0.8537
DW	-0.005	-0.016	-0.005	-0.010	0.064	0.022	1.000	DPCI	0.1419	0.1478	0.8999	0.9759	0.8732	0.8989	1.0000	0.9470
	OV	MW	KO	BGFI	W/D	RFG	DW		MI	MII	PI	PII	EFI	EFII	DPCI	DPCII

Figure 4.12. Correlation matrix between appliances. On the left, residential use case. On the right, industrial use case.

After assessing the possible reasons for the low performance of the algorithm in the industrial domain, one appliance of each kind was selected to evaluate the capabilities of the proposed solution by removing the simultaneous events caused by the correlated temporal dependency of devices of the same class. The aggregated power signals are now just the contributions of MI, PI, EFI, and DPCI. The event detection performance is summarized in Table 4.10.

The quality of the inferred events increased for all appliances, especially for MI. The considerable improvement is attributed to the fact that the extracted power transition intervals are not the sum of two machines (for simultaneous events), so the first association step is correctly assigned to the actual machine. However, 109 predicted labels are related to PI instead of MI, the real label. There are still similarities that characterize MI with PI. Unfortunately, low-power equipment (EFI and DPCI) display the lowest performance. The actual events are not recognized, causing a performance drop.

Appliance	TP	FP	FN	Precision	Recall	F-score
MI	2463	188	479	0.93	0.84	0.88
PI	38	183	242	0.17	0.14	0.15
EFI	1	3	93	0.25	0.01	0.02
DPCI	21	1	371	0.95	0.05	0.10
Average				0.58	0.26	0.29

Table 4.10. Event detection performance only considering one machine of each.

4.4.4 Energy Disaggregation

The energy disaggregation procedure in Section 3.4 is tested on the events found in the aggregated active cycles of the industrial power signals. Table 4.11 compares the true and predicted energy consumption for the machines in the factory. The actual energy expenditure strengthens the evidence of highly related tasks to appliances of the same kind. Moreover, the similar energy percentage among the same equipment indicates that they work in parallel with similar schedules. Despite being the second most power-hungry devices, the milling machines represent 3 % of the total consumption, below the 5.9 % coming from the low-consuming exhaust fans. The previous observation along with the consideration that the milling machines are the most frequent-event appliances could signify that the active cycle of the milling is shorter and highly repeated compared to that of the exhaust fan.

Comparing the actual number of events for the double pole contactors and exhaust fans, the energy underestimation may be due to the insignificant amount of predicted events associated with them. The overestimation for PI can be related to the events that corresponded to the MI but were assigned to PI (see Table 4.11).

Contribution	PI	PII	MI	MII
True	0.478	0.422	0.017	0.013
Estimated	0.567	0.230	0.142	0.060
	DPCI	DPCII	EFI	EFII
True	0.006	0.006	0.020	0.039
Estimated	0.001	0.001	0.001	0.000

Table 4.11. True vs. predicted energy consumption industrial use case.

4.5 Real-time evaluation

The active cycle identification not only was meant to reduce the computational burden but also pointed at a real-time feedback solution (see Section 3.3). A real-time or online disaggregation provides instant measurement awareness to the end-user. It is possible to evaluate if the algorithm can provide this kind of service by calculating the active cycle duration.

Figure 4.13 compares the time duration of the active cycle for the residential and industrial environments studied here.

Based on the definition given by [32], in a house like REDD 1, the solution can be considered near-real-time. The active cycle durations are concentrated around the five-minute mark, so after that time, the end-user can be updated about the consumption status. However, there are also cases where the active cycle lasts longer for appliances with extended duration modes. On the other hand, the active cycle duration between 300 to 700 minutes is more frequent for the Brazilian factory, which is coherent since some machines operate during the whole working period. In this case, near-real-time cannot be implemented. The active cycle average duration does not permit it.

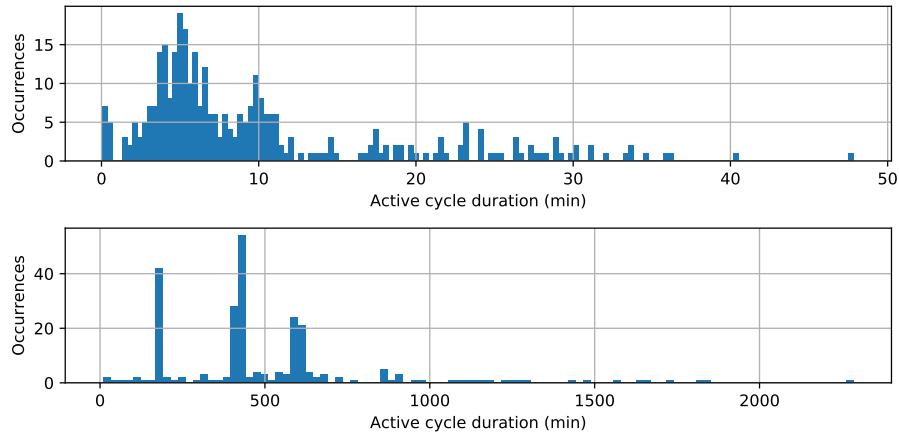


Figure 4.13. Active cycle duration. On the top, residential duration. On the bottom, industrial duration.

Chapter 5

Conclusions and Future Work

In this thesis work, a semi-supervised event pairing method was proposed to address the NILM problem. The algorithm design was aimed at separating the power profile of electricity-powered devices in different application environments in four main phases: (i) a noise-resistant cluster-based event detection; (ii) the extraction of steady, transient, and non-traditional features from a small appliance-specific training set; (iii) a classification procedure which exploits the extracted features to estimate the most probable appliance through an event pairing method; and (iv) an individual rectangle-shaped load consumption estimation for the active electric devices in an establishment. The solution as a whole was validated on the REDD and IMDELD datasets, comparing how the method adjusted for the residential and industrial domains, respectively.

In general, the solution proves to detect events accurately even with a noisy signal. However, the algorithm could improve in the classification step, where the kernel density estimation showed that is not the most appropriate approach to search for feature separation. The features are still very similar in their specific categories causing a false connection of events to appliances that do not belong. Moreover, this tiebreak procedure could suffer from a low estimated probability for one or more features. As it is calculating the joint probability of the occurrence of all features together, a low number could ruin the prediction. A bigger and more meaningful training sample (i.e., supervised method) could be extracted from the signal to mitigate this so that low probabilities are associated with not corresponding appliances.

Furthermore, active cycle detection allows in the residential case a near-real-time solution, as the ON interval does not last long and power consumption is calculated after that. This is not the case for the industrial setting, as the active cycle can last at least one full day because the candidate ground state is found in the last falling edge of a working day. So, the energy consumption estimation will be given daily instead of near-real-time.

5.1 Future Work

The semi-supervised event pairing method presented here is not adequate for an industrial setting that has simultaneous events. The algorithm does not distinguish which appliances are operating as the rising/falling edge does not correspond to the learned power intervals. Moreover, the complexity of the disaggregation task increases if more machines of the same kind are operating, and it is hard to tell the difference as they exhibit similar characteristics, like the one presented in Section 4.4.2. The overlap of the same equipment affects the algorithm's performance.

The sampling frequency may have an impact to treat high event parallelism like the one studied in this investigation with the factory in Brazil. The work by [15] affirms that a high-frequency sampling rate is a requirement to discriminate the possible simultaneous events. The events could be distinguished with a faster sample rate, and the simultaneity likelihood will decrease. Provided that over time, computational capabilities to process elevated amounts of inputs and compressing methods to store massive amounts of data are being improved continuously, data collection campaigns should aim at higher frequency data.

In summary, the challenge of recognizing machines of the same kind is still an open issue. Discriminating devices that perform the same assignment adds another complication to the disaggregation task because their power signature is indistinguishable. Unlike the diverse residential appliances which possess a unique fingerprint. Promising approaches that can consider this matter can be deep neural networks, as they can extract features not so obvious, thanks to their multi-neuron-layer structure.

On the other hand, in real settings would be hard to find disaggregated power signals to train the algorithm, so research in this area should also be pointing at solutions that need only the aggregated power signal. Promising advancements have been made with unsupervised approaches, but they still need to overcome the various cluster formations from multi-mode appliances. Each of the modes of the appliance is identified as a different device, which is considered as one of the major challenges faced with unsupervised solutions.

Bibliography

- [1] AZIZI, E., BEHESHTI, M. T. H., AND BOLOUKI, S. Event matching classification method for non-intrusive load monitoring. *Sustainability* 13, 2 (2021).
- [2] BANDEIRA DE MELLO MARTINS, P., BARBOSA NASCIMENTO, V., DE FREITAS, A. R., BITTENCOURT E SILVA, P., AND GUIMARÃES DUARTE PINTO, R. Industrial machines dataset for electrical load disaggregation. In *IEEE Dataport* (2018).
- [3] BARANSKI, M., AND VOSS, J. Nonintrusive appliance load monitoring based on an optical sensor. In *2003 IEEE Bologna Power Tech Conference Proceedings*, (2003), vol. 4, IEEE, pp. 8–pp.
- [4] BARSIM, K. S., STREUBEL, R., AND YANG, B. An approach for unsupervised non-intrusive load monitoring of residential appliances. In *Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring* (2014), pp. 1–5.
- [5] BATRA, N., KELLY, J., PARSON, O., DUTTA, H., KNOTTENBELT, W., ROGERS, A., SINGH, A., AND SRIVASTAVA, M. Nilmtk: An open source toolkit for non-intrusive load monitoring. In *Proceedings of the 5th International Conference on Future Energy Systems* (New York, NY, USA, 2014), e-Energy '14, Association for Computing Machinery, p. 265–276.
- [6] BISCHOF, S., TRITTENBACH, H., VOLLMER, M., WERLE, D., BLANK, T., AND BÖHM, K. Hipe – an energy-status-data set from industrial production. In *Proceedings of ACM e-Energy (e-Energy 2018)* (New York, NY, USA, 2018), ACM, pp. 599–603.
- [7] CARRIE ARMEL, K., GUPTA, A., SHRIMALI, G., AND ALBERT, A. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy* 52 (2013), 213–234. Special Section: Transition Pathways to a Low Carbon Economy.
- [8] DE BAETS, L., DEVELDER, C., DHAENE, T., AND DESCHRIJVER, D. Detection of unidentified appliances in non-intrusive load monitoring using siamese neural networks. *International Journal of Electrical Power and Energy Systems* 104 (2019), 645–653.

- [9] DE MELLO MARTINS, P. B., PINTO, R. G. D., AND E SILVA, P. B. Load disaggregation of industrial machinery power consumption monitoring using factorial hidden markov models. In *4th International NILM Workshop* (2018).
- [10] FANG, X., MISRA, S., XUE, G., AND YANG, D. Smart grid — the new and improved power grid: A survey. *IEEE Communications Surveys Tutorials* 14, 4 (2012), 944–980.
- [11] FAUSTINE, A., MVUNGI, N. H., KAIJAGE, S., AND MICHAEL, K. A survey on non-intrusive load monitoring methodologies and techniques for energy disaggregation problem, 2017.
- [12] FILIP, A., ET AL. Blued: A fully labeled public dataset for event-based nonintrusive load monitoring research. In *2nd workshop on data mining applications in sustainability (SustKDD)* (2011), p. 2012.
- [13] GONÇALVES, H., OCNEANU, A., BERGÉS, M., AND FAN, R. Unsupervised disaggregation of appliances using aggregated consumption data. In *The 1st KDD workshop on data mining applications in sustainability (SustKDD)* (2011).
- [14] GOPINATH, R., KUMAR, M., JOSHUA, C. P. C., AND SRINIVAS, K. Energy management using non-intrusive load monitoring techniques-state-of-the-art and future research directions. *Sustainable Cities and Society* (2020), 102411.
- [15] HAQ, A. U. *Appliance Event Detection for Non-Intrusive Load Monitoring in Complex Environments*. PhD thesis, Technische Universität München, 2018.
- [16] HART, G. W. Nonintrusive appliance load monitoring. *Proceedings of the IEEE* 80, 12 (1992), 1870–1891.
- [17] HE, K., JAKOVETIC, D., ZHAO, B., STANKOVIC, V., STANKOVIC, L., AND CHENG, S. A generic optimisation-based approach for improving non-intrusive load monitoring. *IEEE Transactions on Smart Grid* 10, 6 (2019), 6472–6480.
- [18] HENAO, N., AGBOSSOU, K., KELOUWANI, S., DUBÉ, Y., AND FOURNIER, M. Approach in nonintrusive type i load monitoring using subtractive clustering. *IEEE Transactions on Smart Grid* 8, 2 (2015), 812–821.
- [19] JON, H. Hierarchigal grouping to optimize an objective fun ction. *Journal of the American Statistical Association* 58, 301 (1963), 236–244.
- [20] JOSHI, H. *Residential, Commercial and Industrial Electrical Systems: Network and installation*, vol. 2. Tata McGraw-Hill Publishing Company, 2008.

- [21] KALINKE, F., BIELSKI, P., SINGH, S., FOUCHÉ, E., AND BÖHM, K. An evaluation of nilm approaches on industrial energy-consumption data. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems* (2021), pp. 239–243.
- [22] KELLY, J., AND KNOTTENBELT, W. The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes. *Scientific data* 2, 1 (2015), 1–14.
- [23] KELLY, J., AND KNOTTENBELT, W. J. Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature. *CoRR abs/1605.00962* (2016).
- [24] KIM, H., MARWAH, M., ARLITT, M., LYON, G., AND HAN, J. Unsupervised disaggregation of low frequency power measurements. In *Proceedings of the 2011 SIAM international conference on data mining* (2011), SIAM, pp. 747–758.
- [25] KOLTER, J. Z., AND JAAKKOLA, T. Approximate inference in additive factorial hmms with application to energy disaggregation. In *Artificial intelligence and statistics* (2012), PMLR, pp. 1472–1482.
- [26] KOLTER, J. Z., AND JOHNSON, M. J. Redd: A public data set for energy disaggregation research. In *IN SUSTKDD* (2011).
- [27] KONG, W., DONG, Z. Y., MA, J., HILL, D. J., ZHAO, J., AND LUO, F. An extensible approach for non-intrusive load disaggregation with smart meter data. *IEEE Transactions on Smart Grid* 9, 4 (2016), 3362–3372.
- [28] LU, M., AND LI, Z. A hybrid event detection approach for non-intrusive load monitoring. *IEEE Transactions on Smart Grid* 11, 1 (2019), 528–540.
- [29] MAKONIN, S., ELLERT, B., BAJIĆ, I. V., AND POPOWICH, F. Electricity, water, and natural gas consumption of a residential house in canada from 2012 to 2014. *Scientific data* 3, 1 (2016), 1–12.
- [30] MAKONIN, S., POPOWICH, F., BAJIĆ, I. V., GILL, B., AND BARTRAM, L. Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring. *IEEE Transactions on smart grid* 7, 6 (2015), 2575–2585.
- [31] MARTINS, P. B., GOMES, J. G., NASCIMENTO, V. B., AND DE FREITAS, A. R. Application of a deep learning generative model to load disaggregation for industrial machinery power consumption monitoring. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)* (2018), IEEE, pp. 1–6.

- [32] MENGISTU, M. A., GIRMAY, A. A., CAMARDA, C., ACQUAVIVA, A., AND PATTI, E. A cloud-based on-line disaggregation algorithm for home appliance loads. *IEEE Transactions on Smart Grid* 10, 3 (2018), 3430–3439.
- [33] NAIT MEZIANE, M., RAVIER, P., LAMARQUE, G., LE BUNETEL, J.-C., AND RAINGEAUD, Y. High accuracy event detection for non-intrusive load monitoring. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017), pp. 2452–2456.
- [34] REVUELTA HERRERO, J., LOZANO MURCIEGO, Á., LÓPEZ BARRUSO, A., HERNÁNDEZ DE LA IGLESIA, D., VILLARRUBIA GONZÁLEZ, G., CORCHADO RODRÍGUEZ, J. M., AND CARREIRA, R. Non intrusive load monitoring (nilm): A state of the art. In *Trends in Cyber-Physical Multi-Agent Systems. The PAAMS Collection - 15th International Conference, PAAMS 2017* (Cham, 2018), F. De la Prieta, Z. Vale, L. Antunes, T. Pinto, A. T. Campbell, V. Julián, A. J. Neves, and M. N. Moreno, Eds., Springer International Publishing, pp. 125–138.
- [35] ROGELJ, J., DEN ELZEN, M., HÖHNE, N., FRANSEN, T., FEKETE, H., WINKLER, H., SCHAEFFER, R., SHA, F., RIAHI, K., AND MEINSHAUSEN, M. Paris agreement climate proposals need a boost to keep warming well below 2 c. *Nature* 534, 7609 (2016), 631–639.
- [36] SCHIRMER, P. A., AND MPORAS, I. Statistical and electrical features evaluation for electrical appliances energy disaggregation. *Sustainability* 11, 11 (2019), 3222.
- [37] SINGHAL, V., MAGGU, J., AND MAJUMDAR, A. Simultaneous detection of multiple appliances from smart-meter measurements via multi-label consistent deep dictionary learning and deep transform learning. *IEEE Transactions on Smart Grid* 10, 3 (2018), 2969–2978.
- [38] TABATABAEI, S. M., DICK, S., AND XU, W. Toward non-intrusive load monitoring via multi-label classification. *IEEE Transactions on Smart Grid* 8, 1 (2016), 26–40.
- [39] TABATABAEI, S. M., DICK, S., AND XU, W. Toward non-intrusive load monitoring via multi-label classification. *IEEE Transactions on Smart Grid* 8, 1 (2017), 26–40.
- [40] WONG, Y. F., AHMET ŞEKERCİOĞLU, Y., DRUMMOND, T., AND WONG, V. S. Recent approaches to non-intrusive load monitoring techniques in residential settings. In *2013 IEEE Computational Intelligence Applications in Smart Grid (CIASG)* (2013), pp. 73–79.

BIBLIOGRAPHY

- [41] ZEIFMAN, M. Disaggregation of home energy display data using probabilistic approach. *IEEE Transactions on Consumer Electronics* 58, 1 (2012), 23–31.
- [42] ZEIFMAN, M., AND ROTH, K. Nonintrusive appliance load monitoring: Review and outlook. *IEEE transactions on Consumer Electronics* 57, 1 (2011), 76–84.
- [43] ZOHA, A., GLUHAK, A., IMRAN, M. A., AND RAJASEGARAR, S. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors* 12, 12 (2012), 16838–16866.