جامعة سيدي محمد بن عبد الله بفاس
UNIVERSITÉ SIDI MOHAMED BEN ABDELLAH DE FES

كلية العلوم والتقنيات فاس
FACULTÉ DES SCIENCES ET TECHNIQUES DE FÈS

N° d'ordre :74/2023

Centre des Etudes Doctorales : Sciences et Techniques et Sciences Médicales
Formation Doctorale : Sciences de l'Ingénieur, Sciences Physiques, Mathématiques et Informatique
Discipline : Génie électrique
Spécialité : Systèmes énergétiques et Informatique
Laboratoire : Systèmes Intelligents, Géo ressources et Energies Renouvelables

Thèse pour l'obtention de DOCTORAT

Energy disaggregation with computational intelligence techniques

Préparée par : Mr Inoussa Habou Laouali

Soutenue le 21/10/2023 devant le jury :

Nom et Prénom	Grade	Etablissement	Qualité
Mostafa MRABTI	PES	Ecole Nationale des Sciences Appliquées de Fès	Président
Bekkay HAJJI	PES	Ecole Nationale des Sciences Appliquées d'Oujda	Rapporteur
Mohammed BERRADA	PES	Ecole Nationale des Sciences Appliquées de Fès	Rapporteur
Abdellah MECHAQRANE	PES	Faculté des Sciences et Techniques de Fès	Rapporteur
Maria da Graca Cristo Dos Santos Lopes Ruano	PES	University of Algarve Portugal	Examinateur
Antonio Eduardo de Barros Ruano	PES	University of Algarve Portugal	Examinateur
Hakim EL FADILI	PH	Ecole Nationale des Sciences Appliquées de Fès	Co-Directeur de thèse
Saad BENNANI DOSSE	PES	Ecole Nationale des Sciences Appliquées de Fès	Directeur de thèse

Abstract

The world is facing challenges related to climate change, the growing increase in global energy consumption and the increasingly worrying decrease in fossil energy resources. The public's awareness of environmental problems and the high costs of energy has increased in recent years. Meanwhile, a significant portion of electrical energy consumption is still associated with the improper use of electrical appliances.

With the widespread use of smart meters, allowing fine-grained monitoring of electricity consumption, energy disaggregation in buildings has become possible, based on a single measurement of aggregated data (current, voltage, active and/or reactive powers, etc.). Non-intrusive load monitoring (NILM) refers to a set of approaches for predicting the electrical energy consumption of devices based on measurements of aggregated power demand data from multiple devices at the building's energy distribution system. It has become one of the most practical solutions for energy disaggregation. Computational intelligence techniques are the most cost-effective and efficient way to fulfill energy-saving goals using NILM. Indeed, it has been shown that appliances exhibit unique energy consumption patterns. NILM techniques enable to identify the fingerprints of each device from the aggregated powers consumption using computational intelligence approaches.

This thesis is focused on the analysis of the practical challenges of deploying low-cost NILM hardware and low-complexity disaggregation algorithms. Specifically, we focused on the use of low-frequency power signals and methods for designing offline NILM algorithms with high-level computational performance and very fast online operation. The first part is focused on the analysis and collection of data from a typical residential house located in the Algarve, Portugal, in the context of a real-life scenario. Then, an optimization approach based on deep learning methods is proposed. The framework relies on a hybrid deep learning architecture based on a convex hull data selection approach using low frequency (1 Hz) power data. It is based on the use of active and reactive powers features as inputs. The results achieved demonstrated the efficiency of the proposed approach, reaching F1 values up to 99% and estimation accuracy values up to 98%. It was shown that using the proposed data selection approach significantly improves the results of the designed models, especially for multi-state devices. However, to achieve satisfactory performance, these techniques need a substantial training data. Moreover, these methods benefit greatly from models with many training parameters, which in most cases require expensive or not readily available processing capacity. To tackle these challenges, we proposed a NILM framework based on a 1 minute sampling rate,

thus enabling the use of low-cost meters, and using shallow and low complexity neural network models. The framework is based on the design of radial basis function neural networks by a multi-objective genetic algorithm (RBFNN-MOGA), with design data selected by an approximate convex hull algorithm. Many experiments were conducted. A comparative analysis was conducted to evaluate the RBFNN-MOGA framework against the proposed approach. The results of the analysis demonstrated that the RBFNN-MOGA model does not require too much training data, while achieving similar or better performance than approaches using more training data. Furthermore, the effectiveness of the proposed approach in obtaining the best estimate of the energy consumed by each device in the house was highlighted by comparing it with other state-of-the-art methods using both distinct and common data.

Overall, the approaches proposed in this thesis to disaggregate the electrical energy consumption have been effectively tested on real-world data, and we are optimistic that they will contribute to solving efficiency problems in households.

Keywords: energy disaggregation, non-intrusive load monitoring (NILM), neural networks, low frequency power data, multi-objective genetic algorithm, convex hull algorithms

Résumé

Le monde est confronté à des défis liés au changement climatique, à l'augmentation croissante de la consommation énergétique mondiale et à la diminution de plus en plus inquiétante des ressources en énergie fossile. La sensibilisation du grand public aux problèmes environnementaux et aux coûts élevés de l'énergie a augmenté ces dernières années. Cependant, une part importante de la consommation d'énergie électrique est encore liée à une mauvaise utilisation des appareils électriques.

Avec la généralisation des compteurs intelligents, qui permettent un suivi fin de la consommation d'électricité, la désagrégation de l'énergie dans les bâtiments et les maisons est devenue possible, sur la base d'une seule mesure de données agrégées (courant, tension, puissances active et/ou réactive, etc.). La surveillance non intrusive de la charge (NILM) fait référence à un ensemble d'approches permettant de prédire la consommation d'énergie électrique des appareils à partir de mesures de données agrégées de demande de puissance provenant de plusieurs appareils au niveau du système de distribution d'énergie du bâtiment. Elle est devenue l'une des solutions les plus pratiques pour la désagrégation de l'énergie. Les techniques d'intelligence informatique constituent le moyen le plus rentable et le plus efficace d'atteindre les objectifs d'économie d'énergie en utilisant la MNIL. En effet, il a été démontré que les appareils présentent des schémas de consommation d'énergie uniques. Les techniques de la NILM permettent d'identifier les empreintes de chaque appareil à partir de la consommation d'énergie agrégée en utilisant des approches d'intelligence computationnelle.

Cette thèse se concentre sur l'analyse des défis pratiques du déploiement de matériel NILM à faible coût et d'algorithmes de désagrégation à faible complexité. Plus précisément, nous nous sommes concentrés sur l'utilisation de signaux de puissance à basse fréquence et sur les méthodes de conception d'algorithmes NILM hors ligne avec des performances de calcul de haut niveau et un fonctionnement en ligne très rapide. La première partie est axée sur l'analyse et la collecte de données provenant d'une maison résidentielle typique située dans l'Algarve, au Portugal, dans le cadre d'un scénario réel. Ensuite, une approche d'optimisation basée sur des méthodes d'apprentissage profond est proposée. Le cadre s'appuie sur une architecture hybride d'apprentissage profond basée sur une approche de sélection de données par coque convexe à partir de données électriques basse fréquence (1 Hz). Il est basé sur l'utilisation des caractéristiques des puissances actives et réactives comme entrées. Les résultats obtenus ont démontré l'efficacité de l'approche proposée, atteignant des valeurs de F1 jusqu'à 99% et des

valeurs de précision d'estimation jusqu'à 98%. Il a été démontré que l'utilisation de l'approche de sélection des données proposée améliore considérablement les résultats des modèles conçus, en particulier pour les dispositifs à états multiples. Cependant, pour obtenir des performances satisfaisantes, ces techniques ont besoin d'un nombre important de données d'entraînement. De plus, ces méthodes bénéficient grandement des modèles avec nombreux paramètres d'entraînement, qui dans la plupart des cas nécessitent une capacité de traitement coûteuse ou peu disponible. Pour relever ces défis, nous avons proposé un cadre NILM basé sur un taux d'échantillonnage d'une minute, permettant ainsi l'utilisation de compteurs à faible coût, et utilisant des modèles de réseaux neuronaux peu profonds et peu complexes. Le cadre est basé sur la conception de réseaux neuronaux à fonction de base radiale par un algorithme génétique multi-objectif (RBFNN-MOGA), avec des données de conception sélectionnées par un algorithme de coque convexe approximative. De nombreuses expériences ont été menées. Une analyse comparative a été menée pour évaluer le cadre RBFNN-MOGA par rapport à l'approche proposée. Les résultats de l'analyse ont démontré que le modèle RBFNN-MOGA ne nécessite pas trop de données d'entraînement, tout en obtenant des performances similaires ou supérieures à celles des approches utilisant plus de données d'entraînement. En outre, l'efficacité de l'approche proposée pour obtenir la meilleure estimation de l'énergie consommée par chaque appareil de la maison a été mise en évidence en la comparant à d'autres méthodes de pointe utilisant des données distinctes et communes.

Dans l'ensemble, les approches proposées dans cette thèse pour désagréger la consommation d'énergie électrique ont été testées efficacement sur des données du monde réel, et nous sommes optimistes quant à leur contribution à la résolution des problèmes d'efficacité dans les foyers.

Mots clés : désagrégation de l'énergie, surveillance non intrusive de la charge (NILM), apprentissage automatique, réseaux neuronaux, algorithme génétique multi-objectif, algorithmes de coque convexe

Acknowledgments

First, I would like to thank Pr. Antonio Ruano and Pr. Maria da Graça Ruano for all their valuable advice, guidance, support, encouragement and for providing me with their expertise. Thank you for welcoming me to the University of Algarve and for facilitating my integration. Undoubtedly, this thesis would not have turned out the way it did without your support and encouragement.

I would like to sincerely thank my supervisors Pr. Saad Dosse Bennani and Pr. Hakim El Fadili for accepting me as a PhD candidate, for trusting me, for their precious advices, encouragement, and support as well as their guidance throughout this research work. It is obvious that this adventure would not have started and would not have gone as far as it has without you.

I would like to take this opportunity to express my sincere gratitude to the president of the jury, the rapporteurs and the examiners for having accepted to evaluate my work, despite their busy schedules. Thank you for your valuable comments and suggestions.

I would also like to thank all my colleagues from the laboratory SIGER at Sidi Mohamed ben Abdellah University as well as the colleagues from the Center for Intelligent Systems at the University of Algarve. Thank you for contributing to make my research experience easy and enjoyable during all these years.

Last but not least, I would like to thank my parents and my family, who have always supported me and accompanied me in my projects. This work is dedicated to you, thank you for always being present for me. Finally, I would like to thank all my friends and all those who contributed to the realization of this work.

This dissertation and the work necessary for its realization were carried out at the Department of electronics engineering and Informatics of the Faculty of Science and Technology of the University of Algarve under the supervision of Pr Antonio Ruano. I also acknowledge the support of the Portuguese Foundation for Science and Technology (Fundação para a Ciência e Tecnologia) under the Programa Operacional Portugal 2020 and Operational Program CRESC Algarve 2020, grant numbers 39578/2018 and 72581/2020.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AAL - Ambient Assisting Living

ACS-F1 - Appliance Consumption Signatures - Fribourg 1

ADL - Activities of Daily Living

ALM - Appliance Load Monitoring

AMPds - Almanac of Minutely Power datasets

ANN - Artificial Neural Network

AFAMAP - Additive Factorial Approximate Maximum a Posteriori

BLOND - building-level office environment dataset

BLUED - Building-Level fully labeled Electricity Disaggregation dataset

CNN - Convolutional Neural Network

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

DT – Decision Trees

DFT - Discrete Fourier Transform

DISEC - Dataset on Information Strategies for Energy Conservation

DRED - Dutch Residential Energy Dataset

EMC - Electromagnetic Compatibility

EMI - Electromagnetic Interference

EMIL – Electro Magnetic Interference for Load disaggregation dataset

FFT - Fast Fourier Transform

FHMM - Factorial Hidden Markov Model

FN - False Negative

FNN - Feedforward Neural Network

FP - False Positive

FSM - Finite State Machine

GA - Genetic Algorithm

GPS – Graph Signal processing

GLR - Generalized Likelihood Ratio

GOF – Goodness-Of-Fit

HAC - Hierarchical Agglomerative Clustering

HOS – Higher order statistics

HMM - Hidden Markov Model

HUE - hourly usage energy

HVAC - Heating, Ventilation and Air Conditioning

IAH - Inside Air Humidity

IAT - Inside Air Temperature

iAWE - Indian Dataset for Ambient Water and Energy

ILM - Intrusive Load Monitoring

IWS - Intelligent Weather Station

K-NN - K nearest neighbors

LM - Levenberg Marquardt

LMS - Least Mean Square

LS - Least Square

MAD - Mean Absolute Deviation

MAE - Mean Absolute Error

MEULP - Measured end-use electric Load profiles

MLP - Multi-Layer Perceptron

MOGA - Multi Objective Genetic Algorithm

NN – Neural Network

NARX -Non-linear AutoRegressive with eXogenous inputs

NILM - Non-Intrusive Load Monitoring

OAKM - Optimal Adaptive K-Means

OAT - Outside Air Temperature

PCA - Principal Components Analysis

PFC - Power Factor Correction

PFM - Pulse-Frequency Modulation

PLAID - Plug-Level Appliance Identification Dataset

PWM - Pulse-Width Modulation

QUD - Qatar university dataset

RAE - Rainforest Automation Energy

RBFNN - Radial Basis Function Neural Network

REDD - Reference Energy Disaggregation Dataset

RMS - Root Mean Square

RMSE Root Mean Square Error

SMPS - Switched-Mode Power Supply

SP - Smart Plug

SPWS - Self-Powered Wireless Sensors

STFT - Short-Time Fourier transform

SVMs - Support Vector Machines

TN - True Negative

TP - True Positive

UK-DALE - UK Domestic Appliance-Level Electricity dataset

WBs - Circutor Wibees

Chapter 1 Introduction

1.1. Context

The world is facing challenges related in particular to climate change, the growing increase in global energy consumption and the increasingly worrying decrease in fossil energy resources. Indeed, several studies have highlighted the impact of energy consumption on CO2 emissions [1]–[5] while warning against the necessity of developing and implementing long-term energy policies. Besides, climate change caused by greenhouse gas emissions, primarily CO2 pollutions, poses unprecedented threats to human progress and survival, including extreme weather conditions, species eradication, and food scarcity [6], [7]. This issue has become a priority for a growing number of governments leading to agreements like the one in Paris (Paris agreement 2015, [8]) aimed at limiting global warming to 2°C compared to an unmitigated scenario of 4.5°C warming [9]. Figure 1.1 illustrates future global greenhouse gas emission scenarios based on a set of hypotheses. (i) if there are no climate policies enacted; (ii) if current policies are maintained; (iii) if all nations reach their current future emission reduction targets as well as essential pathways that are compatible with keeping warming to 1.5°C or 2°C [10].

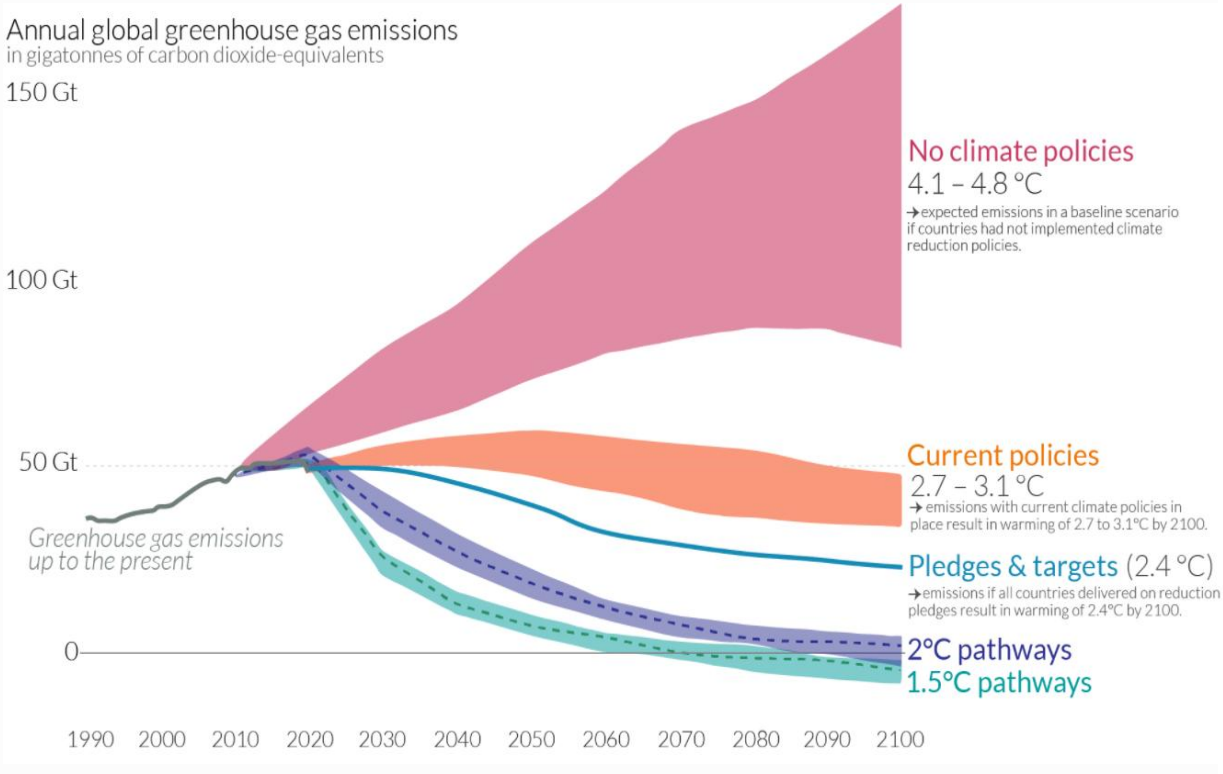


Figure 1-1. Global greenhouse gas emissions and warming scenarios. From [10].

Furthermore, low-carbon economies have been the focus of global attention as a means of reducing energy usage and greenhouse gas emissions. For example, the Chinese government has stated that peak CO2 emissions and carbon neutrality would be reached respectively by 2030 and 2060 in China [11]. Similarly, the European Union (EU) makes a concerted effort to mitigate climate change by enacting a series of strategic policies [12], [13]. Moreover, according to studies reported in [14], global energy consumption would rise from an estimate of 106 quadrillion Btu between 2010 and 2020, to 820 quadrillion Btu by 2040. This translates to a 56 % increase in energy usage over the next 30 years. Therefore, developing countries would account for 85 % of this increase owing to sustained economic expansion and population growth [15]. One of the challenges is to fight effectively against greenhouse effects and move towards the consumption of clean and renewable energy. Meanwhile, energy consumption is one of the main impact factors regarding CO2 emissions [16]. Meeting these challenges therefore requires controlling energy consumption and developing energy management strategies.

One of the major parts of total energy consumption in most countries is energy consumption in the building sector. For instance, in the United States the building sector represents for approximately 76% of electricity consumption and 40% of all primary energy consumption [17]. China ranked first in the world in energy consumption and CO2 emissions, accounting for 24% of global consumption, with energy consumption taking just 20 years to double at an average growth rate of 3.7% [4], [18]. Also, in Europe the building sector represents almost 40% of total energy consumption [19], [20]. It is identified as one of the most promising targets with great energy saving potential associated with the improvement of infrastructure and equipment by the European Union [9], [21]. It is therefore of paramount importance to examine residential energy consumption to promote energy savings and predict CO2 emissions. Figure 1-2 shows the evolution of global final electricity consumption by sector from 1974 to 2019. As it can be seen, since 1974, the residential sector, as well as commercial and public sectors, have accounted for a large proportion of the increase in power consumption especially in the organization for economic cooperation and development (OECD) [22].

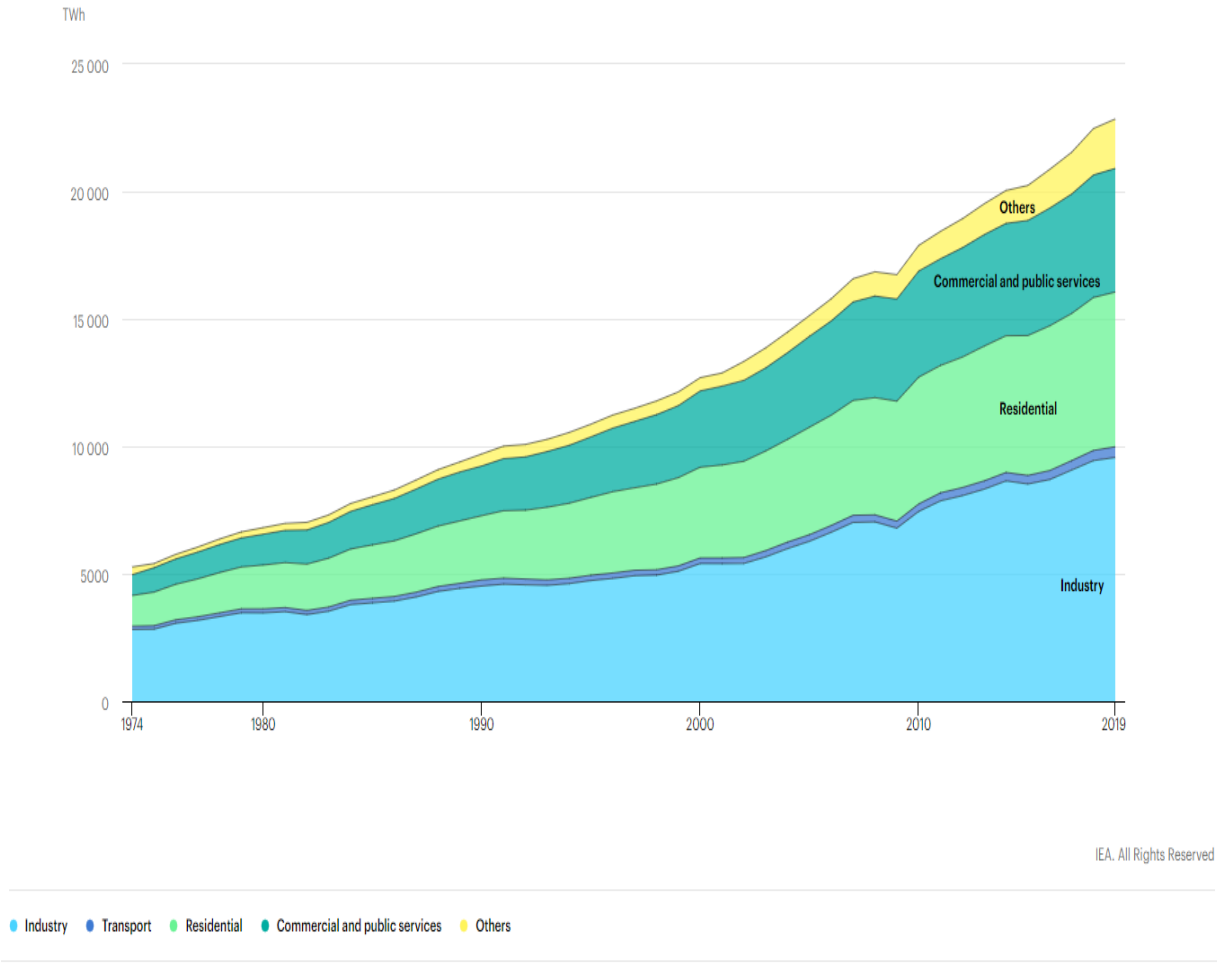


Figure 1-2. Global final electricity consumption by sector, 1974-2019. From [22]

In fact, there are numerous opportunities for lowering building electricity consumption [23]. Several studies revealed that providing direct and real time feedback about residential electrical demand can lower energy consumption by 10-15% [24]–[26]. Indeed, users will be able to better manage their usage and save energy. Even though, studies indicate that aggregate feedback can result in considerable energy savings [27], [28]. However, the lack of specificity limits its usefulness. Thus, while consumers may realize that their home is consuming considerably higher energy at any particular moment, they should test numerous devices to identify which one is primarily responsible. Besides, as some major loads, like water heaters, refrigerators, and ovens, cannot be switched on or off directly by the user, determining their power usage and providing appropriate feedback may be problematic if just a real-time estimate for the entire home is available [23]. Moreover, other studies claim that if consumers had access to effective, real-time, disaggregated power data, they could save even more energy [23]. According to [29], the most significant energy reductions may be achieved by monitoring

energy use at appliance level. Therefore, energy disaggregation techniques should be used to analyze energy at the appliance level. The disaggregation consists of a set of statistical techniques for separating device-level data from a global aggregate data from whole-building energy signal.

One of the most important factors in reducing energy consumption in the buildings sector is consumers awareness about their electricity consumption [30]. Although energy behaviors are a significant determinant of building energy usage, the potential for energy savings owing to behavior is often overlooked, despite being reported as being as considerable as that of technological solutions [9].

Moreover, device-specific data has numerous benefits as highlighted in [29]. First, direct feedback and personalized recommendations. It may be analyzed which specific HVAC systems, or devices among the many available could most efficiently lower energy consumption for a certain house, and then automate the transmission of supplementary information to eliminate hurdles and stimulate action. Thereafter, there are benefits in terms of research and development. The diagnostics can be used to perform automatic commissioning, which recommends changes to building operations to improve quality and performance, as well as fault detection, which alerts users when a device needs to be repaired as a result of a malfunction leading it to use more energy than it should. In addition, there are the utility and policy benefits. Once people acknowledge where they are spending energy, they are more motivated to demand control systems, and demand response programs [31], [32].

Furthermore, the emergence and widespread usage of smart meters has opened new possibilities, such as collecting electrical energy consumption at considerably finer temporal resolutions, on the range of a few seconds to a few minutes at building level or house level [33]. For instance, 96 million of smart meters had been installed in China and 70 million, and 2.9 million respectively in USA and UK by the end of 2016 [34]. One topic to consider is how to use the vast amounts of data generated by smart meters to encourage and enhance energy efficiency? Smart meters can therefore no longer be used solely for billing. Meanwhile, consumers' electricity consumption habits and lifestyles can be gleaned from high-resolution data collected by smart meters [34]. Moreover, the data collected could be readily transferred to the data centers for processing and storage. This provides customers and grid operators with tremendous potential to evaluate, analyze, and integrate data, as well as providing new energy-based services [33], [34].

1.2. Motivation

In general, load monitoring is the process of detecting and estimating the load in a power system. These load measurements will be used to estimate the energy consumption and state of the devices, helping users to better understand how much energy each load in the system uses [35]. The primary purpose of appliance load monitoring is to supply energy feedback to the homeowners allowing them to have a detailed energy analysis and information on the distribution of the energy consumed. Meanwhile, it is also beneficial in several areas due to its applicability in home energy management systems [30], fault detection [36], ambient assisted living [30], remote load monitoring services [37] and so on. There are two primary types of device monitoring approaches: On one hand, Intrusive Load Monitoring (ILM) techniques based on the installation of sensors on each load of interest in the house and non-intrusive load monitoring approaches (NILM) on the other hand, based on aggregate measurement data gathered from a smart meter at the level of the distribution panel allowing the detection of the specific appliance's consumption. ILM determines the operating conditions of each appliance in the house using one or more sensors attached to each appliance. This method can properly estimate the consumption of each device. However, the need to install multiple sensors and the intrusive nature of ILM approach limits its practical application. These disadvantages include the cost and complexity of installation and huge configurations effort to set up multiple sensors for each device, especially when the monitoring scenario involves multiples devices, and some privacy concerns [30], [35], [37], [38]. Therefore, research is more focused on non-intrusive approaches that allow for the disaggregation of each device's usage while maintaining consumers privacy and using a single sensor installed at the house's distribution panel.

With the widespread use of smart meters, energy disaggregation in buildings and houses has become feasible, based on the readings of aggregate current, voltage, or powers. NILM refers to a set of approaches for predicting the energy consumption of appliances based on the aggregate data readings demand across multiple appliances at the building's power distribution system. It has become one of the most practical solutions for energy disaggregation [30]. One of the advantages of NILM is the ability to offer customers real-time power usage feedback, allowing them to make behavioral changes that will save resources and money. Moreover, power companies, users, home energy managements systems, and so on, can utilize the disaggregated data to enhance the quality of electricity use and better understand how it is being expended.

In fact, the multitude of distinct households' electrical devices continues to increase. As a result, it becomes particularly important to realize the potential for energy conservation and demand control, i.e., the ability to anticipate or postpone energy use in order to establish power grid stability. Hence, it is essential to have thorough information of not just the devices present in a house, but also their specific power consumption and operating periods [39]. Computational intelligence techniques are the most cost-effective and efficient way to fulfill energy-saving goals using NILM. Indeed, it has been shown that appliances exhibit unique energy consumption patterns [40]. NILM techniques enable to identify the fingerprints of each device from the aggregated powers consumption using computational intelligence approaches. However, some appliances have been observed to be difficult to identify based just on the aggregated energy use [30], [41]. Most NILM algorithms in the literature are suitable for two-state devices owing to their basic architecture. Nonetheless, the disaggregation of multistate devices remains a challenging topic.

Furthermore, numerous challenges must be overcome in order for NILM to become a feasible and practical solution for energy disaggregation [42]. These challenges include the proper disaggregation of devices, given the aggregated data, when more than two devices with comparable electrical signatures are turned on in the same electrical circuit, high precision in detecting and estimating device consumption, low complexity, and cheap cost. Another challenge for NILM approaches is the data requirements. Many datasets are now accessible to the general public. However, numerous disadvantages, such as missing data, diverse data formats, gaps, limited labeling data and covering only a few developed countries, limit their practical application [41].

1.3. NILMforIHEMS Project

The experiments presented in this thesis work were conducted as part of a research project funded by the Portuguese Foundation for Science and Technology (Fundação para a Ciência e Tecnologia (FCT) Grant number: SAICT-ALG/39578/2018), implemented in the Faculty of Science & Technology of the University of Algarve. The official project name is **Non-Intrusive Load Monitoring applied to Intelligent Home Energy Management Systems (NILMforIHEMS)** and aims to improve the disaggregation results of existing algorithms and the efficiency of energy systems in homes [43].

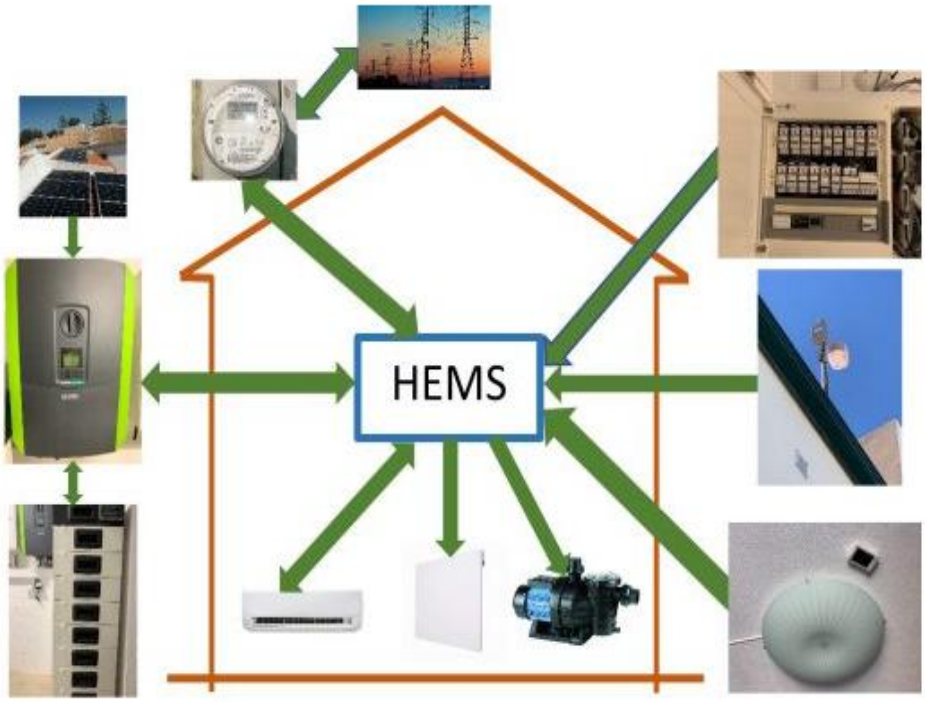


Figure 1-3. NILMforIHEMS project [43]

The project is divided into 6 activities:

- Implementation of data acquisition systems
- Non-intrusive load monitoring
- Forecasting of energy consumption
- Home energy management systems
- Tuning and Testing
- Project management

1.4. Research Goals and Contributions

One of the main objectives of this thesis is to analyze the practical challenges of deploying low-cost NILM hardware and low-complexity disaggregation algorithms. Specifically, we will focus on the use of low-frequency power signals and design methods for offline NILM algorithms with high-level computational performance and very fast online operation. The research goals and associated subtasks are described as follows:

- To review the state of the art of non-intrusive load monitoring literature, and discuss the concepts and challenges involved.
- To design disaggregation algorithms using electrical features available in public datasets.

- To collect data in a case study house under a real-life scenario and examine the different electrical features and their potentials.
- To explore methods that have not yet been employed to tackle the NILM challenge and propose new approaches for efficient design of NILM algorithms.

The following papers have been published as a result of the work done throughout this PhD.

- **Journal papers**

Laouali, I.; Gomes, I.; Ruano, M.d.G.; Bennani, S.D.; Fadili, H.E.; Ruano, A. Energy Disaggregation Using Multi-Objective Genetic Algorithm Designed Neural Networks. *Energies* **2022**, *15*, 9073. https://doi.org/10.3390/en15239073

Laouali, I.; Ruano, A.; Ruano, M.d.G.; Bennani, S.D.; Fadili, H.E. Non-Intrusive Load Monitoring of Household Devices Using a Hybrid Deep Learning Model through Convex Hull-Based Data Selection. *Energies* **2022**, *15*, 1215. https://doi.org/10.3390/en15031215.

Bot, K.; Santos, S.; Laouali, I.; Ruano, A.; Ruano, M.d.G. Design of Ensemble Forecasting Models for Home Energy Management Systems. *Energies* **2021**, *14*, 7664. https://doi.org/10.3390/en14227664.

Bot, K.; Laouali, I.; Ruano, A.; Ruano, M.d.G. Home Energy Management Systems with Branch-and-Bound Model-Based Predictive Control Techniques. *Energies* **2021**, *14*, 5852. https://doi.org/10.3390/en14185852.

- **Conference papers**

I. H. Laouali, A. Ruano, M. G. Ruano, S. D. Bennani and H. El Fadili, " MOGA designed neural networks for non-intrusive load monitoring" IEEE 20th Jubilee International Symposium on Intelligent Systems and Informatics (SISY 2022), September 15-17, 2022, Subotica, Serbia.

Laouali Inoussa Habou, Bot Karol, Ruano Antonio, Ruano Maria da Graça, Bennani Saad Dosse, and El Fadili Hakim, "Low frequency-based energy disaggregation using sliding windows and deep learning," E3S Web Conf., vol. 351, p. 1020, 2022, https://doi.org/10.1051/e3sconf/202235101020.

Ruano, A., Qassemi, H., Habou Laouali, I., Marzouq, M., Fadili, H.E., Dosse, S.B. (2022). A Model-Based Predictive Control Approach for Home Energy Management Systems. First Results. In: Bennani, S., Lakhrissi, Y., Khaissidi, G., Mansouri, A., Khamlichi, Y. (eds) WITS 2020. Lecture Notes in Electrical Engineering, vol 745. Springer, Singapore. https://doi.org/10.1007/978-981-33-6893-4_67.

Laouali, I.H., Qassemi, H., Marzouq, M., Ruano, A., Dosse, S.B., El Fadili, H. (2022). A Non-Linear Autoregressive Neural Network Model for Forecasting Appliance Power Consumption. In: Bennani, S., Lakhrissi, Y., Khaissidi, G., Mansouri, A., Khamlichi, Y. (eds) WITS 2020. Lecture Notes in Electrical Engineering, vol 745. Springer, Singapore. https://doi.org/10.1007/978-981-33-6893-4_69.

I. H. Laouali, H. Qassemi, M. Marzouq, A. Ruano, S. D. Bennani and H. El Fadili, "A Survey on Computational Intelligence Techniques for Non-Intrusive Load Monitoring," 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), 2020, pp. 1-6, https://doi.org/10.1109/ICECOCS50124.2020.9314383.

1.5. Thesis Outline

The rest of the content of this thesis is structured as follows. In Chapter 2 a comprehensive and detailed overview of the current state of the art of energy disaggregation is presented. The chapter starts with a brief introduction of load monitoring concepts and an overview of the evolution of energy disaggregation research. Following this, the main components of NILM and related work are presented. Specifically, the chapter discusses the data acquisition process, extraction of device features and electrical signatures, pattern learning, and the classification processes for load recognition reviewing both event-based and non-event-based NILM techniques. Finally, the chapter concludes with a discussion about the major research efforts conducted to benchmark the performance of different NILM techniques, with an overview of different accuracy measures and a review of the different public datasets available in the literature.

Chapter 3 presents a brief overview of the theoretical background behind the development of this thesis work. First, the basic concepts of machine learning methods were introduced. Then, the models used, including CNNs, LSTMs, SVMs, KNNs, and DTs, are briefly described.

Then, multi-objective optimization approaches with emphasis on evolutionary algorithms and in particular the basic concept of genetic algorithms are discussed. The design of the radial basis function neural network by the multi-objective genetic algorithm is then illustrated. Finally the data selection approach used is introduced.

Chapter 4 deals with NILM approaches based on deep learning techniques. First, the chapter tackles the NILM formulation problem and presents the proposed optimization approach. The proposed framework relies on a hybrid deep learning architecture based on a convex hull data selection technique using low frequency power data. The methods used, including the data selection approach, LSTM, and CNN models, are then presented. The data are gathered from a residential house located in Faro, Portugal. Therefore, the case study house and data collection method are described. The results of the experiments and a discussion including results from the literature are reported at the end of chapter.

In chapter 5, a NILM framework based on the design of radial basis function neural networks by a multi-objective genetic algorithm (MOGA), with design data selected by an approximate convex hull algorithm is presented. The framework uses a low frequency sampling rate, thus allowing the use of low-cost meters, and shallow neural network models. Following this, the design of radial basis function neural networks using the multi-objective genetic algorithm (RBFNN-MOGA) approach for energy disaggregation is detailed. Finally, the chapter presents the experiments conducted and the corresponding results and discussions.

Chapter 6 tackles the application of MOGA design on a long period of data. First a statistical analysis has been conducted in the case study house considering a long period of data. Then the experiment was conducted using one year of data. The results including the performances and limitations of the models were discussed at the end of the chapter.

Chapter 7 summarizes the contributions of this thesis, while discussing the perspectives and challenges for future work.

Chapter 2 Energy Disaggregation: State of the art

2.1. Introduction

This chapter presents a comprehensive overview of the current state of the art of NILM. First, the concept of load monitoring and an overview of energy disaggregation is briefly introduced. Then, the main components of NILM and related work are presented. Specifically, the chapter discusses the process of data acquisition, the extraction of possible device characteristics and electrical signatures, the pattern learning and the classification processes for load recognition, reviewing both eventless and event-based state of the art NILM methods. Finally, the main research efforts conducted to benchmark the performance of different NILM methods with an overview of different accuracy metrics and a survey of different public datasets available in the literature are discussed.

2.2. Notion of Load Monitoring

The notion of load monitoring has gained increasing interest in this area of research driven by parallel advancements in artificial intelligence, internet of things, data communication and networks, sensing technology and machine learning techniques [37]. Load monitoring is a specific process of detecting and collecting load measurements in an electrical system [35]. The primary purpose of any load monitoring system is to offer a fine-grained energy breakdown per device, allowing for energy consumption optimization while maintaining occupant comfort [44]. These load measurements will provide the energy usage and operating condition of devices to better understand the energy consumption of specific loads in the system. It is a precondition for delivering energy feedback to homeowners, but it is also useful to the industrial sector due to its application in a variety of areas such as fault detection, home energy management systems and remote load monitoring. There are two basic ways to perform device load monitoring: Intrusive Load Monitoring (ILM) and Non-Intrusive Load Monitoring (NILM), which are referred in literature as distributed sensing approach and single-point sensing approach respectively [37].

ILM is defined as load monitoring that entails the installation of one or more measurement devices at each load of interest [41]. The operating condition of each load can be accurately predicted using this approach. It can be divided into three segments: the sub-metering system, in which a meter is assigned to a zone of devices, smart outlets, in which devices associated to a specific outlet are assigned a meter for monitoring, and smart devices, in which each device

is assigned a single meter or meters embedded in the devices [45]. The group of devices connected to a single sensor can be used to determine the level of intrusiveness of ILM. In fact, ILM system employs smart plugs to measure the energy usage of single loads, with the most of meters displaying the energy usage on a display screen and some capable of sending the readings to a data center via wireless connection, wired internet or USB [46]. The wireless connections between the sensors and monitoring systems utilize wireless communications like VLAN, Zigbee, WI-FI, and Bluetooth [35]. ILM approaches necessitate the deployment of sensors on each load in order to collect its electrical properties, resulting in high-quality load analysis [47]. However, some disadvantages, such as high costs, the intrusive nature which involve some privacy concerns, multiple sensor configuration, and installation complexity, limit its practical application, encouraging the use of non-intrusive approaches, especially in large-scale installations. As a result, emerging and start-up businesses, as well as academic researchers, have concentrated their efforts on improving non-intrusive techniques in order to make them a feasible solution for practical applications [37], [48].

NILM, on the other hand, is one of the most promising tools for energy disaggregation. It refers to the process of identifying the footprints of specific electrical appliances utilizing only the aggregate data and disaggregation algorithms. It enables users to estimate the consumption of each device in the house using the aggregate data gathered from a smart meter, which is typically placed in a home power panel, while maintaining user privacy [30], [49]. Furthermore, it allows the monitoring of household appliance usage without the necessity for installing single sensors for each appliance, thus, reducing electrical system complexity and associated cost [50]. The main objective could be to assist users in reducing their energy consumption by providing detailed electricity bills, or to detect malfunctioning devices, or to obtain feedback about device usage behavior or to help grid operators in controlling the grid [51].

2.3. Overview of Energy Disaggregation

Research on energy disaggregation started with the pioneering work of George Hart in the mid-1980s [40], [52]. In their approach, NALM was defined as a non-intrusive appliance load monitor that predicts the usage of specific devices turned on and off in an electrical load, based on an itemized analysis of the total load voltage and current, measured at the interface of the power panel. The original NALM methods were based on the concept that every change in a building's aggregate consumption occurs as a result of an electrical device changing its state, such as a device turning on or off. The approach was basically designed for utilities to collect

statistics on energy usage. It's referred to as nonintrusive to distinguish it from prior methods of acquiring device load data, which required installing sensors on single device and, therefore an invasion in the energy consumer's privacy. Hart used a model-based method to describe specific appliances and their combination to accurately breakdown the overall load into its components. These models generate certain signatures that may be identified in the total load to determine the activities of the individual components. For instance, if the house includes a fridge that consumes 250 W and 200 VAR, then an increase of this feature size means that the fridge is turned on, and a reduction of this size means that the fridge is turned off. Other devices have other feature signatures. After determining the exact on and off times of the signature events, all desired statistics can be tabulated.

The suggested Hart algorithm entails taking real and reactive power measurements from the electricity grid at 1 second intervals. The measurements are then normalized and utilized in the edge detection phase which will seek to identify when the devices have changed their operating state. The total power change is calculated by subtracting the stable power level before the change from the constant power level after the change stopped for each power change identified. The observed differences are then clustered based on how much each measure has changed. The two-dimensional space represented by the real and reactive powers is subjected to a cluster analysis to pinpoint these changes. Clusters with opposite signs and similar magnitudes are matched. A best likelihood algorithm is used to associate unmatched clusters with existing or new clusters. The resulting clusters were then utilized to assign the ON and OFF clusters within each device based on their time of occurrence, with an assumption that each ON/OFF pair represents a single device cycle of operation. The power consumption was then calculated by taking the portion of positive power change and multiplying it by the time between the ON and OFF events. Afterwards, each pair of ON/OFF clusters was associated with a device name by comparing its attributes to all device classes listed in a separate database.

In fact, the proposed algorithm could be vulnerable to power fluctuations. Hart recommended the following normalization to reduce the observed power variability due to voltage fluctuations:

$$P_{norm}(t) = \left[\frac{120}{V(t)}\right]^2 \mathrm{P(t)} \quad (1)$$

Where P(t) denotes power and V(t) represents voltage. The linearity of the admittance for devices linked in parallel is used to normalize the data. The value of 120 V refers to the nominal voltage in the American electrical distribution system.

The approach can identify and track on-off devices (two state) pretty easily, but it appears to have difficulty identifying variable-load and multi-state devices. Similarly, devices with similar power consumption, such as computers and incandescent bulbs, cannot be distinguished. Moreover, the matching approach assumes that the positive power change (start on) corresponds to the negative power change (end on) (turn off). However, many devices change resistance after they are turned on, and the offset due to this power drift can be up to 10%[48], [53].

To tackle these above challenges, Hart et al. presented an upgraded version of the proposed NALM system that included multi-state device disaggregation [40], [54]. In the approach, they suggested modeling multi-state devices as finite state machines. However, these reports focus on the mathematical foundations of the problem and reveal minimal insight regarding the practical application of the principle.

Numerous research works have been conducted in recent years to enhance Hart results, and energy disaggregation is now a significant aspect of Smart Grid technology [41]. The state-of-the-art NILM techniques contain some basic concepts, although they are based on different approaches. The first step is data acquisition. Afterward, certain device characteristics, or signatures, must be designated and defined mathematically. The features are then detected in the whole aggregated data using a mathematical algorithm [48]. The key difference between existing approaches, however, is the models and features employed to detect the appliances.

The NILM state-of-the-art can be classified in several ways. Meanwhile, it is frequently categorized into event-based and eventless-based approaches. The event-based methods seek to identify and classify transient states (such as on/off events or device state changes) in the aggregated signal using either supervised or semi-supervised algorithms. Event-less techniques, on the other hand, use statistical and probabilistic machine learning algorithms to match each sample of aggregate data to the usage of a single appliance or a group of appliances. The training phase doesn't need any labeled transitions.

2.4. Data Acquisition

Data acquisition is the initial step in any monitoring system. It is frequently linked to a device or system that is highly comparable to existing electrical installations, and where different techniques can be used to collect the measurements of certain variables such as currents and/or voltages in a house [30]. Moreover, other variables derived from these current and voltage signals, such as I-V trajectory, apparent power, active power, power factor, reactive powers, harmonic distortion, or transient as well as their variation in time, can be determined and used as features [30].

Data gathering is an important factor that determines how the NILM algorithms will subsequently behave and what kind of challenges or applications they will be able to address [55]. NILM methods rely heavily on the sampling rate, which is the frequency at which the meter gathers data [30], [41]. The function of the data acquisition system is to collect aggregated data measurements at a suitable rate so that specific device patterns can be detected [37]. The sampling rate of the acquired data must be highlighted because it defines what type of data may be captured from the electrical signals [56]. There are two main approaches for data collection in the literature [37], [41], [48], [57]. The data may be collected either at a low frequency sampling rate or at a high frequency sampling rate.

2.4.1. Low Frequency sampling rate

The employment of a low frequency sampling rate leads to the normalization of the usage of smart meters. Macroscopic features are considered in this class since the fundamental period is either 60 Hz (USA) or 50 Hz (EU) [38]. Low frequency metering approaches are considered because they are more realistic considering today's smart meter technology and massive data management challenges [57]. The meters available on the market for data collection present a range of sampling rates. These offer pre-built platforms with rudimentary capability, but with some disadvantages, particularly in terms of flexibility and sampling rate [30]. The NILM system may be readily implemented on actual meters, while the analysis and data transmission would be effective [58]. However, the classical features are typically simplex, and the information included in these features is limited owing to the low sampling rate (1 Hz or less), thus constraining the performance attained and the ability to employ them in certain types of applications [30], [58].

In fact, power features like apparent power (S), active power (P), and reactive power (Q) are typically utilized in low frequency sampling approaches [41]. The authors of [59]

investigate sampling rates from 10 Hz to 0.03 Hz for energy disaggregation using active and reactive powers. In their study they considered 3 appliances including washing machine, television, and rice cooker. They found that when the data sampling rate is too low, the NILM performance may be significantly limited. They showed that to prevent the deterioration of NILM performance, the sampling rates should be at least 1-3 Hz. The authors of [60] focused on the disaggregation of multi-state appliances (dishwashers, washing machines and washer dryer). They proposed a practical solution based on convolution neural networks and a new post-processing technique. They tried to understand how the resolution of the measurements would influence the accuracy of the suggested NILM framework. They conducted a comparative study of the results obtained with sampling intervals of 10 s and 1 min. They found that the disaggregation results of dishwashers remained comparable while the performance of washing machine and washer dryer deteriorated significantly. The authors argue that the decrease is due to the fact that dishwasher consumption profiles are very stable, whereas washing machine consumption patterns fluctuate more frequently. These small distinguishing characteristics for washing machines can be lost in lower resolution data, resulting in performance deterioration. Therefore, they highlighted that for better results a certain level of resolution is needed for devices with fast power changes. The authors of [61] have evaluated the influence of the temporal resolution of the data on the accuracy of NILM algorithms. They noted that favorable low frequency sampling rates are between 1 Hz and 1/30 Hz. They concluded that a sampling rate of 1/30 Hz appears to be sufficient to fully exploit the positive effects of reduced execution times and memory utilization, in addition to providing more historical data to NILM algorithms at the expense of low degradation scores. The authors of [62] studied the performance of two NILM algorithms throughout a range of sampling intervals, which varied from 6 seconds to 15 minutes. They showed how the performance of NILM classification deteriorates in a non-linear fashion as the sampling rate diminishes and how this deterioration is variable according to the type of device.

2.4.2. High Frequency sampling rate

Most studies suggest using both the microscopic characteristics and macroscopic features to obtain high accuracy in device identification [30], [48], [57], [63]. Microscopic features like harmonics and signal waveforms can only be extracted at high frequency. The Nyquist sampling theorem or Nyquist-Shannon theorem states that the highest harmonic in a signal could be acquired by sampling at more than twice the frequency of the signal [64]. If a portion of a waveform appears to be problematic, the rule of thumb recommends that around 20 data

points per fundamental period are necessary [63]. Assuming that no harmonic greater than the 11th harmonic is of interest [48], a minimum sampling rate of roughly 1.2–2 kHz is required to acquire microscopic characteristics. In general, harmonics are used as additional characteristics, in complement to active and reactive powers change. The Fourier transform can generate a set of harmonics that can better characterize the signal than a single harmonic [38], [48]. The authors of [53] proposed to integrate the harmonics in the extension method of G. Hart [40]. The harmonic analysis considers only transient signals in order to save computational resources and enhance performance. The study reveals that the characteristics of the third harmonic are advantageous for better discrimination between incandescent light bulbs and computers. The approach employs the concept of spectral envelope, which refers to the first coefficients of a short-time FFT vector. The usage of spectral envelopes could be quite beneficial to NILM. Since variable-load devices may not be detected using macroscopic characteristics, spectral envelopes can distinguish aspects of devices, especially variable-load devices. However, merely detecting variable-load devices is insufficient for load monitoring since the power consumption of the devices must also be estimated. A solution to this issue has been suggested by the author of [65], [66]. They explored the correlations of the 5th and 7th harmonics and the active/reactive powers used by a variable load device. A polynomial function may be used to model them using the least squares approach because there is such a strong correlation [66]. The method's robustness for any variable-load appliance, however, is unknown because of the ambiguous physical nature of the correlation.

A least squares procedure is used in the spectral envelope approach for NILM [65]–[67]. This procedure applies a current spectral envelope to each of the database's "signature" envelopes. The distance metric is then used to classify the envelope: it is assigned to the signature envelope with the minimum distance. The methodology of spectral envelope for NILM can detect a range of appliances, including variable loads, but it appears to have various flaws [48]. Prior to classification and monitoring, each device must undergo extensive training. Furthermore, even if the macroscopic characteristics can be grouped to allow for automatic device detection, there is no evidence that the spectral characteristics can be clustered. Next, numerous practical cases have yet to be described in terms of performance accuracy. Finally, it is unknown whether the technique is robust, such that, how the performance of the approach is impacted by the presence of a new device with an unknown signature [48].

One of the advantages of using high frequency data is that it preserves the entirety of the signals, enabling for the extraction of the maximum amount of information [68]. The author of

[69] investigate the impact of sampling frequency on classification accuracy. They discovered that sampling frequencies higher than 4 kHz are required to develop more practicable and reliable device classifiers. The authors of [70] assert that it is even possible to identify two devices of the same type using raw data sampled at 1 MHz. In order to increase the distinction between the load characteristics, the authors of [71] extracted the power spectrum of the transient current waveform. Since transient fingerprints have a shorter length and differ for each device, they can increase load detection performance. However, extracting the transient features necessitates a high sampling frequency and substantial data storage capacity, leading to higher hardware costs.

In fact, utilizing high frequencies entails high costs in terms of both hardware and software complexity, and it also necessitates a large communication bandwidth to establish a connection to any monitoring or centralized station [41]. Although it is technically possible to overcome these challenges today, implementing these improvements into commercial smart meters will almost surely result in an increased ultimate cost [30]. One solution may be to determine features on-the-fly, employing a microprocessor embedded in the data gathering system and retaining only a small amount of data. Furthermore, the data gathering system might be designed to only collect and process important signal changes [48].

2.5. Feature extraction and device signature

The NILM's concept is based on the fact that each device has its own fingerprints or signatures [58]. Its effectiveness is strongly dependent on the uniqueness of device features and the approach used to extract them. The feature extraction technique is often considered as signal processing since it extracts crucial data from the raw current and voltage signals [58]. It is defined as the process of choosing the appropriate characteristics so that the power event fingerprints are reliable and have sufficient distinguishing characteristics between different devices. The electrical signature of each device, which is characterized by the properties extracted from the raw data is the key element for effective device detection. It denotes the nature of operation of the device and therefore its electrical properties. Its functioning behavior can provide useful information for device identification. The latter depends mainly on the device signatures, which are defined more precisely by the device category [37]. Hart was the first to classify the devices into 3 types [40].

- Type I: ON/OFF devices that have only two distinct states of operation, such as toasters, lamps, kettles.

- Type II: These are finite state machines (FSM), which are multi-state devices with a finite number of operating states. This includes appliances like washing machines, washer dryer, dishwasher, burner stove, refrigerators and so on. These devices' switching patterns are also consistent, making it easy for the NILM algorithm to identify their operating condition. However, detection does not only translate into which devices are turned on, but also into their states [30]. Furthermore, some devices may be replicated (for example, a house may have two refrigerators), and it may be essential to determine the operating state of each appliance using comparable load signatures.
- Type III: These are continuous variable appliances with an endless number of states and the power draw varies over time but not in a periodic manner. Light dimmers and power tools are examples of type III appliances. In general, disaggregating these types of devices from aggregate load measurements using NILM approaches is quite challenging.

Besides these categories of devices, the authors of [48] introduced a fourth type known as permanent consumer appliances or always-on devices. Once turned ON, these devices operate 24/7 with constant energy consumption until they are turned off again.

- Type IV: This type comprises devices like hardwired smoke alarms, telephones, and some external power supplies. Furthermore, since these devices often have a modest and constant energy consumption, they might be seen as a constant offset to the NILM challenge and hence aren't considered and analyzed further.

The authors of [72] introduced another classification of devices based on power types and device operating styles, which the authors used to extract relevant information for their detection. Furthermore, the authors argue that devices may have multiple operating modes based on user personalization and work styles, which must be considered throughout the process of feature extraction. Figure 2-1 illustrates the power consumption patterns of different categories of loads, which can be interpreted as a device characteristic to distinguish distinct types of devices.

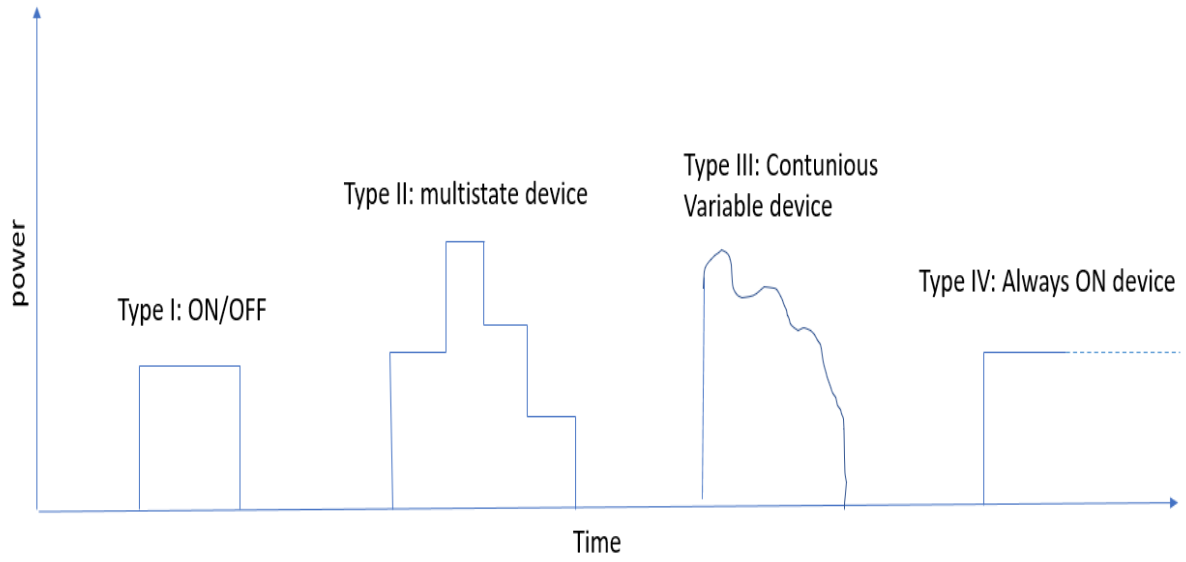


Figure 2-1. Different device types based on their power consumption pattern.

Energy disaggregation is performed by identifying active appliances from the aggregated signal. A set of features that uniquely characterize device behavior should be defined and closely related to both data collection and the methodologies that will be utilized for device recognition [30]. The characteristics depend highly on the sampling rate employed, which defines the frequency at which the device measures the data that will be used for NILM algorithms, and not as the sampling rate of the voltage and current that make up the appliance input [30]. As mentioned in the previous paragraph, the use of 1 s sampling period threshold allows the distinction between the high frequency (in the range of kHz) or microscopic features and low frequency (1Hz or less) or macroscopic features. In fact, device characteristics are classified into steady state signatures, transient features signatures, and non-traditional signatures.

2.5.1. Steady state signatures

The steady state features refer to the load characteristics retrieved when the device is operating in steady state. For example, a variation in the active power consumption from a lower value to a higher value in steady state may detect whether the device is switched on or off. In fact, the energy disaggregation algorithms based on steady-state analysis utilize steady-state characteristics acquired from the devices' steady-state operation [37]. These includes, steady state active and reactive powers, rms current, rms voltage, harmonics, and power factor [35]. The most common steady-state breakdown is based on characteristics collected from low-frequency data samples [58]. The NILM system can be simply added to existing meters thanks

to these features and thus facilitating data transmission and analysis. Hart's seminal NILM technique is based on the steady-state exploration of active and reactive power changes [40]. The approach performed well for on/off devices, but it has always struggled with multi-state and continuously variable devices.

NILM approaches utilizing active power as a single feature has been the subject of several research papers [51], [60], [73]–[75]. They noted that high-powered appliances with distinguishable energy consumption patterns, like water pumps and electric heaters, may be readily detected. This technique, on the other hand, ignores devices with comparable power consumption profiles [37]. Furthermore, simultaneous state transitions of devices provide inaccurate outputs. To tackle some of these challenges, it has been demonstrated that high-power devices and multi state devices may be detected easily by examining gradual changes in real and reactive power characteristics [40], [76]–[78]. However, the power change approach has difficulty distinguishing between overlapping devices in the P-Q function space, particularly low-power devices [37]. Subsequently, researchers attempted to analyze the voltage and current waveforms to extract device-specific characteristics like peak and root mean square (RMS) voltage and current values, phase difference information, and power factor (PF) to uniquely identify device activity and address the limitations of power-based approaches [79]–[81]. Experimental evaluations conducted in [82] have shown the efficacy of time domain features in identifying distinct loads. The root mean square features were reported to be more efficient than peak values; however, the experiment data does not include type-III devices [37]. Furthermore, there is no mention of detecting multiple device activation sequences working at the same time. The authors of [83], [84] developed a voltage noise approach based on Electromagnetic Interference (EMI) patterns. In their approaches, simultaneous events might be readily identified, however, devices without switched mode power supply (SMPS) could not be identified.

In [85], steady state features based on empirical mode decomposition (EMD) were employed to minimize the effect of voltage fluctuations in the power signal for accurate device recognition. In order to enhance appliance recognition performance, a significant number of time series steady state features were utilized by the authors of [86]. In their approach, signature analysis employs various features including chaotic time series analysis, model fitting and forecasting, correlations and stationery, time series distribution, and information theoretic and complexity measurements. Experiments conducted using the suggested time series features lead to an enhancement in the recognition accuracy of the device. The authors of [87] suggested the

usage of 7 current waveform characteristics including number of semi-steady states (duration in state ranging from 1 to 5 s); total time in semi-steady states; number of spikes; number of steady states (duration in state longer than 5 s); number of states per time window; total time in steady states; and the presence or absence of repeating patterns. They showed a good identification accuracy in the test dataset.

The usage of Fourier series analysis to assess current harmonics has been discussed in [53], [88]–[90]. Non-linear loads that generate non-sinusoidal current while operating may be uniquely characterized by current harmonics. The latter have been employed in combination with active and reactive power characteristics to increase the identification algorithm's effectiveness in [80], [91]. It was demonstrated that devices functioning in parallel exhibit different steady-state harmonic signatures depending on their configuration. Although this method can distinguish Type I and Type IV devices, it necessitates the development of unique sets of harmonic features for all potential appliance combinations in order to accomplish load identification [37]. Fast Fourier Transform (FFT) is used in [92] to extract significant information from the harmonic components of the current waveform. This information is subsequently utilized to train models for device identification using SVMs and ANNs. However, required training with as many device combinations as possible, making it an impractical solution for a real-world household.

The high sampling rate enables fine granularity characteristics like steady-state harmonics and V-I trajectory to be captured [93]. It has been shown that converting the V-I trajectory to an image representation and supplying it to machine learning classifiers enhances classification accuracy [93]–[99]. Shape-based features obtained from V-I trajectory were utilized as input to a machine learning classifier in [100], which was the first time V-I based features were employed for device classification. In [101], many features derived from V-I waveforms, like mean line, asymmetry, and self-intersection assessment, were utilized to categorize devices. The method, on the other hand, condenses the information in the V-I-trajectory into a subset of features obtained solely from deep engineering knowledge. In [98], the V-I trajectory is converted into weighted pixelated V-I images and classification was performed using a CNN model. The authors of [94], [102] showed that decomposing the current into active and non-active components employing the Fryze power theory may increase the uniqueness of the V-I binary image and, thus, identification accuracy. The reported studies, on the other hand, utilize single-label training, which presumes that only one device is active at any given time. This

technique ignores the reality that numerous appliances may be active at the same time, as well as the fact that certain device usage may be mutually dependent.

2.5.2. Transient signatures

The temporary occurrence preceding steady state, which is the result of an abrupt shift in the circuit, is referred to as transient state of a device. It refers to the time between the device's shutdown state (OFF state) and its steady-state activity [35]. A device's turning ON (or OFF) relates to a transition between two steady-states. Electrical signals undergo changes throughout this transition, which might define a device. Following the changes in the state of the load, the transient may be either ON transient or OFF transient. Most electrical devices have a distinct transient, making them useful for load detection. The transient signature of an appliance is made up of the different characteristics extracted from the signals analysis throughout this transition. It is primarily used to characterize and analyze the transition of different load patterns before and after an electrical switching operation [58]. It includes a wide range of features, including the power change amount, the harmonic change amount, the impedance change amount, the power peak stability ratio, and so on. The authors of [75] suggested that the shapes of transient occurrences may be employed as a characteristic for detecting devices. Other work presented by the authors of [103] has shown that the energy computed during a transient " switch-ON " event may be used to distinguish devices. In [104], significant information from the waveform of turn ON and turn OFF transients are extracted using the wavelet transform and short-time Fourier transform in the time-frequency domain. The analysis indicated that the signatures used had an impact on load detection and computational cost. A multi-scale transient event detection algorithm was suggested in [105]. Rather than looking for whole transient patterns, the algorithm looks for temporal patterns of segments in the signal that show considerable fluctuation. A change-in-mean detector is used by the algorithm to identify such segments. Transient changes are then identified using a pattern matching technique and groups of segments previously computed as features for certain events.

The dynamic time warping (DTW) distance-based template matching approaches are most often used in load transient detection models [106]. The DTW distance can be considered as a shallow signature. The authors of [107] utilized Dynamic Time Warping (DTW) in their load transient detection. The dynamic time warping distance of the time series between the device to be detected and the model devices was computed, the model with the smallest distance was considered as the target device. The authors of [108] explored three different types of DTW

models and presented a multivariate fusion approach. Transient signatures were generated using both active and reactive power. The dynamic time warping (DTW) model is introduced to quantify the similarity between the raw transient power waveform sample of variable length and the template time series for load transient detection. They used a nearest neighbor transient detection approach to classify the device that generates the transient power waveform sample. They showed that the suggested load transient detection technique enhances the accuracy of the model. In fact, the performance of DTW-based models is greatly influenced by the parameters entailed and the number of samples in the pattern library, although using the multivariate fusion technique or augmenting the number of samples might result in a significantly higher computational load during the application phase [106]. In order to maximize the efficiency of detection models, it is worthwhile to examine more relevant features.

The authors of [109] suggested an approach for recording the transient responses of attached electrical devices using a pulsed voltage signal. The detection of transients in aggregate electrical load is enhanced by the development of a dictionary of transients. The quality of the generated features is tested in a classical NILM task with the goal of detecting the state of a specific device based on feature analysis by three separate algorithms. The suggested approach's effectiveness was demonstrated by the experimental results. The authors of [110] used a high sampling frequency to collect the current of appliances in transient states. The currents gathered when the device was turned ON were used to develop the transient state model. Wavelet transform coefficients (WTCs) were utilized to determine the energy spectra. The energy distribution in the sub bands was then used to determine which devices were active. The authors of [111] suggested to use a transient signal to detect electric noise on the power line induced by the switching state of device in a socket. They considered that the noise feature of certain devices is dependent on both the load and the interconnecting power line's transmission behavior. However, the device signature may be affected by the electrical wiring in the home, so a device plugged into a distinct socket may not be identified correctly. An approach that relies on a characterizing response related with a transient energy signature is presented in [112]. In their experiment, ANNs are combined with turn-on transient energy analysis in order to enhance the detection performance and computing speed. A coreless Hall CT is used to identify non sinusoidal waves. The NILM performance is increased by minimizing the distortion problem in current readings caused by the hysteresis of typical current transformer (CT) iron cores. The results of the experiments show that including the turn-on transient energy technique into NILM improves the detection performance and computing

speed significantly. The authors of [113] utilized power spikes during the device's transition phase as a feature to distinguish appliances. Although these techniques have been shown to be effective for load detection, the requirements for processing this type of feature are higher than for step changes, particularly in terms of the required sampling rate. The transients may be more challenging to catch but offer valuable information that can be used to improve steady-state signatures [38].

One of the key benefits of the transient signature is that it is tied to the internal operation of each device, so it could identify two devices with similar descriptions in the steady state. However, since transient pattern lasts only a few seconds, thus, necessitating high sampling frequencies. Furthermore, the costly process of transient analysis may make this type of signature a worthless solution for NILM [70], [111].

2.5.3. Non-Traditional signatures

Non-traditional signatures are being employed in disaggregation methods, in addition to transient state and steady state features [37]. The non-traditional characteristics such as time of day, temperature, start time, light sensing, end time and peak time are utilized to enhance traditional signatures [35]. They could be used to address some of the shortcomings in detection that traditional signatures cannot address. Moreover, the mix of transient and steady state signatures can lead to the creation of non-traditional characteristics resulting in a single robust device signature. The authors of [114] suggested to use a combination of two simple units, rectangles, and triangles, to describe the power consumption of household device while ignoring tiny changes and flaws. In their approach, peak value, peak time, start time and end time are used to define the triangle unit, whereas peak value, steady power, peak time, steady time and start time are used to describe the rectangle unit. They employed the mean-shift clustering technique to quantify the suggested units that constitute the operating style of different device types. The latter are classified by combining their characteristic features with their operating styles during the load recognition process. They demonstrated that this technique does not necessitate any training or supervision. Furthermore, the authors claimed that using this technique will help to address the problem of overlapping device features. The transient response time was explored using DWT and STFT in [115]. In their approach, the load operation and power demand are detected by analyzing the transient energy and transient response time characteristics [116]. They showed that transient response time and transient energy are more robust than steady-state characteristics in terms of improving detection

performance and lowering computing requirements in NILM systems. Moreover, this signature may be used to identify loads with identical active and reactive powers. The authors of [117], [118] reported two approaches to address the disaggregation challenge through pattern recognition and optimization. The approaches are based on the utilization of various characteristics like the active and reactive power variations, current waveform, power waveforms, eigenvalues, instantaneous admittance, harmonics, and the transient switching waveform. The correlation between the use of several devices and their usage frequencies have been explored in [117]–[119]. The flexibility of a device using the extrapolated energy usage as well as the consumption behavior of the homeowner was examined in [116]. The controllability and estimated power in the specified time slot were used to compute the flexibility. They used a case study to illustrate the effectiveness of the suggested flexibility analysis. The authors of [120] have suggested the usage of occupancy information which may be either inferred or measured in different manners to reduce the complexity of NILM algorithms.

2.6. Event-based approaches

Energy disaggregation techniques based on events are generally designed in the same manner as Hart's original NALM [40]. It entails determining an individual device's consumption by analyzing the traces of each state transition using event detection and classification. The NILM framework based on event detection typically involves 4 processes as depicted in figure 2-2 [30], [57]:

- Data acquisition: collection of electrical signals (current, voltage, power data, etc.)
- Event detection: it is the process of detecting consumption variations that are expected to occur because of devices switching their state. Any change in the state of a device over time is referred to as an event. It denotes changes in power signal that may be recognized in electrical signals using thresholds.
- Feature extraction: it is the extraction of significant events and/or features to generate a power event signature that will be utilized to identify the loads responsible for each occurrence. The load features provided by devices may be utilized to differentiate them from each other.
- Device classification or load identification: machine learning algorithms are employed to classify which devices are active at a given time and/or their states, based on the previously detected features.

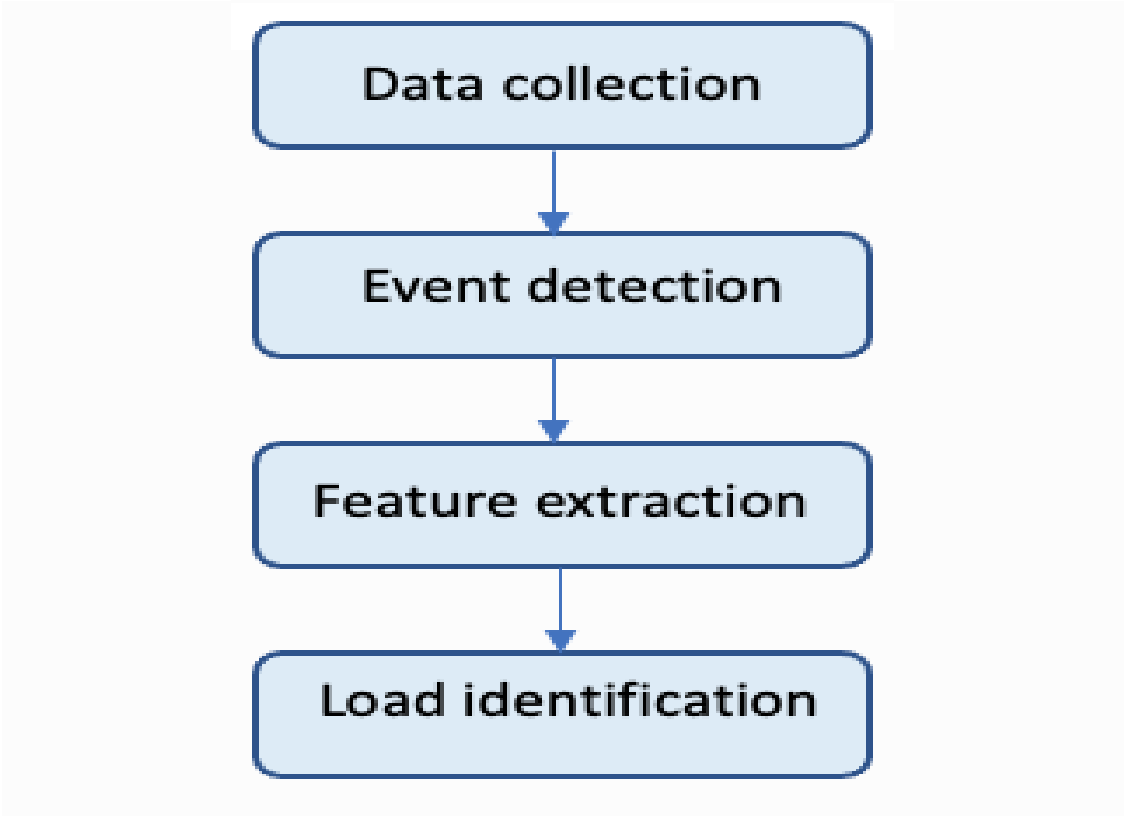


Figure 2-2: Event based NILM framework.

2.6.1. Event detection

Event-based methods rely on the concept of device signatures and the original NILM premise that each change in total power occurs as a result of a change in device operating state [40], [121]. A signal switching from one steady state to another is referred to as an event in NILM. The process of detecting changes in the aggregate data that are thought to occur as a result of devices changing their mode of operation is known as event detection [121]. In fact, the latter is usually associated with high frequencies. This is a necessary condition when processing the signal to obtain an appropriate performance [30]. It's important to mention that most earlier event detectors used current signals rather than voltage signals because events are more easily identifiable in current signals [30]. According to prior research [30], [121], [122] as well as the current progress of research [58], event detectors typically employ four main approaches: expert heuristics, probabilistic models, matching filters, and hybrid models.

2.6.1.1. Expert heuristics models

The main focus of the expert heuristic is the establishment of a set of rules for each device. Some parameters like power variation and total power demand, should normally be initialized [30]. Expert heuristic algorithms are perhaps the simplest, and they work on the concept of

scanning time series data seeking variations that are higher than a specific threshold. The original approach employs a sliding window to detect variations in the power signal's root mean square (RMS) [121]. It was designed to detect events at low sampling frequencies (less than 1 Hz). Its configuration consists of 3 parameters, including the power threshold, the minimum time elapsing between occurrences and the number of seconds preceding the evaluated second. However, it was enhanced with three extra parameters in order to support datasets with higher sampling frequencies (> 1 Hz) including event edge and pre- and post-event window lengths. The event edge parameter is utilized to allow for the comparison of the results obtained to actual data. The pre- and post-event window lengths are utilized to determine the number of samples to be averaged in order to identify the difference in amplitude between distinct points in time. The procedure can be illustrated as follows [121]: In the first stage, the proportion of power change is determined for each power sample by subtracting the mean power before and after that sample. Following that, the power variations exceeding a predetermined threshold are highlighted as potential power events. Lastly, only the highlighted power events separated by at least the minimum elapsed time between events are validated as events, while the rest are rejected.

The authors of [123] suggested an event detection technique that identify changes in the envelope of the current signal. Its primary benefit is the high performance with which the start times of events can be identified. An envelope extraction approach was presented by the author of [124]. The envelope of the current was computed using the root mean square function. To determine the change in the envelope, a derivation was used. Several peaks were created from the changes. Lastly, the events were defined as peaks with amplitudes higher than the threshold. The authors of [125] filtered the power signal first to remove noise and minimize the probability of false positives. The power events are then detected in a second phase by calculating the absolute differences between two successive samples and picking the indices where this difference is higher than a pre-determined threshold.

2.6.1.2. Probabilistic models

The change in the operating state of the device involving the change in the aggregated power consumption defines the events. Comparing the power change deviations of adjacent samples is the most direct technique to identify the event [58]. The latter is triggered when the power change exceeds a predetermined threshold. However, in the real-world application, the power time series is impacted by a multitude of factors and fluctuates randomly. By using a single

threshold, it becomes difficult to separate events from noise [58]. In this context, probabilistic models have been proposed. The GLR is a common probabilistic model that was first designed to address the problem of abrupt changes [30], [122], [126]. The GLR specifies the detection window, the post-event and pre-event averaging window. Two Gaussian distributions are defined using the pre-event and post-event averaging windows. The probability density functions of the two Gaussian distributions are used to assess the likelihood of an event occurring within the detection window [58].

Probabilistic models enable us to obtain a probability that may be utilized to make a choice about the likelihood of certain events occurring. They need to go through a training process to adjust some factors and learn specific statistical models for devices and settings for this aim [30]. A Generalized Likelihood Ratio (GLR) approach is presented by the authors of [122], [127]. The authors showed that in order to optimize performance, it necessitates a training procedure that tunes five parameters whose values fluctuate based on the statistic. The author of [128] suggested an enhanced version of the GLR model that included a voting method to boost performance. The GOF approach was also used to analyze event detection models [129]. The difference in distributions before and after the event is used to detect the occurrence. The authors showed that the GOF-based event detection model outperformed the expert heuristics method in terms of false-positive performance.

2.6.1.3. Matching filters

Matching filters detect the presence of the mask in an unknown signal by correlating a known signal (commonly referred to as the mask) with an unknown signal [122]. The masks may be template power signals that contain the start-up or turn-off transients of devices, while the unknown signal is the overall power usage. The signal waveforms are extracted and correlated with known patterns to classify them. Although no prior training or knowledge of the devices or settings is required, this method usually requires high sampling frequencies [30]. The following are the main phases of template matching-based approaches [58]: (i) computing the spectral envelope of the time series, the actual current and voltage with high frequency are transformed to signals with evident features in transients; (ii) during the training process, the frequent occurrences of the turn on and turn off procedure are extracted and standardized by building up the template library; (ii) The template matching procedure is triggered by a change-of-mean procedure, and the distance between the total power signal and the template power signals is computed, an event is identified if the distance is less than the threshold. Distance

computation is proven to be an important stage in the template matching based event detection approach. The most used distance is the Euclidean distance, which was employed in the template matching procedure in [130]. However, the phase of the signal affects the Euclidean distance, therefore two comparable signals with a phase difference may have a significant Euclidean distance [58].

The authors of [131] suggested an event detector that uses two transversal filters that try to match initial transient segments (obtained via training) to the aggregated signal. The first filter searches for transient shapes in the aggregate signal, while the second filter verifies that the matches represent the real transients rather than random noise. The instantaneous current captured at 20 kHz was given a Hilbert transform by the authors of [132]. Following that, the modified signal is subjected to a mix of averaging and derivation filters in order to reflect only the transitions of interest (power events). The authors of [133] proposed experiments using envelope extraction, in which spectral envelopes from many harmonics are generated and a Kalman Filter is employed to detect occurrences.

2.6.1.4. Hybrid models

The authors of [134] proposed a kernel Fisher discriminant analysis (KFDA)-based unsupervised NILM event detector that generates proper start and end times for so-called active sections. The latter are the extensions of traditional NILM events that were designed to consider complex situations (such as peaks, brief pulses, and fluctuating load). The suggested approach's effectiveness is benchmarked using the BLUED dataset. The authors of [135] presented an adaptive event detection system based on GPS, with an iterative clustering scheme for determining the threshold. The primary objective is to integrate adaptive thresholding, pattern matching, and signal clustering to the emerging subject of GPS. They established the method's performance boundaries and demonstrated its practical utility. They illustrated the efficiency of the suggested technique for standard smart meter using two public datasets REFIT and REDD with active power readings down sampled respectively to 8-s and 1 min resolution. In [136] a variation of the goodness-of-fit ($\chi 2$ GOF) -based on cepstrum smoothing is presented. In their approach, the authors used a voting mechanism to enhance the typical GOF algorithm, a median filter to eliminate impulsive noise from the power time series and a surrogate-based optimization approach to tune the parameters of the hybrid model. The suggested hybrid approach appears to be more effective than the classical GOF model, according to the experimental results. The authors of [137] suggested a DBSCAN based event detection method

that detect two adjacent steady states and the timestamp. They demonstrated good detection accuracy using publicly available dataset BLUED and PLAID. The authors of [138] explored a hybrid framework in which the model's parameters might be defined automatically in different conditions. The suggested unsupervised approach uses automated clustering to find recurrent motifs, which are depictions of the devices' transient power draw characteristics in a particular environment, and proximity-based motif matching to detect occurrences. The approach was tested using the EMBED dataset, which is a public dataset obtained from three houses with various device types. The results demonstrated that the suggested event detection technique performs better than the classical event detection in identifying the operating states of various load types in a variety of scenarios.

2.6.2. Features extraction

The process of extracting the optimum characteristics so that the power event footprints are robust and discriminative enough between devices is known as feature extraction. It is quite crucial in establishing the energy consumption footprint of each appliance and will further enable the identification of the particular signatures of each appliance, which can therefore be used for appliance detection [139]. The features can be either data-driven features or engineered features [140].

The data-driven features are extracted directly from the data using methods like unsupervised learning. The authors of [141] pred a generic data-driven technique for NILM using low frequency sampling rate. They partitioned a set of observations into clusters using the k-means clustering approach, with each object belonging to the cluster with the closest mean value. The authors of [142] suggested a data-driven procedure using an unsupervised feature representation based on extreme machine learning technique for efficient energy disaggregation. In [95], [98], the usage of VI binary image for device identification is proposed. The (V-I) trajectories are converted into binary images after being normalized in amplitude.

Engineered features on the other hand are retrieved using domain knowledge derived from electrical power and device properties. Examples are characteristics related to harmonic components [65], [66], amount of power change [136], transient forms [105], [115], [143]. Typical low-sampling applications use characteristics derived from power time series variables [30], like current, voltage, active power reactive power, apparent power, power factor, power phase angle, and so on. Furthermore, several additional features were captured from the

electrical signal in the frequency domain, including electromagnetic interference [70], electrical noise using FFT [70], [144], wavelet transforms [104], [110] and so on.

2.6.3. Load identification

The purpose is to identify the devices involved in the changes in the measured aggregate signal operating at any given moment using features extracted from the gathered data, including the device's electrical signature [30]. Within the energy disaggregation contexts, studies have primarily concentrated on supervised approaches rather than unsupervised approaches for this identification [37], [48]. The supervised learning approaches adapted for NILM are to date either pattern recognition based or optimization-based approaches. The latter are used in the supervised approaches to find a set of devices whose corresponding load demand matches the observed load in the aggregated signal as closely as possible. These include genetic algorithm [145], segmented integer quadratic constraint programming [146], hybrid programming [147] and so on. However, these approaches necessitate the availability of datasets including all potential associations of power demands for the different devices. This is a disadvantage because the existence of unknown devices in the aggregated signal would make the recognition based on the combination of known appliances challenging [40], [117]. Furthermore, their heavy computational burden limits their practical application [30]. Besides, supervised techniques are exploring pattern recognition approaches capable of recognizing the extracted features (device signatures) in the aggregated data. Offline training is typically used in these techniques to create a dataset of information that is employed to develop classification algorithms. The device database encompasses a series of device features that are utilized to construct the identification algorithm's design and characteristics. Examples of some commonly used techniques include the k nearest neighbors (KNN) [148], [149], the artificial neural network (ANN) [110], [115], decision trees [144], [150], support vector machines [79], [88], [115], [151], [152], convolutional neural network [50], [93], [98], [153]–[156], naïve bayes classifiers (NB) [149], [157], [158], and deep learning [60], [98], [155], [159]–[163]. Hart [40] introduced a clustering-based technique wherein devices generate their own distinct clusters in the P-Q plane. A feature space is mapped using the electrical signal's steady-state variations. The new feature vector is then clustered using the distance metric to see whether it belongs to one of the previously identified clusters. This approach has been commonly employed in NILM owing to its simplicity. However, the technique's failure to identify devices with overlapping P-Q features and susceptibility to power drifts are some of the significant disadvantages of the approach. The authors of [164] proposed a linear-chain conditional random fields, which can

handle multi-state loads by considering the influence of prior states on the current state. The authors of [165] compared the performance of 3 classifiers including multilayer perceptron, radial basis function NN and SVM with distinct kernels, using the odd harmonics of the current waveform. They found that all models perform well in classification and correctly identify the relevant appliances, which demonstrates the applicability of the suggested approach. In [166], [167], the effectiveness of ensemble-based techniques, in which distinct algorithms are associated to enhance the overall accuracy of the classifier, was examined. The authors of [158] suggested a two-stage technique for NILM. In the first stage, a classifier seeks to classify the device by type (purely inductive, capacitive, or resistive) thus in the second stage, a second classifier is trained with only those characteristics that are considered meaningful for identifying devices of that type. The authors of [168] evaluated the performance of several classifiers, including feedforward ANNs, hybrid neural networks, SVMs, and adaptive boost (AdaBoost) models. The method is based on the use of Wave-Shape (WS) characteristics based on V-I trajectories. The models performed well for two households in the REDD dataset [169]. However, the result deteriorated drastically when gaussian noise was added. They found that the waveform features proved to be effective for load classification. The authors of [170] employed the Bayesian approach to identify the most probable device states using state change information. A naïve Bayesian classifier was trained for each device, and a set of learned classifiers was employed to distinguish device-specific states from aggregate data readings. However, the authors assumed that the state of the devices was independent from each other, which, according to [37], is not the case because the operation of consumer devices is frequently correlated in the household (e.g., the use of TV and DVD player). A heuristic technique that uses the histogram thinning method for clustering of P and Q events was compared to the Bayesian technique. They showed that the Bayesian method outperforms the heuristic technique, especially when the energy behavior of the devices is stable.

In fact, pattern recognition methods are generally privileged by researchers for the task of load recognition [37]. Meanwhile, the necessity of having labeled data which is unfortunately not always available and training data are some of the main constraints to the large-scale adoption of supervised approaches. The authors of [171] suggested combining a supervised learning approach on accessible labeled datasets with an unsupervised training method on aggregated unlabeled data. The authors of [172] have also explored semi-supervised methods that seek to employ both labeled and unlabeled data for training NILM algorithms. In these approaches, semi-supervised learning approaches seek to exploit the potential of unlabeled data

by employing small sets of labeled samples to deduce labels from unlabeled samples and subsequently utilize them as training sets.

Unsupervised approaches, on the other hand, do not necessitate any prior training. This is a significant benefit since it entails less effort from the consumer and reduces the intrusiveness of data collections [30]. For this reason, unsupervised approaches have attracted increasing attention [37]. They often don't use events and try to disaggregate the signal directly from the aggregated signal without using event detection.

2.7. Non-event based approaches

Non-event-based techniques rely primarily on statistical and probabilistic models to breakdown aggregated power signals without using any type of event detection [37], [173]. Indeed, these approaches do not necessitate prior training of the machine learning algorithms to recognize appliance specific power change in the aggregate data. Rather, these techniques mainly exploit existing knowledge about the operation of individual devices by employing various techniques like probabilistic models, HMMs, HOS, motif mining, and blind source separation.

In fact, the NILM challenge is a time-dependent problem wherein statistical models are widely targeted as a relevant solution [173]. HMM-based approaches have attracted a lot of interest in the past few years. The authors of [174] developed a hierarchical approach that uses a Bayesian consideration of HMMs to model diverse appliances. The different HMM settings are then combined to create a specific appliance consumption model. They demonstrated that general models of devices can be built utilizing data from only a small number of devices. In [175], the authors developed a sparse Viterbi technique that employs a super-state HMM to conserve load relationships and identify multi-state devices while ensuring efficient and robust accurate inference. The sparse Viterbi approach is used to efficiently generate sparse matrices with multiple superstates that are used to achieve the NILM task. They demonstrated that their model could work in real time. The authors of [176] presented a generic technique based on temporal sequence categorization methods. Its goal is to identify low-resolution device states and use those features to build device types and classify prospective devices. The applicability of FHMM for low power device monitoring was investigated in [177]. The authors concentrated on identifying both binary and multi-state device operation through the use of suitable sets of characteristics. Since the number and types of devices are known a priori, each target device is modeled as a single HMM with several states that characterize its working behavior. In the FHMM, each chain has its own Markovian dynamics. Following that, based on the observation

sequence, the most likely hidden state sequence occurring therein is deduced. They showed that feature concatenation can enhance the model's disaggregation accuracy, enabling it to accurately detect the states of appliance. In [178], several HMMs extensions were explored to model the data. The conditional factorial HMMs (CFHMMs), which enhance factorial HMM by adding features like time of day, device dependency and other sensor measurements; the Hidden Semi-Markovian Factorial Model (FHSMM) which is a variant of the FHMM that improves the fit of probability distributions of appliance state occupancy periods and the mix of CFHMM and FHSMM, which considers additional features as well as more accurate probability distributions of the occupancy times of the device states were studied. The basic concept involves representing each device as a distinct parallel HMM that contributes to the total power. The authors designed a probabilistic model of device behavior that incorporates steady state characteristics. The training of model parameters is unsupervised, and several parameters are learnt from the data. The learning procedure entails determining the parameters from the observations in order for the model to accurately characterize the data. The hidden variables, which are the statuses of the devices, are estimated using the model through these parameters. The framework is analyzed in the low frequency power load data by considering multidimensional characteristics as input. The authors demonstrated the viability of the proposed NILM technique. An approach based on the combined utilization of active and reactive powers using additive FHMM model was proposed in [78]. In the suggested method, the device model is illustrated by a bivariate HMM, where the emitted symbols are the combined active-reactive power signals. To handle the bivariate HMM models, the authors proposed an alternate formulation of the AFAMAP technique to disaggregate the data. The approach relies on modeling the aggregate power readings as a mixture of appliance operating states, and then reconstructing the time series of the state progression for each appliance in the finite state machine. They showed that the usage of reactive power improves the accuracy of the model. In [179], a NILM technique that accounts for mutual interactions between appliances and incorporates information about the interactions between appliances into the FHMM representations of the aggregated data was proposed. In fact, appliance interaction is a power quality issue that impacts an appliance's measured power consumption when other appliances are connected to the network. The Viterbi algorithm was employed to deduce the hidden states in the proposed FHMM. Additional chains were added to FHMMs to model any interactions between the appliances. They showed that the suggested technique outperformed the standard FHMM model, and that observing interactions between appliances enabled more accurate disaggregation and estimation of specific appliance energy usage. The authors of [180]

suggested an Infinitely Factorial Unbounded state HMM (IFUHMM), with potentially unbounded number of parallel HMMs and states in each HMM. The approach is based on a non-parametric Bayesian a priori using integer matrices, with rows representing temporal indexes, columns representing Markov chains, and integers representing the state of each chain and temporal instant. The approach starts by extending the existing infinite factorial binary state HMM model to include any number of states. Then they tweak the model to enable an infinite number of states and create an (Markov chain Monte Carlo) MCMC-based inference algorithm that efficiently handles the trade-off between infinite states and chains. They found that inferring rather than fixing the number of chains and states in the FHMM enhances performance of the model. The authors of [181] developed a hierarchical FHMM to circumvent the assumption of appliance independence and the requirement of one appliance at a time. The approach works on the steady-state signal by grouping the signals of correlated appliances and then generating HMM models based on the detected clusters (called "super appliances"). The super appliances are inferred using AFAMAP algorithm. A state relation table is generated during the training phase and used to map the result back to the original appliance. They showed that inference on super-appliances is way more efficient than inference on original appliances since they only had to deal with a smaller number of "appliances" that met the independence and single-time constraints. In [182], an approach that combines the utilization of FHMMs with contextual information associated to the consumer's occupancy in the household and the hourly usage of devices was examined. The authors examined how the usage of these additional contextual knowledge features could enhance the NILM performance. Experiments using the Tracebase public dataset and synthetic contextual data yielded promising results. The authors of [183] suggested an additive FHMM for NILM. They developed an estimate inference approach that takes use of the FHMM additive structure and employs an efficient convex quadratic programming relaxation. The approach relies on inferring the hidden states of an FHMM in order to address unsupervised NILM and extract individual devices usage from the aggregate signals. The researchers stuck to the original FHMM approach, wherein the output is an additive function of all hidden states. The approach has achieved good results while remaining computationally feasible. The work presented in [181], [182] shows that the consideration of inter-device dependencies, in particular, leads to an enhancement in the effectiveness of NILM. On the other hand, the complexity of HMM models increase exponentially as the number of target devices increase. Furthermore, their scalability and generalization capabilities are challenging and existing inference approaches for state estimation are also extremely sensitive to local optima, which limits their practical application.

The authors of [184] suggested a data-mining-based time-dependent load disaggregation technique. They employed clustering techniques to identify different power levels and characterize the periodic cycles of occurrence of these control levels. Each episode must meet some specific criteria to be considered part of a device. The minimum episode completion criterion, for example, is employed to detect episodes completed by a single appliance. They showed that motif mining is effective in recognizing appliances with different power levels and in characterizing the combinatorial operations of appliances. In [185], a blind source separation approach was proposed to address the problem of NILM. The authors employed steady-state active and reactive powers variations (P and Q) to generate clusters of devices, each of which is supposed to match to one device state transition. Then, the original source of each cluster is reconstructed using a matching pursuit algorithm (MP). The authors of [186] suggested a non-intrusive technique to categorize electrical devices using higher order statistics (HOS). In the proposed approach, Genetic Algorithms (GA) and Fisher's discriminant ration were used to select a finite set of the most relevant characteristics from those generated by HOS in order to lower the computational cost. The main goal of GA was to lower the data dimension while also improving the effectiveness of the classifier. They used multilayer perceptron (MLP) and decision tree (DT) classifiers to perform the final classification. The proposed classifiers used only 6 extracted features (second and fourth order cumulants) that could be suitable for a real-time application. The approach was tested using experimental signals obtained from the main electrical power source of a household and performed very well. In [187], An approach based on powerlets learning and sparse coding scheme was proposed. The power fingerprints of each appliance are automatically extracted during the learning phase by modeling the appliance as a combination of dynamic systems. Relevant dissimilarities between device power samples were developed and employed in a subset selection system to extract the usage signatures of a device corresponding to its distinct operating modes. The extracted energy signatures for all appliances were used to create a dictionary. The authors presented NILM as an optimization over a representation in the trained dictionary, with numerous new priorities such as appliance disparity, knowledge of which appliances operate together and which do not, and disaggregated solution temporal consistency. They demonstrated that the proposed approach could achieve promising performance.

2.8. Machine learning approaches to load disaggregation

Numerous techniques have been presented in the literature to tackle, on the one hand, the challenges faced by the pioneering NILM approach and, on the other hand, the problem of

NILM in general. Initially, the research focused on HMM models and their variants (as detailed in section 2.7). However, their applicability was limited due to their complexity, generalization problem and their sensitivity to local optima. Recently, scientists have resorted to machine learning and deep learning approaches. These include methods like SVM [148], [151], [188], KNN [149], DT [58], [189], Graph signal processing [190], [191], k means clustering [192], etc.

The deep learning framework has proven its ability to extract sophisticated characteristics of complex data automatically which enables it to play an essential role in energy disaggregation problems [193]. Several deep learning architectures have been proposed to tackle the NILM challenge. The research conducted in [51], [194] was among the first to employ deep learning approaches for NILM. In [194], the authors introduced a new approach for supervised NILM based on the LSTM architecture. The primary goal was to monitor the energy usage of significant loads in a house in order to achieve energy savings. The target power signal of the device or sub-circuit is extracted from the aggregated power signal using the LSTM network based on supervised learning. The authors showed the feasibility of deep learning approach for non-event based NILM applications. The results of the tests conducted in REDD's low-frequency public data set, involving only real power measurements, are promising, and have shown a new way to design NILM systems. In [51], the authors analyzed three deep learning algorithms including LSTM coupled with CNN, denoising autoencoder and an estimation algorithm that determines the start time, end time, and the average power demand of each appliance. They defined the NILM task as a denoising operation, in which the clean signal is depicted by the power demand of the target appliance, and the aggregated signal is the background "noise" created by the presence of other appliances. They disaggregate the power demand of five devices from the aggregated data of the whole house and obtain good results. They train the models with the same architecture for all devices, regardless of their type. The models were compared on the UK-DALE public dataset, and the study revealed that the denoising autoencoder model performed better than the other neural network architectures and the state-of-the-art FHMM and combinatorial optimization techniques in the Non-intrusive Load Monitoring Toolkit (NILMTK). In [195], a group of auto associative neural networks is trained to be tuned to the features of a specific device. In the proposed approach, the auto associative NNs are disposed in a competitive parallel architecture that puts them against each other. The closest identification is then used to detect the specific appliance when a new input vector is introduced. The system employs the transient power signal acquired from the device's

on/off events to recognize the specific types of device. The results of tests on three public datasets demonstrated the method's applicability for NILM systems. In [196], the authors presented a sequence-to-point technique based on CNNs. The model used an aggregate active power window as input and a single point of the target device as output. It performed better than the sequence-to-sequence state of the art method on REDD and UKDALE public datasets. A CNN, a LSTM and random forest (RF) classifier are used to form a deep learning model for NILM in [197]. A multi-feature and high-dimensional approach are used to extract characteristics in the data preprocessing step, and a pre-training method is employed to generate correlations in electricity usage habits. The authors discussed the problem of label correlations and evaluated their approach on REDD and Pecan Street datasets. They showed that the proposed method may successfully increase device detection performance. The authors of [198] presented a bidirectional LSTM model. They developed a multi-feature input by combining multiple electrical features. The electrical parameters that had the significant impact on each target device's power usage were selected using the mutual information technique. Then, to remove unnecessary predicted sequences, a second post-processing classification algorithm was suggested. Low frequency data (UKDALE and ECO datasets) were used to evaluate the model. They demonstrated improved performance with the post-processing algorithm. The authors of [199] presented a capsule network based NILM technique. CNN was used to extract prospective characteristics from a collection of non-overlapping segments of energy readings data. The capsule architecture is intended to estimate specific segment class probabilities. Then, based on the estimated class probabilities of the segments, a decision-making process is suggested to establish the final identification. They compared and validated the proposed solution’s sustainability against current literature. A CNN technique was suggested in [200]. They used a features space that had been enhanced by the incorporation of a temporal pooling module to categorize the states of the devices. In their approach, a time interval of the aggregated active power consumption is used as input. The method uses a multi-label classification to predict the device's consumption as a constant average value throughout the device's activation. The authors tested their model using the UKDALE low frequency dataset and demonstrated good generalization properties. In [60], the authors focused on the disaggregation of multistate appliances. A practical approach based on deep CNNs using a data augmentation technique is proposed. A post processing method was used to categorize the activations predicted by the convolutional model. The authors showed that the usage of the post processing algorithm significantly increases the disaggregation results. In [201], a technique based on the pinball quantile loss function for guiding a deep NN for NILM was proposed.

They used the pinball and mean square error loss functions to compare the suggested network to three distinct deep NN. They demonstrated that the method using the pinball loss function consistently outperforms the method employing the MSE loss function. The authors of [202] presented an event detection method based on a transient signal. They used active power signal to extract the suggested transient signal in low frequency sampling rate. The events were classified using DT and LSTM models. They demonstrated that the use of transient signal allows for good performance. In [203], the authors described a NILM technique that generates synthetic data for training deep CNN models via data augmentation. The suggested data augmentation technique combines ON and OFF-durations of a target device from (REDD, UKDALE and ECO) datasets in low frequency sampling rate, resulting in unified sub-meter profiles and complete synthetic aggregate. They demonstrated that training the algorithm on generated synthetic data improves its generalizability. The authors of [204] suggested an Active Deep Learning-based load detection approach. The high-dimensional device characteristics are collected from the original current signals using the discrete wavelet transform (DWT). The features were then learned and high-value samples suitable for labeling were selected using a pool-based active deep learning model. Three sampling approaches were employed to test the suggested approach using a mixed dataset made up of 3 public datasets. The results demonstrated that the suggested approach might greatly lower the cost of labeling huge datasets. A deep NN technique for NILM that presents a tailored attention mechanism framework was presented in [205]. To address the NILM issue, the proposed model includes an estimation subnetwork combined to a classification subnetwork. In the regression subnetwork, the authors employed convolutional and recurrent layers to improve feature extraction and enable the creation of better appliance models. They showed that the suggested model increases accuracy and model generalization capability on the public datasets REDD and UK-DALE. In [206], the authors developed a technique for device detection and classification of unidentified devices in NILM. In their approach, devices are defined by their V-I trajectory and mapped to a previously learned feature space built by a Siamese NN, resulting in samples of the same device forming tight clusters. Device samples are then assigned to clusters or labeled as "unidentified" using the clustering performed through DBSCAN. They showed using the PLAID and WHITED datasets that an F1-Macro of 0.85 and 0.90 can be achieved for categorizing unidentified devices. In [207], a NILM technique based on online learning has been designed to infer a range of devices from a transferred model using low sampling data. The proposed framework, in particular, combines a transfer learning approach with deep NN. First, they extract low-level spatial and temporal features from the gray-scale images produced

by the power readings using the LSTM. Then, the classification of the device type and the transfer between devices is performed using a probabilistic neural network. The algorithm's effectiveness has been validated through the use of real-world smart outlet hardware. A NILM approach based on graph theory for matching load events is proposed in [208]. The method is based on the enhanced Kuhn-Munkras (KM) algorithm. The ON-OFF state of the load is detected using an adaptive fitting approach employing a time window. A kernel density is constructed using a number of independent characteristics of device turn-on, such as active and reactive power features. The load events are thus classified according to their characteristics using the load signature distribution. The enhanced KM technique for load matching is then used to address the matrix generated by the load event's matching probability. The results of the experiments utilizing laboratory and REDD datasets show that the suggested technique may be effective for load event matching and enhance load detection performance.

Even though existing Deep learning-based disaggregation techniques have demonstrated good scalability and learning of characteristic device signatures, the practical feasibility of these techniques remains an open problem [161], [198]. Indeed, deep NNs tend to estimate irrelevant power signals that are not relevant to the target appliance. The work reported in [60], [198] tackled the problem by proposing a post-processing model, which included employing a separate algorithm to categorize the estimated device activations. The approach consists in determining if the estimated sequence corresponds to the actual activation of a target appliance. However, these techniques these approaches require a significant amount of training data to attain satisfactory performance. This poses a significant challenge due to the lack of high-quality datasets, in terms of both timing and the quality of labels. Furthermore, these methods greatly benefit from a large number of trainable parameters, but this requires a lot of processing power that is neither low-cost nor easily accessible in most situations.

2.9. Evaluation metrics

One of the primary challenges in NILM research nowadays is the lack of a formal approach for evaluating the effectiveness of the wide range of solutions presented in the literature. The relevant literature has a wide range of metrics that span a broad set of evaluation methodologies. Despite the fact that most of the NILM techniques report different performance metrics, it is not possible to make meaningful comparison since there is no agreement on the metric to be used to assess the effectiveness of an NIM algorithm. Moreover, owing to the ambiguity of the

evaluation’s metrics, which are typically data and experiment context dependent, a direct comparison of NILM methods proposed in the literature is not feasible [37], [48].

Many evaluations metrics have been proposed since the beginning of the NILM research. For instance, Hart employed both the events correctly detected and the percentage of time that a given device was active as an accuracy metrics in the seminal work. A few years later, the authors of [75] suggested a comparison between the actual and predicted power of each appliance. In [113], the percentage of energy consumption properly assigned to each appliance under consideration with error related to the duration of the load and the time intervals of the "on" working modes were used to evaluate the proposed NILM algorithm. The authors of [117] suggested the use of metrics in terms of event detection accuracy, classification accuracy and overall accuracy. They also discussed the possibilities of Type I (false detection) and Type II (missed detection) errors, as well as equations to represent the amount of true, incorrect, and missing events, which can be global or device specific. The authors of [122] suggested using the total power change and the average power change in order to readily understand the interaction between detection errors and actual power consumption. In [209], the authors suggested a metrics that reflects the total error in allocated energy (TEAE), standardized by the actual energy usage in each time slice averaged across all devices. A similar metric was introduced in [183], which this time calculates the individual device error (IDTEAE) instead of the average of all devices, thus reducing the risk of pointing out huge errors in certain time slots owing to poor performance of some devices. Receiver operating curves (ROCs) are commonly used by pattern recognition researchers to assess the accuracy of different models. The ROC curve illustrates the trade-off between specificity and sensitivity. According to [48], ROC curves can potentially be used as a benchmark evaluation approach to assess the accuracy of NILM approaches.

Although the question of NILM performance indicators is under ongoing discussion, most researchers tend to agree on two main categories of evaluation’s metrics, namely energy estimation and event detection [41], [161], [210], [211]. The choice of the most suitable category depends strongly on the application for which the NILM is intended. In event-based approaches, each event is evaluated and compared to the ground truth to ensure that it is accurately categorized. Many researchers employ accuracy to evaluate an algorithm's effectiveness in identifying the ON/OFF states of specific appliances.

$$acccuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (2)$$

The accuracy evaluates the number of correctly attributed matches in comparison with the total of all possible matches (True positives (*TP*) which are switching events that have been accurately identified and classified, True negatives (*TN*) which denotes the case where a device did not switch and the algorithm also asserts that there was no switching, False positives (*FP*) which represents the case where an appliance has not switched, but the algorithm wrongly detects a switch and False negatives (*FN*) refers to the case wherein the algorithm fails to identify a switch, despite the fact that the appliance has switched). However, the high accuracy values are not necessarily related to the correct detection of positive samples. In the case of unbalanced data, the accuracy may present a near flawless classification as a bad one. For instance, if all the samples are only assigned to the negative category in a binary classification context where the number of negative samples is much higher than the number of positive samples, the accuracy value would be close to 1 despite the fact that no positive samples have been detected. Since there are devices that operate very occasionally or are almost always on, using accuracy as a metric can lead to misleading results [178], [212]. In order to address this challenge, the authors of [213] suggested *F-Measure* to assess the estimation of these devices with more reliability. The *F-measure*, also known as F_β-score, is a trade-off between recall and precision. It is given as the harmonic mean of Recall and Precision.

$$F_\beta = (1+\beta^2) * \frac{Precision * Recall}{(\beta^2 * Precision) + Recall} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

Precision is the proportion of true positives (*TP*) in the universe of all positive samples in the data set, with emphasis on false positives (*FP*), whereas Recall is the proportion of true positives (*TP*) in the universe of all positive samples in the data set, with emphasis on false negatives (*FN*). β represents a weight coefficient that allows β times more significance to be attached to *Recall* than to Precision. For instance, if β =2 (*F2-score*), *Recall* is twice as significant as *Precision*, whereas if β = 0.5 ($F_{0.5}$*-score*), Precision is twice as significant as Recall. Usually, β =1 (*F1-score*) is used, and thus *Recall* and *Precision* have the same weight.

In fact, the *F-score* is not a rigorous fit for multi-state devices since it only considers the ON/OFF states of devices. The authors of [178] claimed that the *F-score* consider only binary results of the classifier and that power signals are not binary. They argued that not only the accuracy of the classification of the device status will need to be measured, but also the accuracy of the intended usage of the device. They introduced the modified *F-score* (*M F-score*), which they claimed is more suitable for assessing the accuracy of the NILM algorithm. Consequently, *Precision* and *Recall* are reformulated by accounting the number of states of each device by computing the accurate true-positives (*ATP*) and inaccurate true-positives (*ITP*). Let y be the estimated values, and y_0 the ground truth values.

- True negative (*TN*), if $y = 0$ and $y_0 = 0$
- False negative (*FN*), if $y = 0$ and $y_0 > 0$
- False positive (*FP*), if $y > 0$ and $y_0 = 0$
- Accurate true positive (*ATP*), if $y > 0$, $y_0 > 0$ and $\frac{|y-y_0|}{y_0} \leq \partial$
- Inaccurate true positive (*ITP*), if $y > 0$, $y_0 > 0$ and $\frac{|y-y_0|}{y_0} > \partial$

Where ∂ denotes the threshold.
Precision and *Recall* are redefined as follows:

$$Precision = \frac{ATP}{ATP+ITP+FP} \tag{6}$$

$$Recall = \frac{ATP}{ATP+ITP+FN} \tag{7}$$

M F-score remains the harmonic mean of the redefined *Recall* and redefined *Precision*.

The performance in terms of consumption estimation should be defined to reflect the ability of the NILM approach to predict the amount of energy used compared to the actual power consumption. This is essential since disaggregating systems should inform consumers of how much electricity each device contributes to the system. Moreover, consumers should know how much they could have saved if certain devices had not been used during the peak period, especially under time-of-use billing. The Mean Absolute Error (*MAE*) which evaluates the reliability of the power consumption estimate of the appliance has been widely used [41], [161].

$$MAE^{(i)} = \frac{1}{T}\sum_{t=1}^{T}\left|\hat{y}_t^{(i)} - y_t^{(i)}\right| \tag{8}$$

Where *T* is the number of values that represent the load data, $y_t^{(i)}$ are the actual values, $\hat{y}_t^{(i)}$ are the estimated values of device i at time t.

Signal Aggregate Energy (*SAE*) that evaluates the relative error in predicting the quantity of energy consumed during the entire evaluation period [161], [198].

$$SAE^{(i)} = \frac{|\hat{E}^{(i)} - E^{(i)}|}{E^{(i)}} \tag{9}$$

Where $\hat{E}^{(i)}$ is the total estimated energy, and $E^{(i)}$ is total ground truth energy for device i.

The root mean square error (*RMSE*) is commonly employed in the NILM community by considering the difference between the actual energy consumption $y_t^{(i)}$ and predicted energy consumption $\hat{y}_t^{(i)}$ for each time interval t.

$$RMSE^{(i)} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(\hat{y}_t^{(i)} - y_t^{(i)}\right)^2} \tag{10}$$

where T represents the total number of load data values. Since the *RMSE* value is not standardized, it is difficult to benchmark the disaggregation results of one device to another [212]. To deal with this issue, other studies [44], [183] have used the normalized disaggregation error, resulting in a normalized measure of the difference between the predicted and the ground truth energy consumption of the i-th device.

$$Normalized\ Disaggregation\ Error = \sqrt{\frac{\sum_{t,i}\left\|\hat{y}_t^{(i)} - y_t^{(i)}\right\|^2}{\sum_{t,i}\left\|y_t^{(i)}\right\|^2}} \tag{11}$$

The authors of [209] suggested the total energy correctly assigned (E_{acc}) to report the estimation performance. It is one of the most commonly used metrics. The estimation accuracy E_{acc}, in particular, assesses the quality of energy disaggregation and relates the difference between the ground truth and estimated consumption to the performance of disaggregation, making it an excellent choice for evaluating different NILM approaches. It is given as follows:

$$E_{acc} = 1 - \frac{\sum_{t=1}^{T}\sum_{i=1}^{M}\left|\hat{y}_t^{(i)} - y_t^{(i)}\right|}{2\sum_{t=1}^{T}\sum_{i=1}^{M}\left|y_t^{(i)}\right|} \tag{12}$$

where $\hat{y}_t^{(i)}$ represents the algorithm's estimate for the i-th device at the t-th time step, $y_t^{(i)}$ is the actual power values for the i-th device at the t-th time step, T is the time sequence length, and

M denotes the number of devices. The factor 2 in the denominator is due to the fact that the absolute value will "double count" the errors, since $\sum_{i=1}^{M} y_t^{(i)} = \sum_{i=1}^{M} \hat{y}_t^{(i)}$.

The estimation accuracy can be computed for each device using this measurement by removing the summations over *M* appliances as shown in the following equation (13).

$$E_{acc}^{(i)} = 1 - \frac{\sum_{t=1}^{T} \left| \hat{y}_t^{(i)} - y_t^{(i)} \right|}{2 \sum_{t=1}^{T} \left| y_t^{(i)} \right|} \tag{13}$$

The authors of [214] introduced the estimated energy fraction index (*EEFI*) which yields the fraction of energy attributed to the i-th device. It is given as follows:

$$EEFI = \frac{\sum_{t=1}^{T} \hat{y}_t^{(i)}}{\sum_{t=1}^{T} \sum_{i=1}^{M} \hat{y}_t^{(i)}} \tag{14}$$

The *EEFI* should be compared with the actual energy fraction index (*AEFI*) which yields the actual fraction of energy consumed by the i-th device. It is defined as follows:

$$AEFI = \frac{\sum_{t=1}^{T} y_t^{(i)}}{\sum_{t=1}^{T} \sum_{i=1}^{M} y_t^{(i)}} \tag{15}$$

The *EEFI* provides consumers with information on how much energy each device consumes, allowing them to get customized suggestions for lowering their energy usage.

The relative square error (*RSE*) which yields a normalized measure of the difference between the ground truth and predicted energy consumption of the i-th appliance was used in [214]..

$$RSE^{(i)} = \frac{\sum_{t=1}^{T} (y_t^{(i)} - \hat{y}_t^{(i)})^2}{\sum_{t=1}^{T} (y_t^{(i)})^2} \tag{16}$$

The R2 coefficient [214], which is defined as:

$$R_i^2 = \frac{\sum_{t=1}^{T} (y_t^{(i)} - \hat{y}_t^{(i)})^2}{\sum_{t=1}^{T} (y_t^{(i)} - \bar{y}_i)^2} \tag{17}$$

where $\bar{y}_i = (1/T) \sum_{t=1}^{T} y_t^{(i)}$. The *R2* coefficient and the relative square error (*RSE*) are both used to determine how well the predicted power profiles reflect the ground truth power profiles

over time. A good estimate of power usage profiles over time is necessary to advise customers of potential savings by deferring the use of a particular device to off-peak hours [214].

2.10. Public datasets

The availability of data for parameterization of algorithms is necessary for any problem that can be tackled with data mining and machine learning approaches. The recent development of smart meters that capture measurements much more frequently (for example, per second, minute, or hour) instead of reporting a house's energy consumption to users once per billing period (for example, once per month) makes disaggregation of whole-building energy data more accessible than ever. Energy disaggregation techniques require higher granularity energy readings since it involves extracting more information from the measurements than just the overall energy usage of the house. Energy consumption datasets are essential for NILM researchers in order to design and assess NILM algorithms. An energy disaggregation dataset is a gathering of electrical energy readings conducted in households under real-life scenarios, without perturbing the daily routine of households, and while maintaining the data as close to reality as possible [41]. It typically includes aggregated data (measured at the main power panel) and specific device data, which are gathered by either metering each specific device at the outlet or monitoring the circuit to which the load is attached. However, in a real-world setting, many loads are typically attached to the same circuit; subsequently, the latter technique does not always ensure that the specific consumption of all separate loads will be available [210].

In fact, the scope and resolution of datasets are most often constrained to the resources required to meet the experiments' purposes and are further driven by personal experience, equipment availability, budget, and time constraints [215]. Therefore, many differences between the datasets available in the literature can be observed, e.g., the granularity of the data (from a few Hz to several kHz), the number of sub-meter deployed (circuit level, device level and/or a single sensor for the entire house), the type of readings available (current, voltage, apparent, active and reactive power …), the duration of the dataset (from a few hours to several years), the storage formats, the region of collection or the availability of ground truth annotations [215], [216]. Although all of these characteristics are of paramount importance in categorizing state of the art datasets, the co-existence of device-specific consumption and aggregated consumption is typically utilized to classify the different energy disaggregation datasets, as this aspect has a significant impact on the potential applications of different

datasets. For instance, algorithms for identifying devices and recognizing activities may only be analyzed in datasets that include specific device consumption data [216].

There are several datasets available in the literature, each with its own set of characteristics and limitations, making it difficult to choose a dataset for dealing with energy efficiency challenges [217]. In most cases, these datasets only include a power reading for the entire household and/or many households loads due to the high expenses of purchasing and installing equipment. The authors of [209] presented REDD (Reference Energy Disaggregation Data Set). It's a collection of electricity consumption data for the entire household and specific circuits/appliances data acquired over several weeks from the 6 real households. The whole electrical signal of the house (voltage monitor on one phase and a current monitor on both phases of the power supply) at a high frequency (15kHz) was recorded. Electricity consumption measurements of up to 24 appliances are measured at 0.5 Hz sampling period each labeled with its category of device or devices. Moreover, the load schemes of over 20 plug-level monitors in the households, measured at 1 Hz, with an emphasis on recording electronic devices where multiple appliances are grouped on a single circuit are observed. However, the authors of [213] noted that the REDD whole-house readings were given in apparent power, while specific circuits were given in real power. As a result, the sum of different circuits did not match the total of the entire house. In [127], a building-level fully labeled electricity disaggregation (BLUED) dataset was presented. Voltage and current readings from a single house in Pittsburgh, Pennsylvania, USA, are resumed. The data were collected at a sampling frequency of 12 kHz over a period of seven days. It contains a large number of events (2845) with timestamps, device identifiers and phase labels. There are 50 different devices and 30 different device categories involved in the monitoring operation. Each state change of each device in the household was tagged and time-stamped, which provided the ground truth needed to evaluate the event-based methods. The authors of [218] presented the TRACEBASE dataset which contains energy consumption trends for many appliances, allowing disaggregation analysis. The measurement was taken at a 1 Hz sampling frequency. The data set can be used to improve energy efficiency. However, since no information about the appliances being studied or their features is provided, it cannot be used for appliance recognition or preference detection. Moreover, it collects data from 43 different devices, each with different recordings over several days and in several houses. In [219], the UMASS Smart* Home Data Set that continuously collects a wide range of environmental and operational data in three real households was presented. The data includes the average electrical energy usage readings of a house every

second, as well as the operation of every circuit and almost every outlet, data on electricity generation from solar panels and wind turbines installed on site, indoor room temperature and humidity data etc. Moreover, the authors also presented 24 hours of minute-level electrical data from 443 anonymous households in the United States. The Measured end-use electric Load profiles (MEULPv.1 and MEULPv.2) datasets were presented in [220], [221]. MEULPv.1 reports the electrical energy usage readings of 12 Canadian homes. Data were collected at both the aggregate and device levels using a 1-minute sampling frequency. During the data collection procedure, a total of 8 devices are tracked. MEULPvi.2, on the other hand, monitors 23 homes for one year at a 1-minute sampling frequency, identifying both aggregate and device-level consumptions. The authors of [222] presented a large-scale dataset (SustData). The dataset contains energy consumption and related information from 50 households and over 50 million individual records of electrical energy data, covering a totally 1144 different days. The Almanac of Minutely Power datasets (AMPdsv1 and AMPdsv2) were presented in [223], [224]. They include 11 one-minute interval measurements for 21 sub-meters over a one-year (AMPdsv1) and two-year (AMPdsv2) period. AMPds also contain water and natural gas consumption data. The authors of [225] presented the Indian Dataset for Ambient Water and Energy (iAWE) which collects energy usage readings at 1 Hz sampling periods. Measurements of electricity, water, and ambient parameters were conducted as part of the IAWE initiative in a three-story pilot house in Delhi for 73 days between May 2013 and August 2013. Moreover, 33 sub-meters were deployed throughout the household measuring these parameters, gathering a total of about 400 MB of data per day. In [226], an appliance Consumption Signatures - Fribourg 1 (ACS-F1) database was presented. ACS-F1 captures the energy usage in a set of houses at the device level. Electrical sub-meters were used to track the electrical energy usage readings of 100 households' appliances representing 10 different categories. The sub-metering of electricity is performed using 10 seconds sampling intervals for one hour period. The dataset is particularly suitable for device detection applications. The GREEND dataset is presented in [227] to characterize the itemized energy consumption patterns gathered during an experimental campaign in Austria and Italy via the analysis of the energy consumption of many specific devices. Eight households are observed during the collection campaign, each with up to nine distinct appliances. Device-level energy consumption patterns are collected at 1-Hz sampling rate over a six-month period. The authors of [228] presented ECO (Electricity Consumption and Occupancy) dataset. It is a comprehensive measurement campaign conducted to gather detailed information on the consumption patterns in six Swiss houses over a period of eight months using a sampling rate of 1 Hz. The dataset also includes information on the occupancy

of the observed houses. The Plug-Level Appliance Identification Dataset (PLAID) which presents energy consumption profiles for about 56 different household appliances representing about 11 appliance groups was introduced in [229]. The data were gathered over a three-month period in the summer of 2013 in Pittsburgh, Pennsylvania, USA using a sampling rate of 30 kHz, which is considered as one of the highest resolution frequencies employed in existing datasets of energy consumption in buildings. Pecan Street Inc. introduced the Dataport dataset [230]. It is the largest collection of disaggregated consumer energy data in the world. The dataset includes electricity data from 722 households across the United States, including 631 in Texas, 42 in California and 49 in Colorado. There are 501 single-family residences, 183 apartments, 35 townhouses and 3 mobile homes among the houses observed. In most houses, both mains' circuits and specific device circuits were recorded. Each circuit's average power demand was monitored at one-minute intervals. The REFIT dataset was presented within the framework of the UK REFIT project [231]. Data were gathered from 20 households over a two-year period using an 8 second sampling rate, providing electricity, gas, and heating data that can be viewed over multiple seasons. In [120], the Dutch Residential Energy Dataset (DRED) collects occupancy patterns, energy, and environmental data from a pilot household in the Netherlands. Sensors were deployed to track overall energy consumption as well as the electrical consumption of specific devices. Actually, 12 distinct household appliances are recorded at one-minute sampling periods, with sampling rates of 1 Hz used for aggregated data collection. The authors of [232] presented the United Kingdom-Domestic Appliance Level (UK-DALE) dataset. The voltage and current characteristics of three households are collected at 16 kHz sampling periods and two other households at 1 Hz sampling rates. In addition, specific device patterns from five different houses are recorded at a sampling interval of 6 seconds over periods ranging from 39 to 655 days. The Rainforest Automation Energy (RAE) dataset was presented in [233]. It contains aggregate and sub-measured data from two houses, which were taken at a sampling rate of 1 Hz over a period of 72 days for household 1 and 59 days for household 2. Environmental and home thermostat sensor data are included in addition to the energy consumption data. In [234], the Dataset on Information Strategies for Energy Conservation (DISEC) was presented. It is a project that collects data from 19 apartments in an Indian college housing complex over a period of 284 days. Various characteristics such as energy and weather are sampled every 30 seconds and aggregated into intervals of 15, 30 and 60 minutes. Weather changes are also updated using atmospheric conditions derived from nearby station data. The authors of [235] presented the building-level office environment dataset (BLOND). It is intended to provide voltage and current readings at the aggregate and

appliance levels. 53 appliances representing 16 device groups are monitored. It consists of two main data stores: BLOND-50 and BLOND-250. The former includes 213 days of energy usage readings collected at 50kSps (aggregated data) and 6.4kSps (specific devices data) while the latter includes usage patterns over a two months period, collected at resolution intervals of 250 kSps at the aggregate level and 50 kSps at the device level. In [236], an hourly usage energy (HUE) dataset is presented. It collects long-term energy consumption patterns from five houses using an hourly resolution. Moreover, the data from household 2 are collected for a period of one year using a sampling rate of 1 Hz while the appliance-level usage data from household 1 are gathered for a two-year period with a sampling interval of one minute. The authors of [217] presented the Qatar university dataset (QUD). It is gathered using a system that includes sub-metering modules recording the footprint of power usage in terms of energy consumption and other climate variables. Data were collected using a sampling interval ranging from 3 s to 30 min over a one-year period.

Table 1 provides a summary of the different data sets available in the literature. The analysis is based on various features gathered from each dataset, such as voltage (V), current (I), real power (P), reactive power (Q), apparent power (S), Temperature(T), Frequency (F), Energy (E), Power Factor (PF), phase angle (α), Weather (WT), occupancy (O), humidity (H), light level (L), the collection period and region, the number of monitors per house, the sampling rate, and the number of monitoring houses.

Table 1. Overview of the different datasets available in the literature and their characteristics.

Acronym	Release	Country	Features	#Houses	#Sub-meters	Sampling rate	Duration
REDD [209]	2011	USA	P, I, V	6	24	1/3Hz, 15kHz	2-4 weeks
HES [237]	2011	UK	P, I, V, T	26/251	23	10 min	1 year
IHEPCDS [238]	2012	France	P, I, V, Q	1	3	1 min	47 months
SMART* [219]	2012	USA	P, S	8	25	1Hz	3-4 months
MEULPv 1 [221]	2012	Canada	P	11	8	1 min	1 year

BLUED [127]	2012	USA	P, Q, I, V,	1	30	12 kHz, 60Hz, 1Hz	1 week
TRACEBASE [218]	2012	Germany	P	-	43	1-8 s	1883 days
PSD [239]	2012	USA	P	10	-	1 min	1 week
DATAPORT [230]	2013	USA	P	1400+	70	1 min	4 years
IAWE [225]	2013	India	P, I, V, Q, F, S, E, α	1	33	1 Hz	73 days
ACS-F1 [226]	2013	Switzerland	P, I, V, Q, F, α	-	100	10 s	1 hour
AMPds1 [224]	2013	Canada	P, I, V, F Q, S, PF	1	21	1 min	1 year
BERDS [240]	2013	USA	P, Q, S	-	4	20 s	1 year
ECO [228]	2014	Switzerland	P, I, V, α	6	-	1 Hz	8 months
PLAID [229]	2014	USA	I, V	11	60	30 kHz	3 months
SUSTDATA [222]	2014	Portugal	I, V, P, Q	-	17	12.8kHz, 0.5Hz	10 days
AMPds2 [241]	2014	Canada	P, S, I, F, V, PF	1	21	1 min	730 days
UKDALE [232]	2015	UK	P, I, V, Q, S	5	5-53	6 s, 6 kHz (Agg), 1Hz (Agg)	655 days
DRED [120]	2015	Netherland	P, T, H, WT	1	13	1 min, 1 Hz (Agg)	6 months
GREEND [227]	2015	Italy & Austria	P	8	9	1 Hz	6 months
REFIT [231]	2015	UK	P, T, O, L, PF	20	9	8 s	213 days

HFED [242]	2015	India	EMI	-	15	5MHz	-
COOLL [243]	2016	France	I, V	1	46	100 kHz	2 hours
WHITED [244]	2016	Germany	I, V	-	110	44.1kHz	-
MEULPv 2 [220]	2017	Canada	P	23	5	1 min	1 year
RAE [233]	2018	Canada	P, V, Q, S, F, E	1	24	1 Hz	72 days
LILAC [245]	2018	Germany	I, V	-	15	50kHz	-
BLOND [235]	2018	Germany	I, V, P	1	53	6.4 kSps, 54 kSps (Agg)	213 days
HUE [236]	2019	Canada	P	5	-	1 h, 1 min	3 years
ENERTA LK [246]	2019	South Korea	P	22	7	1 Hz, 15 Hz	122 days
QUD [217]	2019	Qatar	P, H, T, O	3	4	3 s–30 min	1 year
CREAM [247]	2020	Germany	I, V	-	2	6.4kHz	-
Dataset* [43]	2023	Portugal	P, I, V, Q, S, PF, F, T, H, WT, α, O	1	-	1 Hz, 1 min	3 years

As can be seen in Table 1, the number of publicly available datasets in literature has significantly increased in recent years. However, only certain datasets contain both aggregate and individual device/circuit consumption (AMPds, REDD, DATAPORT, IAWE, SMART*, ECO, UK-DALE, IHEPCDS, DRED, REFIT). and therefore, can be used for eventless NILM approaches. As such, these datasets can be utilized for both training and validation data whereas

datasets such as HES, TRACEBASE, GREEND, ACS-F1 only contain device-level energy consumption and are therefore not suitable for testing NILM approaches. One technique to use these datasets for NILM algorithms is to aggregate them by summing up each device's power consumption [210]. Nevertheless, this method has several drawbacks that will undoubtedly affect the final result. In fact, the summation of all the specific device consumptions will completely ignore the effects of the non-sub-metered devices, resulting in an oversimplified and unrealistic dataset that will not reflect the intricacies of the residential electrical system. Furthermore, datasets such as PLAID, WHITED, LILAC, CREAM, HFED, COOLL only contain the signatures of startup transients and spectral traces of several individual devices and thus may only be utilized to extract characteristics, design edge detectors, or create transient models. PLAID and WHITED datasets might be used to categorize power events, but OFF transitions cannot be classified since the datasets only contain startup transients. CREAM dataset allows for the extraction of internal operating states, which might be useful in improving complex appliance modeling.

In fact, there are a number of limitations regarding the data collection that make performance evaluation difficult. These limitations include limited labeling, missing data, and significant gaps in the available data sets. Indeed, missing data not only encompasses the gaps that are very common in many data sets, but also loads for which there is no sub-metering data [210]. This is due to a variety of factors, including the difficulty installing outlet-level meters in every load due to the lack of a fixed outlet, as well as the fact that some loads cannot be tracked via outlets. To tackle some of these challenges, a **dataset*** has been collected during almost 3 years in a residential household in Algarve, Portugal. The house includes a multitude of household devices. The data acquisition system will be detailed in the following sections.

Although one of the primary challenges is the complexity of developing hardware configurations to gather data for fully labeled energy disaggregation datasets, the primary challenge remains in the actual labeling step, which still depends largely on the manual inspection of the dataset, which is error prone and time consuming. Indeed, the process of labeling sensor data is a cross-cutting concern in many areas of machine learning since it is difficult to rely on completely automated labeling processes to ensure flawless labels.

Moreover, the large differences in available datasets could be another barrier that limits fair comparisons between NILM datasets. These differences result from several key factors, including the granularity and type of measures available, as well as the different formats in

which the data are provided to the NILM community (e.g., CSV, plain text, MAT, etc.). Meanwhile, relatively little investigation has been undertaken to date to examine the actual impacts of these differences. For instance, it is difficult to quickly determine which methodologies can be evaluated in a given dataset or the metrics that may be computed from a given dataset [210]. Therefore, finding a scalable and easily comprehensible approach to define the relationship between the algorithm, the metrics, and the dataset will be crucial for NILM research. The standardization of dataset descriptions and the design of a uniform metadata schema have recently been the subject of research [41], [248]. The primary motivation behind this is to have a common manner of importing various datasets with widely diverse file structures. The authors of [249] suggested the NILMTK, which may be used to quickly assess NILM algorithms across a variety of datasets. It is an open-source toolkit developed specially to facilitate the benchmarking of NILM techniques in a reproducible fashion. The toolkit consists of analyzers for a set of existing datasets, a set of statistics to define the datasets, a set of preprocessing algorithms, a couple of benchmarks NILM algorithms, and a set of evaluation metrics.

From a geographical perspective, the datasets are in most cases collected from the developed countries (USA, UK, Canada, Germany and so on). Since the measurement from different countries may result in a mismatch of electricity consumption (for instance, the RMS voltage value is 110V in the USA and 220V in Europe), care should be taken when using multiple data sets in the same system design [250]. It's worth noting that the compilation of consumption data for many contributions takes several days or months. Nonetheless, numerous datasets contain one-year or more, allowing researchers to analyze human behavior across time, including the impact of seasonal fluctuations on consumption [250].

2.11. Summary

In this chapter a comprehensive and detailed overview of the current state of the art of energy disaggregation was presented. First, a brief introduction of the concept of load monitoring and an overview of energy disaggregation are presented. The problem of energy disaggregation was then duly stated, followed by a review of the original NILM framework. Next, the main components of a NILM system, including data collection, feature extraction and electrical signatures of devices, pattern learning, and load classification, were discussed. The data collection process, including low frequency data collection and high frequency data collection were introduced while highlighting the relationship between sampling frequency and device

signatures. Following that, the background and research on the electrical features of the devices, particularly steady-state, transient, and non-traditional signatures, were reviewed. Although the majority of the study focused on exploring device signatures, there is still a need to create a comprehensive set of characteristics that may accurately describe all forms of signals. One alternative is the usage of non-power features which might be mixed with the power features to enable a more accurate detection. Regarding the pattern learning model and the load classification process, both event-based and non-event-based approaches for energy disaggregation were surveyed. The former relates signal state changes to device state changes, while the latter aims at estimating the global state of the system using statistical and machine learning approaches.

Finally, a discussion about the primary evaluation metrics typically used and a study of the different public data sets available in the literature were presented. With respect to performance metrics, there is no consensus on the most suitable performance metric to use in order to assess the success of a NILM technique. Therefore, the selection of the most suitable evaluation metric seems to depend on the method proposed. Regarding the publicly available datasets, although several datasets have been presented in the literature, some disadvantages limit their practical use, including limited labeling data, missing data, gaps, different data formats, and covering only some developed countries. Studies towards the standardization of public NILM datasets would be of paramount importance for NILM research.

Chapter 3 Theoretical background

3.1. Introduction

The objective of this chapter is to introduce the basic concepts of the soft computing methods used to develop the presented approaches for energy disaggregation. Specifically, Section 3.2 introduces the basic concepts of machine learning techniques. Then, the methods used, including CNN, LSTM, SVM, KNN, and DT are briefly described in Sections 3.3-3.7. Section 3.8 introduces radial basis function neural networks (RBFNNs) and their typical training and structure selection procedures. Section 3.9 discusses multi-objective optimization approaches with a focus on evolutionary algorithms and in particular the basic concept of genetic algorithms. Since the multi-objective genetic algorithm is employed as a framework to define the parameters and input features of radial basis function neural networks (RBFNNs), its relevant parameters and input characteristics, given the specified objectives, are discussed in Section 3.10. Finally, an overview of the data selection approach used is exposed in section 3.11.

3.2. Basic concepts

In recent years, machine learning has yielded impressive results in many domains, such as the classification of images, the recognition of speech, and so on [161], [251]. Deep architecture demonstrates a great ability to extract high-level features from data with complex structure by mimicking the structure of the human brain [58]. The key concepts were based on newly gained knowledge about the biological functioning of neuronal connections, biological neuron models and their related experiments [252]. Since our entire brain is not active at any given time, the hypothesis concentrated mostly on the significance of neural activity, with neurons having to exhibit a "all or nothing" signal pattern. This means that a neuron must be excited above a certain voltage threshold to be activated; for example, the sum of input connections must translate into a signal at the output, thus establishing a cause-and-effect principle. A biological neural network is a large set of interconnected specialized cells (neurons) that process and store information, thus handling the body processes to which they belong. The biological neuron model consists of, among others, dendrites considered as input connection to a nucleus, a cell body (nucleus) that processes the input signals, an axon that transmit the output signal to other connected neurons and synapses considered as output terminal (with weight) [250]. Fundamentally, a neuron responds to input (either inhibitory or excitatory) from its dendrites by generating an impulse along its axon, which is proportional to the combined influence of the

input signals, weighted by specific factors. The biological neural networks can handle very complex tasks in a very brief period (such as identification, memorization, recognition, etc.). Figure 3-1 depicts the structure of a biological neuron.

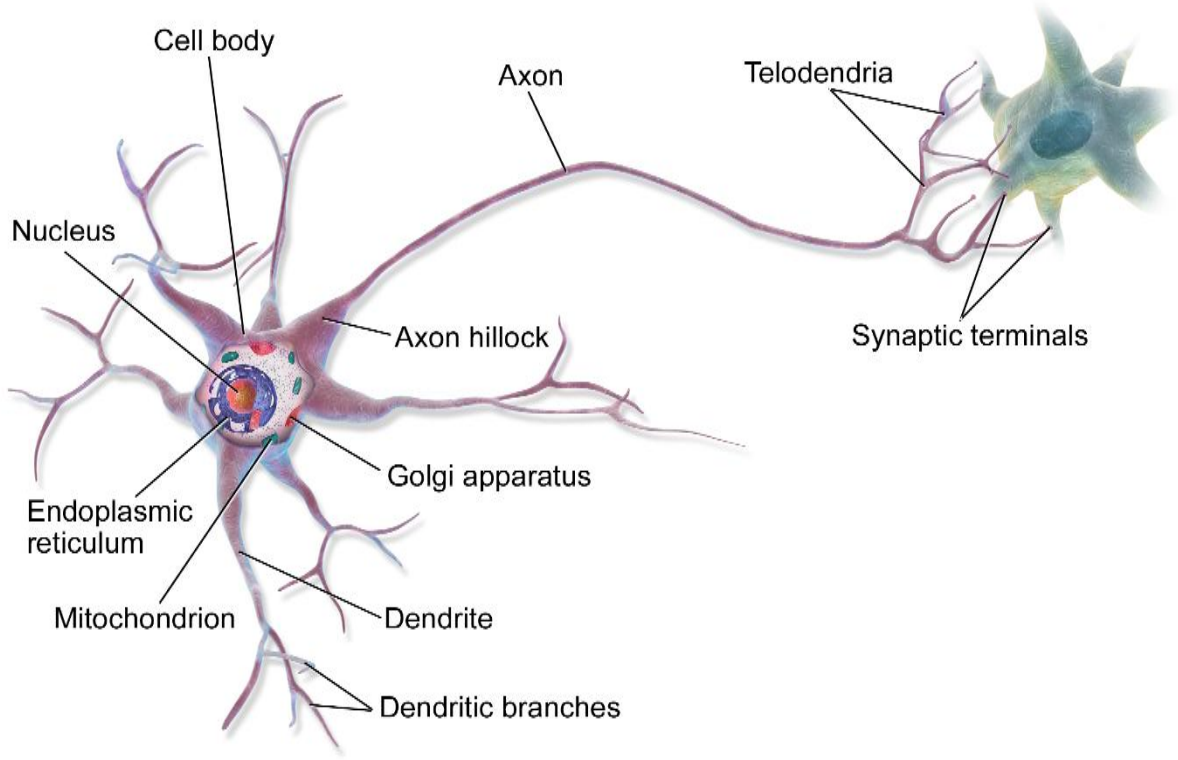


Figure 3-1. Biological neuron [253]

In fact, the brain is made up of many microscopic processing components known as neurons that work in tandem. These neurons are tightly coupled, with each neuron receiving information from a number of other neurons and delivering its results to a number of other neurons. The brain has the ability to learn, which is supposed to be accomplished by changing the strength of existing connections. However, a comprehensive description of the structure and processes of the human brain is not actually known. This results in a plethora of proposals for the model of the neuron, the pattern of interconnection, and most importantly, for the mechanisms of learning. The ANNs are considered as distributed parallel processors consisting of basic processing units that have a propensity to naturally store and make accessible experimental knowledge. [250], [254]. The ANNs acquire their knowledge by recognizing relationships and patterns in data and learn (or are trained) via experience. An ANN is made up of numerous single units, processing elements or artificial neurons, that are linked by coefficients (weights) and structured in layers to form the neural structure. In principle, ANNs do not necessarily model the intricacies of biological neurons, but can be perceived as units that take in signals

from other units, and sends out a signal to other units once it exceeds a certain threshold. Meanwhile, most of the key characteristics of ANN schemes were inspired by biological neural networks. These characteristics include learning process based on the network's knowledge gained from its environment, distributed memory, local information processing, synaptic weight adjustment by experience, use of synaptic weights for the storage of information acquired and so on. A neuron is a fundamental unit of data processing that plays a vital role in the operation of a NN. Its computational power is derived from the neural connection in a network. The structure of a neuron consists of 3 basic components [250]:

- A collection of synapses, each with an associated weight, that act as connecting links.
- An adder that sums the input signals and weights them according to the synaptic strength of the neurons.
- An activation function used to regulate the amplitude of the neuron's output. Commonly, the activation function is represented by a closed interval of [0, 1] or [-1, 1], which normalizes the output range of the neuron.

Given an artificial neuron (Figure 3-2), each input is represented by x_i and each connection by w_{ki}.

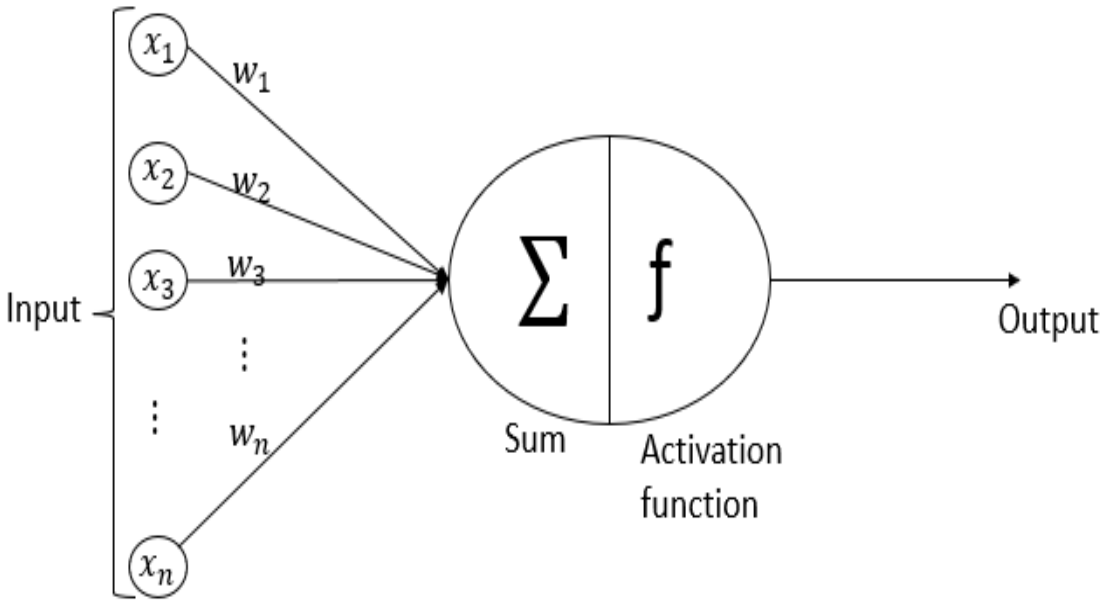


Figure 3-2. Artificial neuron.

The mathematical formulation of the activity of a neuron is as follows:

$$u_k = \sum_{i=1}^{n} w_{ki} x_i \tag{18}$$

$$y_k = \varphi(u_k + b_k) \tag{19}$$

Where b_k denote the bias, u_k is the linear combiner output, $\varphi(.)$ represents the activation function and y_k is the neuron output signal.

The activation function $\varphi(\vartheta)$ can assume different shapes. One is the threshold function, usually called the Heaviside function, and defined as follows:

$$\varphi(\vartheta) = 1 \; if \; \vartheta \geq 0 \tag{20}$$

$$\varphi(\vartheta) = 0 \; if \; \vartheta < 0 \tag{21}$$

3.3. Convolutional Neural Network (CNN)

The natural process of visual perception in living organisms inspired the design of CNN. The authors of [258] reported in 1959 that cells in the visual cortex of animals are responsible for sensing light in receptive fields. Motivated by this finding, the authors of [259] suggested the "*neocognitron*" in 1980, which might be considered as the predecessor of the CNN. In 1990, LeCun et al. [260] introduced the foundational concept of CNN in their seminal paper, which was later improved upon in the study reported in [261]. Recently, CNNs have been successful in achieving high accuracy in large-scale image classification tasks [60]. The CNN-based approaches' performance has constantly improved with substantial margins in the ImageNet challenge (ILSVRC) from 2012 to 2015 [262]. Their performance lies in their ability to emulate non-linear dependencies in a localized manner.

The use of CNN models for NILM was motivated by the similarity of NILM challenges to image recognition tasks [60]. In NILM, the effective detection of subtle local characteristics of target devices during their operation, which include the energy consumption level and the switch between different power levels, is the basis for successful recognition and estimation of target device activations and consumptions. These subtle patterns, such as a persistent tiny variation followed by a huge step shift after a brief time interval, demonstrate substantial local dependency. For instance, in the consumption profiles of many multi-state devices, these

characteristics and dependencies are particularly rich and robust. However, within mixed profiles of simultaneous energy use with other household devices, these subtle characteristics of the target devices are readily overwhelmed. It has been shown that the CNN is able to distinguish classes of images even when different and noisy backgrounds [262] overwhelm the main features. Hence, CNN could be a viable solution to the task of disaggregating device loads.

In the literature, there are several different CNN architectures [263]. Typically, the architecture of a CNN consists of 3 layers: the convolutional layer, the pooling layer, and the fully connected layer. The convolutional layer is used to learn the input feature representations. It consists of many convolution kernels that are employed to generate various characteristic maps. Basically, each feature map neuron in the previous layers is linked to an area of surrounding neurons. The new feature map may be generated by first seeding the input with a learned kernel, and subsequently applying a nonlinear feature activation function to each element of the convolved outputs. It's worth noting that the kernel is utilized by all spatial locations of the input in order to build each feature map. Numerous different kernels are used to obtain the full feature maps. The following equation can be used to define the feature value at point (i, j) in the kth feature map of the lth layer [263]:

$$z_{i,j,k}^{l} = W_k^l x_{i,j}^l + b_k^l \tag{22}$$

Where $x_{i,j}^l$ represents the inputs, W_k^l and b_k^ldenote the weight vector and bias term respectively. Since the kernel W_k^l that yields the feature map is distributed, the weight-sharing approach has various benefits, including reduced model complexity and ease of learning the network [263]. The activation function, which is essential for multi-layer networks introduces non-linearities to the CNN in order to detect non-linear features. The nonlinear activation function is defined by:

$$\varphi_{i,j,k}^{l} = \varphi\left(z_{i,j,k}^{l}\right) \tag{23}$$

A suitable activation function considerably increases the efficiency of CNN for a certain operation. Examples of commonly used activation functions are tanh, sigmoid, and ReLU [264]. ReLU is one of the most widely used activation functions in NN. It's defined as follows:

$$\varphi_{i,j,k}^{l} = max\,(z_{i,j,k}^{l}, 0) \tag{24}$$

where $z_{i,j,k}^{l}$ represents the input to the activation function. ReLU is a piecewise linear function that keeps the positive part while pruning the negative part to zero. ReLU's simple max () operation enables it to determine significantly faster than tanh or sigmoid activation functions, and it also generates sparsity in the hidden units, making it easier for the network to get sparse mappings [265]. Despite the fact that the discontinuity of ReLU at 0 may degrade backpropagation efficiency, many studies have demonstrated that ReLU is more effective than tanh and sigmoid activation functions from an empirical standpoint [266], [267].

The pooling layer seeks to achieve translation-invariance by lowering the dimensionality of the feature maps. It allows filter and select the features extracted by the convolutional layer. It is frequently positioned between two convolutional layers. The pooling layer's feature maps are linked to their corresponding feature maps in the previous convolutional layer. For each feature map $\varphi_{:,:,k}^{l}$, the pooling function is given as follows:

$$y_{i,j,k}^{l} = pool\left(\varphi_{m,\mathrm{n},k}^{l}\right), \forall\ (\mathrm{m}, \mathrm{n}) \in R_{ij} \tag{25}$$

where R_{ij} refers to the neighborhood in the area (i, j). Max pooling and average pooling are examples of common pooling operations [263]. In this work, max pooling is employed for the pooling operation. Basically, a max-pooling layer is a top-down resampling procedure that aims to downsize the length of the input sequence. It enhances the effectiveness of the training and lowers the overall number of trainable parameters for the entire CNN. Assume $x \in \mathbb{R}^{T}$ represents the input of the max pooling with stride n and size m. Suppose x [i : j] represents the slice of sequence x between the ith and the jth element and $y \in \mathbb{R}^{\frac{(T-m)}{n}+1}$ the output. The max-pooling layer's element is computed by:

$$y(i) = max\ x[i \times n : i \times n + m - 1] \tag{26}$$

Higher-level feature representations could be progressively extracted by layering multiple convolutional and pooling layers. Then, in order to conduct high-level reasoning, the multiple convolution and pooling layers may be followed by one or more fully connected layers. To establish global semantic information, the fully connected layer relates all neurons of the preceding layer to each neuron of the current layer. The result is generated by non-linearly merging the extracted features with the fully connected layer. It should be noting that a fully

connected layer is not necessarily required; a convolution layer 1x1 can be used instead [263]. The last CNN layer is an output layer. The SoftMax function is employed for classification tasks [265]. The optimal parameters for training the CNN may be found by minimizing a suitable loss function corresponding to the task. Consider that θ stands for all the parameters of the CNN. Considering a set of K desired input-output relations $\{(x^{(k)}, y^{(k)})$, k = $[1, \cdots, K]\}$ where $x^{(K)}$ represents the kth input data, $y^{(k)}$ is the matching target label, and $o^{(k)}$ is the CNN output. The formula for determining the CNN loss function is as follows:

$$\Gamma = \frac{1}{K}\sum_{k=1}^{K} \ell(\theta, y^{(k)}, o^{(k)}) \quad (27)$$

CNN training is a global optimization problem [263]. The process of finding the optimal set of parameters for the CNN involves minimizing a suitable loss function.

3.4. Long Short-Term Memory (LSTM)

LSTM was pioneered by the authors of [268] to tackle long-term dependency problems. It's a deep NN that handle sequential data and uses a feed-forward variation to addresses vanishing gradient issues. The LSTM design uses combined memory cells to replace the hidden layer of typical NN units. The LSTM architecture consists of a memory cell and three gates: the forget gate, input gate, and output gate. The operation of adding or removing information from the state of the memory cell is performed by the LSTM network using the input gate, forget gate and output gate. The specific function of each gate is determined by both the current external input and the output from the preceding cell. The input gate controls the input range of the new value to the LSTM, which is then transferred to the memory cell. The forget gate defines which data from the previous state of the cell should be discarded and which should be saved. It allows the cell to store or forget its previous state depending on the situation. The output gate controls the cell state that should be forwarded to the output. The output of the LSTM block is determined by taking the element-wise product of the current state of the filtered cell and the output gate. It can activate or inhibit the involvement of additional neurons in the cell state. The structure of the LSTM cell is presented in Figure 3-3.

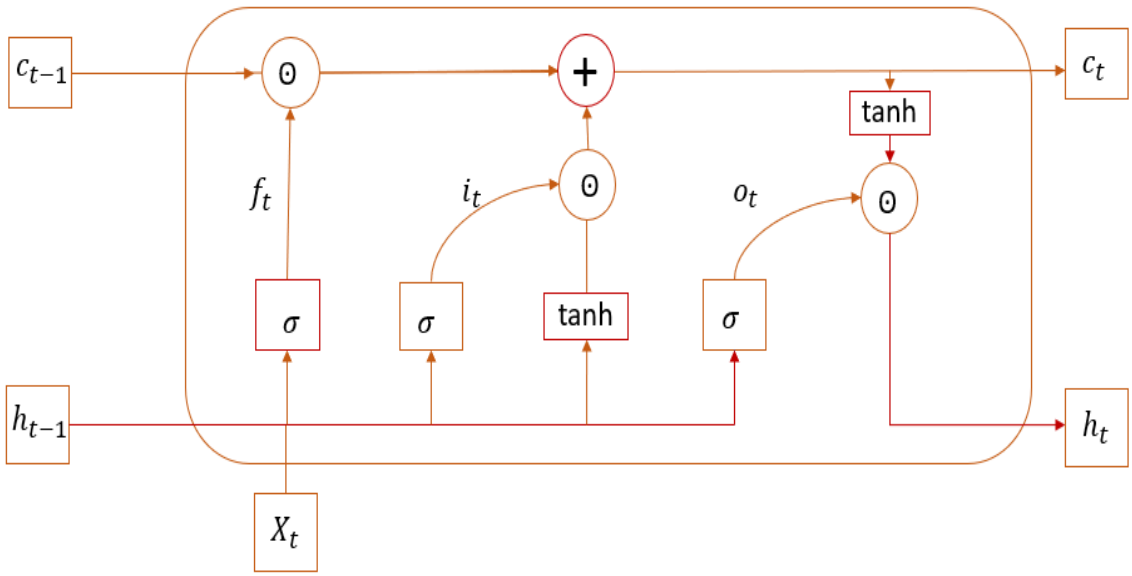


Figure 3-3. LSTM cell architecture [268].

The forget gate output f_t can be computed by considering the bias and weight values of the input sample x(t), and the feedback from the prior time step h_{t-1}.

$$f_t = \sigma\,(W_f.[h_{t-1}, x_t] + b_f) \tag{28}$$

where σ denotes the sigmoid function, W_f and b_f represent the bias and weight values respectively. The forget layer's output can be either 0 or 1. A value of zero (0) indicates that the information is completely discarded, whereas a value of 1 indicates that the information is completely stored. The forget state is assigned a value between 0 and 1 when these values are exposed to sigmoid activation function. Following the discarding of some data from the input sequence, it is decided which new information should be saved at the present time step. This stage is performed in two steps via the input gate and the Tanh gate (Tanh layer). The input gate is similar to the forget gate as it uses the following relationship to determine which values should be updated:

$$i_t = \sigma\,(W_i.[h_{t-1}, x_t] + b_i) \tag{29}$$

Where W_i and b_i denote respectively the bias and weight values in the input gate.

The Tanh gate generates a set of new candidate values that can be used to update the state. Instead of the sigmoid function, the Tanh function is employed in this stage.

$$\widehat{c_t} = tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{30}$$

Where $\widehat{c_t}$, W_c , b_c represent respectively the output, the weight and the bias values for the Tanh gate.

In the next step, the memory cell is updated from c_{t-1} to c_t. The following formula is used to determine the new state of the cell c_t using the data from the input gate i_t, forget gate f_t, and the prior state of the cell c_{t-1}:

$$c_t = f_t \odot c_{t-1} + i_t \odot \widehat{c_t} \tag{31}$$

Finally, the output gate o_t and the current cell state $\widehat{h_t}$ are multiplied to update the activation function.

$$o_t = \sigma\,(W_o.[h_{t-1}, x_t] + b_o) \tag{32}$$

$$\widehat{h_t} = \tanh(c_t) \tag{33}$$

$$h_t = o_t \odot \widehat{h_t} \tag{34}$$

The LSTM's cell state is updated in a single hidden unit using equations (29) to (34). The LSTM deep architecture is obtained by stacking multiple hidden LSTM layers sequentially. The use of bidirectional layers is one of the techniques for improving the deep recurrent NN models [51]. In fact, one recurrent NN processes the input sequence forward, whereas another processes it backward. The output from the forward and backward halves of the network are combined using an element-wise sum. Since the energy consumption data being studied by the energy disaggregation process involves signals with diverse patterns throughout different time intervals, it is essential to analyze the data over the long term [269]. The bidirectional LSTM network enables to estimate the actual value based on prior and future information, which makes it a suitable model for the NILM task [198]. It is composed of 3 layers: the input layer, the hidden layer, and the output layer. The hidden layer is composed of two identically designed LSTM layers that propagates the information in the opposite sense but use the same input. The vector formulas describing the forward LSTM layer for the propagation process are as follows [268]:

$$f_t = \sigma\,(W_f.[h_{t-1}, x_t] + b_f) \tag{35}$$

$$i_t = \sigma\,(W_i.[h_{t-1}, x_t] + b_i) \tag{36}$$

$$\widehat{c_t} = tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (37)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widehat{c_t} \quad (38)$$

$$o_t = \sigma\,(W_o.[h_{t-1}, x_t] + b_o) \quad (39)$$

$$\widehat{h_t} = \tanh(c_t) \quad (40)$$

$$h_t = o_t \odot \widehat{h_t} \quad (41)$$

Where $\odot$ represents element-by-element multiplication. b_f, b_i, b_c, b_o and W_f, W_i, W_c, W_o denote respectively the bias vectors and the weight matrices for different gates in the forward LSTM layer.

In the backward LSTM layer, the information propagation process is as follows:

$$f_t' = \sigma\,(W_f'.[h_{t+1}', x_t] + b_f') \quad (42)$$

$$i_t' = \sigma\,(W_i'.[h_{t+1}', x_t] + b_i') \quad (43)$$

$$\hat{c}_t' = \tanh(W_C'.[h_{t+1}', x_t] + b_c') \quad (44)$$

$$c_t' = f_t' \odot c_{t-1}' + i_t' \odot \hat{c}_t' \quad (45)$$

$$o_t' = \sigma\,(W_o'.[h_{t+1}', x_t] + b_o') \quad (46)$$

$$\widehat{h_t'} = \tanh(c_t') \quad (47)$$

$$h_t' = o_t' \odot \widehat{h_t'} \quad (48)$$

Where W_f', W_I', W_C', W_O' and b_f', b_i', b_c', b_o' represent respectively the weight matrices and bias vectors for the different gates in the backward LSTM layer.

The bidirectional LSTM output vector is given by:

$$\hat{y}^b = W_s.(h_t \oplus h_t') + b_s \quad (49)$$

where W_s represent the weight matrix, b_s denotes the bias vector and $\oplus$ is the vector splicing.

The loss function employed is given as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i^b - y_i^b)^2 \quad (50)$$

3.5. Support Vector Machines (SVM)

SVMs are a class of efficient and powerful supervised learning algorithms employed to tackle regression and classification tasks where the data are non-linearly separable with regard to the target characteristics. They stem from the ideas and theories of Vladimir Vapnik regarding the advancement of statistical learning in the 1990s [285]. Assuming a problem of two-class classification, the basic principle of SVMs is to convert the initial feature space into a higher dimensional space where the data may be linearly separated, and then to seek the best hyperplane that separates the two classes in the new space. Actually, the goal is to find the hyperplane that maximizes the distance between the closest data points of each class known as support vectors and the distance the maximum margin. Figure 3-4 illustrates an optimal hyperplane and the corresponding support vectors. The widest gap between the two classes, known as the margin of separation, is indicated by the support vectors, which are represented by red circles.

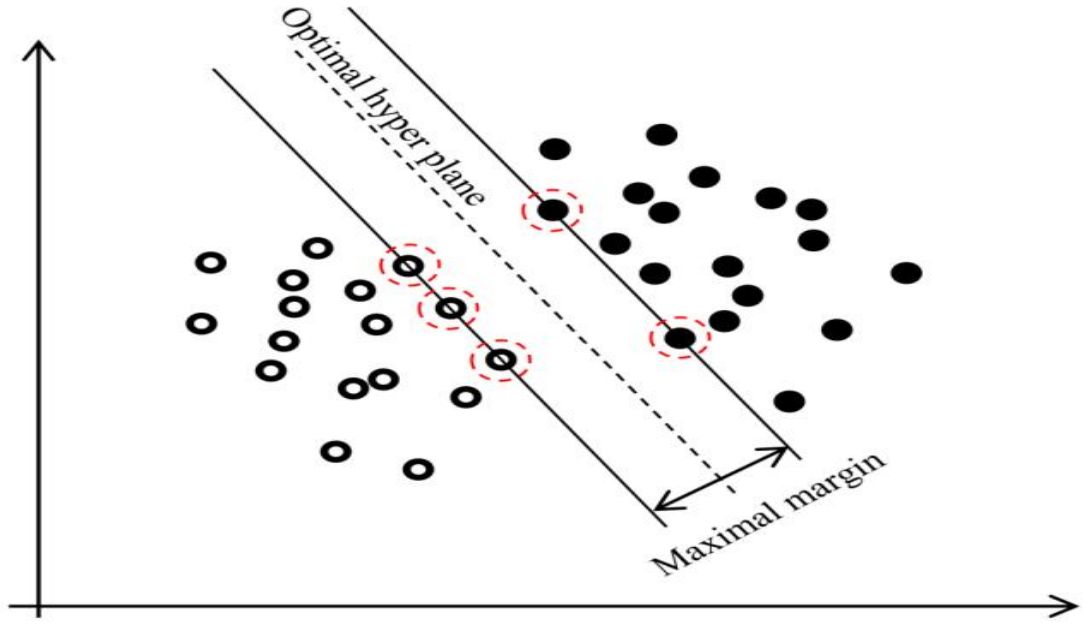


Figure 3-4. Illustration of hyperplane separation by SVM [285].

SVMs use a constrained quadratic optimization problem to address the large margin hyper plane determination [285]. In fact, the Lagrangian given in equation (51) must be maximized with respect to α_i under the conditions shown in equation (52) using Kuhn-Tucker theory.

$$L = \sum_{i=1}^{N} \alpha_i - \sum_{i,j=1}^{N} \alpha_i \, \alpha_j y_i y_j K(x_i, x_j) \tag{51}$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \; and \; 0 \leq \alpha_i \leq C \tag{52}$$

where N and α_i stand for the data points in the training set and the corresponding Lagrange multiplier associated with the i-th point. x_i and $y_i \in$ {-1, +1} denote the i-th input feature and the associated target class, respectively, C commonly referred to as the regularization parameter is a user-defined penalty parameter, and $K(x_i, x_j)$ are the inner product kernels given by the following equation:

$$K(x_i, x_j) = \sum_{z=1}^{m} \Phi_z(x_i) \Phi_z(x_j) \tag{53}$$

where m represents the dimensionality of the feature space that allows for linear separation of the transformed samples and $\Phi_z(x_i)$ corresponds to the zth dimension of the transformed sample x_i. There are many well-known kernel functions, such as:

- Gaussian radial basis function $K(x_i, x_j) = e^{-\frac{1}{2\sigma_i^2}\|x - x_i\|^2}$ (54)
- Homogeneous polynomial $K(\ x_i, x_j\ ) = (\ x_i, x_j)^d$ (55)
- Inhomogeneous polynomial $K\ (x_i, x_j) = (\ x_i . x_j + 1\ )^d$ (56)

The decision function can be described as follows:

$$f(x) = sign(\sum_{i \epsilon sv} y_i \alpha_i^* \, K(x, x_i) - \theta) \tag{57}$$

Where $\alpha_i^* s$ represent the solutions of the constrained optimization problem, SVs denote the indices of the support vector and θ is a user-defined threshold.

3.6. K Nearest Neighbors (KNN)

The KNN technique is a simple but effective non-parametric method used in supervised classification and regression tasks [286]. The technique consists in identifying the k number of training samples that are closest to the new input data in terms of Euclidean distance. The output for the new input is then determined by the majority vote of the k nearest training samples. The selection of an appropriate value for k is a challenging task, since the usage of a large value of k may decrease the precision of the algorithm, whereas the usage of a small value of k may make the results more sensitive to noise. Specifically, the k-NN technique is used to measure

the distance between a new input and the existing data points in the dataset, in order to determine the output for the new input. Any new query situation is subjected to a distance comparison of its features with all prior observed situations in the training dataset. The classification process involves assigning the new query to the class that is most commonly represented among its k-nearest neighbors in the dataset [287]. The distance used for this purpose is given by the following equation (59) for any input x_i:

$$d(x_i) = \min\{d(x_i, x_1), d\,(x_i, x_2), \dots, d(x_i, x_n)\} \tag{59}$$

The Euclidean distance is employed in this work. It is computed using the following equation:

$$d_E\big(\{x_1, x_2, \cdots, x_N\}; \{y_1, y_{2,\cdots,} y_N\}\big) = \sqrt{\sum_{1}^{N} (x_i - y_i)^2} \tag{60}$$

where N denotes the total number of dimensions.

3.7. Decision Trees (DT)

DT is an algorithm for developing a knowledge-based system through inductive inference of case histories. The structure of a DT may be regarded as a tree-like illustration of the decision-making process [288]. It organizes data according to tree characteristics, with leaf nodes representing a set of records subject to constraints. The process of constructing the decision tree involves developing a branch of the tree based on distinct record field values. In fact, a decision tree is created by repeatedly including additional branches and sub nodes to each subset of branches. The selection of different record field values is one of the critical steps in creating the decision tree. The structure and rate of development of the decision tree are influenced by the quality of the selection process. The DT's classification performance is also affected. The growth rate of the decision tree is slowed down by an unsatisfactory branch value, which also results in a branch that is too thin and can lead to the discarding of some relevant information rules [288].

The ID3 DT classification algorithm was first presented by Quilan in 1986 [289]. The inability of the ID3 algorithm to handle discrete features and its propensity for overfitting led to the subsequent development of the C4.5 algorithm. It consists in automatically selecting discriminating factors from unstructured and possibly very large data. Thus, logical concepts of cause and effect that were not obvious in the raw data may be deduced through decision trees. The different potential decisions are made at the ends of the branches and closely follow

the decisions made at each level. The following is a definition of the C4.5 algorithm: Considering a training set consisting of A data points in the form $(V_1, V_2, \cdots, V_a, C)$, where V_i denotes the sample attribute value and C its category. Assuming that the discrete attribute A has k different values, the sample set T is split into subsets $T_1, T_2, \cdots, T_k$. The information gain rate is then given as *ratio (A, T)* resulting from the division of the sample set T by the discrete attribute A.

$$ratio\ (A,T) = \frac{Gain\ (A,T)}{spilt(A,T)} \tag{61}$$

The *Gain (A, T)* and the *spilt (A, T)* can be defined as:

$$Gain\ \ (A, T) = inf\ (T) - \sum_{i=1}^{k} \frac{|T_i|}{|T|} \times inf\ (T_i) \tag{62}$$

$$spilt\ (A, T) = -\sum_{i=1}^{k} \frac{|T_i|}{|T|} \times log_2 \left(\frac{|T_i|}{|T|}\right) \tag{63}$$

where *inf(T)* denotes the information entropy of T.

3.8. Radial basis function neural network (RBFNN)

The RBFNN is a high-dimensional interpolation technique introduced in [290]. It can be used to tackle a wide range of applications, including time series problems, classification, prediction, interpolation, and many other applications. It has many benefits, including high precision, fast learning, and a significant potential for self-adaptation. The RBFNN consist typically of 3 functionally different layers. The input layer consists merely of a set of sensory units made up of input features. The first layer's features are mapped directly, without any weights, to all the hidden layer's neurons. The second layer is a hidden layer. It is designed to have a suitable number of hidden units that perform nonlinear transformations on the input data, transforming it into a new space. Each neuron in the hidden layer implements a single radial basis function. The final layer, which is the third layer, performs a linear transformation from the hidden unit space to the output space. Figure 3-5 illustrates the topology of a RBFNN with one hidden layer.

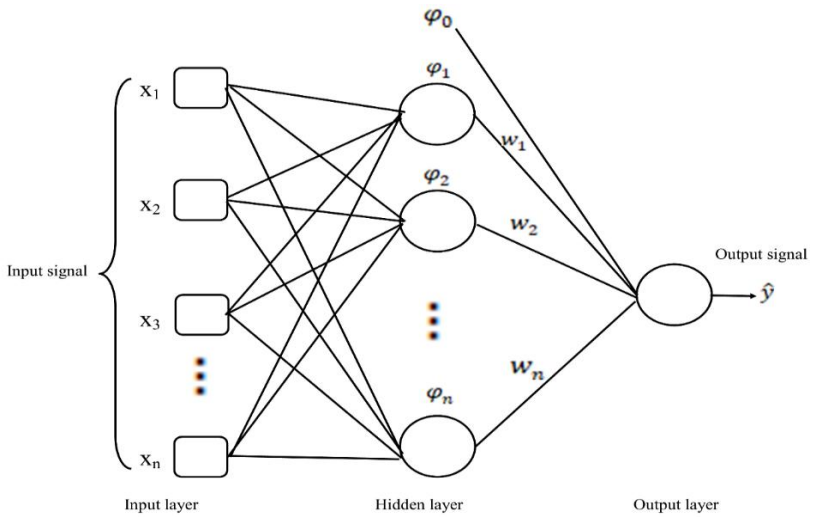


Figure 3-5. Radial basis function neural network structure.

The output of the model is given as follows:

$$\hat{y}(\mathrm{X}) = \sum_{\mathrm{i}=1}^{n} W_i \, \varphi_i(\|X - C_i\|) + \; b \tag{64}$$

Where W_i and b represent the weights and bias term, n denotes the number of neurons in the hidden layer, $\|.\|$ represents the Euclidean norm and $\varphi\{\|X - C_i\|\}_1^n$ are nonlinear radial basis activation functions centered at points C_i. The most used nonlinear activation function is a Gaussian function defined as follows:

$$\varphi(\|X - C_i\|) = \; e^{-\frac{1}{2\sigma_i^2}\|X - C_i\|^2} \tag{65}$$

Where σ_i represents a spread parameter.

The training (offline) of an RBF neural network consists in finding the values of C, W, and σ to minimize (66) for a given number of neurons and a given set of inputs [291]:

$$\Phi(\mathrm{X}, \mathrm{W}, \mathrm{C}, \sigma) = \frac{\|y(X) - \hat{y}(X, W, C, \sigma)\|^2}{2} \tag{66}$$

where y represents the actual output vector of the system given the input data X.

Since the outputs of the hidden layer are combined linearly to form the model output, equation (66) may be formulated as follows:

$$\Phi(\mathrm{X}, \mathrm{W}, \mathrm{C}, \sigma) = \frac{\|y(X) - \varphi(X, C, \sigma)W\|^2}{2} \tag{67}$$

Determining the global optimal value (W*) of the linear parameters W with regard to the nonlinear parameters C and σ as least squares solution.

$$\mathrm{W}^* = \varphi^+(X, C, \sigma)\; y(X) \tag{68}$$

where "+" represents a pseudo-inverse, and replacing (68) in (67), the training criterion to compute the nonlinear parameters C and σ is:

$$\Phi(X, C, \sigma) = \frac{\|y(X) - \varphi(X, C, \sigma)\varphi^{+}(X, C, \sigma)\, y(X)\|^2}{2} \tag{69}$$

Equation (69) is now independent of the linear parameters W. It is minimized using the Levenberg-Marquardt approach [292]. A clustering method known as the optimal adaptive k-means algorithm (OAKM) [293] is used to initialize the hidden layer function center positions. Equation (70) is applied to define the initial spreads of the neuron activation functions [294].

$$\sigma_i = \frac{d_{max}}{\sqrt{2\pi}}, i = 1, \dots, n \tag{70}$$

where d_{max} represents the largest distance between the centers found by the OAKM algorithm.

The training procedure described so far for the estimation of parameters assumes that the relevant inputs and the network topology (number of neurons) have been defined. However, in order to determine the parameter vector W, the number of hidden units, the number of neurons and the range of input features must be specified. Generally, the set of d input features, $x_{k,i}$, must be chosen from a larger set, F, whose dimension is often much larger than the maximum dimension d_M intended for the input vector.

The problem can be specified as follows: select d ($d_m < d \leq d_M$) input features from the set F, an appropriate number of neurons n ($n_m < n \leq n_M$), and determine the parameter vector w, so that the best RBFNN mapping, given in (64), is obtained.

The problem of designing an ANN can be clearly separated into two distinct subproblems, each highlighting certain design features [295]:

1. **ANN parameters** refer to the parameters of the network. They are calculated using a learning approach.
2. **ANN structure** refers to the network topology. It consists in particular in selecting appropriate range of inputs and an adequate number of neurons.

Several approaches have been suggested to tackle these two sub-problems, either jointly or separately, some of them failing to exploit fully the existing techniques that are deemed more suitable [296]. The first is a problem of nonlinear optimization of parameters, for which nonlinear gradient-based approaches have proven to be effective. The second one requires a multi-objective optimization approach because it is a combinatorial optimization challenge.

The problem decomposition reveals the existence of quality metrics for each sub-problem and provides detailed guidance on how to select them [296]:

1. **ANN parameters** the training stage's effectiveness and the quality of the mapping generated by the computed parameters should be reflected in the quality metrics.
2. **ANN structure** the quality metrics should indicate the suitability of the ANN structure for the application at hand.

This decomposition in the nature of the quality metrics enables us to define a two-component quality vector as follows:

$$\mu(F, \Lambda, n, W) = [\mu^p, \mu^s] \tag{71}$$
$$\mu^p = [\mu_1^p, \mu_2^p, \dots, \mu_u^p],$$
$$\mu^s = [\mu_1^s, \mu_2^s, \dots, \mu_v^s],$$
$$\Lambda = [\lambda_1, \lambda_2, \dots, \lambda_d]$$

where μ^p and μ^s include the quality metrics u and v associated with each of the subproblems, the superscripts p and s represent the quality metrics associated with the training phase of the ANN network and the suitability of the ANN network for the specific application, respectively. Λ refers to the index vector of the columns of F which defines the input features under consideration.

Considering that the quality metrics in $\mu(F, \Lambda, n, W)$ are well-defined quantities that specify the objective functions that should be minimized to achieve the "best ANN for the application at hand", the problem can now be formulated as follows: select d $\in [d_m, d_M]$ input characteristics from F, n $\in [n_m, n_M]$ neurons, and determine W, such that $\mu(F, \Lambda, n, W)$ is minimized.

$$\min_{\Lambda, n, W} \mu(F, \Lambda, n, W), \text{ given: (F, y), d} \in [d_m, d_M] \text{ and n} \in [n_m, n_M] \tag{72}$$

Based on the definition of $\mu(F, \Lambda, n, W)$, it is plausible that certain objective functions are conflicting, for instance, in μ^s, there may be an objective of minimizing the complexity of the ANN network, to express the objectives of enhancing performance while lowering the network size. Since there is no single solution that minimizes all components of $\mu(F, \Lambda, n, W)$ simultaneously, the problem described in (72) is therefore a combinatorial multi-objective optimization problem. The set of Pareto points in the design space that determines the Pareto

front in the objective space is the actual solution. This means that the ANN model designer must choose a specific ANN by evaluating the tradeoffs in the Pareto front objectives.

The solution lies in performing an exhaustive search on the search space represented by (F, d ∈ $[d_m, d_M]$, $[n_m, n_M]$), thus exploring the true Pareto front. However, this is generally not feasible in a reasonable time due to the difficulty of assessing $\mu(F, \Lambda, n, W)$ and the size of the search space. Trial and error procedure may provide a strategy for conducting the search, but since there are so many potential outcomes, it is possible to perform many tests without discovering any suitable objective values for $\mu(F, \Lambda, n, W)$. Furthermore, since the the connection between the search space and the objective space is unknown, the test results could easily lead the designer astray to some undesirable local minima [296].

Although many approaches were suggested over the years to tackle multi-objective challenges, it is only recently that the ability of evolutionary algorithms (EA) to approximate the Pareto front has been realized, leading to the development of the research area now referred to as multi-objective evolutionary optimization (MOEA) [296]. Therefore, to tackle the challenge of finding the structure of ANNs, Multi-Objective Optimization Algorithm are used, since they have been shown to be reliable when dealing with challenges with multiple conflicting objectives [291], [297], [298].

3.9. Multi-Objective Optimization

In classical optimization tasks, there is a single objective and a single ideal solution [299]. This can be formulated as follows:

$$F(x) = \min_x(f_1(x)) \ \ s.t. x \ \epsilon \ X, \tag{73}$$

Where x denotes the decision vector and X ⊆ $\mathbb{R}^D$ refers to the parameter space. Moreover, the objective space is given by Y such that F: X → $Y \subseteq \mathbb{R}^D$.

Multi-objective optimization (MOO) is commonly used to tackle problems with many objectives. In such problems, the objectives are usually not well correlated with each other, making it impossible to represent them as a single objective. Consequently, the ideal solution is usually not a single point in space, but rather a set that characterizes the possible alternatives. The MOO problems can be expressed as follows [300]:

$$F(X) = \min_x(f_1(x), \cdots, f_z(x)) \ \ s.t. x \ \epsilon \ X, \tag{74}$$

Where F: X → $Y \subseteq \mathbb{R}^z$ involves z objectives.

The definitions of some relevant terms that will be used later are introduced below [300].

- **Dominate solution:** a solution x^1 dominates solution x^2 ($x^1 \leq x^2$), if and only if $f_i(x^1) \leq f_i(x^2)$ for i ϵ $(1,2,\cdots,Z)$ and $\exists\, j \,\epsilon\, (1,2,\cdots,Z)$ such that $f_j(x^1) < f_j(x^2)$.
- **Non-dominated solution:** a solution x^1 and solution x^2 are non-dominated with regards to each other if and only if $\nexists\, i \in (1,2,\cdots,Z)$ such that $f_i(x^2) < f_i(x^1)$ and $\nexists\, j \in (1,2,\cdots,Z)$ such that $f_j(x^1) < f_j(x^2)$. i.e., $x^1 \leq x^2$ and $x^2 \leq x^1$ are not satisfied.
- **Pareto-Front:** it is a set of all possible non-dominated solutions. Each non-dominated solution x^0 is in the Pareto-Front if $\nexists\, x \,\epsilon\, X$ such that $(x \leq x^0)$.

The optimization of a design problem is a challenging task in the real world, as it involves multiple objectives, many of which are often in conflict with each other. This implies that enhancing one may harm the other. In this situation, there is a set of non-dominated or Pareto-optimal solutions, and none of them is better than the others in terms of objectives. Figure 3-6. illustrates an example of an optimization problem with two objectives f1 and f2. Solution (1) dominates solutions (4) and solution (2), solution (3) dominates solution (2), and solutions (1) and (3) are non-dominated with respect to each other.

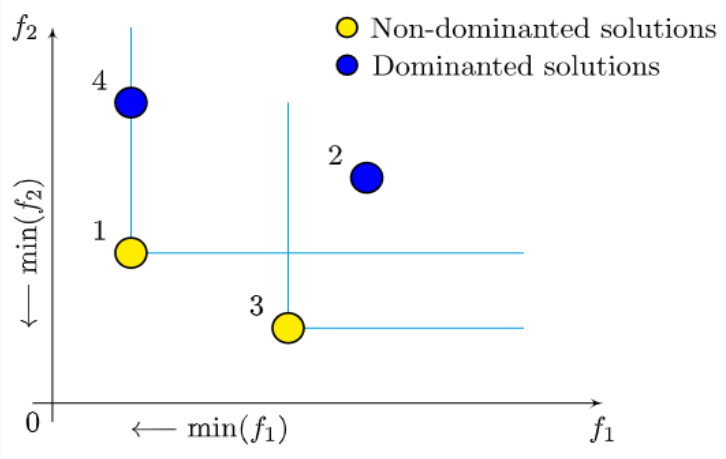


Figure 3-6. Bi-objective minimization problem [299]: solution (1) dominates solution (4) and solution (2), solution (3) dominates solution (2), and solution (1) and solution (3) are non-dominated.

The multi-objective optimizer aims to enhance the surface of the non-dominated solutions so that it is as close as possible to the origin.

3.9.1. Evolutionary algorithms (EA)

Comparing each solution to all other solutions is a simple and relatively slow approach for finding the Pareto front set. This procedure has a complexity level of O (ZN^2), with N denoting

the number of solutions and Z the number of objectives. Hence, to enhance the effectiveness of the search for the most suitable solutions, EAs have been introduced [296], [299]. They exploit the natural evolution principle: the survival of the fittest. These approaches generally employ a population of randomly selected solutions. Each random solution is referred to as an offspring, and a set of these solutions are referred to as a generation. Each iteration consists of updating, improving, and performing several tasks including crossover (combining solutions), mutation (adding new solutions), fitness assignment (computing the cost function for each solution), and selection (selecting the best solution for the next iteration).

3.9.2. Genetic algorithms (GA)

GAs are popular evolutionary algorithms whose objective is to get an approximate solution to an optimization problem when there is no exact method (or the solution is unknown) to address it in a reasonable time. The GA was proposed by *John Holland et al.* based on the natural process of evolution to determine an ideal solution in a search space [301]. It relies on two key concepts: (i) Survival of the fittest or competition and (ii) Inheritance of the child from the parents' genetic heritage. The GA initiates its operation by generating a certain number of potential solutions, referred to as the initial population. Following this, the initial population is expected to evolve over a certain number of generations. Each potential solution, also referred to as an individual is subsequently assessed and assigned a fitness measure since the strategy is to utilize the elites to generate the next generation. The replacement operators, mutation, parental recombination, and mating selection are all used in the canonical GA. The concept of evolution involves producing successive generations with the goal of generating individuals with a relevant fitness level after a certain number of generations. By mating the elitists from the initial population, a new generation is developed to replace the previous population, with the expectation that the new generation will be more suitable for the challenge.

The problem that the GA seeks to define is considered as a black box system with one output parameter and many input parameters. Specifically, a representation referred to as a chromosome encodes the input parameters that characterize a potential solution, while the output parameter is typically the outcome of a function that illustrates the fitness of the potential solution to the problem at hand. The input parameters are regarded as genes, and a specific combination of genes may generate a chromosome that is a potential solution to the problem being tackled. The objective is to determine a combination of input parameter values that generates a suitable value for the output parameter [302].

Once the genetic mapping of the solution range has been defined, the individuals of the first generation are generated, usually randomly, thus integrating the entire search space. A portion of the current population of each generation should be chosen to raise the following generation. The primary objective is to enable the genes of the best individuals to be inherited by the next generation. For this purpose, a fitness value is assigned to each individual thanks to a fitness function.

3.9.2.1. Selection

To develop a new generation, some of the individuals in the generated initial population should be selected once each has been assigned a fitness value. Several well-known selection techniques can be employed, such as roulette wheel (also referred to as proportional fitness selection), tournament selection, stochastic universal sampling (SUS), and truncation selection.

- **Roulette wheel**

In this approach, individuals are first sorted in ascending order according to their fitness scores. Assume that the i-th individual receives the fitness value f_i. The second stage involves computing the relevant normalized fitness value for each individual as follows:

$$P_i = \frac{f_i}{\sum_{k=1}^{I} f_i} \tag{75}$$

where I represent the number of individuals. The probability of choosing the i-th individual is actually indicated by the normalized fitness value P_i. In the third phase, a probability interval is calculated for each individual with respect to the accumulated normalized fitness scores. it is defined as follows:

$$[l_i, l_i + P_i] = \left[\sum_{j=1}^{i-1} P_j \, , \sum_{j=1}^{i-1} P_j + P_i\right] \tag{76}$$

where l_i stands for the interval's lower bound for the i-th individual. It should be noted that l_1 is equal to zero for the first individual. In the fourth stage, the individual whose interval contains the random number k chosen in the interval [0,1] is then chosen for mating. In order to create a pool of potential parents for mating, this phase is performed multiple times. The individuals with high fitness values have a greater chance of being chosen than the rest using this approach. Moreover, they stand a good chance of being chosen more than once during the selection process.

- **Tournament selection**

Using the tournament selection approach, multiple attempts are conducted to randomly select a subset of size s from the current population. Within each tournament, the most deserving individual is selected to be the parent who will raise the next generation. The most vulnerable individuals are less likely to be chosen if s is a large value. In contrast, when s = 1, the tournament approach behaves like the random selection approach.

- **Stochastic Universal Sampling (SUS)**

The SUS technique is an extension of the roulette wheel approach that selects an individual using several evenly spaced points at a time instead of just one. Therefore, more than one individual is selected simultaneously from the SUS. The benefit of the SUS is that it gives weaker individuals a stake to be chosen as well. This approach mitigates the unfairness of the roulette wheel approach, which largely overlooks the weaker individuals. In the SUS approach, a random variable r is first chosen from the range [0, k], where k is calculated as follows:

$$k = \frac{F}{I} \tag{77}$$

where F and I, respectively, represent the total fitness values and the number of individuals that the technique is intended to select. I pointers are produced as follows in the second phase:

$$pointers = [p_1, p_2, \cdots, p_N], \qquad p_i = r + (i-1) * k \qquad for\ i = 1\ to\ I \tag{78}$$

Similar to the roulette wheel approach, the individual for which the interval contains p_i is chosen for mating for each pointer p_i.

3.9.2.2. Crossover

Crossover is a type of genetic operator that combines genes. The basic goal of crossover is to mix the beneficial gene combinations of the two parents to generate a new individual that benefits from both parents' gene sets. The parents' genes are combined to create new chromosomes, which are then used to generate offspring. The crossover operator uses a probability known as the crossover rate, which often requires a high probability value (for example, 0.7). Many crossover techniques have been presented including one-point, multipoint, Cut and Splice and Uniform crossover techniques.

- **Single-point crossover**

In the single-point crossover approach, the two mating chromosomes, or the chosen parental population, are cut at a randomly selected point called the crossover point or pivot point. To generate two offspring chromosomes (children), the genetic material to the left (or right) of the

point is exchanged between the two parent chromosomes at this cut. The rest of the genes after the designated point are interchanged between the two parents, leading to the creation of two offspring that have the same size as their parents. Figure 3-7 illustrates the single-point crossover operator.

Figure 3-7. Single-point crossover operator.

- **Multipoint crossover**

Multipoint crossover, unlike single-point crossover, works with multiple crossover points. Therefore, the original parent chromosomes are more fragmented. The genetic information between two selected pivot points that are to the right of an even (or odd) number of crossover points is swapped to generate two distinct children. Figure 3-8 illustrates a two-point crossover approach.

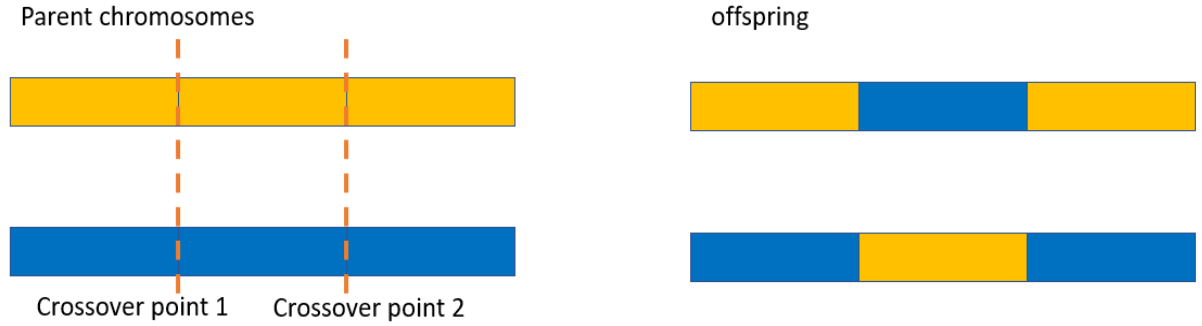


Figure 3-8. Two-point crossover operator.

- **Cut and Splice crossover**

The Cut and Splice technique is comparable to the single point technique; however, each parent has a distinct pivot point. The two parents swap each gene after each point, resulting in two children of distinct lengths. Figure 3-9 depicts an example of the Cut and Splice technique.

Figure 3-9. Cut and Splice crossover.

- **Uniform crossover**

In contrast to previous techniques, in the uniform technique the parents contribute to the generation of the offspring at the gene level rather than at the segment level. Specifically, it is decided for each gene whether it should be swapped with the corresponding gene of the other parent or whether it should remain unchanged. In this technique, each gene is exchanged with a fixed likelihood, often 0.5, so that each offspring inherits approximately 50% of the genes from the first parent and 50% from the second. Figure 3-10 illustrates an example of the uniform crossover technique.

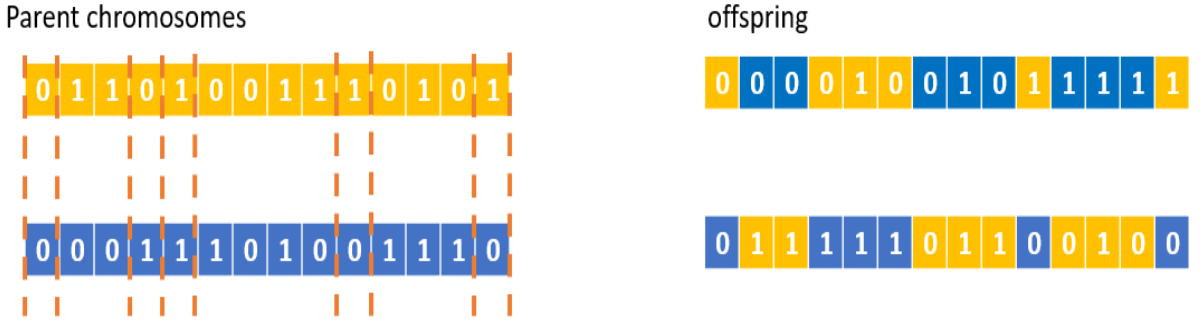


Figure 3-10. Uniform crossover.

3.9.2.3. Mutation

A second genetic operation called mutation is applied to a generated offspring to add diversity to the next generation. In this operation, one or more genes of the relevant chromosome are mutated. The mutation rate has a much smaller range than the crossover rate. The typical approach to mutation affects the value of a gene with a probability of $\frac{1}{l}$, where l represents the length of the chromosome.

3.9.2.4. Replacement

It involves deciding which individuals from the current generation will pass to the next after the offspring have been generated and evaluated. The most common method is to completely

replace the current population with the new offspring. However, in some advanced techniques, the parents of the current population are given the opportunity to compete with the offspring and only a portion of the current population migrates to the next generation.

3.9.2.5. Termination criteria

Typical termination criteria for genetic algorithms are as follows [303]: (i) the specified maximum number of generations is attained, (ii) the maximum number of fitness function evaluations is reached, (iii) the probability of obtaining relevant changes in the following generations is extremely low.

To define the upper bound of the first two criteria, an appropriate maximum search length should be estimated, which requires some knowledge of the problem to be tackled. The third method (iii), on the other hand, does not necessitate any knowledge of the problem. It has two variations: phenotypical and genotypical termination criteria. The phenotypical termination criterion evaluates the progress made over the previous generations. The average fitness value over the last n generations may be used to indicate progress. The algorithm ends if the average exceeds a pre-specified threshold. The genotypical method ends the genetic algorithm when the present population reaches a certain level of convergence with regards to the population' s chromosomes. Specifically, the criterion verifies whether a certain proportion of genes in the population have converged. The designer of the genetic algorithm specifies the convergence of a gene to a certain value by establishing a threshold that must be reached. For instance, a gene is considered to have converged to x if 90% of the chromosomes of a given gene have the same value as x. The program then stops when, for example, 80% of the genes have converged.

The operating procedure of the Genetic algorithm is summarized in figure 3-11 flowchart.

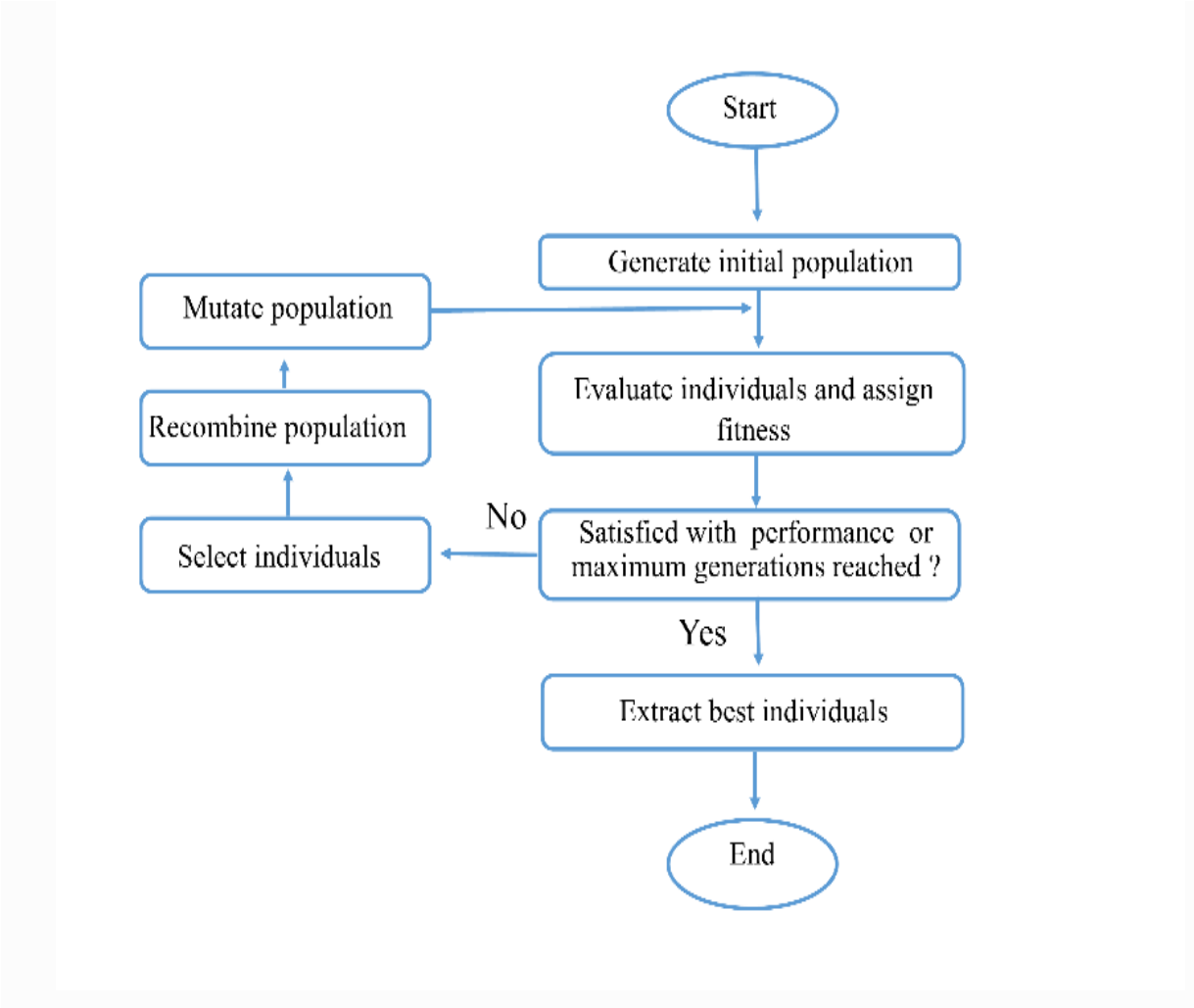


Figure 3-11. Genetic algorithm flowchart.

3.9.3. Multi-Objective Genetic Algorithm (MOGA)

Genetic algorithms are well-suited tools for dealing with multi-objective optimization tasks [304], [305]. Within the MOGA, each individual in the population is assessed in the space of multiple objectives rather than a single objective. Moreover, rather than generating a solution that is superior to all others, a set of Pareto solutions is proposed at the end of the MOGA cycle. Indeed, A parallel search is conducted in the space bounded by the objectives thanks to the behavior of the population-based genetic algorithm. The population of potential solutions to the challenge is produced by the succession of generations. The challenge consists in assigning a fitness value to express how well a solution has already optimized the specified objectives [306]. The simplest method is to use the weighted sum strategy. In this method, weight is given to each objective before the research process begins. The weights reflect the relative importance of the different objectives. This method has some disadvantages, such as the possibility of an incorrect search due to inadequate weights. On the other hand, finding the right weights requires many runs of the optimizer since it is only possible to establish the optimal weights after some

search. Furthermore, small adjustments to the weights can result in large adjustments to the objective values, and conversely.

An effective Pareto-based approach for ranking was suggested in [307]. In this approach, the ranking is based on the number of individuals that dominate each individual. This means that the number of other individuals that dominate it determines the individual's rank. For instance, if any other individual does not dominate an individual, its rank will be 0. If two other individuals dominate it, its rank will be 2, and thus if it is dominated by k individuals, its rank will be k. The concept of Pareto ranking for two goals is presented in figure 3-12.

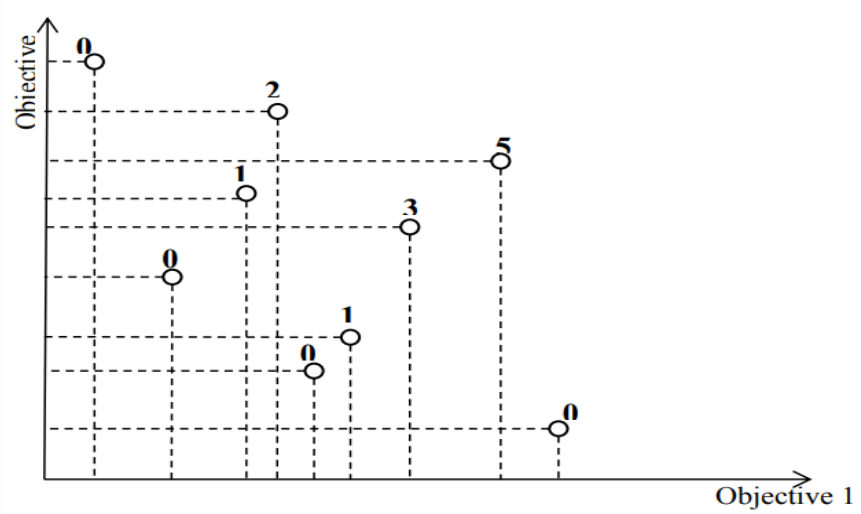


Figure 3-12. Pareto ranking [308]

It is common in most applications to establish multiple or varying priorities for the objectives that the MOGA seeks to fulfill. If all the objectives have the same importance, those who satisfy all of them will have corresponding ranks that are proportional to the number of individuals who dominate them. Those who do not fulfill certain objectives are penalized with a higher rank. A Pareto-based ranking example for two goals with the same priorities is presented in figure 3-13.

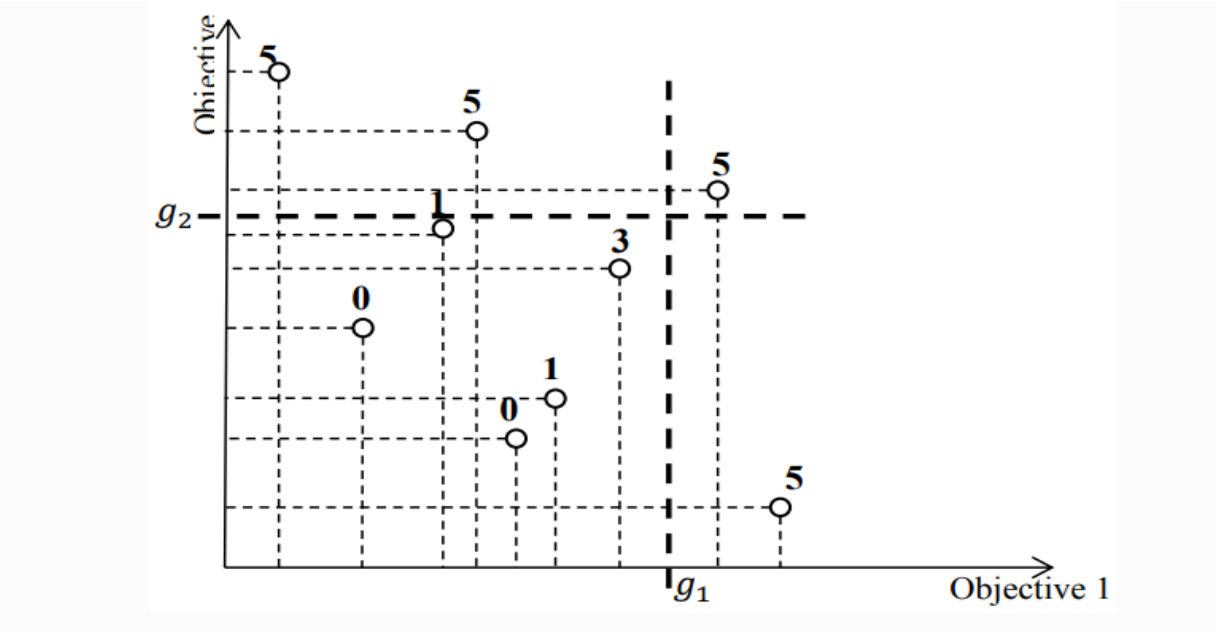


Figure 3-13. Pareto-based ranking [307]: both objectives have the same priority and must fulfill the defined goals.

Assuming that the priority of objective 2 is higher than the priority of objective 1, individuals satisfying objective g2 are ranked according to their ability to optimize objective 1. In contrast, other individuals failing to satisfy objective g2 are assigned the worst rank, regardless of their accuracy in optimizing objective 1. Figure 3-14 present an illustration of a Pareto-based ranking for two goals, with objective 2 being given a higher priority than objective 1.

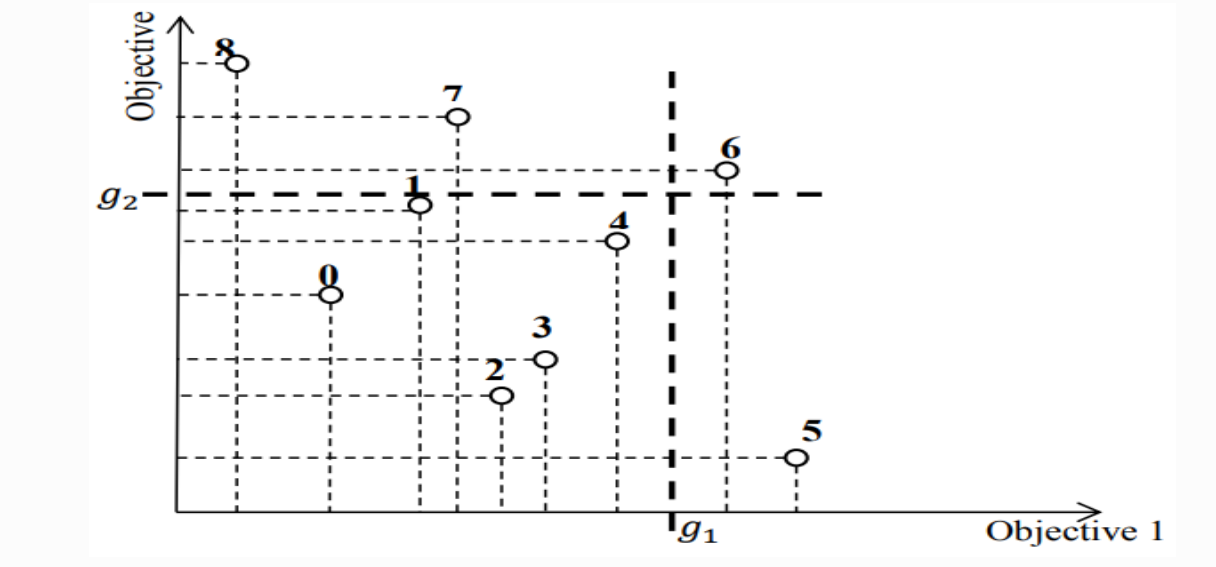


Figure 3-14. Pareto-based ranking [307]: objective 2 being given a higher priority than objective 1and both objectives must satisfy the specified goals.

Following the ranking of individuals, the MOGA attributes a fitness value to each individual according to its associated rank. Then, the individuals are arranged according to their rank, and their fitness value is estimated based on a linear or exponential relationship from the best individual (i.e., rank=0) to the worst. For individuals with the same rank, a single fitness value is computed using an averaging approach. The average value is assigned to individuals with the same rank, to ensure the same likelihood of being chosen as the next generation's parent [306]–[308].

3.10. Radial Basis Function Neural Networks designed by Multi Objective Genetic Algorithm (RBFNN-MOGA)

The configuration (parameters and architecture) of the neural network (i.e., those to be established from the data) are often identified in an iterative, ad hoc manner, with the focus on parameter determination. Indeed, the number of possibilities for selecting model structure and inputs is usually very high. Furthermore, the design objectives may involve multiple conflicting goals, making the problem of identifying the model a complex task that requires multi-objective combinatorial optimization [306].

To determine the best possible parameters and structure of the RBFNN, this study exploits the multi-objective NN model identification approach suggested in [296]. The approach also enables us to tackle conflicting objectives that occur at the same time. For example, the need to diminish the complexity of the model and improve its reliability at the same time. Another illustration would be the goal of having a model with a strong generalization capability while maintaining an extremely low error in the training set.

The task of designing a NN based model may be separated into two sub-problems, as described in section 3.8.

- NN structure: It specifies the inputs of the network, the number of hidden layers and the number of neurons in each layer.
- NN parameters: They are usually defined by an appropriate learning process and depend on the model that has been selected.

The MOGA framework is used to design the RBFNN model. The approach combines a genetic algorithm with a derivative-based algorithm. The final outcome of the design process is a set of non-dominated models that have been optimized with respect to multiple conflicting

objectives, which are known as Pareto-optimal solutions. These solutions are obtained after a specified number of iterations set by the user. The derivative algorithm estimates the parameters of each individual model while the genetic algorithm searches the space of inputs and neurons. Specifically, each potential NN topology is formulated as a chromosome. The first component of the chromosome represents the number of neurons in the hidden layer, and the following components represent the indices of a selected number of features from the preliminary feature space. Figure 3-15 illustrates the topology of the chromosome.

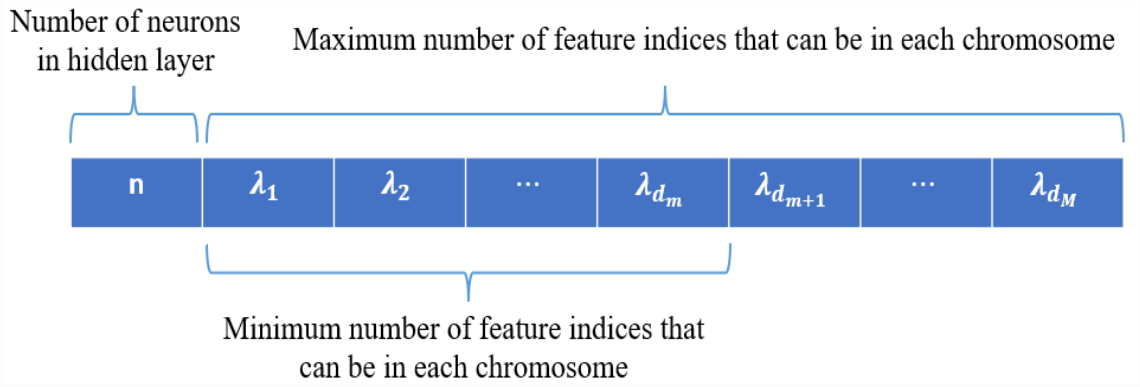


Figure 3-15. The topology of the chromosome

The parameters of each model are estimated using a modified version of the Levenberg-Marquardt (LM) approach [309]. It takes advantage of the linear/nonlinear separability characteristic of the NN, which results in high accuracy and a fast convergence rate. The nonlinear parameters (C and σ) initial values (see Equation (65)) are either selected randomly, or by using a clustering algorithm, W is established by a linear least-squares solution, and the process is terminated using the early stopping method within a maximum number of iterations [282].

Assuming D = (X, y) is a set made up of N input-output pairs, which is divided into three parts: the training set D_{tr}, the test set D_{te}, and the validation set D_{va}. Assume that L is a set of all possible input characteristics. The design parameters given to MOGA include the dataset, the allowed range of hidden neurons $n \in [n_m, n_M]$, and the input feature range $d \in [d_m, d_M]$ from L. Afterwards, MOGA develops a non-dominated set of RBFNN designs minimizing $[\mu_p, \mu_s]$, where μ_p and μ_s represent a set of objectives with respect to the parameters p of the RBFNN and its structure, respectively. The only objective in μ_s is generally the model complexity, which depends on the number of hidden neurons and the number input features.

$$\mu_S = [O(\mu)] \tag{79}$$

Equation (80) formulates the complexity of the model (O(μ)):

$$O(\mu) = \mu_1 \times (\mu_2 + 1) \quad (80)$$

Where μ_1 represents the number of neurons in the hidden layer and μ_2 is the number of input features.

The objectives of μ_p vary depending on the specific problem being addressed. Different problems may have different requirements and goals, leading to different objectives for the optimization process.

3.10.1. Objective of μ_p in the classification problem

For classification tasks, the objectives to be minimized are mainly the false positives (FP) and false negatives (FN) criteria. Hence, the associated objectives for μ_p are given as follows:

$$\mu_p = \left[FP_{D_{tr}}, FN_{D_{tr}}, FP_{D_{te}}, FN_{D_{te}}\right] \quad (81)$$

Where $[FP_{D_{tr}}, FP_{D_{te}}]$ represent the false positives and $[FN_{D_{tr}}, FN_{D_{te}}]$ the false negatives for training and testing set, respectively.

3.10.2. Objective of μ_p in the estimation task

In the case of estimation tasks, the objective of μ_p relies on lowering the error between the model outputs and the target values. A nonlinear autoregressive configuration with eXogenous inputs (NARX) is employed. Hence, the corresponding objectives for μ_p are given as follows:

$$\mu_p = [\varepsilon(D_{tr}), \varepsilon(D_{te}), \varepsilon(D, PH)] \quad (82)$$

Where $\varepsilon(D_{tr})$ and $\varepsilon(D_{te})$ stand for RMSE of the model with respect to the training set D_{tr} and the testing set D_{te}. $\varepsilon(D, PH)$ refers to the forecast error. It is calculated by summing the RMSEs over the prediction horizon (PH):

$$\varepsilon(D, PH) = \sum_{i=1}^{PH} RMSE(E(D, PH), i) \quad (83)$$

$E(D, PH)$ is an error matrix defined over the simulation set D, as described in Equation (84), where D consists of a number of consecutive samples with regard to each time instant.

$$E(D_s, PH) = \begin{bmatrix} e[1,1] & e[1,2] & \cdots & e[1, PH] \\ e[2,1] & e[2,2] & \ddots & e[2, PH] \\ \vdots & \vdots & \cdots & \vdots \\ e[p - PH, 1] & e[p - PH, 2] & \ldots & e[p - PH, PH] \end{bmatrix} \quad (84)$$

Where $e[i, j]$ represents the prediction error of the model observed at time i from D to step j over the prediction horizon PH.

3.11. Data selection approach

Artificial neural networks approaches are well known learning techniques for addressing classification and regression problems. In fact, the models designed by these techniques are data-driven and their performance is therefore affected by the quality of the training data set. The data should be appropriately selected to cover the entire input range for which the model will be applied. In this section we will present the data selection approach used. It is based on a randomized convex hull approximation algorithm, Approxhull introduced in [255]. The method can handle high dimensional data efficiently in a reasonable runtime and minimal memory demands thus facilitating the model design.

3.11.1. Convex hull definition

In Euclidean space, an object is convex if for every couple of points inside the object, each point on the straight-line segment connecting them is also inside the object. The convex hull of a set of data *S* is the smallest convex set (area) that includes S. A set *S* is convex if, for each couple *(x, y)* $\in S$, and any t $\in$ [0,1], the point *(1 − t) x + ty* is in *S*. Moreover, if *S* is a convex set, for any $u_1, u_2, \cdots, u_j \in S$, and any nonnegative numbers $\{\beta_1, \beta_2, \cdots, \beta_j\}$: $\sum_{i=1}^{j} \beta_i = 1$, the vector $\sum_{i=1}^{j} \beta_i u_i$ is referred as convex combination of $u_1, u_2, \cdots, u_j$. Figure 3-16 presents an illustration of convex and non-convex sets.

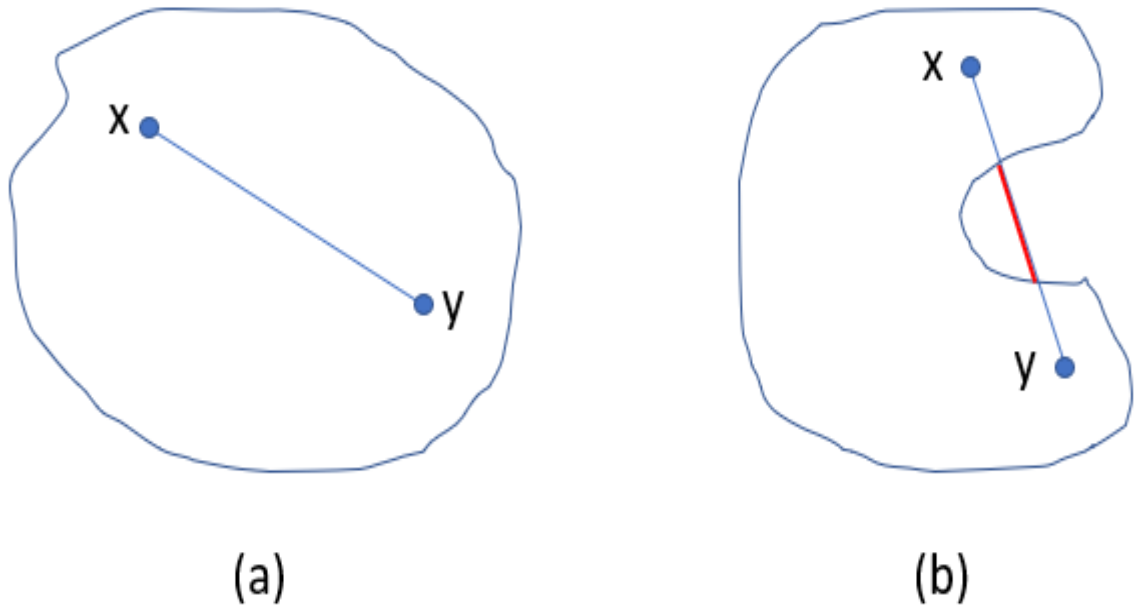


Figure 3-16. (a) Convex, (b) Nonconvex

From the given definitions, the convex hull of a set S of points in Euclidean space can be described in terms of convex sets or combinations of convex [255], [257]:

- the minimal convex set containing S, or
- the intersection of all convex sets containing S, or
- the set of all convex combinations of points in S.

From the convex hull definition, a k-simplex is a polytope of dimension k that is the convex hull of k + 1 affinely distinct points. Basically, a point, a line segment, a triangle and a tetrahedron correspond to 0-simplex, 1-simplex, 2-simplex, and 3-simplex respectively. In principle, a k-simplex is made up of i-faces, where $i \leq k - 1$. Each data sample in the convex hull set is referred to as a convex vertex or a convex point. Given a convex hull, it can be defined using vertices and facets, with vertices denoting the boundary points of the data set and facets denoting the connections between vertices. The facet's dimension is the same as the dataset's dimension. Figure 3-17 illustrates the vertices and facets of the convex hull of a set of points.

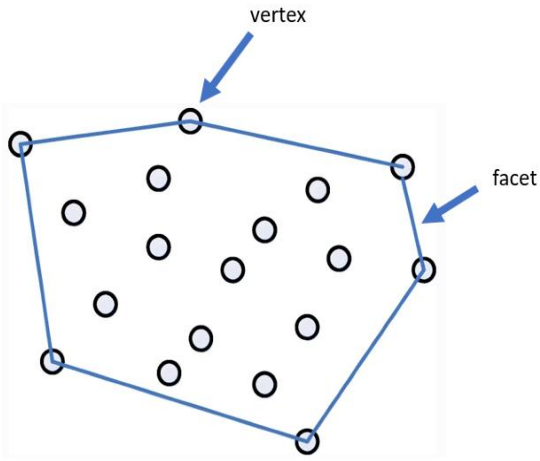


Figure 3-17. Vertices and facets of the convex hull of a set of points

3.11.2. Approxhull algorithm

An efficient data selection method that enables processing of large datasets by using an approximation approach based on randomized convex hulls. The algorithm reduces the computational time and memory usage, making it an ideal solution for handling large data [255]. The algorithm uses hyperplanes and convex hull distances from computational geometry to efficiently handle large datasets by approximating the real convex hull and finding informative vertices. This helps to reduce the memory complexity and processing time of the

data selection process [257]. In the following sections, we will first explain these two notions of computational geometry, before tackling ApproxHull algorithm.

3.11.2.1. Hyperplane Distance

Assuming $O = [O_1, O_2, \cdots, O_n]^T$ is a point, f is a n-vertex facet (each facet of n-dimensional convex hull entails exactly n vertices) and H is the corresponding hyper-plane of facet f in a n-dimensional Euclidean space. The equation of the hyperplane H is defined as follows:

$$\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n + \beta = 0 \tag{85}$$

Where $N = [\alpha_1, \alpha_2, \cdots, \alpha_n]^T$ denote the normal vector and β is the offset of H.

The distance from the point O to the hyperplane H can be calculated as follows:

$$d_s(O, H) = \frac{\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n + \beta}{\sqrt{{\alpha_1}^2 + {\alpha_2}^2 + \cdots {\alpha_n}^2}} \tag{86}$$

3.11.2.2. Convex Hull Distance

Considering a set $S = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$ and a point $x \in \mathbb{R}^d$, the Euclidean distance between x and the convex hull of S, represented by $conv(S)$, may be calculated by addressing the following quadratic optimization problem:

$$\min_a \frac{1}{2} a^T Q a - c^T a \tag{87}$$

$$s.t. e^T a = 1, a \geq 0$$

where $e = [1,1,\cdots,1]^T$, $Q = X^T X$ and $c = X^T x$, with $X = [x_1, x_2, \cdots, x_n]$.

Assume that the optimal solution of (87) is a^*; then the distance of point x to $conv$ (S) is given by:

$$d_c\big(x, conv(S)\big) = \sqrt{x^T x - 2c^T a^* + a^{*T} Q a^*} \tag{88}$$

3.11.2.3. Algorithm

ApproxHull is an incremental algorithm that starts by constructing an initial convex hull, and then gradually extends it by incorporating new vertices in each iteration. It is characterized by two main properties that enable it to overcome memory and time complexity constraints in high

dimensions. It is primarily an approximation algorithm that extracts a subset of the most informative vertices from the real convex hull considering a pre-defined threshold. Moreover, the ApproxHull convex hull is given only by the vertices. The algorithm computes the hyperplane distance of each sample to the facets of the real convex hull in order to identify the vertices of the real convex hull. It involves five main steps [255]. The first step involves normalizing each dimension to the range [-1, 1]. In the second stage, the maximum and minimum samples for each dimension are determined. These samples are regarded as vertices of the initial convex hull. Step 3 generates a population of k facets using the current convex hull vertices as a baseline. Each iteration involves checking the validity of all formed facets. A facet made up of n points in n dimensions is considered to be valid if the matrix representing the facet has full rank. Two tasks are implemented to ensure that the population comprises valid facets. In the first stage, the validity of a facet of the population is verified when n vertices of the current convex hull are chosen to form the facet. Invalid facets are disregarded and substituted with others set of n vertices from the current convex hull until a valid facet is identified. In the second stage, there is a possibility that invalid facets may be generated iteratively; the jogging technique may be used as an optional step in the data preprocessing phase to minimize the time spent ignoring invalid facets and creating valid substitutes. In computational geometry, input jogging is employed to address the accuracy error. For overcoming the problem of coplanar points that may generate invalid facets, jogging (also known as random perturbation) consists of changing the input of each cell in the dataset by a small random quantity (negative or positive). In the fourth step, the algorithm identifies the farthest points of each facet in the current facet population and considers them as new vertices of the convex hull, as long as they have not been discovered previously. First, the equation of the hyperplane corresponding to the facet given by Algorithm 1 in terms of offset and normal vector is used to find the furthest points. Afterwards, the hyperplane distance of the samples from the corresponding hyperplane is calculated by equation (86). Actually, Algorithm 1 relies on the fact that the distance between the center point and the hyperplane is negative to calculate the equation of the corresponding hyperplane of the facet. It is in the form $\alpha x = \beta$ where α and β are the normal and the offset of the hyperplane equation, respectively.

ALGORITHM 1: OBTAINING THE CORRESPONDING HYPERPLANE OF A FACET

Input: $\boldsymbol{DS} = \{\boldsymbol{x_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{n}} \subseteq \mathbb{R}^{\boldsymbol{d}}$ as a set of samples and $\boldsymbol{F} = \{\boldsymbol{\gamma_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{d}}$ as a particular facet so that $\boldsymbol{\gamma_i}$ is a row vector which represents a specific sample in DS.

1. Let c be a row vector that represents the center point of all samples in DS.
2. $\boldsymbol{U} = \{\boldsymbol{u_i} | \boldsymbol{u_i} = \boldsymbol{\gamma_i} - \boldsymbol{c}\}_{\boldsymbol{i=1}}^{\boldsymbol{d}}$
3. $\alpha = \{\}$
4. $\beta = \{\}$
5. $\alpha = \boldsymbol{U^{-1}e}$ where e $[\boldsymbol{1}, \boldsymbol{1}, \cdots, \boldsymbol{1}]^{\boldsymbol{T}}$
6. $\beta = 1 + c\alpha$
7. $t = \sqrt{\sum_{\boldsymbol{i=1}}^{\boldsymbol{d}} \boldsymbol{a_i^2}}$ where $\boldsymbol{a_i} \in \alpha$ for i=1, 2, …, d
8. $n\alpha = \left\{\frac{\boldsymbol{a_i}}{\boldsymbol{t}} | \boldsymbol{a_i} \in \boldsymbol{\alpha}\right\}_{\boldsymbol{i=1}}^{\boldsymbol{d}}$
9. $n\beta = \frac{\boldsymbol{\beta}}{\boldsymbol{t}}$

Output: $n\alpha$ and $n\beta$.

Finally, in step 5, the current convex hull is updated by integrating newly identified vertices into the current set.

Steps 3 to 5 are executed iteratively until one of the two termination criteria is satisfied:

- ✓ There are no newly identified vertices in Step 4
- ✓ If there are new vertices as a result of step 4, and the difference between the maximum and minimum of dc (with dc being the maximum of the approximate distances of the furthest points of the current convex hull at each iteration) over the last w iterations is less than a user-defined threshold β (default value 0.1) and there is fluctuation in value of dc in this w-sliding window, the algorithm ends.

The main concept of the second criterion is to avoid generating new vertices that are very close to the current convex hull, thus not providing new information in a meaningful way. The dc shows a decreasing trend over the iterations, as the convex hull formed by ApproxHull grows iteratively. Therefore, when the difference between the maximum and minimum of dc over the last w iterations is not large, the newly detected vertices can be discarded and ApproxHull can be ended.

Since calculating the distance of a point to the current convex hull by resolving the quadratic optimization problem stated in equation (87) is complex and time-consuming in high dimension, the approximate distance of a newly found vertex to the current convex hull is calculated based on 2*d vertices in the current convex hull that are the nearest neighbors of the

newly found vertex, where d refers to the dimension. The Approxhull algorithm is summarized in Algorithm 2 [257].

ALGORITHM 2: APPROXHULL ALGORITHM

Input: $\boldsymbol{DS} = \{\boldsymbol{x_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{n}} \subseteq \mathbb{R}^{\boldsymbol{d}}$ as a set of samples, k represents the population size of facets in d-dimensional space and w is an integer value as width of the sliding window.

Scale each dimension of DS to the range [-1, 1]
Let V represents the maximum and minimum samples with regards to each dimension in DS;
NotFound = False; Diff = False; iteration = 1; DC = {};
While (not NotFound and not Diff) **do**
    Let P be an empty population.
    For (i = 1; i ≤ k; i + +) **do**
        Let F be an empty facet. j = 1;
        While ($j \leq d$) **do**
            Select randomly a vertex v from V;
            If (v is not in F) **then**
                F = F ∪ {v}; $j = j + 1$
        $P = P \cup \{F\}$;
    new$V$ = {};
    For each facet F in P **do**
        Let FP be the furthest points to facet F.
        For each point $\boldsymbol{f_p}$ in FP **do**
            If ($\boldsymbol{f_p}$ is not in V) **do**
                new$V$ = new$V$ ∪ **{$\boldsymbol{f_p}$ };**
    If (new$V$ = {}) **then**
        NotFound = True;
    If (not NotFound) **then**
        Let dc be the maximum of the approximated distances of vertices in new$V$ to the current convex hull.
        DC = DC ∪ {dc}
        If (iteration ≥ w) **then**
            Let $\boldsymbol{dc_{min}}$ be the minimum of dc in DC over w last iterations.
            Let $\boldsymbol{dc_{max}}$ be the maximum o of dc in DC over w last iterations.
            If (fluctuating observed in value of dc over w last iteration and ($\boldsymbol{dc_{max} - dc_{min}}$)< 0.1) **then**
                Diff = True**;**
            Else
                $V = V \cup \{$new$V$ };
    iteration = iteration + 1

Output: V

The process results in the generation of training, testing, and validation sets. Figure 3-18 illustrates a flowchart summarizing the ApproxHull algorithm.

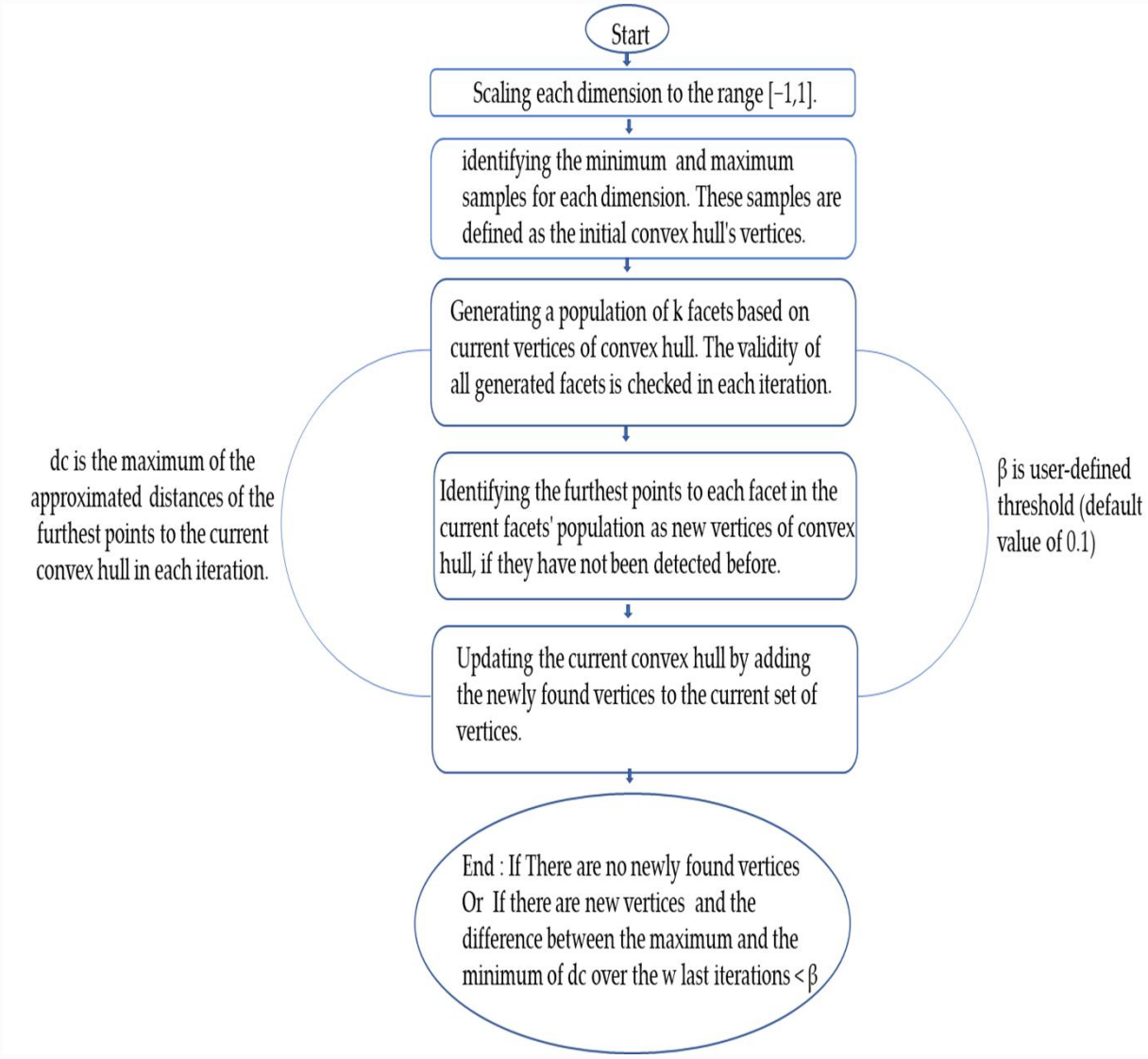


Figure 3-18. ApproxHull algorithm flowchart [161].

Chapter 4 Deep learning approach for non-intrusive load monitoring

4.1.Introduction

This chapter focuses on deep learning-based NILM techniques. In fact, the accuracy of energy estimation and generalization are the most important two capabilities that an NILM algorithm must fulfill to be viable in practical application [198]. Although training deep learning algorithms on a large quantity of data with many characteristics does not necessarily ensure optimal performance owing to potential presence of misleading and unimportant characteristics, however the deep learning models can achieve good generalization and high accuracy if trained on a reasonable amount of data and/or by optimizing the hyper parameters [51], [203]. Moreover, it is possible to achieve low time complexity and good performance with limited data by selecting the most efficient features. Owing to the inefficiency of the selected feature(s), the disaggregation accuracy of specific loads may deteriorate in some cases. Hence, it is necessary to define a comprehensive set of characteristics that can help in identifying all types of devices with a satisfactory performance and minimal generalization error.

To tackle some of these challenges, we propose a hybrid deep learning architecture based on a convex hull data selection technique using active and reactive low frequency power data. The framework is based on a hybrid architecture consisting of two models. A classification model based on a CNN trained with an estimation model based on a bidirectional LSTM. The rest of chapter is structured as follows: Section 4.2 discusses the NILM problem formulation and the proposed optimization approach. Section 4.3 presents the case study house and data collection method. The results of the experiments and a discussion including results from the literature are reported in Section 4.4. Section 4.5 summarizes the chapter.

4.2. Proposed Optimization

The basic objective of energy disaggregation is to break down the aggregate energy of the house into specific appliance contributions. This involves identifying the operating state of each device and assessing its contribution to the overall energy consumption of the household. The problem can be defined as follows: considering a sequence of aggregated data $X_t = \{x_1, \cdots, x_T\}$, the NILM task is to estimate the energy consumption of each specific device i, $y_t^{(i)} = \left\{y_1^{(i)}, \cdots, y_T^{(i)}\right\}$, where $i=\{1, \ldots, N\}$ is the index of the device (from a set of N devices), and t =

{1, …, T} is the time-index of the sequence with length T. The aggregated data can be described as the sum of the contributions of the specific devices plus an unspecified part owing to noise, in the following terms:

$$X_t = \sum_1^N y_t^{(i)} + \partial_t \tag{89}$$

where ∂_t denote the noise term.

The contribution of specific device $y_t^{(i)}$ is defined by:

$$y_t^{(i)} = \psi(X_t) \tag{90}$$

where ψ, is the operator that, when employed to the aggregated data, yields the best prediction of the contribution of each specific target device. The problem of finding an estimate of the operator ψ can be regarded as a supervised learning task. The proposed optimization is based on a supervised deep learning hybrid architecture using a data selection algorithm, introduced in [255]. The hybrid network architecture is composed of two subnetworks inspired by the studies reported in [205], [256]. The method employs a distinct classification subnetwork that is formed jointly with an estimation subnetwork. The output of the two subnetworks is combined to form the network output. With the spreading of parameters over associated tasks, these learning methods enable the models to generalize more efficiently. Furthermore, deep neural networks have a penchant for detecting irrelevant and unrelated appliance activations [60], [198]. The coupled classifier eliminates these irrelevant predictions, thus considering only the predictions where the appliance is active. In the general framework of the proposed optimization each subnetwork is regarded as a subtask of NILM task [256]. On the one side, the CNN model is used to classify the appliance's states. The objective is to determine whether the appliance is turned ON or OFF. The appliance is deemed to be "ON" when its energy usage surpasses a predetermined threshold. The outcome of the classification subnetwork is the state of the appliance $s_t^{(i)} = \{s_1^{(i)}, \cdots, s_T^{(i)}\}$, where $s_t^{(i)} \in \{0, 1\}$ denotes the ON-OFF state of appliance i at time t. The CNN hyper parameters were tuned using the trial-and-error procedure [161]. The architecture of CNN with the best hyper parameters is presented in table 2.

Table 2. Classification subnetwork CNN architecture

Layer index	Layer
1	Input shape (length defined by the appliance data)
2	1D CNN layer (filters=32, kernel size=3, activation = 'ReLu')
3	1D CNN layer (filters=64, kernel size=3, activation = 'ReLu')
4	1D CNN (filters=128, kernel size=3, activation = 'ReLu')
5	Maxpool layer
6	dense layer (activation ='ReLu', number of units = 1024)
7	dense layer (number of units = 1)

The power sequence of the appliance, on the other hand, is predicted using a bidirectional LSTM model. The use of recurrent NN allows the model to achieve a reliable estimation of the energy usage of each device while maintaining the scalability of the network architecture. Minimizing the MSE loss function yields the best LSTM model fit. The following is the regression model's subnetwork architecture with the best parameters:

Table 3. Regression subnetwork bidirectional LSTM architecture

Layer index	Layer
1	Input shape (length defined by the appliance data)
2	Bidirectional LSTM layer (activation = 'ReLu', number of hidden units = 32,)
3	Dropout layer with dropout = 0.3
4	Bidirectional LSTM layer (activation = 'ReLu', number of hidden units = 64,)
5	Dropout layer with dropout = 0.3
6	Bidirectional LSTM layer (number of hidden units=128, activation= 'ReLu')
7	Dropout layer with dropout = 0.3
8	dense layer (activation ='ReLu', number of units = 1024)
9	dense layer (number of units = 1)

Considering $\hat{s}_t^i = f_c^i(X_t)$ where f_c^i denotes the appliance state classification model and X_t is the input of the models. Assume that $\hat{p}_t^i = f_r^i(X_t)$ where f_r^i represent the estimation model. The outcome of the classification sub-network is multiplied by the estimated power values of the estimation sub-network, as shown in (91), to obtain the overall disaggregation output.

$$\hat{y}_t^{(i)} = \hat{s}_t^i \odot \hat{p}_t^i \tag{91}$$

where $\hat{y}_t^{(i)}$ denotes the disaggregation output of device i at time t, $\hat{p}_t^i$ and $\hat{s}_t^i$ represent the regression and classification subnetworks output respectively. Figure 4-1 presents the full architecture of the proposed optimization model.

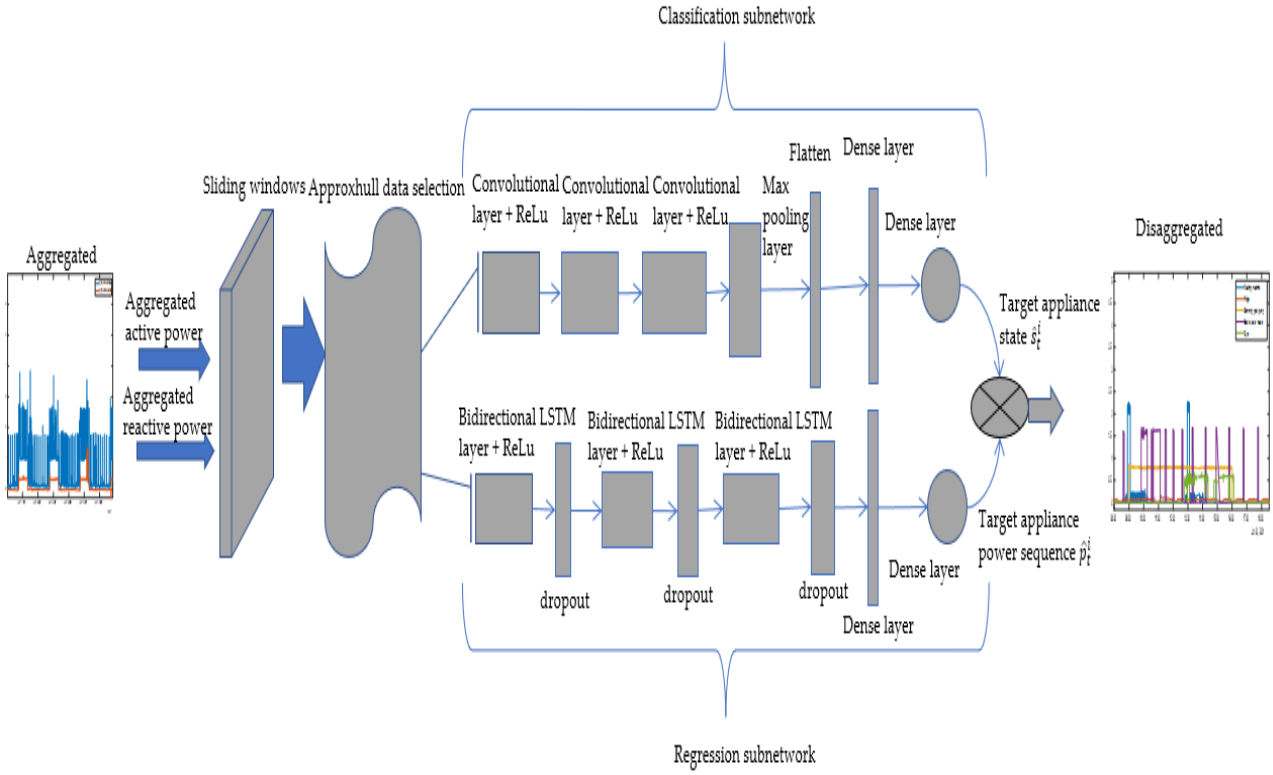


Figure 4-1. Architecture of the proposed optimization model

4.3. Data Collection in the case study house

The study employs data gathered from a residential household located in Algarve region of southern Portugal. It is a single-family household, with two floors and 20 different spaces (such as halls, garden, etc.). The house contains a solar panel installation consisting of 20 Sharp NU-AK panels [270] organized in two strings and with a maximum power of 300 W per panel. The battery is a BYD Battery Box HV H11.5 [271], which has a storage capacity of 11.5 kWh. It is controlled by an inverter called Kostal Plenticore Plus converter (KI) [272]. Figure 4-2 depicts the view of the building's PV system.

(a) (b) (c)

Figure 4-2. Case study house details: (a) photovoltaic panels, (b) inverter, (c) battery.

Actually, the house is equipped with a variety of household electrical appliances, and a json file was generated using the NILM toolkit format [249] to describe the house's rooms and equipment. The electrical panel of the house includes a Schneider panel with one triphasic and 16 monophase circuit breakers. An Intelligent Weather Station (IWS) [273], a few TP-Link HS100 Wi-Fi Smart Plugs (SP) [274], and a few Self-Powered Wireless Sensors (SPWS) are also available in the household for monitoring room climate variables. Following that, a monitoring system for several electrical variables was implemented. A Carlo Gavazzi (EM340) [275] 3-phase energy meter supplies the aggregate data required for the NILM task. The electrical measurement is performed using a 2-wire Modbus RTU connection on the Class X certified Carlo Gavazzi meter. 45 separate electrical variables are supplied by the EM340, measured at 1 Hz. For each circuit breaker, additional electrical variables are measured to supply a ground truth approximation for the NILM task. In this case, the measurement tools are Circutor Wibees (WBs) [276]. They are wireless plug-and-play meters that measure electricity usage. Each one of them supplies a hotspot so that the initial configuration can be performed via a manufacturer's mobile app. The WB automatically transfers the collected data to a free web service from the manufacturer. This functionality may be turned off and the collected data are still available via the Modbus IP protocol or via an internal web interface/service. Since the appliances utilize Hall effect technology, certain calibrations are necessary in order to obtain

accurate readings. For the 16 monophasic circuit breakers and subsequently for each phase of the triphasic, variables including current, voltage, frequency, apparent power, active power, reactive power, power factor, active inductive reactive and capacitive reactive energy are measured every second. The WBs measure 198 variables in total every second. It should be noted that the time instants of the circuit breakers are not the same since the measuring meters are not synchronized.

A Kostal smart energy meter (KEM)[277] and inverter provide information on the variables regarding the energy produced by the PV, stored in the battery and injected to the grid. The inverter also provides measurements of the house's power consumption. The Kostal Intelligent Energy Meter (KEM) and the Kostal Plenticore Plus converter (KI) measure a total of 21 and 47 variables, respectively using a sampling interval of one minute. The Modbus IP protocol is used to access data through a cable IP network. Smart Plugs are utilized for on/off control of certain devices. They also enable separate measurement of sockets that are part of the same Circutor Wibees. These tools utilize an initial access point and a mobile app from the manufacturers to connect to an already-established wireless network. A cloud API or a direct internal web service might be used to read and control them. Presently, 4 Smart Plugs are used in the house, allowing 6 variables to be measured for each plug every second.

The Intelligent Weather Station (IWS) is used to monitor relative humidity, air temperature, and global solar radiation and forecast their changes over a self-defined time period. A two-step method is used to make these predictions. The nearest neighbor technique is used when minimal measurement data are available while neural network based predictive models are automatically built offline and loaded onto the IWS for real-time use when sufficient data are available. For further information about the operation of the IWS, please see [273].

The Self-Powered Wireless Sensors (SPWS) measure room data including relative humidity, air temperature, door status, and window status (open or closed), light, walls temperature, and room movement. They are Ultra-Low-Power equipment that uses the 2.4 GHz or 868 MHz ISM radio band to communicate. They are intended to measure thermal comfort for use in developing predictive control of certain air conditioners in the household. More details regarding the use of SPWS for HVAC predictive control can be found on [278].

Data transfer from and to the measurement equipment is handled by Gateways and a technical network. A wireless network and a technical IP cable network were established using a network router installed in an extension of the electrical panel box. The home network is separated by the router from the technical network. All the equipment is connected to this network, except for the SPs and SPWS. EasyGateway [279] are utilized to perform data

acquisition from existing appliances. It is a fault tolerant IoT gateway that supports a range of receive/acquire protocols such as SNMP, Modbus, serial http, and Easy modules, as well as a set of data delivery connectors (DDCs) generally used in IoT environments like AMPQ and MQTT. Five of these gateways are employed inside the electrical panel box to ensure the necessary acquisition rates, and the readings from the EM340, WB, KI, and KEM have been distributed within and between them. There is also a separate internal EasyGateway for the weather station. In order for the SPWS to communicate, an additional EasyGateway is placed in a centralized position in the residence. Since EasyGateways always operate at a one-minute rate, the data collected by the measurement equipment attached to each EasyGateway are packed and transferred at that rate.

The gateways can provide information to up to three separate DDCs. In this case, two DDCs and two IoT platforms are deployed. One platform employing an EasyMqs DDC, is used inside the house, and another, employing a Generic AMPQ DDC, in the cloud. The same IoT platform is used within the cloud and inside the house. It collects information from the configured message queue servers. A set of plug-ins configured for each type of entity on the platform is used to pass the arrived data. A comprehensive description of the IoT platform can be found in [279].

The system includes a web page where the end user can configure a set of specifications for data management and storage. The data may be downloaded in 4 common formats: csv, xlsx, mat, and npz. It also provides data visualization using graphs organized by sensor category. Figure 3-9 illustrates a diagram of the data acquisition system.

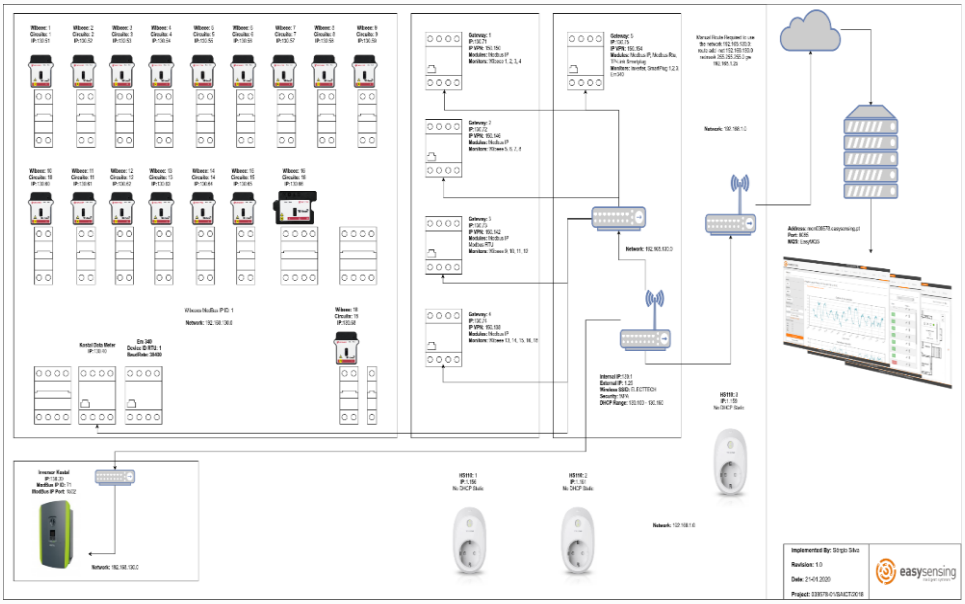


Figure 4-3. diagram of the data acquisition system [280].

The distribution of some appliances with respect to the circuit breakers of the house is shown in table 4.

Table 4. Distribution of appliances through circuit breakers

Breaker #	Devices
1	Alarm
2	Swimming Pool Pump (utility,4), Incandescent Lamp (outdoors), Incandescent Lamp (outdoors), Linear Fluorescent Lamp (utility,4), LED lamp (outdoors), LED lamp (outdoors), Linear Fluorescent Lamp (outdoors), Hair Dryer (outdoors), LED Lamp (utility,5)
3	LED Lamp (bedroom,4), Compact Fluorescent Lamp (bedroom,2), Compact Fluorescent Lamp (bedroom,3), LED Lamp (bathroom,2), LED Lamp (bathroom,2), LED Lamp (hall,2), LED Lamp (hall,2), Compact Fluorescent Lamp (hall,2), Compact Fluorescent Lamp (study), Compact fluorescent lamp (study), Compact Fluorescent Lamp (study), Compact Fluorescent Lamp (study), Compact Fluorescent Lamp (study)
4	Linear Fluorescent Lamp (garage), Compact Fluorescent Lamp (hall,1), Compact Fluorescent Lamp (hall,1), LED Lamp (bathroom,1), LED Lamp (bathroom,1), Compact Fluorescent Lamp (bedroom,1), Compact Fluorescent Lamp (bedroom,1), Running Machine (bedroom,1), Clothes Iron (bedroom,1), Electric Air Heater (bathroom,1), Hair Dryer (bathroom,1)
5	Air Conditioner (study)
6	Motor_1 (garage), Motor_2(garage)
7	Air Conditioner (bedroom,3), Air Conditioner (bedroom,2)
8	Electric Water Heating, Washing Machine, Broadband Router, Dryer Machine (utility, 2),
9	Dish washer (kitchen), Microwave (kitchen), Fryer (kitchen), Kettle (kitchen)
10	Air Conditioner (bedroom,4)
11	Fridge Freezer (kitchen), Coffee Maker (kitchen), Toaster (kitchen), Food Processor (kitchen), Small Cooking Device (kitchen), Television (kitchen), Fridge Freezer (hall, 2), LED Lamp (outdoors), Compact Fluorescent lamp (hall, 2), Compact Fluorescent Lamp (utility, 1)

	Compact Fluorescent Lamp (utility, 2), Compact fluorescent lamp (dining room)
12	Compact Fluorescent Lamp (lounge), Television (lounge), Audio System and Set top box (lounge)
13	Compact Fluorescent Lamp (bedroom,4), Compact Fluorescent Lamp (bedroom,4), LED Lamp (bedroom,4), Compact Fluorescent Lamp (bedroom,3), Compact Fluorescent Lamp (bedroom,2), Broadband Router (bedroom, 2), Electric Air Heater (bathroom,2), Electric Air Heater (hall,3), Immersion Heater (study), Aquarium Pump (study), Compact Fluorescent Lamp (study), Desktop Computer (study), Laptop Computer (study), Laptop Computer (study), Television (study), audio system (study), Printer (study), Broadband router (study)
14	Air Conditioner (lounge)
15	Linear Fluorescent Lamp (kitchen), Compact Fluorescent Lamp (lounge), Compact Fluorescent Lamp (lounge), LED Lamp (outdoors)
16	Burner Stove_1 (kitchen)
17	Oven (kitchen)
18	Burner Stove_2 (kitchen)
19	Data acquisition system

Data are being collected since November 2019 totaling over 2 years of data. The aggregated data are collected using the three-phase EM340 energy meter. The distribution of devices associated with each phase according to their circuit breakers is presented in table 5.

Table 5. Distribution of devices through EM340 phases

Phase	Breakers #
I	1, 4, 7, 10, 13, 16
II	2, 6, 8, 11, 15, 17
III	3, 5, 9, 12, 14, 18

The distribution of energy consumption in the household during the period from January 2020 to December 2021 (2 years of data) in terms of phases is presented in figure 4-4.

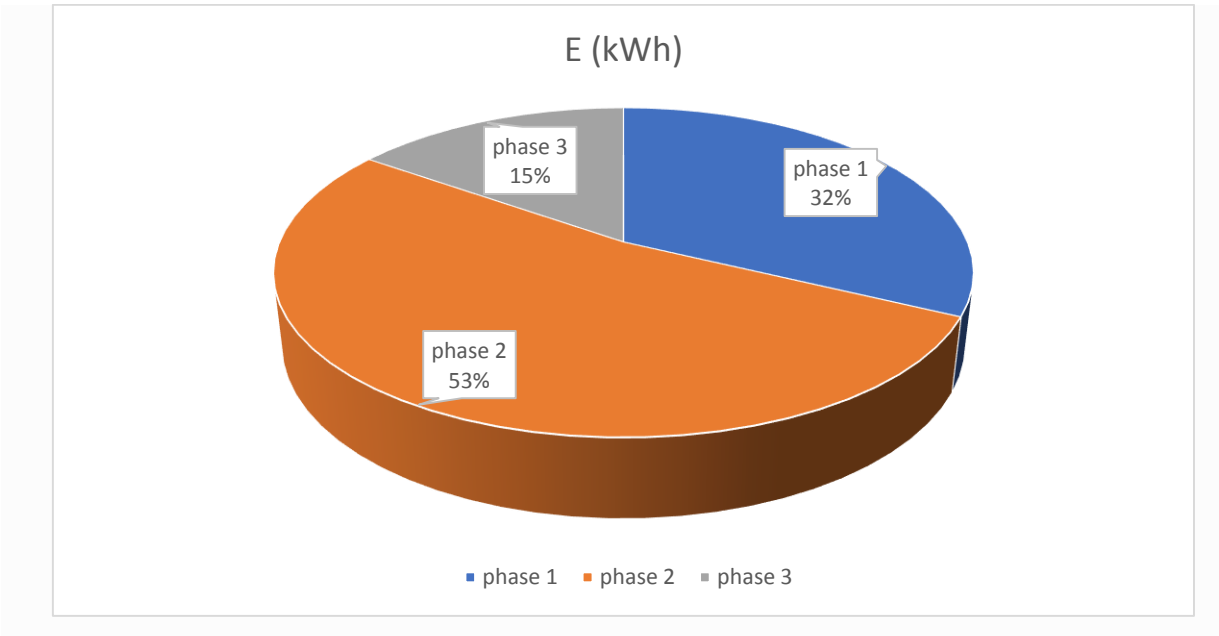


Figure 4-4. Distribution of energy consumption in the case study house.

4.4. Experiments and Results

4.4.1. Data Pre-Processing

Data were gathered over several weeks in the household presented in Section 4.3 using a 1 second sampling rate. Although only 1 month of data is considered in this experiment, each device actually accounts for roughly 2.4 million samples. The EM340 phase 2's aggregated active and reactive powers readings are retrieved. A sliding window of 20 variables is created at each time point, 10 of which are lagged active power values and 10 lagged reactive power values. The ApproxHull algorithm outlined in Section 3.5.1 receives these data along with the label (device ON/OFF) if a classification is desired or the current active power if an estimation is sought. Four relevant appliances (swimming pool pump, washing machine, Fridge, and electric water heater) are considered to evaluate the proposed optimization. These appliances are among the main consumers in the case study household. Each device's actual power series is extracted manually using data from the smart plugs and Circutor Wibee readings. An approach similar to that employed in NILMTK [249] is used to define the ON-OFF state of each appliance. Basically, the device is deemed ON when its energy consumption surpass a predefined threshold value for a minimum period of time. In fact, for certain multi-state devices, the power consumption may decrease below the threshold value for a short period of time without being turned OFF. In this case a minimum time threshold is predefined during which the power consumption is maintained below the threshold. If the period of time during which the power does not exceed the power threshold is longer than the predefined time threshold,

then the device is considered OFF. Moreover, the ON state is defined considering strictly consecutive samples above the predetermined threshold, and the data sequences below the predefined time threshold are excluded. The parameters used to create the appliance labels, some statistics regarding the number of activations, the maximum and average activation time, and the total active energy consumed by each appliance during the considered time period are summarized in Table 6.

Table 6. Appliances statistics

Devices	Max Power (W)	Power threshold (W)	Time threshold (s)	Activations	ON duration(s)		Energy (kWh)
					Max	average	
Fridge	200	50	60	708	7045	2100	36.1
WM	2500	20	3	17	8696	2250	9.5
EWH	1700	1200	3	663	10169	561	122.9
SPP	1200	500	3600	28	25202	24985	159.1

WM: Washing machine, SPP: Swimming pool pump, EWH: Electric water heater.

The entire dataset is processed using the ApproxHull algorithm to create training, testing, and validation sets for each device (1 month of data sampled at second interval, totaling around 2.4 million samples). A separate dataset is generated for each device. Following that, the convex points covering the entire input range where the model is intended to be applied are created. The identified convex points are subsequently incorporated into the training set for each device data. The rest of samples are then randomly shared over the remaining training, testing, and validation sets in the proportions of 60%, 20%, and 20%, respectively. Table 7 displays the average number of vertices found and the dimensions of the training, testing, and validation sets for each device dataset.

Table 7. Size of training, testing, and validation sets.

Appliances	CH Vertices	Training set	Testing set	Validation set
Washing machine	170	1451513	483839	483837
Electric water heating	118	1375956	458652	458652
Swimming pool pump	109	1375956	458652	458652
Fridge	106	1375956	458652	458652

4.4.2. Training and Testing

The studies were carried out considering data from March 1, 2021, to March 28, 2021, for the washing machine and data from June 2, 2021, to June 30, 2021, for the other appliances. A separate model is trained for each appliance. The best hyperparameter settings were obtained through hyperparameter tweaking. The model is trained using Adam's optimization approach [281] with a learning rate of 5e-5 and a batch size of 512. For the classification subnetwork and the estimation subnetwork, respectively, the binary cross-entropy loss function and the MSE loss function were employed. The loss functions of the two subnetworks are combined to obtain the loss function for the design process. To avoid overfitting, early stopping approach [282] was used. This is also referred to as implicit regularization, as it terminates training when the error on the validation set starts to increase. Tables 8 and 9 present the results obtained in the test dataset for state detection and power consumption prediction of each device, both with and without the ApproxHull data selection strategy respectively.

Table 8. Performance evaluation results without ApproxHull data selection approach

Appliance	TP	TN	FP	FN	R	P	F1	MAE (W)	SAE	EA
Washing machine	9872	469495	3959	511	0.95	0.71	0.81	8.90	0.46	0.66
Electric water heating	74424	380307	3107	814	0.98	0.95	0.97	18.5	0.007	0.96
Swimming pool pump	126772	331301	379	200	0.99	0.99	0.99	6.6	0.016	0.98
Fridge	291218	105227	28595	33612	0.89	0.91	0.90	17.5	0.17	0.83

R: Recall, P: Precision, SAE: Signal Aggregate Error, MAE: Mean absolute error, and EA: Estimation Accuracy.

Table 9. Performance evaluation results using ApproxHull data selection approach

Appliance	TP	TN	FP	FN	R	P	F1	MAE (W)	SAE	EA
Washing machine	13034	469737	547	519	0.96	0.96	0.96	1.64	0.050	0.93
Electric water heating	55157	403053	158	284	0.99	0.99	0.99	5.0	0.007	0.98
Swimming pool pump	134803	323699	64	86	0.99	0.99	0.99	4.92	0.002	0.98
Fridge	326212	99553	25862	7025	0.97	0.92	0.95	12.72	0.090	0.88

The results presented in tables 8 and 9 show the reliability of the model in identifying and predicting the energy usage of each appliance. The proposed data selection strategy has considerably improved the performance of the model compared to the approach without data selection strategy, especially for multi-state appliances (fridges and washing machines) which

have complex signature behavior and, therefore, are not easily identifiable (wort F1 score values).

Actually, the data selection strategy enabled to reduce the MAE for the fridge and the washing machine by 27% and 81%, respectively. For the swimming pool pump and the electric water heater, the MAE was lowered by 25% and 73%, respectively. Regarding the F1 score, the identification performance of the washing machine and the fridge were enhanced by 18% and 5% respectively using the proposed data selection strategy. The identification performance of the electric water heater was improved by 2%. In terms of estimation accuracy (EA), the prediction performance was improved by 40% for the washing machine and 6% for the fridge by using the data selection strategy while the performance of prediction of electric water heater was improved by 2%.

The swimming pool pump and electric water heater could be identified with quite good accuracy owing to their basic architecture. Nonetheless, using the data selection strategy, the overall performance showed improvements with slightly higher values. The comparison of the EA and F1 score ((i) without the data selection strategy, and (ii) using the data selection strategy) is shown in Figure 4-5 and Figure 4-6 for all appliances involved in the experiment.

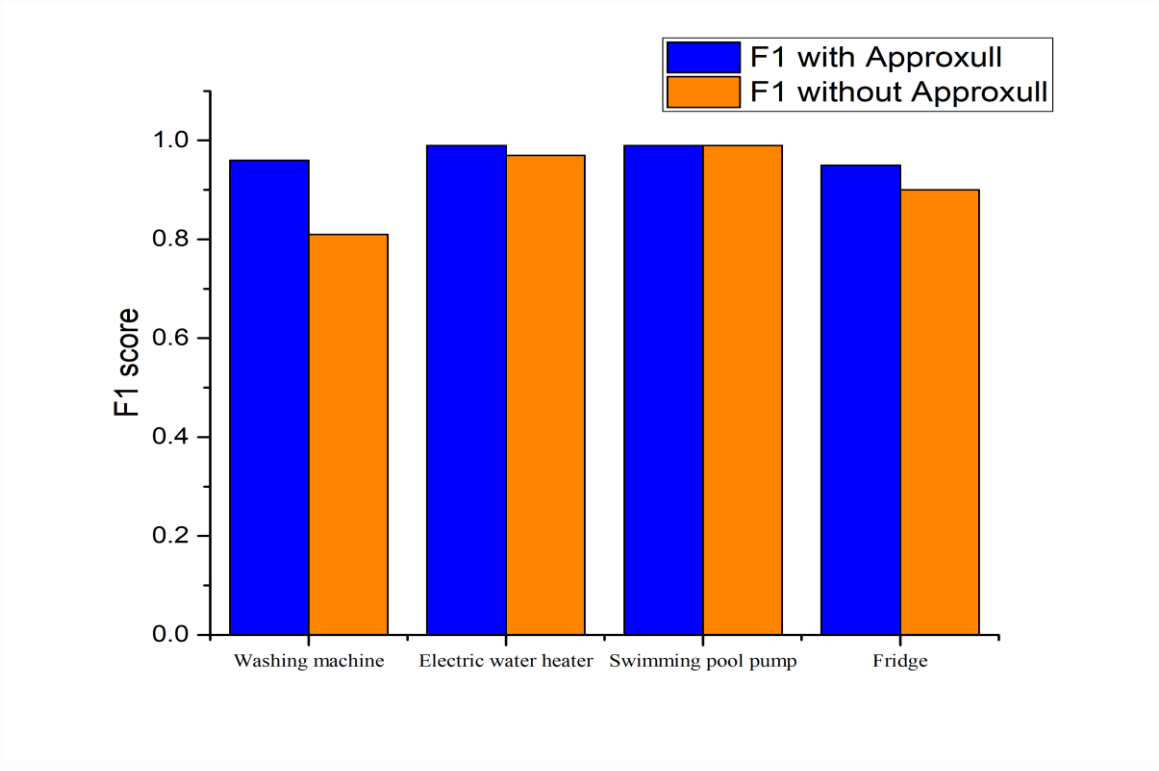


Figure 4-5. Comparison of the F1 score with data selection and F1 score without data selection strategy.

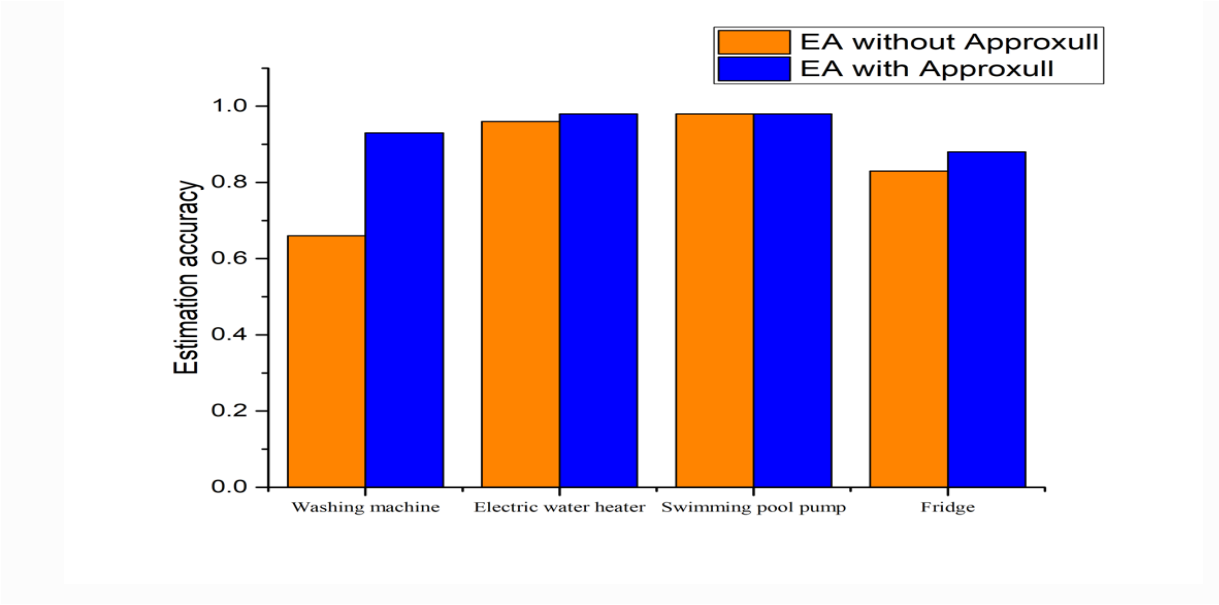


Figure 4-6. Comparison of estimation accuracy (EA) with and without data selection strategy.

The data selection strategy has resulted in a more accurate disaggregation of energy consumption for each appliance, regardless of the complexity of multi-state appliance architectures. Figure 4-7 to figure 4-10 present some examples of the results of the disaggregation. Figure 4-7 compares the predicted active power consumption of the washing machine to its actual active power and the total aggregated active power. Figure 4-8 shows the fridge's predicted active power versus its actual active power. The aggregated active power can be viewed in figure 4-9 and figure 4-10 where figure 4-9 shows the actual, predicted active power for the electric water heater, and figure 4-10 the actual and estimated active power of the swimming pool pump.

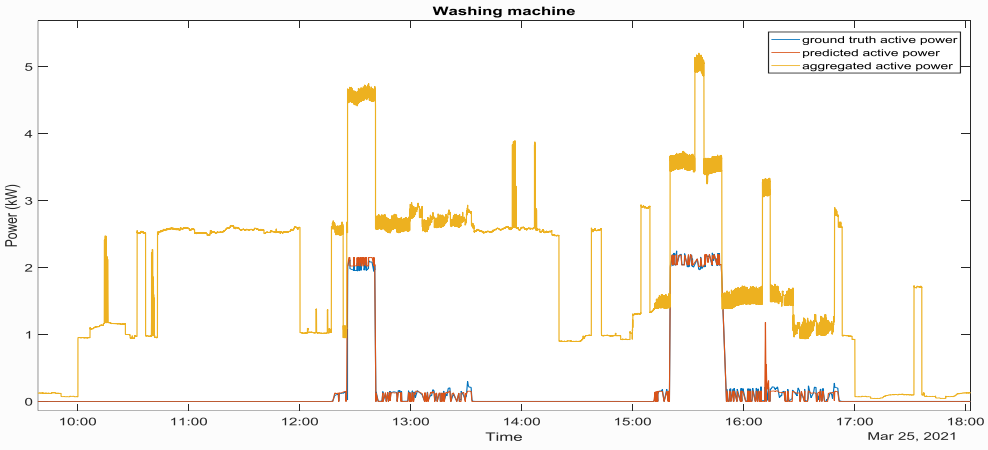


Figure 4-7. Washing machine's disaggregation output. Red, predicted active power; Blue, actual active power; yellow, aggregated active power.

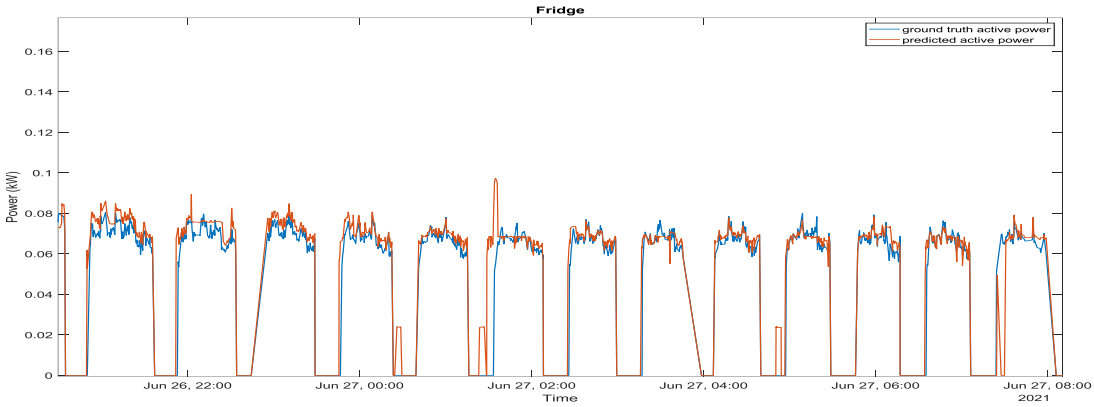


Figure 4-8. Fridge's disaggregation output. Red, predicted active power; blue, actual active power.

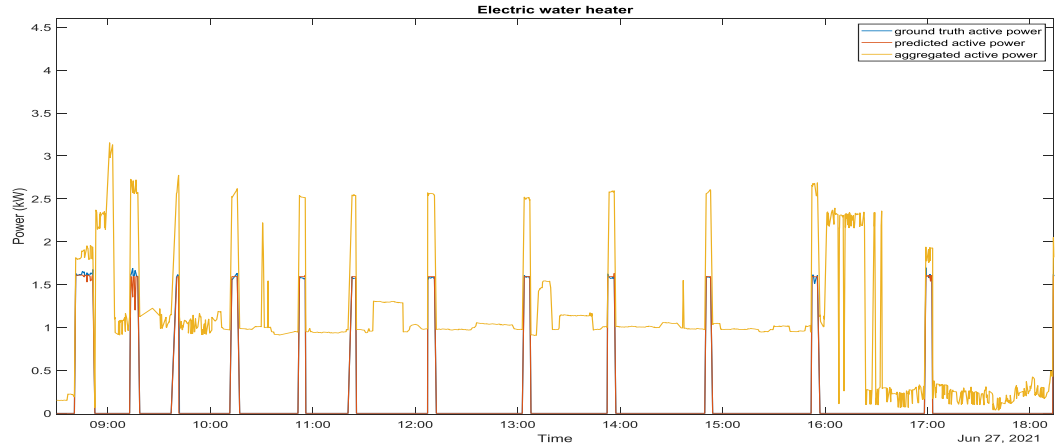


Figure 4-9. Electric water heater's disaggregation output. Red, predicted active power; yellow, aggregated active power and blue, actual active power.

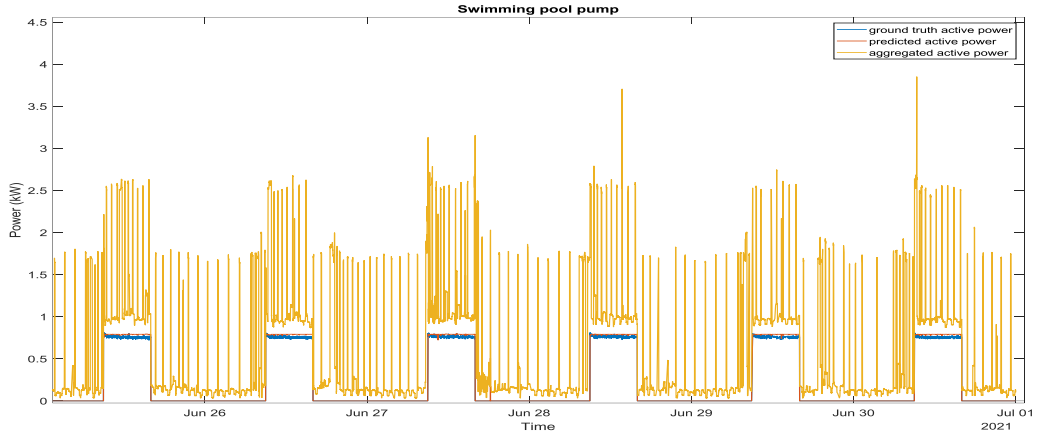


Figure 4-10. Swimming pool pump's disaggregation output. Red, estimated active power; blue, actual active power; yellow, aggregated active power.

Figures 4-7 through 4-10 show that there is a fairly accurate consistency between the predicted and the actual energy consumption for the four devices under study. The predicted and actual active power consumption signals are very close. It is also important to note that each of these devices showed very satisfactory results.

4.4.3. Cross Validation

Five-fold cross-validation was conducted in order to evaluate the reproducibility of the experiment. The design data was divided into five non-overlapping segments of equal size. One segment was used ams the test data to test the model, while the remaining four segments were utilized to calculate the model's parameters and determine its accuracy. Similar to the initial experiment, training was done using 80% of the data, and testing with 20%. The test subset is circulated alternately k times in the process. Table 10 displays the F1 score results of the five-fold cross-validation for the fridge and washing machine.

Table 10. F1 score using Cross validation

k	1	2	3	4	5
Washing machine	0.963	0.961	0.959	0.958	0.959
Fridge	0.944	0.938	0.950	0.934	0.947

4.4.4. Comparison with Other State-of-the-Art Techniques

As discussed in chapter 2 section 2.9, it is difficult to directly compare NILM approaches due to the ambiguity of the performance indicators, which depend on the data and the context of the studies. The methods presented in this experiment employ data gathered from a household using a sampling rate of 1 Hz. Both active and reactive powers are used as inputs in the proposed model design. However, only active power readings are available in the majority of currently publicly available low-frequency NILM datasets [209], [232]. Therefore, a reasonable qualitative comparison of the approaches using different data sets is not feasible. Meanwhile, to illustrate its effectiveness, a comparison was undertaken with state-of-the-art techniques to benchmark the proposed approach. Two commonly used multi-state appliances for assessing NILM approaches are washing machine and fridge. Table 11 contrasts the presented NILM approach to the current state-of-the-art NILM techniques for washing machine. The comparison of the fridge's performance with the state-of-the-art methods can be seen in Table 12.

Table 11. Results of comparison with existing NILM approaches for washing machine.

Method	Recall	Precision	F1	MAE (W)	SAE	EA
Online-NILM [283]	**1**	0.60	0.70	118.11	-	-
MFS-LSTM [198]	-	-	0.76	14.42	0.51	0.74
CNN [284]	0.78	0.20	0.32	18.38	-	-
LDwA [205]	-	-	0.69	11.17	-	-
CNN [60]	0.98	0.87	0.92	-	-	0.92
TP-NILM [50]	0.86	0.87	0.86	8.31	**0.01**	-
Proposed	0.96	**0.96**	**0.96**	**1.64**	0.05	**0.93**

Table 12. Results of comparison with existing NILM approaches for the fridge.

Method	Recall	Precision	F1	MAE (W)	SAE	EA
Online-NILM [283]	0.73	0.87	0.79	**4.34**	-	-
MFS-LSTM [198]	-	-	0.87	19.60	0.46	0.76
CNN [284]	**0.97**	0.80	0.88	7.90	-	-
LDwA [205]	-	-	0.86	19.81	-	-
TP-NILM [60]	0.89	0.85	0.87	17.03	−0.05	-
Proposed	**0.97**	**0.92**	**0.95**	12.72	**0.09**	**0.88**

The results reported in Tables 11 and 12 demonstrate that the proposed framework is quite successful in detecting the operating states of the appliances and predicting the power consumption of each device. The proposed optimization achieved the best performances in terms of MAE (1.64 W), F1 score (96%), precision (96%), and EA (93%) for the washing machine. However, it achieved slightly lower recall (96%) than the CNN (98%) proposed in [60] and the Online-NILM (100%) presented in [283].

The proposed framework for the fridge shows consistent superiority over state-of-the-art methods in terms of precision (92%), recall (97%), estimation accuracy (88%), and F1 score (95%). On the other hand, the MAE of 12.72 W obtained by the proposed framework is slightly worse than the MAE of 4.34 W achieved by the Online-NILM method [283].

From these overall results we can conclude that the proposed optimization approach successfully disaggregates the contribution of specific appliances with a better F1 score and a more reliable EA compared to state-of-the-art works.

4.5. Summary

This chapter described a NILM approach that utilizes a combination of deep learning and a convex hull data selection strategy. The data were gathered from a household located in the region of Algarve, South Portugal.

The first step was to select the data using a random convex hull approximation. The approach consists in selecting the most informative vertices of the real convex hull. Thereafter, the model is designed by combining two subnetwork models. Each subnetwork is considered as a subtask of the overall NILM task. The main architecture relies on a CNN-based device state identification subnetwork model and an energy estimation subnetwork model based on bidirectional LSTM models. Actually, the energy estimation subnetwork tends to estimate some irrelevant energy consumptions that do not belong to the target appliance. Therefore, the usage of a subnetwork for appliance operating state identification enables the prediction of the ON/OFF state of the appliance and, thus, eliminates all estimates that do not belong to the target appliance. The outcome of the model is generated by merging the outputs of the two sub-models.

The results of the test showed that the data selection strategy has considerably improved the results of the proposed model, especially for multi-state appliances. Furthermore, the results demonstrated that the proposed framework successfully disaggregates the target appliances with better energy estimation reliability and higher F1 score than the state-of-the-art techniques.

Chapter 5 Shallow neural networks approach for non-intrusive load monitoring

5.1. Introduction

A trend in the NILM field is the search for low sampling rate disaggregation approaches to enable potential compatibility with readily available residential smart meters. Recently, an increase in NILM research has been spurred by advances in deep learning methods, with very promising results, as discussed in Chapter 4. However, these techniques require a lot of training data to accomplish a satisfactory performance [201]. Owing to the scarcity of high-quality labels and durable data sets, this represents a significant challenge for NILM algorithms [41]. Moreover, these approaches benefit considerably from a large number of training parameters, which in most cases necessitates processing capacity that is neither cheap nor readily available [201]. Furthermore, NILM framework must meet a number of requirements, including sufficient accuracy to recognize the appliance operating state as well as accurate estimation of power usage, low-complexity models, and inexpensive hardware.

To tackle these challenges, a NILM approach based on a low sampling power data, thus enabling the usage of low-cost meters, and using shallow neural network models with low complexity is proposed. The framework is based on the design of radial basis function neural networks by a multi-objective genetic algorithm (MOGA). The rest of Chapter is structured as follows: Section 5.2 describes the proposed radial basis function neural networks designed by multi objective genetic algorithm (RBFNN-MOGA) approach for NILM. The experiments conducted, the data preprocessing approach, the evaluation metrics used, the parameters as well as the results obtained using the suggested framework are described in section 5.3. Since a comparative analysis was conducted to benchmark the performance of the proposed framework with some classification methods, the results of the developed classification methods are presented in Section 5.4. Sections 5.5 and 5.6 discuss the results of the comparative analysis and the experiments conducted using the public data set. The energy consumption in the case study house is discussed in Section 5.7 and Section 5.8 summarizes the chapter.

5.2. Proposed approach

The NILM process, as described in Chapter 2, encompasses data collection, feature extraction, device classification, and energy estimation [30], [35], [37], [38], [48]. Figure 5-1 presents an overview of the NILM framework. The first step of an energy monitoring system is dedicated

to data acquisition from a smart meter. Then, certain device characteristics must be selected and processed. A learning algorithm is employed to predict which devices are active at a given time, as well as their states. From this knowledge, the energy usage of appliance is estimated.

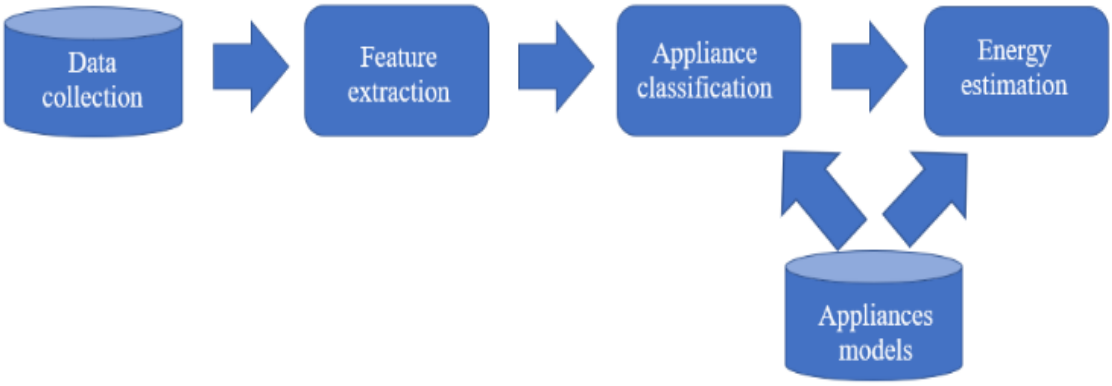


Figure 5-1. NILM Overview

The proposed NILM approach employs a RBFNN-MOGA, with design data selected through an approximate convex hull algorithm. Initially, the data selection method described in Chapter 3 section 3.11 is employed to conduct a data selection based on the identification of the most informative vertices of the real convex hull. Then, an RBFNN-MOGA classifier is designed to classify whether a specific appliance is active or not. The RBFNN-MOGA classifier is designed with the goal of lowering the complexity of the model as well as the number of false positives and false negatives. If the device is active, its energy consumption is then estimated using an RBFNN-MOGA estimator, designed in a similar way to the RBFNN-MOGA classifier. In this case the RBFNN-MOGA estimator is designed with the goal of lowering the complexity of the models and the root means square error (RMSEs) in the design sets. Figure 5-2 presents a flow chart summarizing the processes occurring in the design of the RBFNN-MOGA model. It should be noted that before using MOGA the data selection approach presented in Chapter 3 section 3.11 is applied to create the training, testing and validation set. The MOGA design approach has been utilized in the literature to tackle several classification and estimation problems [297], [298], [310], [311].

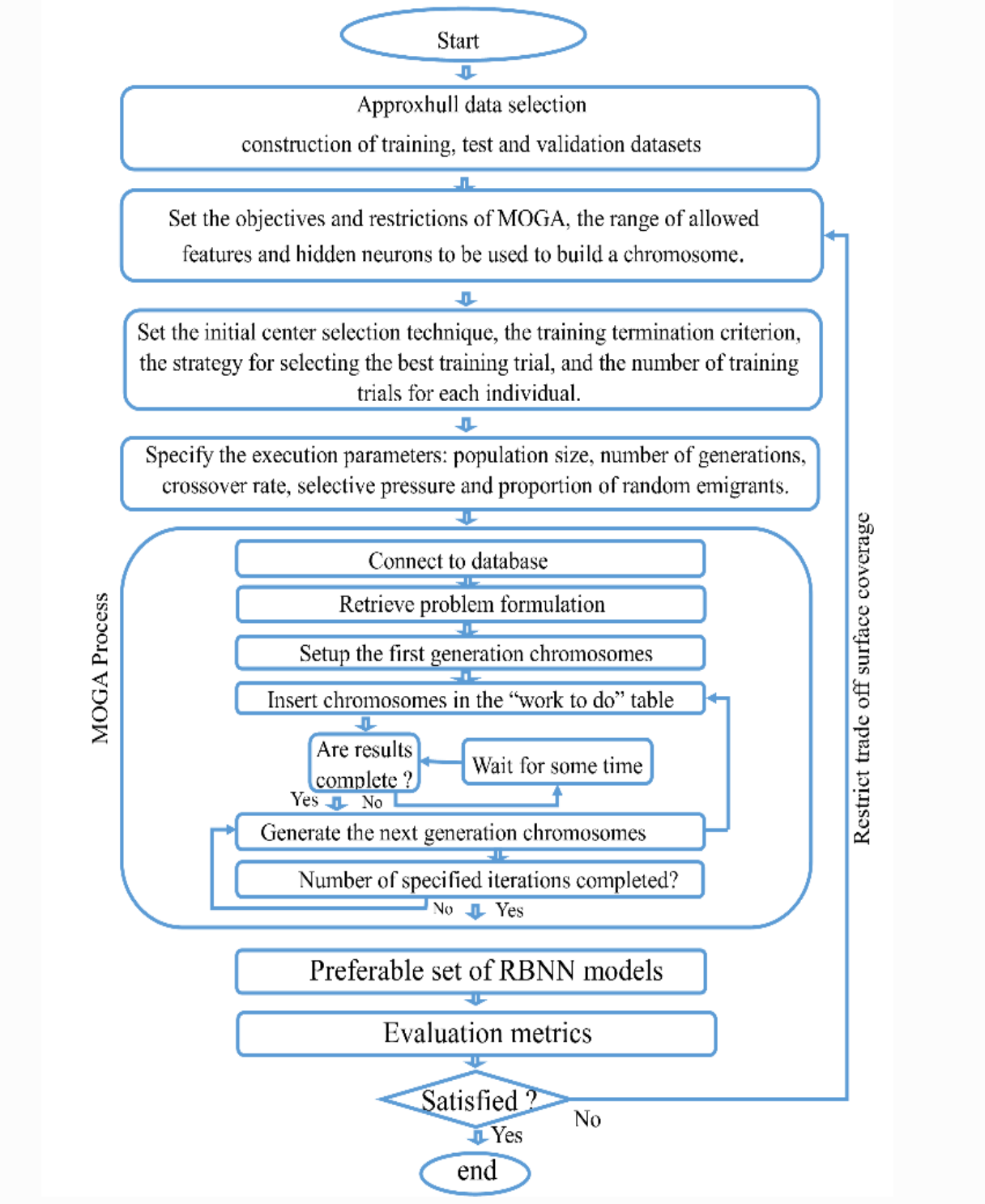


Figure 5-2. Overview of the design approach.

Once the models are designed, the online functioning process is illustrated in figure 5-3 flowchart.

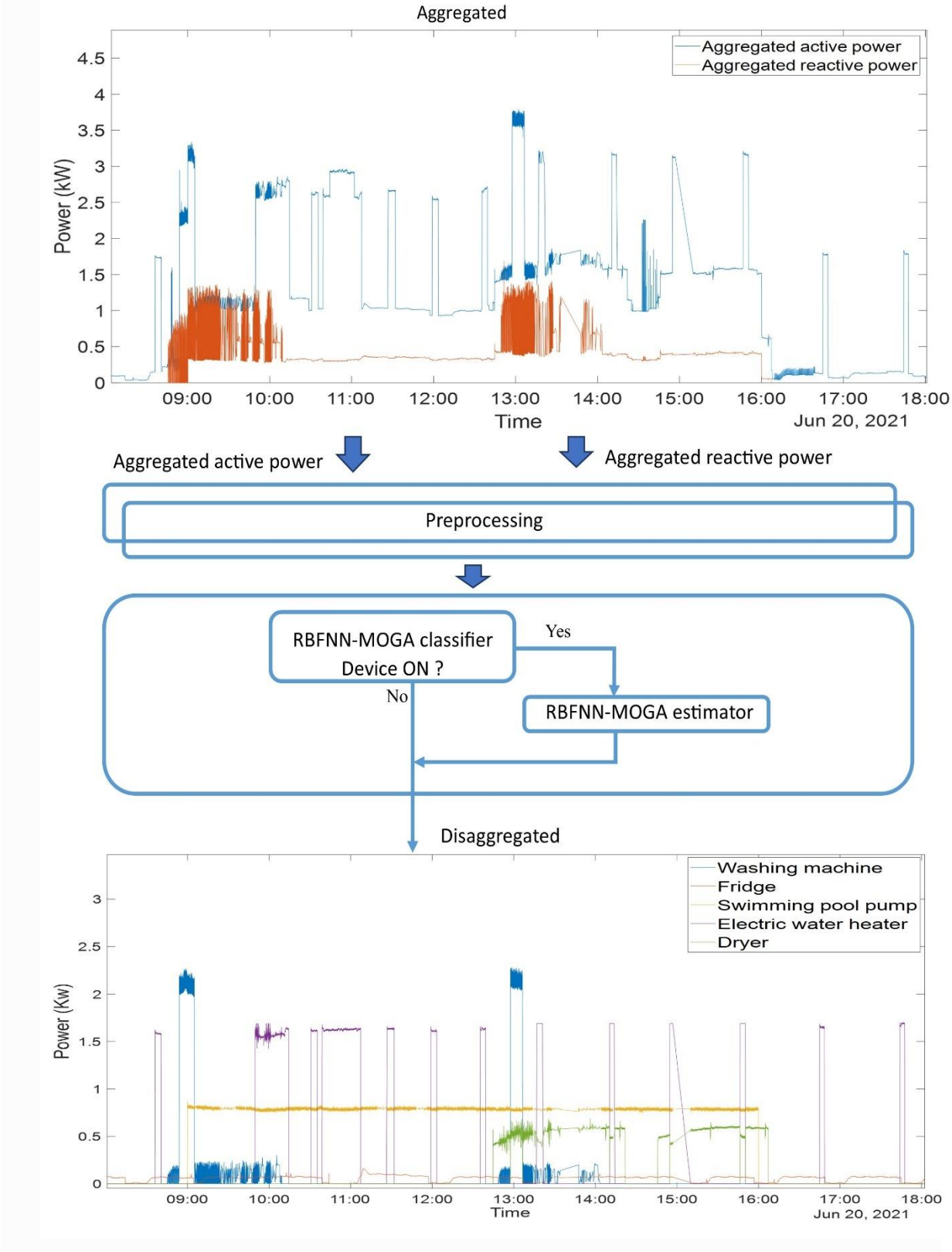


Figure 5-3. Overview of the proposed approach.

5.3. Experimental Results

The following sections report the results of the experiments conducted on the data gathered in the case study household described in Chapter 4, Section 4.3. The data have been gathered for over two years period and are available at [312]. The experiments considered data gathered over a one-month period, from March 1, 2021, to March 28, 2021, for the washing machine and from June 2, 2021, to June 28, 2021, for the other appliances, for a total of roughly 2.4 million samples for each appliance. Since the preliminary data, which were acquired with a one-second sampling period, were too big to be handled by the MOGA framework, the data were resampled at a one-minute rate.

5.3.1. Data preprocessing

Following the data collection, the next step is the pre-processing of the data sets. The total aggregate powers were retrieved from the EM340 meter data. The ground truth active power sequence of the device was extracted manually using the CW and SP data. The ON-OFF labels of the appliances were created using a method similar to that presented in Chapter 4, Section 4.4.1 using a sampling rate of 1 minute. One data set is generated for each device.

5.3.2. MOGA design radial basis function neural network results

To generate the optimal final solution, the system needs to train a considerable number of RBFNN models. On one side, certain limitations need to be set on the size of the datasets that will be fed into MOGA in order to make the process completed within a reasonable time frame, on the other side, to ensure the quality of the training set to be utilized. To do this, the data selection algorithm described in chapter 3 section 3.11. is employed. First the initial dataset is given to the algorithm. For device state identification, the input features are a sliding window containing 20 variables (10 delayed aggregated active power values and 10 delayed aggregated reactive power values). For energy estimation, the delays of two exogenous variables (aggregated active and reactive powers, 10 for each variable), as well as 10 lags of the modeled variable (active power of the device) are provided to the data selection algorithm.

All convex hull points of the relevant data samples are created using the ApproxHull data selection algorithm. The generated convex points and random data samples are then used to construct the MOGA training set, resulting in 60% of the data. The remaining random data samples are used to create test and validation sets for MOGA, 20% for each set. It is important to note that the initial data set is cleaned up before being fed into the ApproxHull data selection

algorithm. Thus, rows with non-numeric values, rows with missing values, equal columns (identical features), duplicate rows (equal samples) are excluded to reduce the risk that ApproxHull generates a singular matrix that corresponds to a random invalid facet. The results created by the Approxhull approach for the classification and estimation models as well as the size of the data set created for each device are presented in table 13 and table 14, respectively.

Table 13. Approxhull results for classification models

Devices	CH vertices	Training	Testing	Validation
Fridge	441	24493	8164	8166
Washing machine	810	24185	8061	8063
Electric water heater	731	8212	2737	2739
Swimming pool pump	673	12655	4218	4220

Table 14. Approxhull results for estimation models

Devices	CH vertices	Training	Testing	Validation
Fridge	842	24185	8061	8063
Washing machine	753	24185	8061	8063
Electric water heater	1497	8212	2737	2739
Swimming pool pump	640	12655	4218	4220

In terms of device state classification, the MOGA models were trained with the goal of lowering the complexity of the model, the number of false positives and negatives in the training and test sets for each device.

In terms of estimation of energy, the objectives that MOGA minimizes include the RMSEs of training set $\varepsilon(D_{tr})$, test set $\varepsilon(D_{te})$, the complexity of the model, and the prediction error ε (Ds, PH). The prediction horizon (PH) was given a value of 1, and Ds stands for a given period retrieved from the active power time series of the device (with s consecutive input-output pairs).

The parameters used for the RBFNN-MOGA are as follows: In order to lessen the complexity of the models in all the experiments, the number of input features was defined in the range of [1, 20] for classification models and in the range of [1, 30] for estimation models, while the number of hidden layer neurons was defined in the range of [2, 30] for both models. The number of generations was defined to 100, with a population size of 100. The proportion of random immigration was set to 0.01. A crossover rate of 70% was adopted, while the selective pressure was set to 2. Each individual in the population was given 10 training sessions under distinct initial conditions, and the best training trial was chosen using the proximity of origin criterion. An early stopping criterion with a limit of 50 iterations is adopted as the

termination criterion. These parameters have proven to be effective in the design of RBFNN by MOGA based on the experiences of previous research works [291], [297], [298], [311].

At the end of one MOGA run, the dimensions of the non-dominated sets generated for the estimation models and for the classification models for each device are presented in Table 15.

Table 15. Dimensions of Non-dominated sets

Devices	Classification	Estimation
Fridge	274	45
Washing machine	89	171
Electric water heater	414	329
Swimming pool pump	153	172

For each device, the performance statistics in the non-dominated sets in terms of minimum and average false positives (FP) and false negatives (FN), as well as the corresponding model complexities in the classification phase are presented in table 16.

Table 16. Classification performance in the non-dominated sets

Devices		Training		Testing		Validation		Model complexity
		FP	FN	FP	FN	FP	FN	
Fridge	Min	1722	0	534	0	550	0	6
	Mean	2638	295	843	96	836	103	139
Washing machine	Min	0	20	0	10	0	7	8
	Mean	24	220	4	56	6	55	282
Electric water heater	Min	58	1	24	0	25	0	6
	Mean	231	231	79	68	60	71	162
Swimming pool pump	Min	6	0	2	0	2	0	6
	Mean	84	49	23	14	23	12	143

Table 17 presents the performance statistics for the training, test, and validation sets performed in the non-dominated sets during the estimation phase in terms of minimum and mean root mean square error values.

Table 17. Estimation performance in the Non-dominated Sets

Devices	RMSE	Training	Testing	Validation
Fridge	ϵ_{min}	0.131	0.124	0.133
	$\bar{\epsilon}$	0.136	0.129	0.136
Washing machine	ϵ_{min}	0.010	0.018	0.010
	$\bar{\epsilon}$	0.031	0.032	0.025
Electric water heater	ϵ_{min}	0.250	0.221	0.239
	$\bar{\epsilon}$	0.298	0.260	1.710
Swimming pool pump	ϵ_{min}	0.063	0.048	0.057
	$\bar{\epsilon}$	0.069	0.048	1.790

After some further evaluation, for each device and each designed model, the optimal model of non-dominated sets with better performance and low complexity is chosen. Once the optimal models are defined, the optimal classification model is mainly used to detect whether the device is active or not in the next sample. If the appliance is active, then the optimal estimation model is used to predict the energy usage of the appliance. Table 18 shows the results of classification on the validation set utilizing the optimal model in terms of *Precision, Recall, F1 score, number of features, number of neurons in the hidden layer* and *model complexity* for each appliance. The results of estimation utilizing the optimal estimation models in terms of *SAE, MAE, EA, number of features, number of neurons in the hidden layer* and *model complexity* are reported in table 19.

Table 18. Device status classification results

Devices	R	P	F1	Number of features	Number of neurons	Model complexity
Fridge	0.98	0.91	0.95	7	30	240
Washing machine	0.97	0.97	0.97	4	30	150
Electric water heater	0.99	0.97	0.98	12	29	377
Swimming pool pump	1	1	1	8	25	225

R: Recall, P: Precision

Table 19. Energy estimation results

Devices	MAE (W)	SAE	EA	Number of features	Number of neurons	Model complexity
Fridge	4.00	0.010	0.96	4	6	30
Washing machine	1.1	0.030	0.93	23	20	480
Electric water heater	66.7	0.043	0.95	27	18	504
Swimming pool pump	1.5	0.004	0.99	7	14	112

The analysis of the results reported in Tables 18 and 19 shows the ability of the models to identify the appliance status and estimate the power consumed by each device. The results of the appliance state classification (Table 18) show an F1 score of over 95% for the fridge, 97% for the washing machine, 98% for the electric water heater and 100% for the swimming pool pump. The complexity of the classification models varies from 150 for the washing machine (with 30 neurons in the hidden layer and 4 input features) to 377 for the electric water heater (using 29 neurons in the hidden layer and 12 input features). For the fridge (30 neurons in the hidden layer and 7 input features) and the swimming pool pump (25 neurons in the hidden layer and 8 input features), the complexity of the classification models is 240 and 225 respectively. The results in terms of energy estimation (Table 19) show that the energy usage of the washing machine was estimated with an EA score of over 93%, the electric water heater was estimated with an EA score of 95%, and an EA score of 96% was observed in the disaggregation of the fridge whereas the swimming pool pump was estimated with an EA score of 99%. The complexity of the estimation models varies from 30 for Fride (6 neurons and 4 input features), 112 for the swimming pool pump (14 neurons and 7 input features), 480 for the washing machine (20 neurons and 23 input features) and 504 for the electric water heater (18 neurons and 27 input features).

An illustrative example of the results of the disaggregation process in terms of active power is presented in figures 5-4 to 5-7. The actual and predicted active power of Fridge is presented in figure 5-4. Figure 5-5 illustrates the estimated and the actual active power of washing machine. Figures 5-6 and 5-7, respectively, show the predicted and actual active power of Electric water heater and Swimming pool pump.

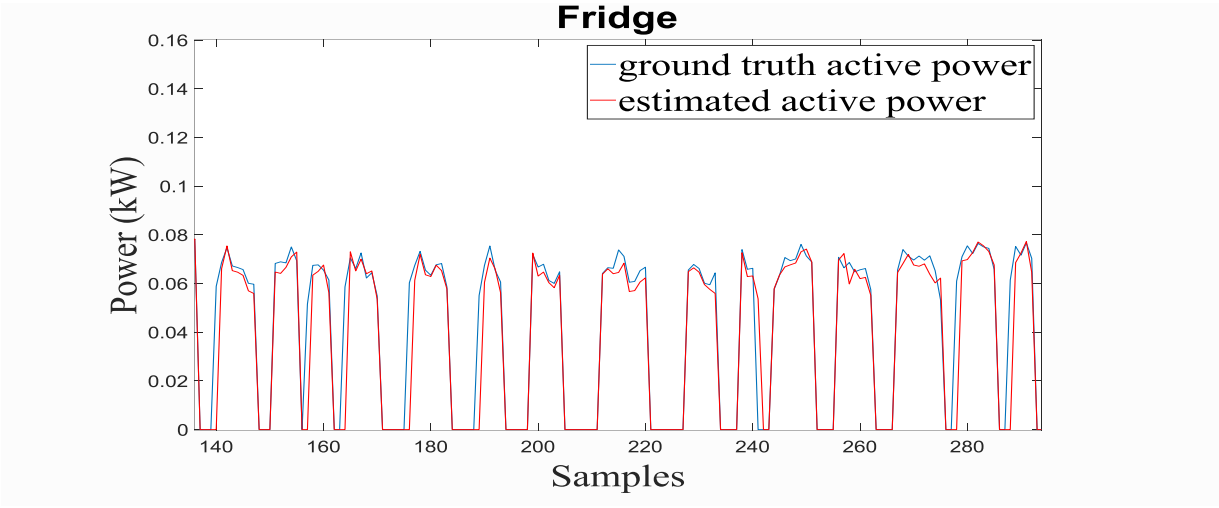


Figure 5-4. Fridge. Red, predicted active power, blue, actual active power,

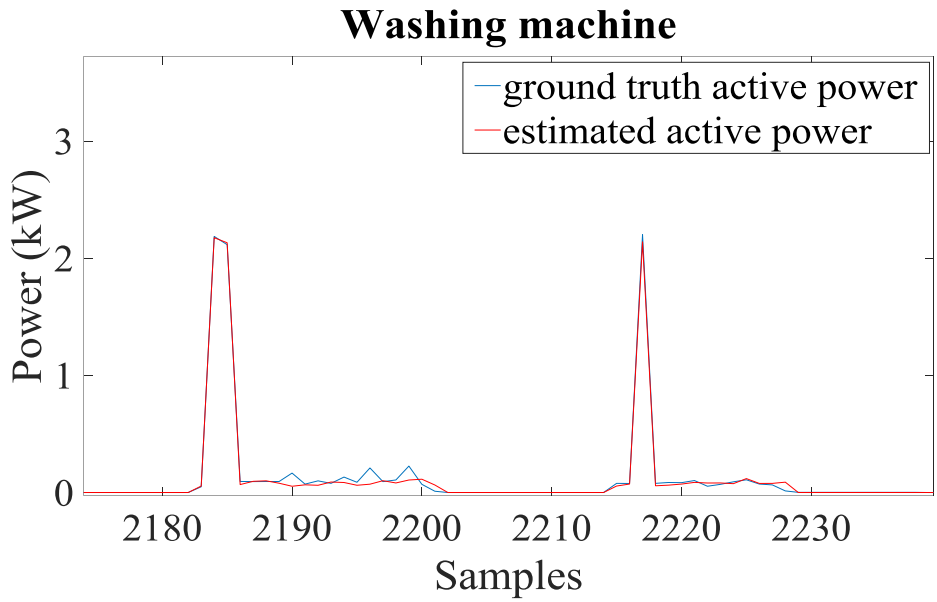


Figure 5-5. Washing machine. Red, predicted active power, blue, actual active power.

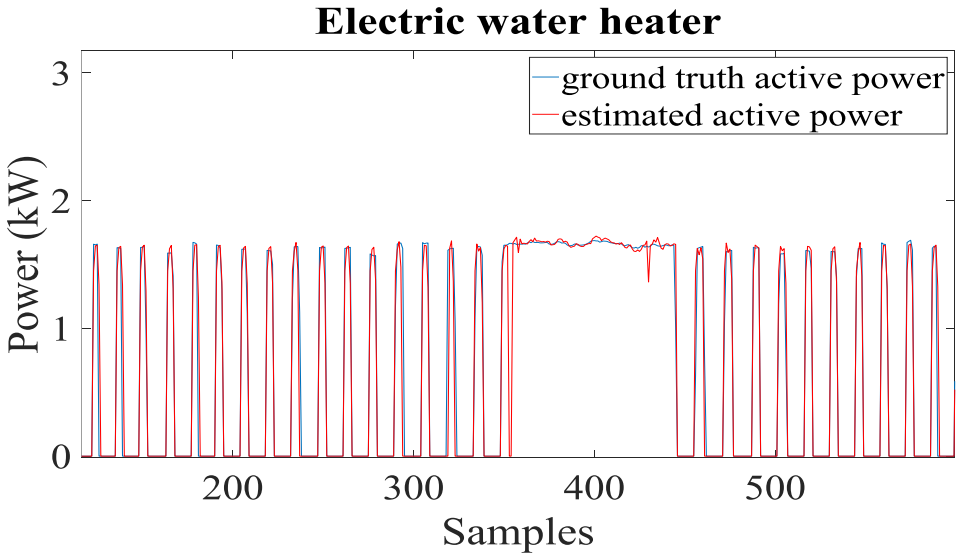


Figure 5-6. Electric water heater. Red, predicted active power, blue, actual active power.

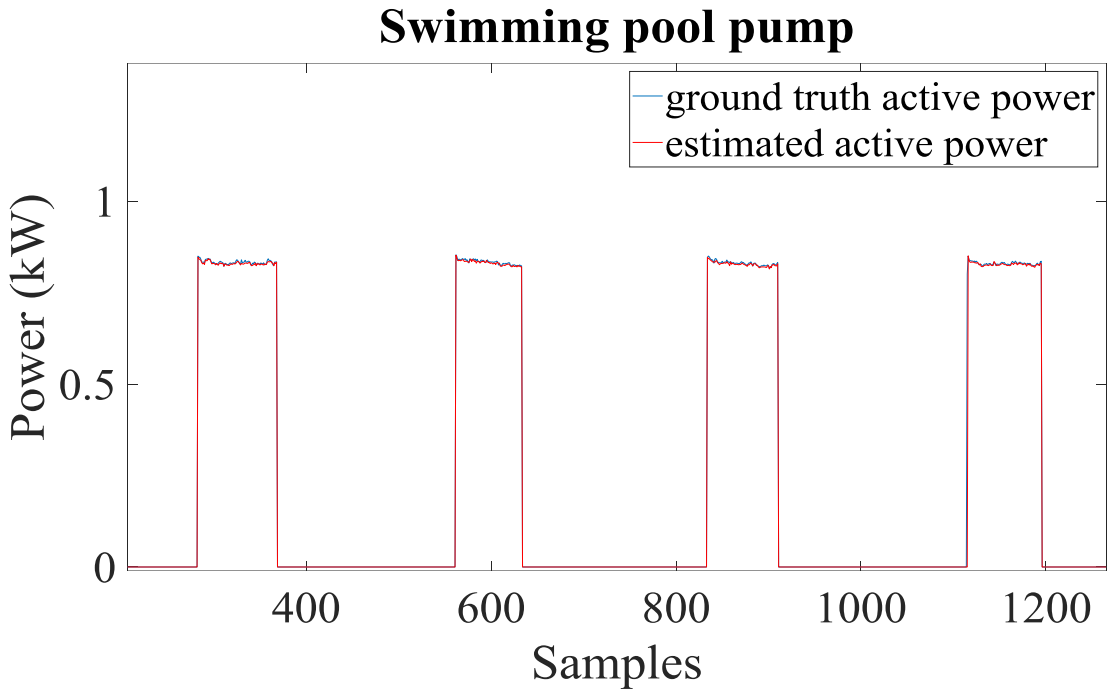


Figure 5-7. Swimming pool pump. Red, predicted active power, blue, actual active power,

The analysis of Figures 5-4 to 5-7 shows an accurate consistency between the predicted and measured active power for all the appliances in the experiment. It can be observed that the estimated power signals are very reliable. In view of these experimental results, we can conclude that the disaggregation results are quite satisfactory, with models of very low complexity and working on one-minute data.

5.4. Results of other implemented classification methods

Five experiments employing a distinct range of classification techniques, including SVM, KNN, DT, LSTM, and CNN classifiers, were conducted on the same one-month of data from the previous experiment. To improve the robustness of the training operation and accelerate its process, the data has been standardized. An approach similar to that described in [51] was used to standardize the aggregated data. The mean and standard deviation of each sequence are first determined. Afterwards, the mean is subtracted from the input sequence which is then divided by the standard deviation to generate the standardized sequence. The process is formulated in the equation (97).

$$X_{standardized} = \frac{(X - \mu)}{\sigma} \sim N(0,1) \tag{97}$$

Where X represents the input sequence, σ the standard deviation of the input sequence, and μ the mean of the input sequence.

For all the models, the data set is separate into training set (80%) and a test set (20%). As in the previous experiment, a sliding window of 20 variables (10 aggregated active power values and 10 aggregated reactive power values) is utilized as input to the models. The output of the models is the ON-OFF status of the devices.

5.4.1. Support Vector Machines (SVM) results

The model was trained to identify the operating states of the appliance. The goal is to detect whether the device is active or not. For each appliance, a single model is trained. The Gaussian rbf kernel was employed to train the model. The spreading parameter σ and the penalty C, are tweaked to minimize the error function described in (98).

$$error\ (x_i, y_i) = \frac{1}{2} w'w + C \sum_{i=1}^{N} \xi_i \tag{98}$$

Where w denotes the weight vector, and ξ_i the slack variables. The search for hyperparameters that reduce the loss of five-fold cross-validation was performed automatically. Table 20 reports the results of the tests conducted in the test dataset in terms of *Precision, Recall* and *F1 score*.

Table 20. SVM results

Devices	Recall	Precision	F1
Fridge	0.91	0.90	0.91
Washing machine	0.91	0.75	0.82
Electric water heater	0.98	0.96	0.97
Swimming pool pump	0.98	0.99	0.98

5.4.2. K Nearest Neighbors (KNN) results

As for SVM, the KNN model was trained to identify the appliance's operating states. For each appliance, one model is trained. Given that k is a user predetermined parameter, a trial-and-error process was performed to select the suitable k that suits the data. The resulting best value obtained was k = 5. Table 21 presents the results within the test dataset in terms of Precision, *Recall*, and *F1 score.*

Table 21. KNN results

Devices	Recall	Precision	F1
Fridge	0.90	0.92	0.91
Washing machine	0.91	0.73	0.81
Electric water heater	0.98	0.95	0.97
Swimming pool pump	0.99	0.99	0.99

5.4.3. Decision Tree (DT) results

The algorithm was trained to identify the operating states of the appliances. For each appliance, a single model is trained. Through a cross validation method, the training data was randomly split into ten subsets. Then, ten new trees are generated within each of the nine subsets of data. Afterwards, using data that was not involved in the training, the prediction accuracy of each new tree is assessed. This provides a reliable estimate of the prediction reliability. The ideal leaf node observations (MinLeafSize) minimizing the cross-validation loss was determined using an automatic hyperparameter optimization. The results of the tests conducted in the test dataset in terms of *Precision*, *Recall* and *F1 score* are reported in Table 22.

Table 22. Decision tree results

Devices	Recall	Precision	F1
Fridge	0.87	0.93	0.90
Washing machine	0.94	0.73	0.82
Electric water heater	0.98	0.93	0.95
Swimming pool pump	0.98	0.99	0.98

5.4.4. Long Short-Term Memory (LSTM) results

The fourth experiment was to design an LSTM classifier to identify the device operating states. The binary cross-entropy loss function and Adam's optimizer were used to train the model. Through a trial-and-error process, the loss function was minimized in order to obtain the set of parameters that best suits the task. The search for hyper parameters which minimize the loss of five-fold cross-validation was performed using hyper parameter optimization. The design parameters used with the suitable hyper parameter values is reported in table 23. Table 24 reports the results of the test performed in the test dataset in terms of *Precision*, *Recall* and *F1 score*.

Table 23. LSTM structure

Layer	number of hidden units	activation
Input		
LSTM	32	ReLu
Dropout (dropout=0.3)		
LSTM	64	ReLu
Dropout (dropout=0.3)		
LSTM	128	ReLu
Dropout (dropout=0.3)		
Fully connected dense	1024	ReLu
Fully connected dense	1	sigmoid

Table 24. LSTM results

Devices	Recall	Precision	F1
Fridge	0.89	0.83	0.86
Washing machine	0.82	0.96	0.88
Electric water heater	0.98	0.94	0.96
Swimming pool pump	0.97	0.99	0.98

5.4.5. Convolutional neural network (CNN) results

In the fifth experiment, a CNN classifier was trained to classify the operating states of the devices. As for LSTM, the Adam optimizer and the binary cross-entropy loss function are used. The design was conducted using a trial-and-error process to find the best suited set of parameters that minimize the loss function. Table 25 presents a summary of the architecture used with the suitable hyper parameters. The results of the tests conducted on the test dataset in terms of *Precision*, *Recall*, and *F1 score* are reported in Table 26.

Table 25. CNN structure

Layer	filters	kernels	activation
Input			
Conv1D	32	3	ReLu
Conv1D	64	3	ReLu
Conv1D	128	3	ReLu
Max pooling1D			
dense (1024)			ReLu

dense (1)	sigmoid

Table 26. CNN results

Devices	Recall	Precision	F1
Fridge	0.92	0.91	0.91
Washing machine	0.96	0.90	0.93
Electric water heater	0.95	0.98	0.96
Swimming pool pump	0.99	0.99	0.99

5.5. Discussion

The results of the experiments reported in tables 18, 20, 21, 22, 24 and 26 show that all the models considered can identify the operational states the different appliances. To highlight the performance of the proposed approach, the obtained F1 score value has been compared to the other implemented classification approaches. The comparative histogram of the different approaches is shown in figure 5-8.

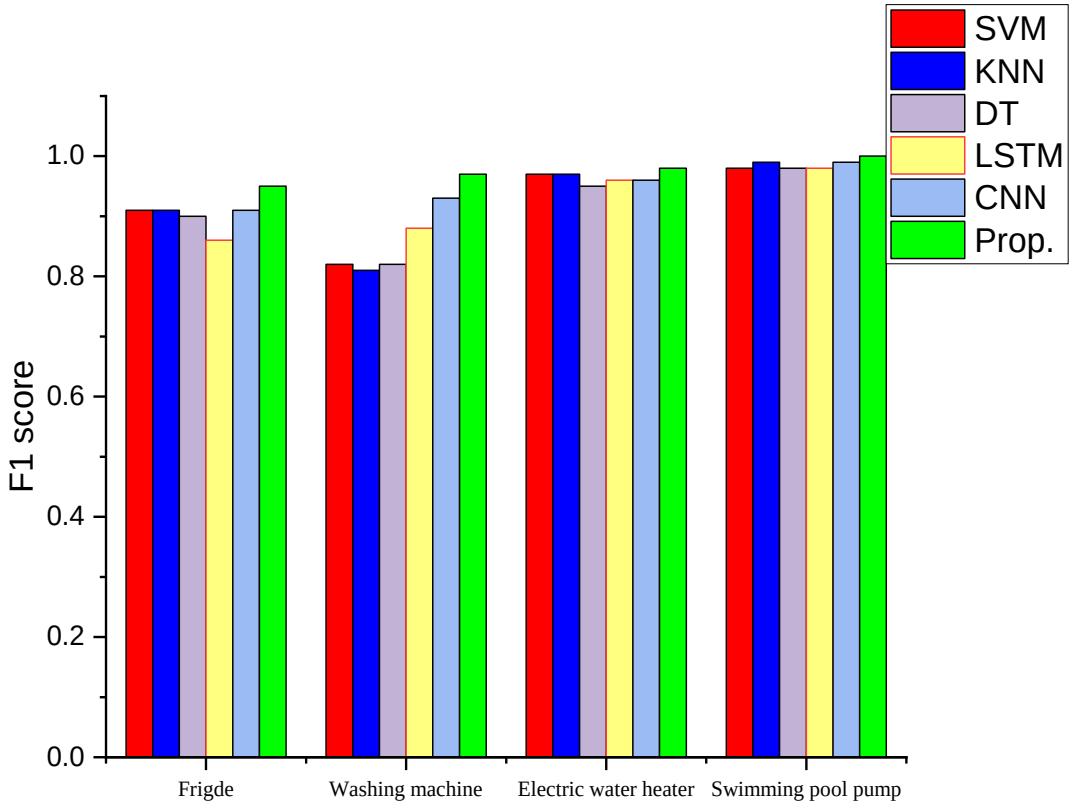


Figure 5-8. Classification performance in terms of F1 score.

From the analysis of the comparative results shown in Figure 5-8, it can be observed that the proposed model obtained the best F1 score of 95% for the fridge, while the LSTM obtained the worst F1 score of 86%. The other classifiers (SVM, KNN and CNN) identified the operating states of the fridge with similar F1 scores of 91% and a slightly lower score of 90% for DT.

For the washing machine, the proposed model achieved the highest F1 score of 97%, whereas the SVM and DT models achieved similar F1 scores of 82%. Also, the deep learning models performed slightly better than the SVM and DT models, with the CNN achieving an F1 score of 93% and the LSTM an F1 score of 88%. The KNN model obtained the worst F1 score of 81%.

For the electric water heater, the MOGA framework achieved an F1 score of 98%, while the SVM and KNN models achieved a comparable F1 score of 97%. The LSTM and CNN models achieve a similar F1 score of 96%. The DT model (95%) obtained the lowest F1 score.

For the pool pump, the proposed framework achieved an excellent F1 score of 100% whereas the other models (SVM, KNN, DT, CNN, LSTM) performed well with F1 scores ranging from 98% to 99%.

Globally, it can be noted that the best performances are achieved in the classification of high-power appliances, while most of the models struggle to correctly classify the operating states of devices like washing machines and fridges. This stems from the complex architecture of multi-state devices, where identification is more challenging, and may lead to a high number of false positives. Overall, it can be concluded that the proposed approach performs better than the other classification methods in terms of identifying the operating states of devices.

Since the same data were used for each of these above six experiments, the performance of these approaches can be qualitatively benchmarked. This is generally not the case when evaluating the performance of state-of-the-art approaches, owing to the distinct nature of the data used and the context of the experiments. However, for illustrative purposes, a comparative study with the most recent state of the art methods has been performed. Two of the most commonly used multistate devices for evaluating NILM state-of-the-art techniques are the washing machine and the fridge, and thus they will be considered in this analysis. Table 27 illustrates a comparison of the proposed NILM framework with some state-of-the-art NILM techniques for washing machine. The effectiveness of the MOGA framework compared to state-of-the-art techniques for fridge disaggregation is presented in Table 28. Also, to be noted

that the optimization approach proposed in Chapter 4 (which we will call deepOpt) uses the same data as those used to design the MOGA framework, but with distinct sampling intervals (1 second for deepOpt and 1 minute for MOGA framework).

Table 27. Washing machine performance comparison

Approach	S (s)	R	P	F1	MAE(W)	SAE	EA
[283]	1	**1**	0.60	0.70	118.1	-	-
[198]	-	-	0.76	14.42	0.51	0.74	
[50]	60	0.86	0.87	0.86	8.31	**0.01**	-
Prop. deepOpt	1	0.96	0.96	0.96	1.64	0.05	**0.93**
Prop. MOGA	60	0.97	**0.97**	**0.97**	**1.1**	0.03	**0.93**

S: sampling frequency, R: Recall, P: Precision, Prop.: proposed

Table 28. Fridge performance comparison

Approach	S (s)	R	P	F1	MAE(W)	SAE	EA
[283]	1	0.73	0.87	0.79	4.34	-	-
[198]	1	-	-	0.87	19.60	0.46	0.76
[50]	60	0.89	0.85	0.87	17.03	0.05	-
Prop. DeepOpt	1	0.97	**0.92**	**0.95**	12.72	0.09	0.88
Prop. MOGA	60	**0.98**	0.91	**0.95**	**4.00**	**0.01**	**0.96**

From the analysis of the performance comparison results shown in tables 27 and 28, the proposed approach outperformed the optimization approach presented in chapter 4 (DeepOpt), especially in terms of estimation accuracy (EA) for the fridge (96% versus 88% for DeepOpt). For the washing machine, the mean absolute error was lowered by 32% using the proposed approach compared to the optimization approach illustrated in Chapter 4 (DeepOpt). It is important to note that the proposed MOGA approach, and the optimization method reported in Chapter 4 performed better than the state-of-the-art approaches reported in Tables 27 and 28. Since MOGA's design of a RBFNN does not involve too much training data (about 8212 samples to 24493 samples), while performing comparably or better than approaches involving more training data (about 1.3 million samples to 1.4 million samples for DeepOpt), it can be seen that resampling the data from 1 second to 1 minute did not alter the performance of the MOGA framework and considerably lowered the proportion of data to be processed.

5.6. Experiment on AMPD public dataset

An experiment was conducted in the public dataset AMPD (Almanac of Minutely Power datasets [224]). The AMPD dataset contains water data, natural gas data, and electricity consumption data from a Canadian household over a period of two years, in which electrical features such as current, voltage, active, and reactive powers are captured at one-minute resolution. The experiment was conducted using data collected from April 1, 2012, to April 30, 2012. The same configuration as the proposed MOGA framework was used to train the models. The fridge and the clothes dryer were considered in order to assess the effectiveness of the proposed technique within the AMPD public dataset. Table 29 reports the results of the test performed on the validation set in terms of power estimation and identification of the operational states of the appliances.

Table 29. Results on AMPD public dataset

Devices	MAE (W)	SAE	EA	R	P	F1
Fridge	9.13	0.10	0.90	0.91	0.93	0.92
Clothes dryer	3.53	0.01	0.94	0.99	1	0.99

The analysis of the results reported in Table 29 demonstrates that the operating states of the clothes dryer and the fridge were identified with a high F1 score of 99% and 92% respectively. The performance in terms of energy estimation shows that the clothes dryer was predicted with an estimation accuracy of more than 94%, whereas the fridge obtained an EA score of 90%. Figure 5-9 depicts some examples of disaggregation outcome for the fridge. An example of disaggregation outcome for the clothes dryer is shown in Figure 5-10.

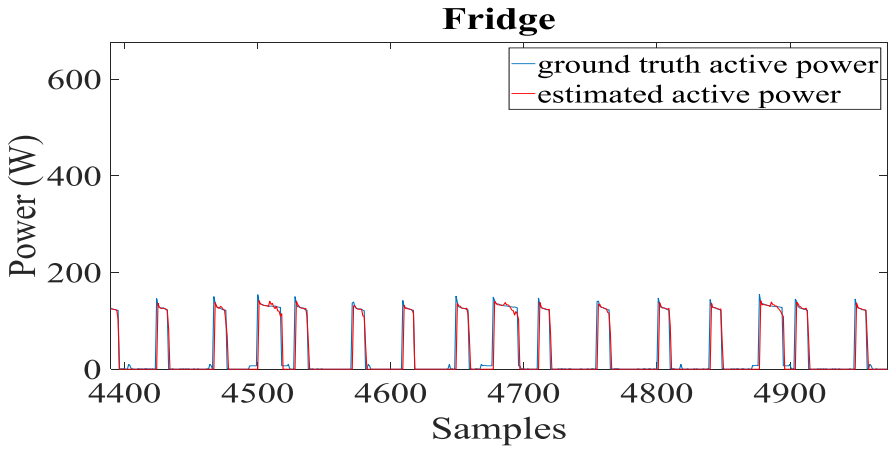


Figure 5-9. Fridge AMPD. Red, predicted active power, blue, actual active power

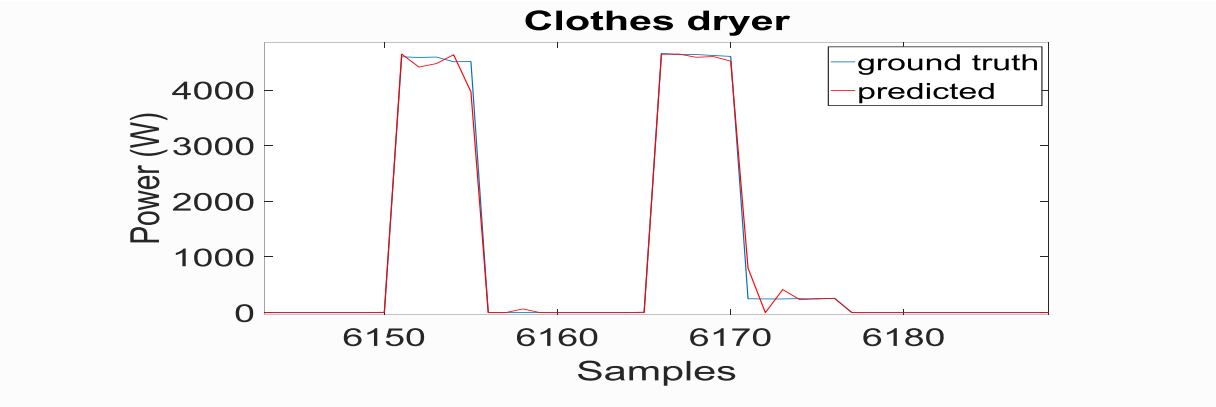


Figure 5-10. Clothes dryer AMPD. Red, predicted active power, blue, actual active power.

From the analysis of Figures 5-9 and 5-10, a reasonable consistency between the actual active power and the estimated active power of the appliances is observed. The estimated active power signals are very trustworthy.

A comparative study was conducted with the work reported in [313] and [314] which use the same AMPD public data sets. Indeed, the authors of [313] suggested a modified cross-entropy (MCE) algorithm based on a combinatorial optimization approach to identify load states. The method involves generating a load decomposition while iteratively updating the likelihood of the appliance operation and considering the penalty function constraint to find the optimal combination of states. In [314], the authors proposed a method based on the probability of time-segmented states. In order to collect the power signatures of the load and count the probabilities of the segmented states over time, they applied an affinity propagation clustering technique. The function chooses the best matrix following the detection of the device state after producing a variety of matrix representations of the device states using the likelihood. Table 30 presents the results of performance comparison of the fridge and clothes dryer in terms of operating state detection (F1 score) with the works reported in [313] and [314].

Table 30. F1 comparison

Devices	[313]	[314]	**Prop. MOGA**
Fridge	**0.92**	0.88	**0.92**
Clothes dryer	0.23	0.29	**0.99**

The analysis of the results illustrated in Table 30 shows that the proposed framework achieved an F1 score of 92% comparable to the F1 score achieved by the work reported in [313] while the work presented in [314] achieved the worst F1 score of 88%. For the clothes dryer,

the proposed framework obtained the best F1 score of 99% while the approaches presented in [313] and [314] obtained the lowest scores of 23% and 29% respectively. From these results it can be concluded that the proposed MOGA framework performs very well.

5.7. Energy consumption estimation in the case study house

The MOGA technique proposed to disaggregate the four devices presented in Section 5.2 was extended to other appliances in the case study home. These appliances are within the range of appliances that consume the most electricity in the house. The experiment was conducted using one month of data. The results obtained from the data selection and the size of the dataset for each appliance in terms of estimation and classification are reported in table 31 and table 32.

Table 31. Data selection results for classification models

Devices	CH vertices	Training	Testing	Validation
AC_1	913	28777	8925	8927
AC_2	1170	28505	9501	9503
AC_3	569	26777	8925	8927
AC_4	799	26777	8925	8925
BS_1	714	26777	8925	8927
BS_2	638	26777	8925	8927
Oven	1164	26777	8925	8927
DM	993	24185	8061	8063
EAH _1	900	26777	8925	8927
EAH _2	1386	26777	8925	8927
EAH _3	1384	26777	8925	8927

AC: Air conditioner, DM: Drying machine, BS: Burner stove, EAH: Electric air heater.

Table 32. Data selection results for estimation models

Devices	CH vertices	Training	Testing	Validation
AC_1	1993	26777	8925	8927
AC_2	913	28505	9501	9503
AC_3	543	26777	8925	8927
AC_4	565	26777	8925	8927
BS_1	881	26777	8925	8927
BS_2	698	26777	8925	8927
Oven	1249	26777	8925	8927
DM	835	24185	8061	8063
EAH_1	1982	26777	8925	8927
EAH_2	1309	26777	8925	8927
EAH_3	2180	26777	8925	8927

The results obtained after one run of MOGA for each device is an ensemble of the non-dominated models. Table 33 reports the dimensions of the non-dominated sets for the device's classification models and the energy estimation models.

Table 33. Dimensions of non-dominated sets

Devices	Classification	Estimation
AC_1	444	145
AC_2	281	86
AC_3	663	126
AC_4	131	159
BS_1	827	81
BS_2	256	256
Oven	539	213
DM	561	59
EAH_1	636	80
EAH_2	654	231
EAH_3	768	443

Once the non-dominated sets are obtained, an analysis is conducted to select the optimal models with both good performance and low complexity among the non-dominated sets.

Therefore, two models (one model for classification and one model for estimation) are selected for each appliance. The results regarding the device operating states identification using the selected models in terms of *model complexity*, *number of features*, *number of neurons in the hidden layer, Precision, Recall* and *F1 score* are presented in Table 34. Regarding device energy estimation, the results using the selected models in terms of *MAE, SAE, EA*, number of features, number of neurons in the hidden layer and model complexity are presented in Table 35.

Table 34. Device state classification results

Devices	Recall	Precision	F1	Nb of features	Nb of neurons	Model complexity
AC_1	0.92	0.94	0.93	13	20	280
AC_2	0.97	0.98	0.97	5	30	180
AC _3	0.94	0.90	0.92	11	28	336
AC_4	1.00	1.00	1.00	9	11	110
BS_1	0.59	0.81	0.68	5	23	138
BS_2	0.86	0.97	0.91	12	29	377
Oven	0.80	0.97	0.88	19	29	580
DM	0.96	0.98	0.97	17	29	522
EAH_1	0.98	0.95	0.96	4	30	150
EAH_2	0.91	0.89	0.90	9	30	300
EAH_3	0.74	0.76	0.75	9	18	180

Table 35. Energy estimation results

Devices	MAE (W)	SAE	EA	Nb of features	Nb of neurons	Model complexity
AC_1	191.0	0.011	0.87	14	18	270
AC_2	11.0	0.002	0.97	3	17	68
AC _3	11.0	4e-4	0.98	7	10	80
AC_4	9.0	0.003	0.97	15	18	288
BS_1	90.0	0.004	0.86	3	20	80
BS_2	37.0	0.002	0.90	3	4	16
Oven	43.0	0.005	0.73	22	19	437
DM	0.009	4e-4	0.98	3	9	36
EAH_1	28.0	0.010	0.95	12	13	169
EAH_2	10.0	0.004	0.99	3	12	48

EAH_3	14.0	0.006	0.87	16	6	102

From the analysis of the classification results reported in Table 34, it is observed an excellent F1 score of 100% for the air conditioner (AC-4), while the poorest F1 scores were obtained in the detection of the electric air heater (EAH-3: 75%) and the burner stove (BS-1: 68%). In the first case, it may be possible to use a higher sampling rate, while in the second case, the appliance is rarely used and there is a scarcity of detectable data. The remaining devices were identified with good F1 values, varying between 88% and 98%. Regarding the energy estimation, the results presented in Table 35 demonstrate that the energy consumption of the electric air heater (EAH-2) was predicted with an excellent estimation accuracy (EA) of 99% while the poorest performance was obtained in the estimation of the energy consumed by the oven (73%). This poor disaggregation performance results from the fact that the oven has many distinct modes of operation, with distinct power consumption, as well as a wide range of temperatures. The other appliances were disaggregated with a good estimation accuracy varying from 86% to 98%. Figures. 5-11 to 5-21 illustrate the examples of disaggregation outcome in terms of estimated active power (red line) and ground truth values of active power consumption (blue line) for the devices considered.

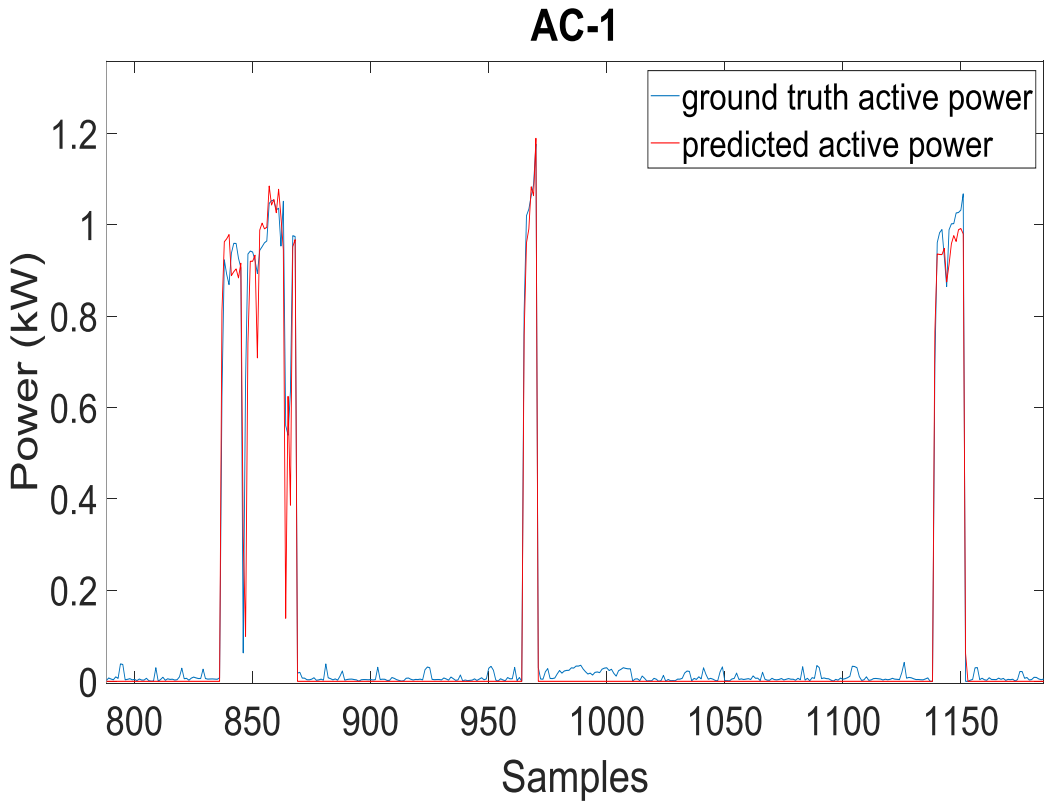


Figure 5-11. Air conditionner (AC-1). Red, predicted active power, blue, actual active power.

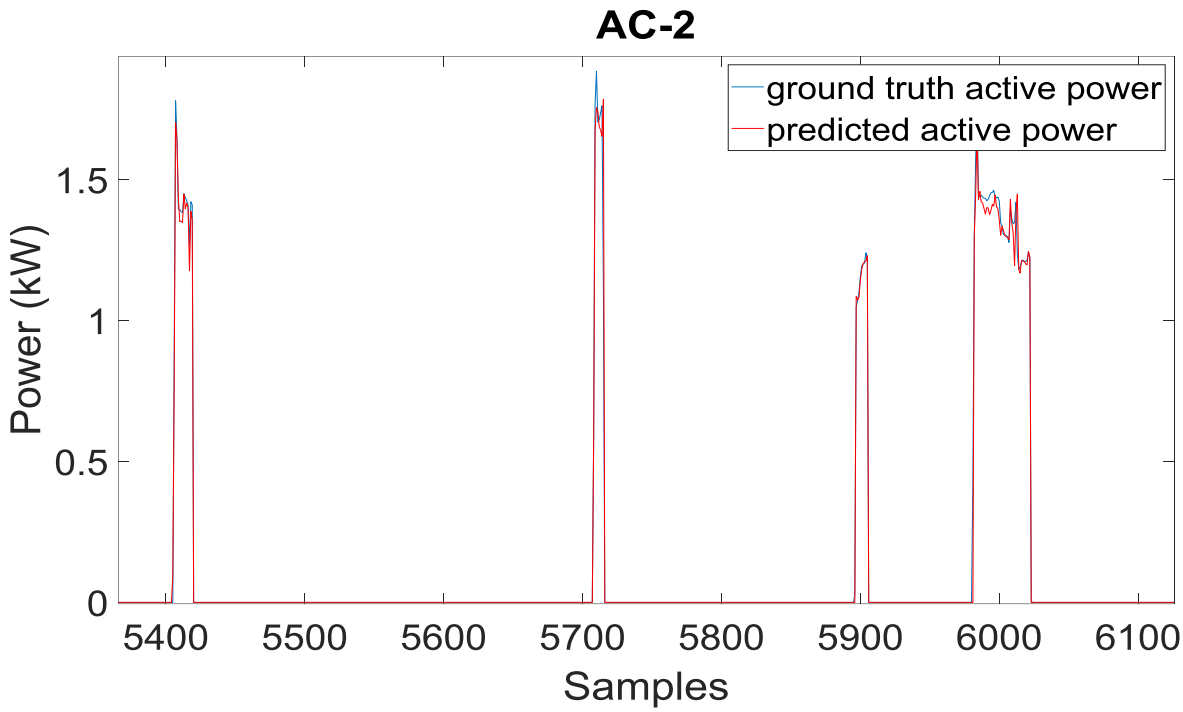


Figure 5-12. Air conditionner (AC2). Red, predicted active power, blue, actual active power.

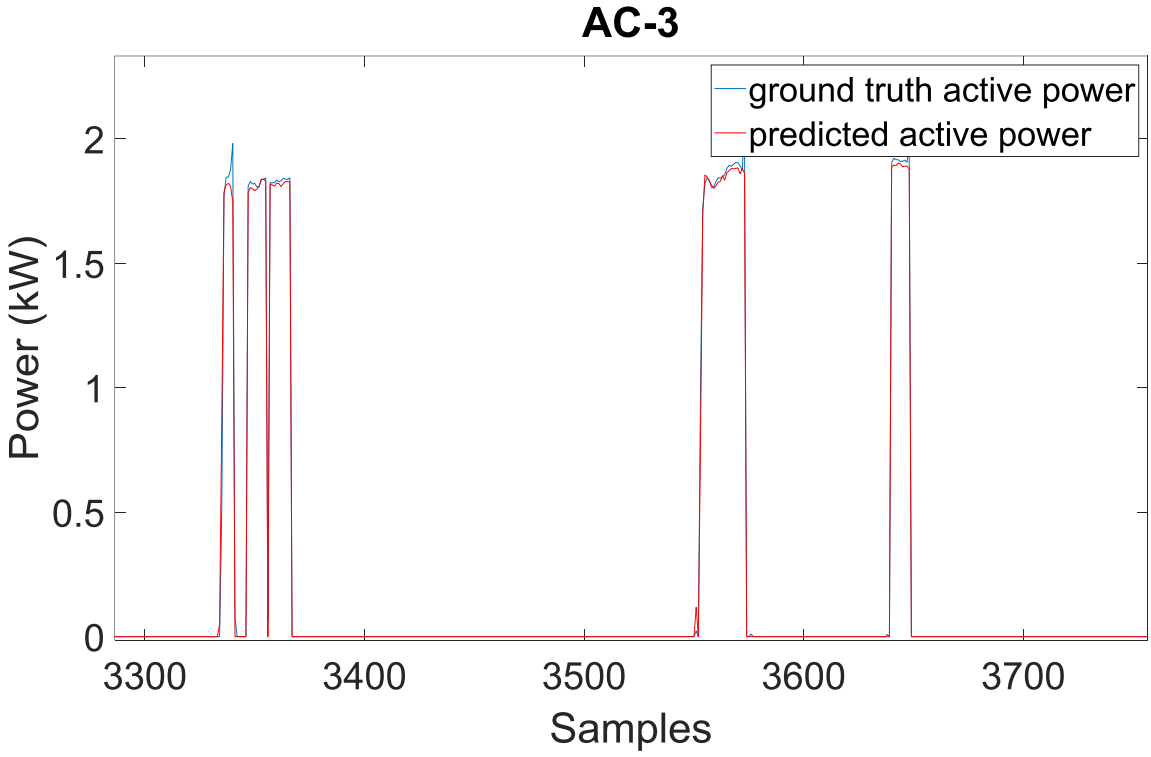


Figure 5-13. Air conditionner (AC3). Red, predicted active power, blue, actual active power.

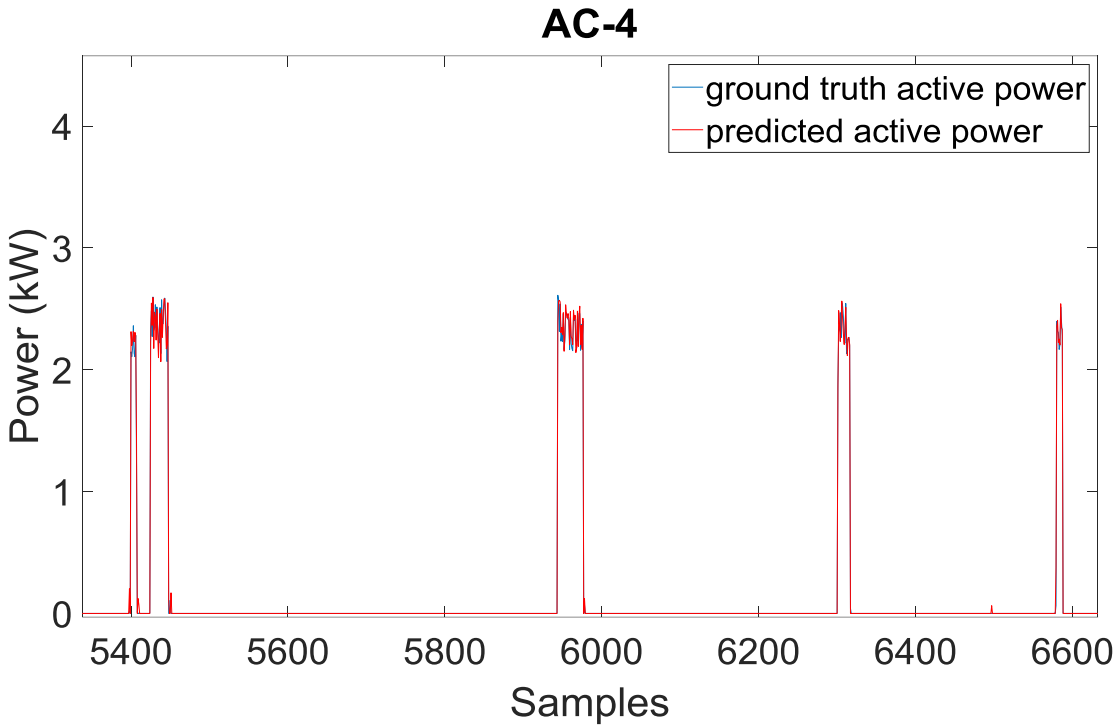


Figure 5-14. Air conditionner (AC3). Red, predicted active power, blue, actual active power.

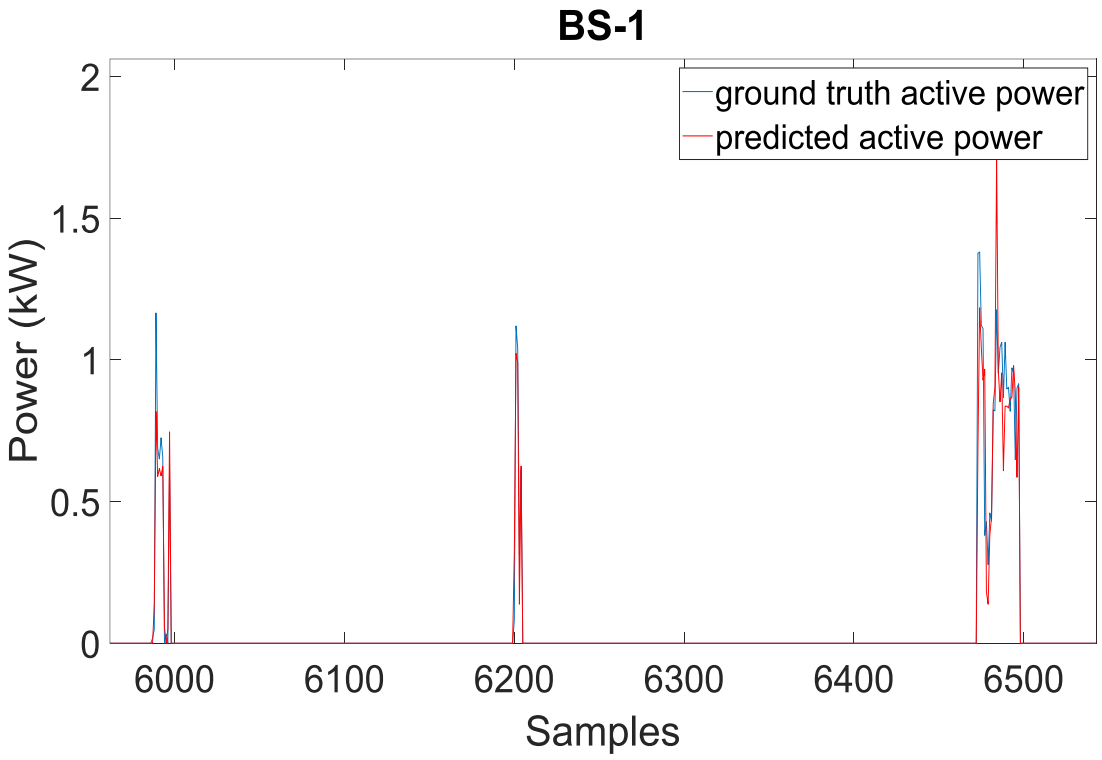


Figure 5-15. Burner Stove (BS1). Red, predicted active power, blue, actual active power,.

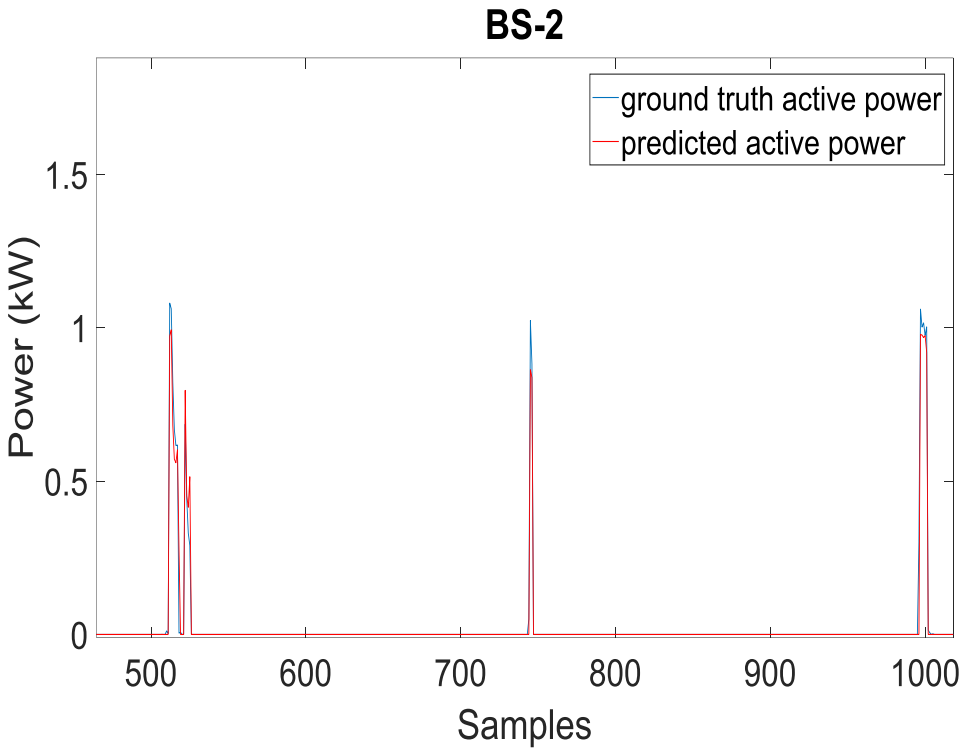


Figure 5-16. Burner Stove (BS2). Red, predicted active power, blue, actual active power

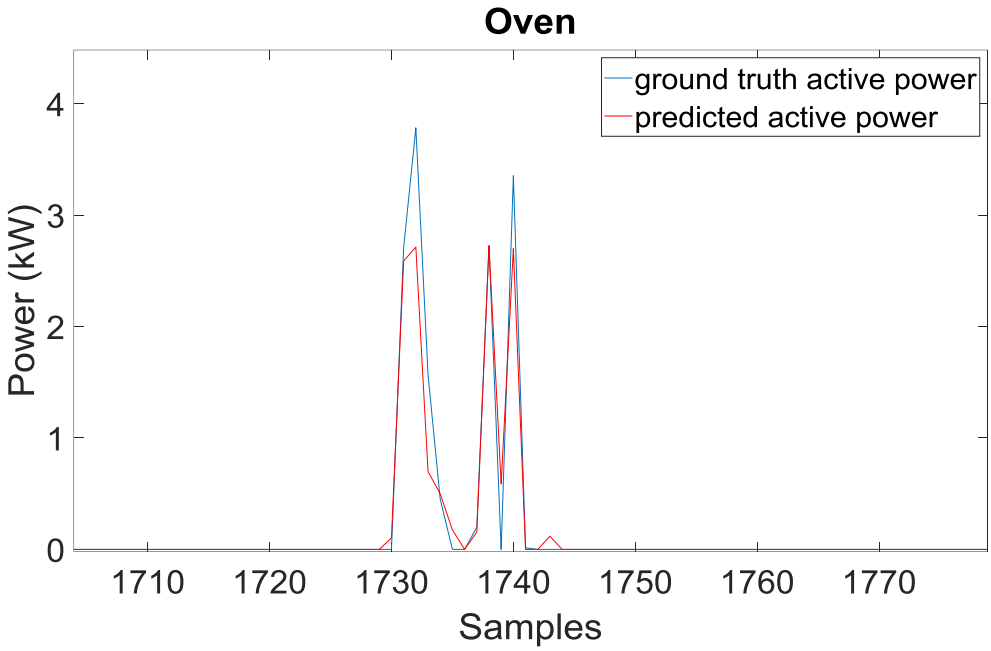


Figure 5-17. Oven. Red, predicted active power, blue, actual active power.

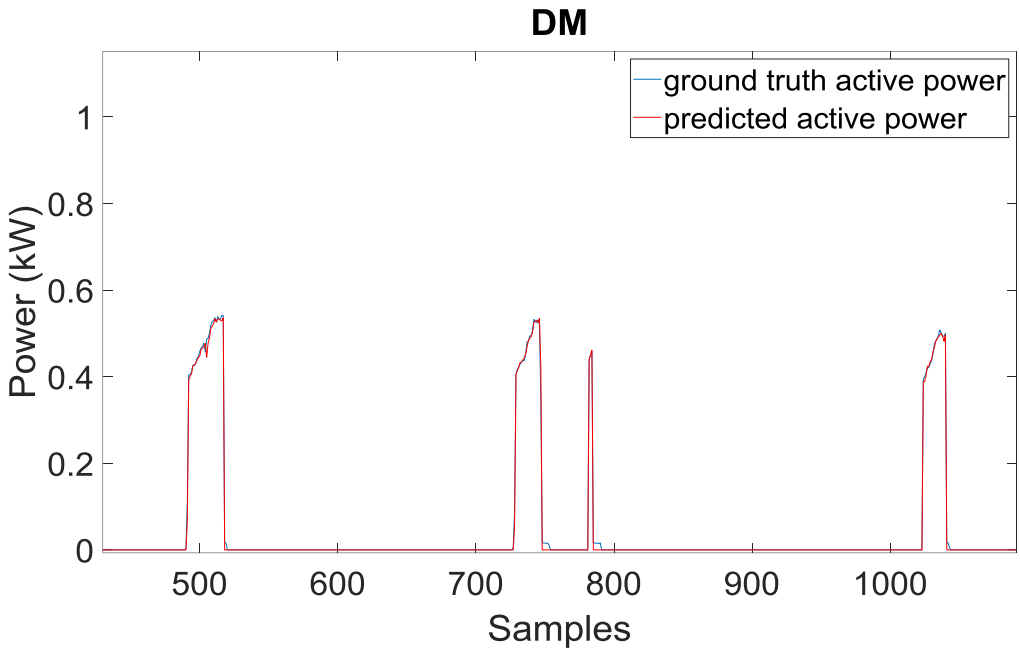


Figure 5-18. Drying Machine. Red, predicted active power, blue, actual active power.

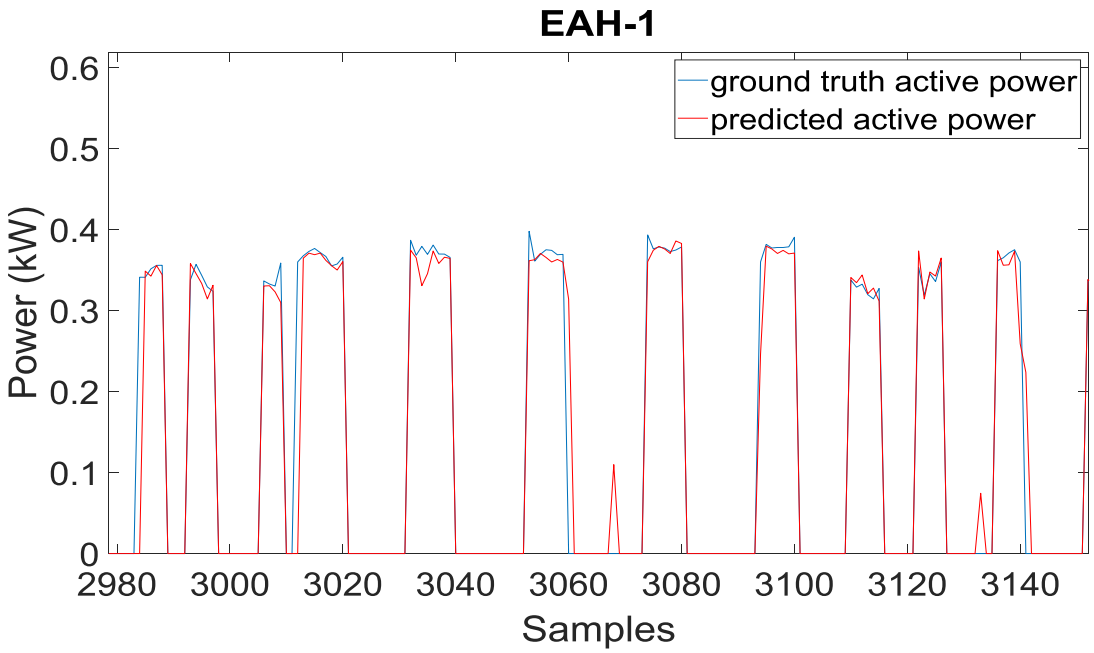


Figure 5-19. Electric air heater (EAH-1). Red, predicted active power, blue, actual active power.

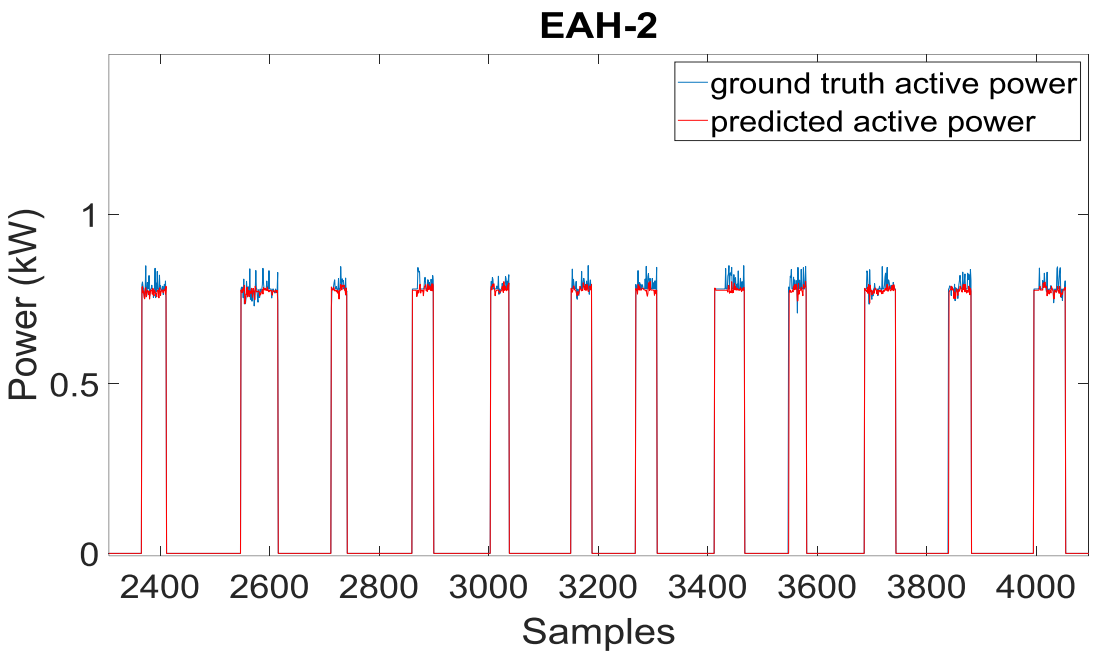


Figure 5-20. Electric air heater (EAH-2). Red, predicted active power, blue, actual active power.

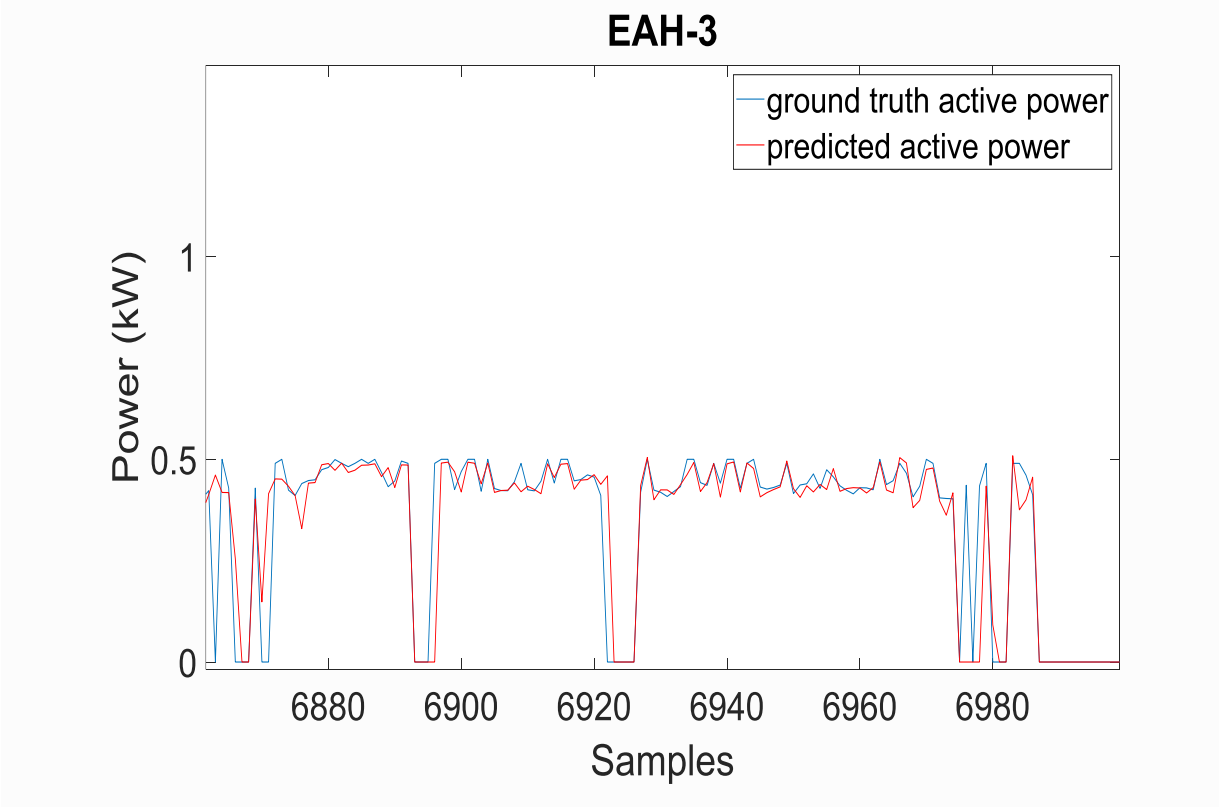


Figure 5-21. Electric air heater (EAH-3). Red, predicted active power, blue, actual active power.

Following the design of the models, an experiment to predict the energy consumption of some devices in the case study house using the designed models was conducted. The experiment considered only the aggregated data of January 2022. The energy disaggregation results

obtained using the designed models are presented in Table 36. The consumption distribution of the devices per phase is reported since the aggregated data were collected using the EM340 3-phase smart meter.

Table 36. Distribution of energy consumption in the case study house

Phase	Appliances	E(kWh)
I	Air conditioner (AC1)	11.97
	Air conditioner (AC2)	91.98
	Burner stove (BS1)	14.42
	Electric air heater (EAH1)	24.56
	Electric air heater (EAH3)	28.40
	Electric air heater (EAH2)	95.52
II	Drying machine (DM)	13.36
	Electric water heater (EWH)	114.54
	Fridge	17.80
	Washing machine (WM)	13.73
	Swimming pool pump (SPP)	153.14
	Oven	19.22
III	Air conditioner (AC3)	16.32
	Air conditioner (AC4)	71.74
	Burner stove (BS2)	19.27

From the results of the energy disaggregation presented in Table 36, the swimming pool pump (153.14 kWh) and the electric water heater (114.54 kWh), account for the highest energy consumption during the month of January 2022 in the case study house. The air conditioners (AC2 and AC4) and the electric air heater (EHA2) have consumed roughly 91.98 kWh, 71.74 kWh and 95.52 kWh respectively during the month of January 2022. The consumption of the remaining devices is estimated in the range of 4.22 kWh to 28.40 kWh. The distribution of electricity consumption in the case study house during the month of January 2022 is summarized in the pie charts presented in Figure 5-22.

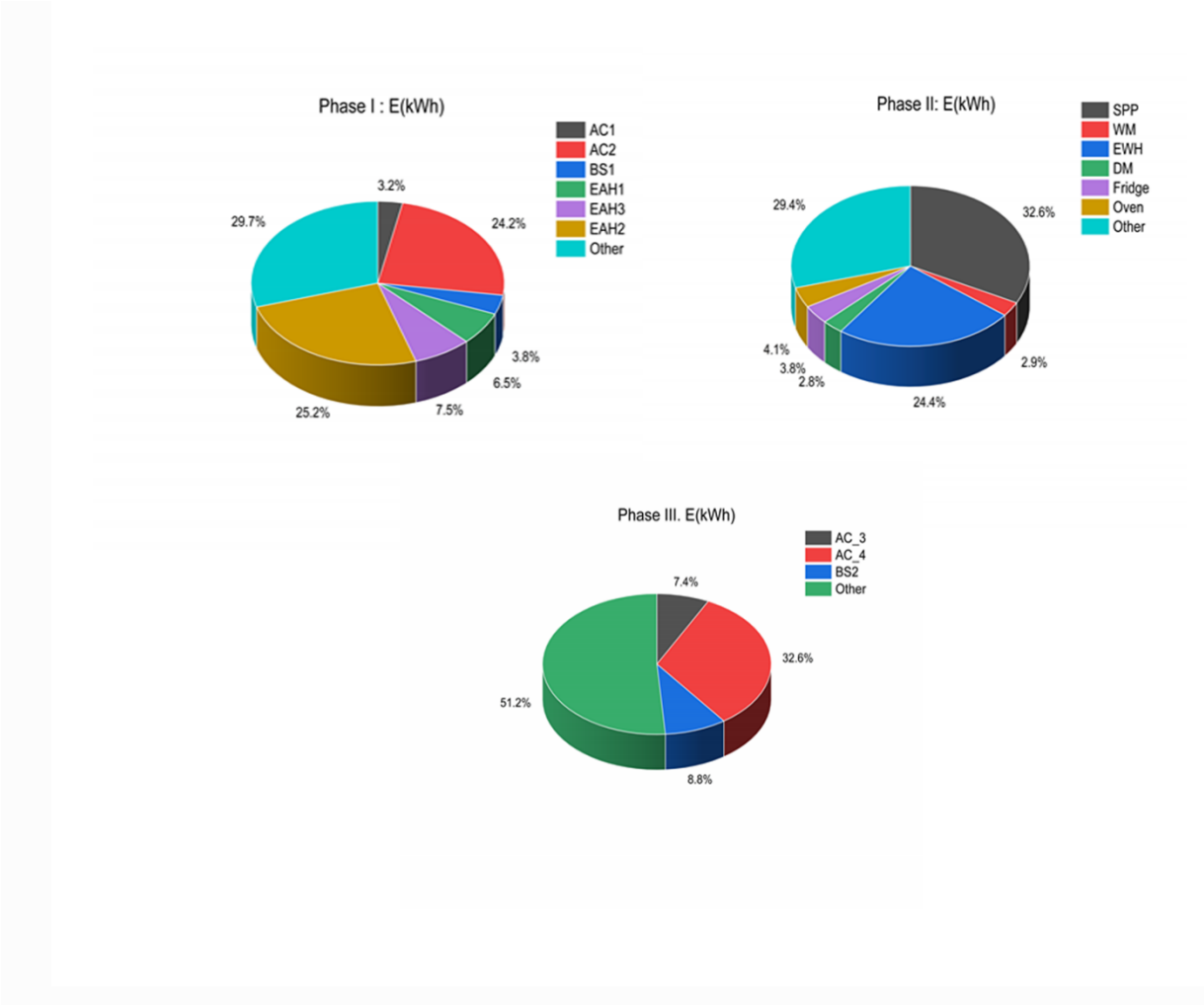


Figure 5-22. Distribution of electricity consumption in the case study house.

The overall consumption of the case study household during the month of January 2022 accounted for 1070 kWh, consisting of 380 kWh for phase I, 470 kWh for phase II and 220 kWh for phase III. The devices considered in the study represent 66% of the total monthly energy usage in the case study household. The majority of these devices are programmable besides being responsible for the largest consumption of electrical energy in the house, which means that it may be possible to modify their operations without creating too much inconvenience for the users. This is particularly the case for HVAC devices (Air conditioners (AC 1-4), Electric air heaters (EAH 1-3), Washing machine (WM), Drying machine (DM), Swimming pool pump (SPP), and to some extent Electric water heater (EWH)), which represent for 60% of the total household consumption. This means that there is a significant flexibility to design the electrical profile of the house to meet energy management objectives. Since the case study house is equipped with photovoltaic panels and an electricity storage system, the online

operation of certain of these electrical devices shall be considered in the new design of the model-based predictive control of the HEMS system proposed in [315].

5.8. Summary

This chapter introduced a low-complexity NILM framework for energy disaggregation based on radial basis function neural networks designed using a multi-objective genetic algorithm (MOGA). The approach involves the collection and selection of residential house data, the extraction of characteristics, the classification of appliances, and the estimation of energy. The data were acquired within a real-life scenario in a household located in Faro, Portugal, employing a low-frequency sampling rate. The proposed framework demonstrated an excellent ability to disaggregate the appliance specific consumption while reducing the data sampling from 1 second to 1 minute to enable the usage of low complexity models (a few hundred parameters) and of low-cost meters.

Moreover, a comparative analysis was conducted with other computational intelligence classifiers in order to identify the operating states of the appliances. The experiment focused on implementing and testing on the same data as those used for the proposed framework, the classification algorithms such as SVM, DT, KNN, LSTM, and CNN. The analysis of the comparative results demonstrated that the proposed framework performed better in terms of identifying the operating states of the appliances. Furthermore, the effectiveness of the proposed framework in obtaining the best estimate of the energy consumed by each appliance in the household was highlighted by comparing it with other state-of-the-art techniques using both distinct and common data.

The proposed NILM framework enabled the disaggregation of the energy consumption of appliances representing 2/3 of the total electrical consumption of the case study household over a one-month period. Additionally, it revealed that nearly 60% of the consumption was associated with programmable devices, providing more flexibility to the HEMS deployed in the house.

Chapter 6 Application of MOGA design to a large amount of data

6.1. Introduction

In this chapter, we will analyze and discuss the application of the MOGA design on a long period of data. Since electricity consumption is influenced by various socio-economic and environmental factors, the distribution of electricity consumption data is not evenly distributed over time and within the consumption pattern. Several works have shown that existing NILM solutions are not well suited for real-world applications due to their low disaggregation accuracy over a larger amount of new data [198]. Indeed, non-electric elements also influence energy consumption in buildings. Energy use patterns, for example, are influenced by building and consumer factors such as number of occupants, housing type, and consumer behavior. In addition, the performance of machine learning algorithms tends to degrade in new contexts, for instance when usage patterns are not similar to those on which the algorithms were trained. Meanwhile, the generalizability and transferability of the models within an ever-changing data distribution is a challenging task for NILM algorithms. Therefore, the ability of the models to adapt to these changes and, obviously, to other houses without the need for intensive retraining should be considered.

To tackle this challenge, we will analyze the application of the proposed MOGA approach designed using a small amount of data (3 weeks) to a long period of data (one year). The remainder of the chapter is structured as follows: Section 6.2 presents a data analysis of the case study house. The results of the experiments conducted are presented in section 6.3 and a discussion in section 6.4. Section 6.5 summarizes the chapter.

6.2. Data Analysis

As previously mentioned, data are being collected in the case study house since November 2019 until today (2023). The data are recorded with a resolution of one second which represents several tens of millions of records. We have considered a year of data from May 2020 to May 2021 to analyze the electricity consumption trend in the case study household. Figure 6-1 presents the total monthly consumption in the case study house during one year.

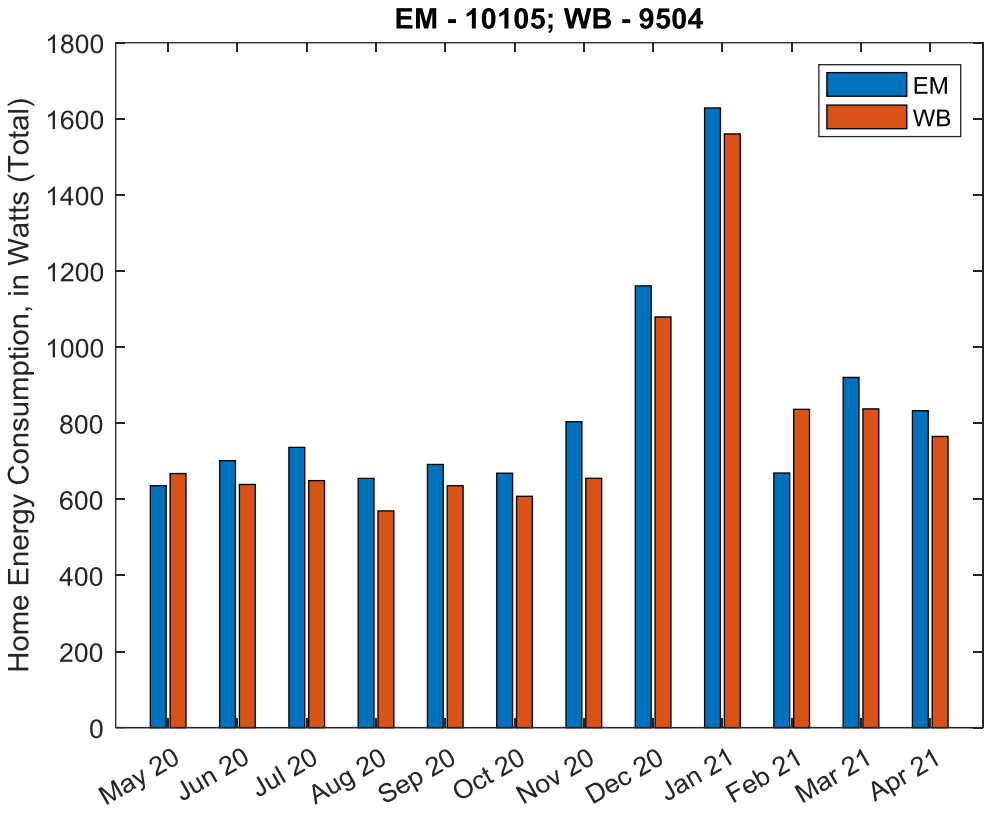
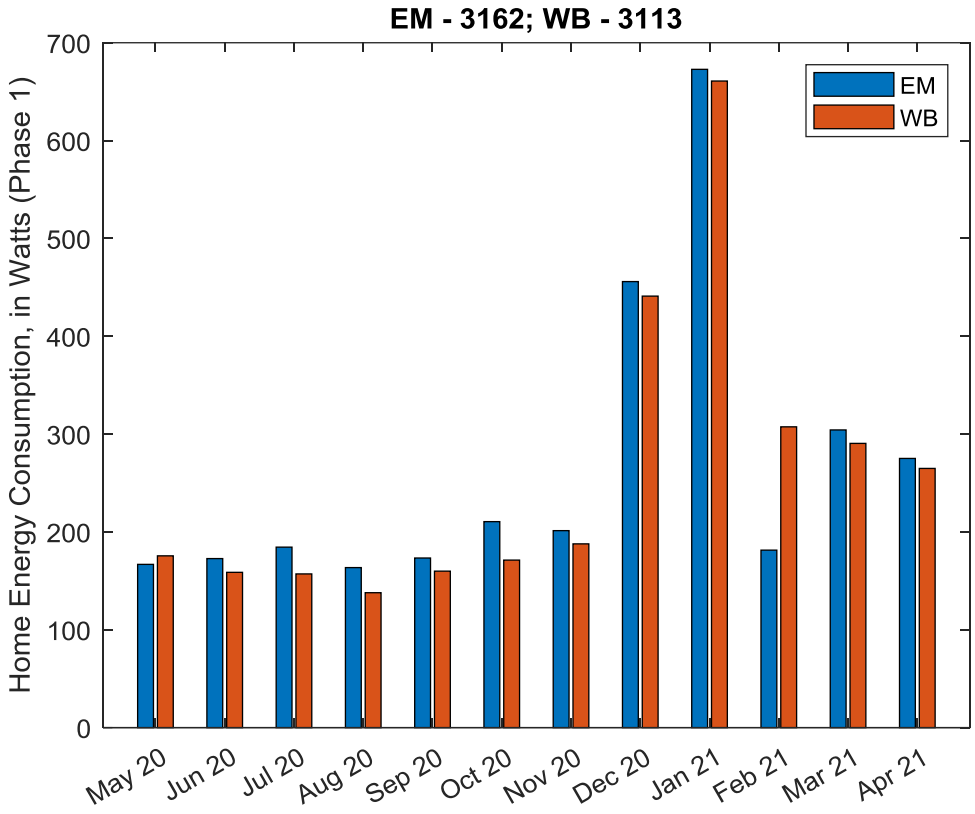
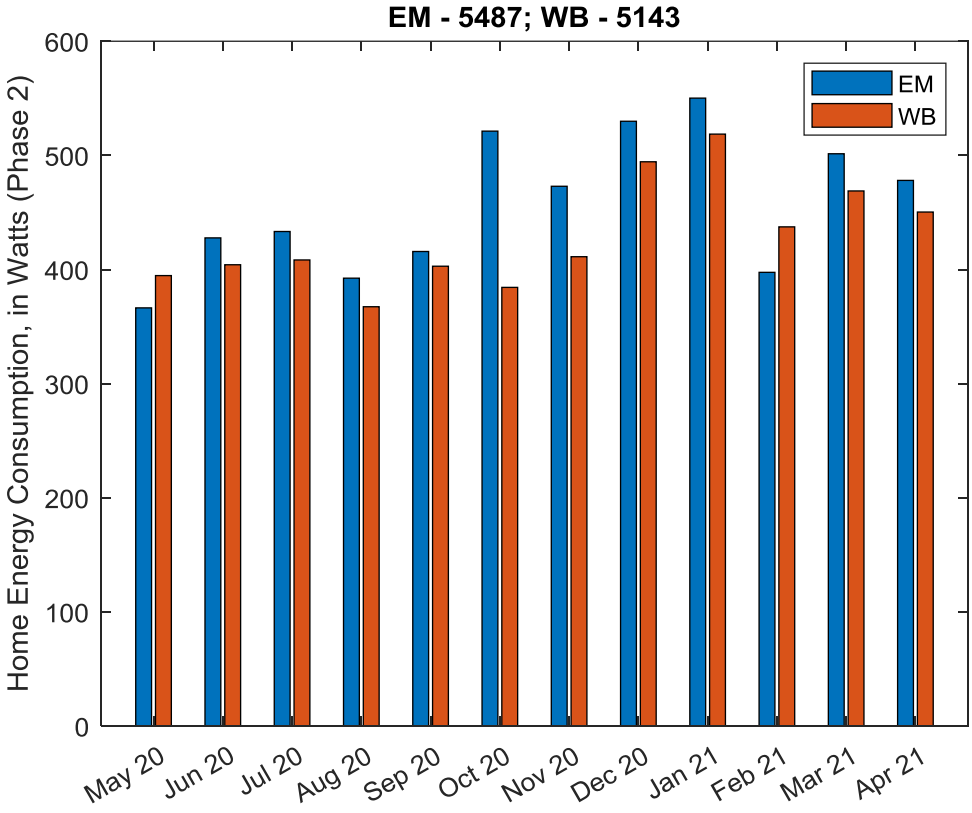


Figure 6-1: Monthly electricity consumption in the case study house. EM represents the data measured by the tri-phasic smart meter and WB (Wibee) represents the data collected by the different circuit breakers.

As it can be seen, the monthly consumption varies from month to month with a large increase in December and a peak in January. This can be explained by the meteorological conditions, which record very low temperatures during these months. Since the smart meter used is a three-phase meter, we can observe the electricity consumption related to each phase. Figure 6-2 depicts histograms of monthly electricity consumption related to each phase during the year.


EM - 3162; WB - 3113
Home Energy Consumption, in Watts (Phase 1)
0
100
200
300
400
500
600
700
EM
WB
May 20
Jun 20
Jul 20
Aug 20
Sep 20
Oct 20
Nov 20
Dec 20
Jan 21
Feb 21
Mar 21
Apr 21

(a)


EM - 5487; WB - 5143
Home Energy Consumption, in Watts (Phase 2)
0
100
200
300
400
500
600
EM
WB
May 20
Jun 20
Jul 20
Aug 20
Sep 20
Oct 20
Nov 20
Dec 20
Jan 21
Feb 21
Mar 21
Apr 21

(b)

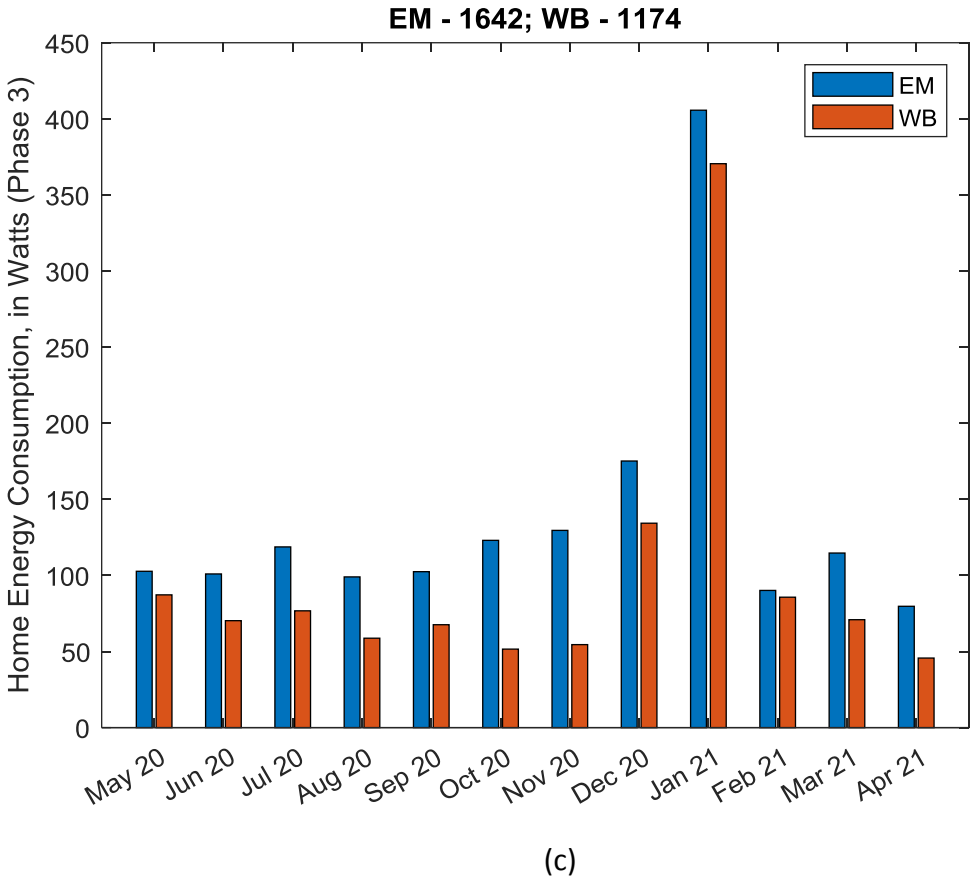


Figure 6-2. Monthly electricity consumption in the case study house by phase. (a) Phase 1, (b) Phase 2, (c) Phase 3.

The trend of consumption differs from one phase to another. This is because different loads are attached to each phase (see Table 4 in chapter 4). The consumption of phases 1 and 3 show the same peaks as the total consumption while the consumption related to phase 2 exhibits a slightly different profile. It can also be observed that the devices related to phase 2 are responsible for the consumption of more than 53% of the total consumption while phase 1 and phase 3 represent roughly 31% and 16% respectively. As already indicated in chapter 4 section 4.3 in table 5 the circuit breakers or wibees (1, 4, 7, 10, 13 and 16) are associated with phase 1, the wibees (2, 6, 8, 11, 15 and 17) are associated with phase 2 and the wibees (3, 5, 9, 12, 14 and 18) with phase 3. Figure 6-3 presents the distribution of the monthly consumption measured by the circuit breakers associated with phase 1. The monthly distribution of electricity consumption measured by the circuit breakers associated with phase 2 and phase 3 are presented in figure 6-4 and figure 6-5 respectively.

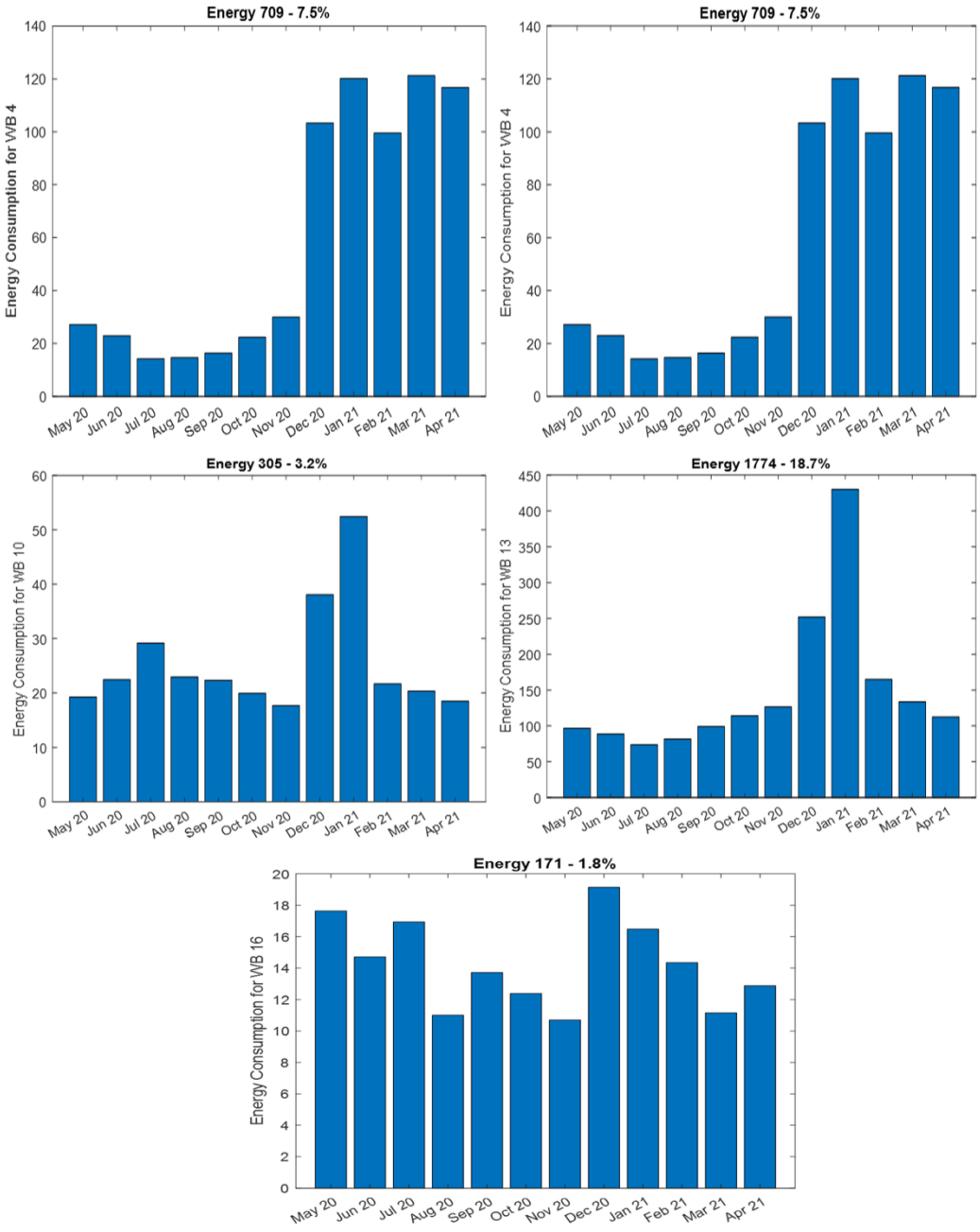


Figure 6-3. Monthly electricity consumption in the case study house by circuits breakers (phase 1).

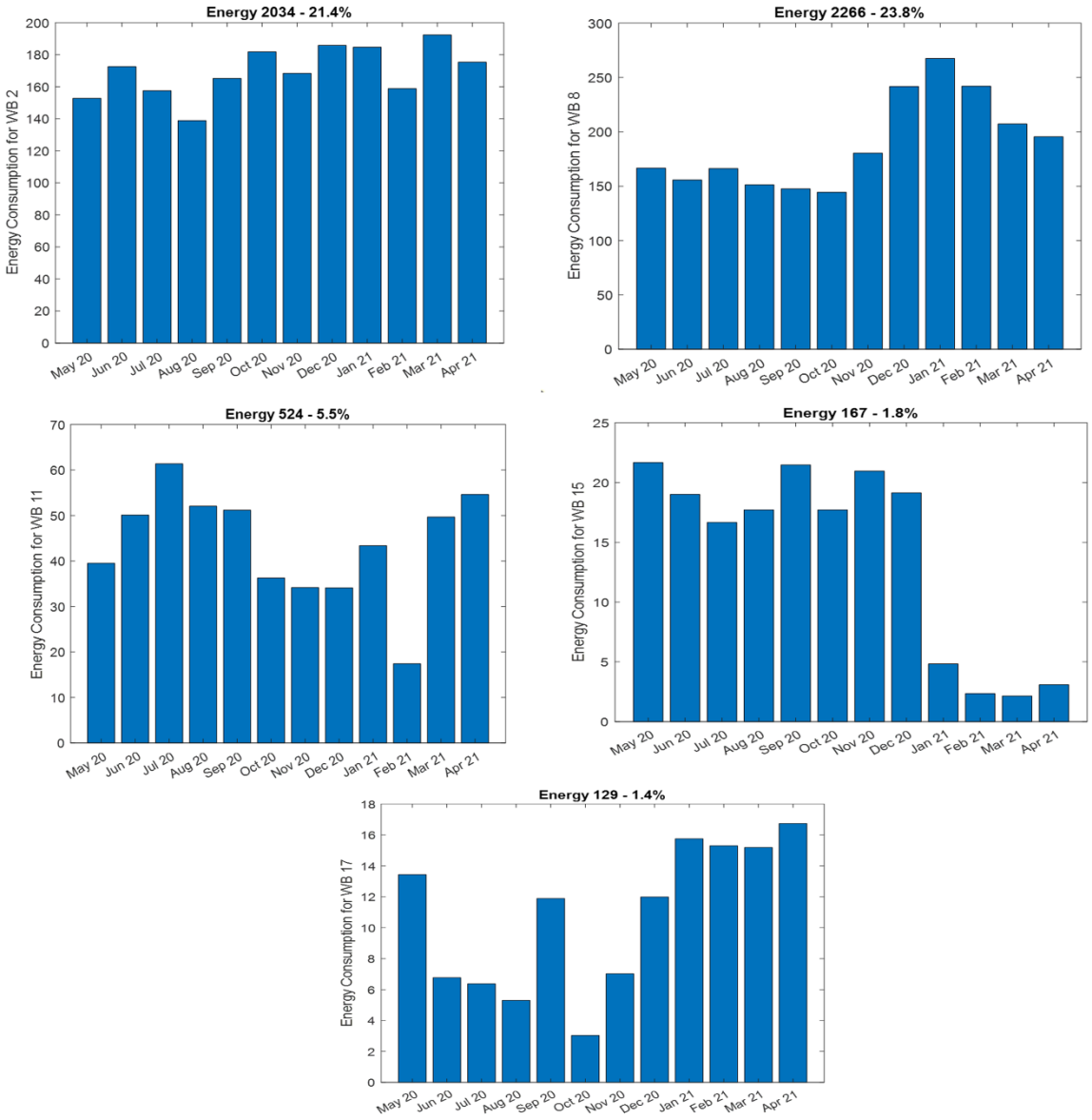


Figure 6-4. Monthly electricity consumption in the case study house by circuits breakers (phase 2).

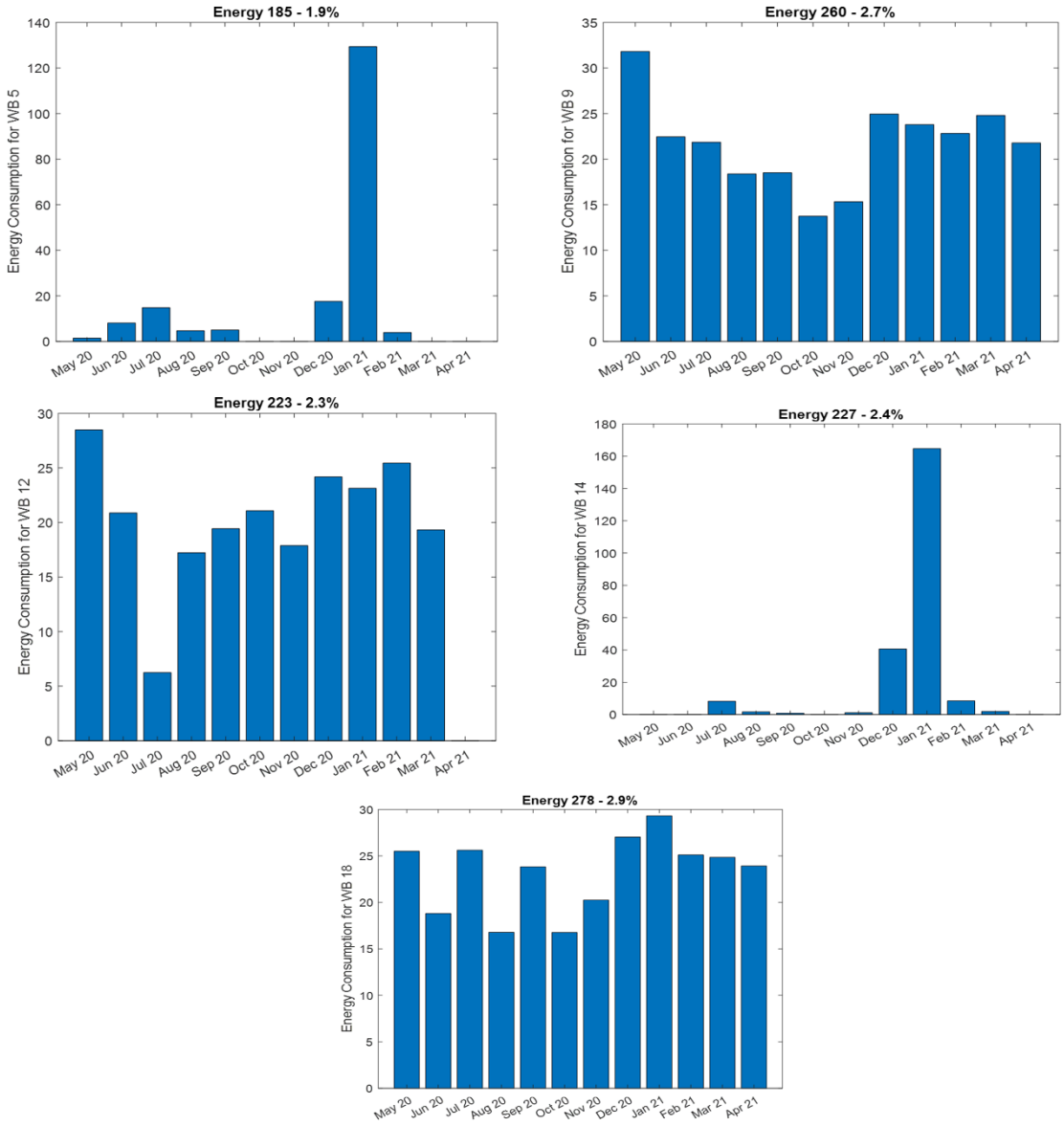


Figure 6-5. Monthly electricity consumption in the case study house by circuits breakers (phase 3).

The distribution of devices by circuit breakers is already presented in table 4, section 4.3 of chapter 4. It should be noted that the circuit breakers (wibees 1, 3 and 6) are not represented because they are responsible for an insignificant part of the consumption (less than 1% of the house consumption). This is because the devices attached to these circuit breakers do not consume much energy (e.g. alarm for wibee 1, lighting for wibee 3 and motors for wibee 6).

6.3.Experiments and Results

In this experiment, we considered the one year of data. A set of appliances within the range of the most consuming ones in the case study house and attached to individual circuit breakers are

used. These include air conditioners, burner stoves, swimming pool pump, and ovens. We have given the one year of aggregated data to the MOGA previously trained with the 3 weeks of data (see chapter 5). The purpose is to identify the operating states of the appliances. The classification results are reported in Table 37.

Table 37. Classification results on one year of data

Devices	Recall	Precision	F1
AC_1	0.50	0.86	0.63
AC_2	0.19	0.29	0.23
AC_3	0.88	0.35	0.51
AC_4	0.95	0.85	**0.90**
BS_1	0.50	0.27	0.35
BS_2	0.24	0.89	0.37
SPP	0.99	0.50	0.67
Oven	0.68	0.76	0.72

6.4. Discussion

The analysis of the results reported in Table 37 reports that the overall performance of the models decreased when applying the models on a large data distribution. Meanwhile, the models were able to identify the air conditioners with an F1 score ranging from 23% for AC_2, 51% for AC_3, 63% for AC_1 and 90% for AC_4. For the burner stoves, the models identified BS_1 with an F1 score of 35% and BS_2 with an F1 score of 37%. The swimming pool pump (SPP) and the oven were identified with an F1 score of 67% and 72% respectively. These low performances, especially for ACs and BSs, can be explained by the fact that these devices are occasionally used in the case study house, resulting in a low number of activations and a high number of false positives. The devices also have a similar consumption pattern that makes it challenging to identify when two of the same type are operating at the same time. Moreover, the aggregate energy consumption distribution changes drastically depending on, for instance, the months of the year, the occupancy of the house, etc., thus changing the consumption pattern and leading to the deterioration of the performance of NILM models.

The MOGA models were designed using the data selection approach, which consists in selecting the most informative convex points from the training set. We analyzed the distributions of the convex points in the training set and the one-year test data, considering only one delay for the active and reactive powers. Figure 6-6 shows the sets of data covered by the training set and the full one year of data, and the distribution of false positives detected in the one year of data for Burner stove.

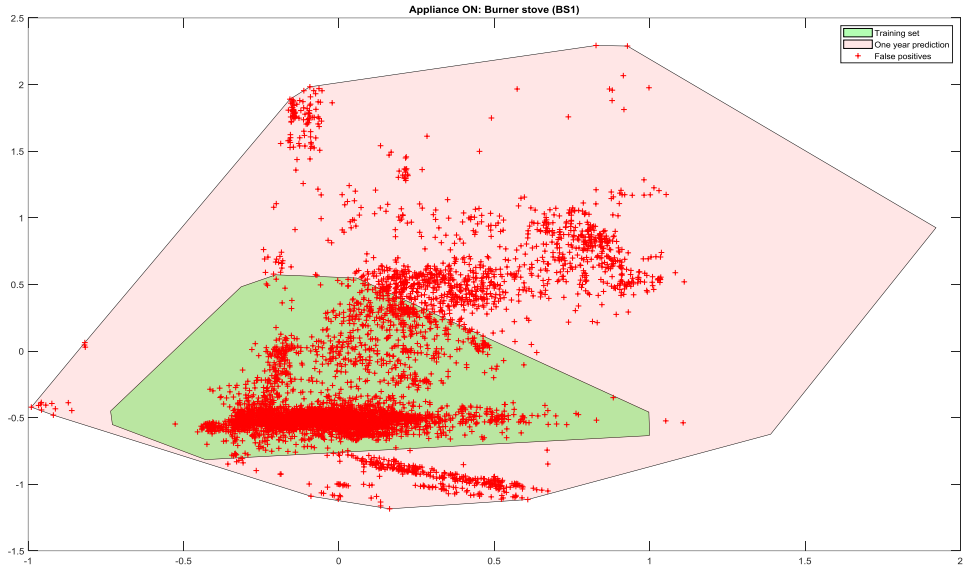


Figure 6-6. Convex points in the training set and the distribution of false positives

Analyzing the figure 6-6 we can see that the convex points of the training set cover only a part of the data when considering the one year data. The features patterns used to train the models have changed radically which leads to a high number of false positives. One alternative is to consider a wider distribution of data to train the models in order to capture the maximum range of feature patterns.

6.5. Summary

This chapter presented the application of MOGA designed models on a long period of data. We first conducted a statistical analysis in the case study house. Following this, the experiment focused on data from one-year period. The results showed the limitation of the models to detect certain types of devices with good performance over a long period of data. There are several reasons for these limitations including the amount of training data representing only a small fraction of the appliance consumption patterns, meteorological and socio-economic factors etc.

Chapter 7 Conclusions and future work

This chapter summarizes the content of this thesis and discusses the limitations of the proposed work as well as the prospects for future work in the deployment and application of NILM systems.

7.1. Conclusions

In this thesis work, computational intelligence techniques for NILM were studied. In fact, the total electricity consumption is usually collected at a single point, usually the interface between the supplier and the customer. Then, using computational intelligence techniques, this aggregated consumption is breakdown into the consumption of specific device attached to the household electric panel, providing the consumer with detailed electricity consumption by device. Device-specific information has been shown to be an effective tool for enhancing energy efficiency at the residential level, which is of the utmost importance in the current context of climate change and environmental concerns. Moreover, this information would help homeowners to understand how their house consumes energy, allowing them to make wise choices about how to reduce energy consumption. This is generally thought to be a very individual and dynamic decision-making process that would consider a variety of factors, including the characteristics of the house, the homeowners' comfort level, financial constraints, etc. Furthermore, device-specific power consumption is also beneficial in several areas due to its applicability to home energy management systems, fault detection, ambient assisted living, human activity recognition, remote load monitoring services, etc.

Within all the approaches explored by the research community, this thesis focused on the optimization of deep learning models and the design of radial basis function neural networks using the multi-objective genetic algorithm for non-intrusive load monitoring, as their capabilities and promising performance are at the forefront of feasible enhancements. Since the primary challenge of NILM involves the disaggregation of energy, the research work of this thesis can be summarized in the following points.

In Chapter 1, the research context marked by the concerns regarding the increasing global energy consumption, the alarming decline of energy resources and climate change was discussed. Then, the importance of NILM technology to optimize the energy use in the

residential sector was highlighted and its main challenges were briefly discussed. The research objectives of this research work were then introduced.

In Chapter 2, a detailed and comprehensive overview of the current state of the art of NILM is presented. First, a brief introduction to the concept of load monitoring is presented. Next, the problem of energy disaggregation has been duly stated, followed by a review of the original NILM framework. Then, the main components of a NILM system were studied. The data collection process, including low frequency data collection and high frequency data collection, was presented while highlighting the relationship between sampling frequency and device signatures. Following this, the background and research on device electrical signatures, particularly steady-state, transient, and non-traditional signatures, was discussed. Finally, a review of the main evaluation metrics typically employed and a survey of the different public datasets available in the literature were outlined. The study exposed a number of observations, including the fact that the majority of studies have focused on exploring device signatures, and that there is still a need to create a comprehensive set of features that can accurately describe all forms of signals. An alternative is to use non-energy features that could be mixed with the energy features to allow for more accurate detection. Regarding the evaluation metrics, there is no consensus on the most appropriate evaluation metric to use in order to report the success of a NLM algorithm. Therefore, the selection of the most appropriate evaluation metrics seems to depend on the method developed. Regarding the publicly available datasets, although several datasets have been proposed in the literature, there are some disadvantages that limit their practical use, including different data formats, missing data, covering only some developed countries, and limited labeling data. Another challenge for state-of-the-art NILM approaches is the disaggregation of multi-state devices, which seems to be much more difficult than the disaggregation of two-state devices.

In Chapter 3, a brief overview of the theoretical background for the development of this thesis work is presented. First, the basic concepts of machine learning methods were introduced. Then, the methods used, CNNs, LSTMs, SVMs, KNNs, and DTs, are briefly described. Since the MOGA is used as a framework to define the parameters and input features of RBFNN, its associated parameters and input characteristics, given the specified objectives, are studied. Finally, an overview of the data selection approach used was outlined.

In chapter 4, machine learning and deep learning-based approaches for NILM were tackled. Many studies in literature have focused on HMMs and their variants generally operating on low

frequency data. However, it has been shown that the complexity of these models grows exponentially as the number of target devices increases. Moreover, their scalability and generalization capabilities are not very promising. Also, the existing inference techniques for state estimation are extremely sensitive to local optima, which limits their application in the real world. Thus, since the complexity of the NILM problem increases exponentially with the number of devices, it is necessary to address the complexity of the model, e.g., by optimizing the solver or by reducing the dimensionality of the input. In this perspective, one of the contributions of this thesis is the proposal of an optimization approach based on a hybrid deep learning architecture and a convex hull data selection approach. To tackle the problem of scarcity and quality of data sets discussed in Chapter 2, low frequency power data were collected in a typical house as part of a real-life scenario in Montenegro, Portugal. The case study house is equipped with a multitude of households load and several variables are monitored. A rigorous analysis of the different electrical features was conducted and thus the aggregated active and reactive powers sampled at 1 Hz were adopted as input to the models. The data selection approach was to select the most informative vertices of the real convex hull by randomly approximating the convex hull. The proposed hybrid deep learning architecture is composed of a classification model based on a CNN jointly trained with an estimation model based on a bidirectional LSTM model. The results achieved demonstrated the efficiency of the proposed approach, reaching F1 values up to 99% and estimation accuracy values up to 98%. It was shown that using the proposed data selection approach significantly improves the performance of the designed model, especially for multi-state loads. Furthermore, the comparative analysis conducted has shown that the proposed optimization approach accurately disaggregates the target appliances with better power estimation reliability and F1 score than the state-of-the-art methods.

Chapter 5 is devoted to the proposal of an energy disaggregation algorithm. In fact, the quest for low sampling rate disaggregation algorithms to enable potential compatibility with widely available residential smart meters is a trend in the NILM sector. Recent developments in deep learning techniques have increased interest in NILM research, with extremely encouraging results, as exposed in Chapter 4. However, to achieve satisfactory performance, these techniques need a lot of training data. This is a significant challenge for NILM algorithms due to the scarcity of high-quality labels and durable datasets. Moreover, these methods benefit greatly from a large number of training parameters, which in most cases require expensive or not readily available processing capacity. Furthermore, NILM algorithms must meet a range of

criteria, such as sufficient accuracy to identify the device's operating state, low-complexity models, and low-cost hardware. In order to tackle these challenges, we proposed a NILM framework based on a low resolution data, thus enabling the use of low-cost meters, and using shallow and low complexity NN models. The framework is based on the design of RBFNN-MOGA, with design data selected by an approximate convex hull algorithm. Many experiments were conducted. The first experiment consisted of designing disaggregation models for devices exposing a range of power characteristics, from two-state devices to complex characteristics of multi-state devices. The proposed RBFNN-MOGA framework achieved very satisfactory performance, with models of very low complexity and operating on one-minute data. A comparative analysis was conducted to evaluate the RBFNN-MOGA framework against the optimization approach proposed in Chapter 4. The results of the analysis showed that the RBFNN-MOGA model does not require too much training data, while achieving similar or better performance than approaches using more training data, therefore the down sampling of the data sampling rate from 1 second to 1 minute did not affect the performance of the RBFNN-MOGA framework and considerably reduced the amount of data to process. The second experiment consisted in using the same data (the data used for the RBFNN-MOGA design) to assess the proposed framework versus the state-of-the-art classification methods. Several classifiers including SVM, DT, KNN, CNN and LSTM have been implemented. The results of the comparative analysis showed that the RBFNN-MOGA framework outperformed the implemented methods in terms of identification of device operating states. Another experiment was conducted in the public dataset AMPD. The RBFNN-MOGA framework was used to train the models. A comparative study was conducted to benchmark the proposed framework with state-of-the-art approaches using the public APMD data. The results of the experiment showed that the RBFNN-MOGA framework performed better than the state-of-the-art approaches on the same datasets. The experiment was then extended to other appliances in the case study household. Considering one month of data, the RBFNN-MOGA framework allowed to identify the most energy consuming appliances (the swimming pool pump with 153,14 kWh and the electric water heater with 114,54 kWh during the month of January 2022). Moreover, it enabled the disaggregation of the energy consumption of devices that were responsible for 66% of the total electrical consumption of the case study household over a one-month period. Furthermore, it revealed that nearly 60% of the consumption was associated with schedulable appliances, thus providing greater flexibility to the HEMS system deployed in the house.

In Chapter 6, the application of the MOGA design over a long period of data is tackled. First, a statistical analysis was conducted in the case study household considering a long period of data to highlight the trend and consumption patterns of the house. Then, the experiment was conducted using one year's data. The results, including the performance and limitations of the models, were discussed. It was shown that the features patterns of the house consumption are influenced by several factors, which leads to different appliance signatures resulting in the deterioration of the models' performance. Most existing NILM models are designed to learn to disaggregate data that are faithful to the target device data used for validation, leveraging adequate and varied training data. However, this constitutes a limitation for NLM models, making it challenging to adapt them to new data. One alternative is to consider invariant features that are more likely to generalize.

In summary, the methods proposed in this thesis have proven to be effective in addressing some of the challenges outlined in Chapter 1 regarding NILM algorithms. The proposed methods could be further investigated from both a scientific and practical application perspective.

7.2. Future works

The research conducted in this thesis has led to some intriguing ideas for further scientific study. One of the challenges of current NILM research is the lack of labeled datasets for a comprehensive evaluation and comparative analysis of NILM approaches. This shortcoming is mainly due to the fact that the existing labeling procedure still relies on laborious, time-consuming, and error-prone manual inspection, which prevents the formation of alternative labeled datasets. Future research should take this into account and consider the development of labeling tools for NILM datasets that would allow users to review automatically created annotations, thereby significantly reducing the workload associated with human annotation. In fact, the study of disaggregation algorithms should focus on the robustness, disaggregation reliability, and scalability of NILM for a larger number of devices so that the detailed device consumption information provided by the available smart meters becomes more tangible.

The discussions in this thesis work showed that computational intelligence techniques achieve very high performance when evaluated on the same data domain. However, further research will be needed, especially to improve the portability of the NILM architecture, but also to acquire more knowledge about invariant feature extraction, in order to generalize the resulting models to other houses. One of the main points we want to study is the state duration,

which represents the probability that a load remains in a given state for a given amount of time. This can help to enhance the classification of loads operating at similar power consumption by introducing an additional singularity.

Finally, although the disaggregation algorithms are primarily focused on single-point electrical energy use, a possible direction of research would be to explore other features such as time of the day, day type (weekday, weekend, or holiday), season and/or local weather information, devices combinations (e.g., stoves and hoods are frequently used together), and occupant preferences [30]. For instance, the average outdoor air temperature and consequently the season have a strong correlation with HVAC system consumption. However, the effort to consider all of these parameters will be a significant challenge. It is also worth noting that the information extracted from NILM techniques has the potential to be used in other applications such as home energy management systems, energy efficiency recommendation systems, fault diagnosis applications, ambient assisted living, recognition of activities of daily living, etc.

References

[1] S. Wang, Q. Li, C. Fang, and C. Zhou, "The relationship between economic growth, energy consumption, and CO2 emissions: Empirical evidence from China," *Science of the Total Environment*, vol. 542, pp. 360–371, Jan. 2016, doi: 10.1016/j.scitotenv.2015.10.027.

[2] S. Shafiei and R. A. Salim, "Non-renewable and renewable energy consumption and CO2 emissions in OECD countries: A comparative analysis," *Energy Policy*, vol. 66, pp. 547–556, 2014, doi: 10.1016/j.enpol.2013.10.064.

[3] M. Salari, R. J. Javid, and H. Noghanibehambari, "The nexus between CO2 emissions, energy consumption, and economic growth in the U.S.," *Econ Anal Policy*, vol. 69, pp. 182–194, 2021, doi: 10.1016/j.eap.2020.12.007.

[4] L. Wen and Y. Cao, "Influencing factors analysis and forecasting of residential energy-related CO2 emissions utilizing optimized support vector machine," *J Clean Prod*, vol. 250, 2020, doi: 10.1016/j.jclepro.2019.119492.

[5] J. L. Fan, B. Zeng, J. W. Hu, X. Zhang, and H. Wang, "The impact of climate change on residential energy consumption in urban and rural divided southern and northern China," *Environ Geochem Health*, vol. 42, no. 3, pp. 969–985, 2020, doi: 10.1007/s10653-019-00430-3.

[6] T. S. Adebayo and D. Kirikkaleli, "Impact of renewable energy consumption, globalization, and technological innovation on environmental degradation in Japan: application of wavelet tools," *Environ Dev Sustain*, no. 0123456789, 2021, doi: 10.1007/s10668-021-01322-2.

[7] K. Dong, G. Hochman, Y. Zhang, R. Sun, H. Li, and H. Liao, "CO2 emissions, economic and population growth, and renewable energy: Empirical evidence across regions," *Energy Econ*, vol. 75, pp. 180–192, 2018, doi: 10.1016/j.eneco.2018.08.017.

[8] R. Warren, J. Price, J. VanDerWal, S. Cornelius, and H. Sohl, "The implications of the United Nations Paris Agreement on climate change for globally significant biodiversity areas," *Clim Change*, 2018, doi: 10.1007/s10584-018-2158-6.

[9] M. A. R. Lopes, C. H. Antunes, and N. Martins, “Energy behaviours as promoters of energy efficiency: A 21st century review,” *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 4095–4104, 2012, doi: 10.1016/j.rser.2012.03.034.

[10] Hannah Ritchie and Max Roser, “CO_2 and Greenhouse Gas Emissions,” *Our World in Data*, 2020. https://ourworldindata.org/co2-and-other-greenhouse-gas-emissions (accessed Sep. 21, 2021).

[11] Q. Zhai, T. Li, and Y. Liu, “Life cycle assessment of a wave energy converter: Uncertainties and sensitivities,” *J Clean Prod*, vol. 298, p. 126719, 2021, doi: 10.1016/j.jclepro.2021.126719.

[12] T. Balezentis, “Shrinking ageing population and other drivers of energy consumption and CO2 emission in the residential sector: A case from Eastern Europe,” *Energy Policy*, vol. 140, no. March, p. 111433, 2020, doi: 10.1016/j.enpol.2020.111433.

[13] I. Santiago, M. A. Lopez-Rodriguez, D. Trillo-Montero, J. Torriti, and A. Moreno-Munoz, “Activities related with electricity consumption in the Spanish residential sector: Variations between days of the week, Autonomous Communities and size of towns,” *Energy Build*, vol. 79, pp. 84–97, Aug. 2014, doi: 10.1016/j.enbuild.2014.04.055.

[14] L. Shen, B. He, L. Jiao, X. Song, and X. Zhang, “Research on the development of main policy instruments for improving building energy-efficiency,” *J Clean Prod*, vol. 112, no. 2016, pp. 1789–1803, 2016, doi: 10.1016/j.jclepro.2015.06.108.

[15] U. S. Briefing, “International energy outlook 2013,” *US Energy Information Administration*, vol. 506, p. 507, 2013.

[16] K. Dong, X. Dong, and Q. Jiang, “How renewable energy consumption lower global CO2 emissions? Evidence from countries with different income levels,” *World Economy*, vol. 43, no. 6, pp. 1665–1698, 2020, doi: 10.1111/twec.12898.

[17] (DOE) US Department of Energy, “An Assessment of Energy Technologies and Research-Chapter5,” *Enabling Modernization of the Electric Power System - Technology Review*, 2015.

[18] L. Pérez-Lombard, J. Ortiz, and C. Pout, “A review on buildings energy consPérez-Lombard, L., Ortiz, J., & Pout, C. (2008). A review on buildings energy consumption information. Energy

and Buildings, 40(3), 394–398. http://doi.org/10.1016/j.enbuild.2007.03.007umption information," *Energy Build*, 2008.

[19] E. Aydin and D. Brounen, "The impact of policy on residential energy consumption," *Energy*, 2019, doi: 10.1016/j.energy.2018.12.030.

[20] D. D'Agostino, B. Cuniberti, and P. Bertoldi, "Energy consumption and efficiency technology measures in European non-residential buildings," *Energy Build*, 2017, doi: 10.1016/j.enbuild.2017.07.062.

[21] E.-E. Commission, "Communication from the Commission Action Plan for Energy Efficiency: Realising the Potential (Energy Efficiency Action Plan)," *COM (2006)*, vol. 545, 2006.

[22] IEA, "Electricity Information: Overview," Paris, 2021. Accessed: Sep. 28, 2021. [Online]. Available: https://www.iea.org/reports/electricity-information-overview

[23] M. E. Berges, E. Goldman, H. S. Matthews, and L. Soibelman, "Enhancing Electricity Audits in Residential Buildings with Nonintrusive Load Monitoring," *Applications and Implementation*, vol. 14, no. 5, doi: 10.1111/j.1530-9290.2010.00280.x.

[24] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? The case of electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013, doi: 10.1016/j.enpol.2012.08.062.

[25] C. Fischer, "Feedback on household electricity consumption: A tool for saving energy?," *Energy Effic*, vol. 1, no. 1, pp. 79–104, 2008, doi: 10.1007/s12053-008-9009-7.

[26] D. Parker *et al.*, "How Much Energy Are We Using ? Potential of Residential Energy Demand Feedback Devices," *Solar Energy*, pp. 1665–06, 2006, [Online]. Available: http://www.fsec.ucf.edu/en/publications/pdf/FSEC-CR-1665-06.pdf

[27] S. Darby, "The effectiveness of feedback on energy consumption," *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, vol. 486, no. 2006, p. 26, 2006.

[28] J. Kelly and W. Knottenbelt, "Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature," 2016.

[29] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, “Is disaggregation the holy grail of energy efficiency? The case of electricity,” *Energy Policy*, vol. 52, pp. 213–234, Jan. 2013, doi: 10.1016/j.enpol.2012.08.062.

[30] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. Garcia, “NILM Techniques for Intelligent Home Energy Management and Ambient Assisted Living: A Review,” *Energies (Basel)*, vol. 12, no. 11, p. 2203, Jun. 2019, doi: 10.3390/en12112203.

[31] R. G. Pratt *et al.*, “The smart grid: an estimation of the energy and CO2 benefits,” Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2010.

[32] K. Ehrhardt-Martinez, K. A. Donnelly, and J. A. Laitner, “Advanced metering initiatives and residential feedback programs: A meta-review for household electricity-saving opportunities,” 2010.

[33] B. Völker, A. Reinhardt, A. Faustine, and L. Pereira, “Watt’s up at home? Smart meter data analytics from a consumer-centric perspective,” *Energies*, vol. 14, no. 3. 2021. doi: 10.3390/en14030719.

[34] Y. Wang, Q. Chen, T. Hong, and C. Kang, “Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges,” *IEEE Trans Smart Grid*, vol. 10, no. 3, 2019, doi: 10.1109/TSG.2018.2818167.

[35] I. Abubakar, S. N. Khalid, M. W. Mustafa, H. Shareef, and M. Mustapha, “Application of load monitoring in appliances’ energy management – A review,” *Renewable and Sustainable Energy Reviews*, vol. 67, pp. 235–245, Jan. 2017, doi: 10.1016/j.rser.2016.09.064.

[36] P. A. Lindahl, D. H. Green, G. Bredariol, A. Aboulian, J. S. Donnal, and S. B. Leeb, “Shipboard Fault Detection Through Nonintrusive Load Monitoring: A Case Study,” *IEEE Sens J*, 2018, doi: 10.1109/JSEN.2018.2869115.

[37] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, “Non-intrusive Load Monitoring approaches for disaggregated energy sensing: A survey,” *Sensors (Switzerland)*, vol. 12, no. 12, pp. 16838–16866, 2012, doi: 10.3390/s121216838.

[38] N. F. Esa, M. P. Abdullah, and M. Y. Hassan, “A review disaggregation method in Non-intrusive Appliance Load Monitoring,” *Renewable and Sustainable Energy Reviews*, vol. 66, pp. 163–173, Dec. 2016, doi: 10.1016/j.rser.2016.07.009.

[39] J. Huchtkoetter, M. A. Tepe, and A. Reinhardt, “The Impact of Ambient Sensing on the Recognition of Electrical Appliances,” *Energies* , vol. 14, no. 1. 2021. doi: 10.3390/en14010188.

[40] G. W. Hart, “Nonintrusive Appliance Load Monitoring,” *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992, doi: 10.1109/5.192069.

[41] I. H. Laouali, H. Qassemi, M. Marzouq, A. Ruano, S. D. Bennani, and H. el Fadili, “A survey on computational intelligence techniques for non intrusive load monitoring,” 2020. doi: 10.1109/ICECOCS50124.2020.9314383.

[42] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, and S. Patel, “Disaggregated End-Use Energy Sensing for the Smart Grid,” *IEEE Pervasive Comput*, vol. 10, no. 1, pp. 28–39, 2011, doi: 10.1109/MPRV.2010.74.

[43] NILMforIHEMS-project, “Non-Intrusive Load Monitoring applied to Intelligent Home Energy Management Systems (NILMforIHEMS),” 2022. https://csi.ualg.pt/nilmforihem/en/inicio-en/ (accessed May 12, 2022).

[44] N. Batra, H. Dutta, and A. Singh, “Indic: Improved non-intrusive load monitoring using load division and calibration,” in *2013 12th International Conference on Machine Learning and Applications*, 2013, vol. 1, pp. 79–84.

[45] A. Ridi, C. Gisler, and J. Hennebert, “A Survey on Intrusive Load Monitoring for Appliance Recognition,” in *2014 22nd International Conference on Pattern Recognition*, 2014, pp. 3702–3707. doi: 10.1109/ICPR.2014.636.

[46] R. M. Bacurau, L. F. C. Duarte, and E. C. Ferreira, “Focus on Energy Efficiency Through Power Consumption Disaggregation,” *Energy Efficiency Improvements in Smart Grid Components*, p. 21, 2015.

[47] A. F. Moreno Jaramillo, D. M. Laverty, D. J. Morrow, J. Martinez del Rincon, and A. M. Foley, “Load modelling and non-intrusive load monitoring to integrate distributed energy resources in

low and medium voltage networks," *Renewable Energy*, vol. 179. Elsevier Ltd, pp. 445–466, Dec. 01, 2021. doi: 10.1016/j.renene.2021.07.056.

[48] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 76–84, 2011, doi: 10.1109/TCE.2011.5735484.

[49] D. Saha, A. Bhattacharjee, D. Chowdhury, E. Hossain, and M. M. Islam, "Comprehensive NILM Framework: Device Type Classification and Device Activity Status Monitoring Using Capsule Network," *IEEE Access*, vol. 8, no. Ilm, pp. 179995–180009, 2020, doi: 10.1109/ACCESS.2020.3027664.

[50] L. Massidda, M. Marrocu, and S. Manca, "Non-intrusive load disaggregation by convolutional neural network and multilabel classification," *Applied Sciences (Switzerland)*, vol. 10, no. 4, 2020, doi: 10.3390/app10041454.

[51] J. Kelly and W. Knottenbelt, "Neural NILM : Deep Neural Networks Applied to Energy Disaggregation," 2015, doi: 10.1145/2821650.2821672.

[52] G. W. Hart, *Prototype nonintrusive appliance load monitor: Progress report 2*. MIT Energy Laboratory, 1985.

[53] C. Laughman *et al.*, "Power signature analysis," *IEEE Power and Energy Magazine*, vol. 1, no. 2, pp. 56–63, 2003, doi: 10.1109/MPAE.2003.1192027.

[54] G. W. Hart and A. T. Bouloutas, "Correcting dependent errors in sequences generated by finite-state processes," *IEEE Trans Inf Theory*, vol. 39, no. 4, pp. 1249–1260, 1993.

[55] Hernández, A. Ruano, J. Ureña, M. G. Ruano, and J. J. Garcia, "Applications of NILM Techniques to Energy Management and Assisted Living," *IFAC-PapersOnLine*, vol. 52, no. 11, pp. 164–171, 2019, doi: 10.1016/j.ifacol.2019.09.135.

[56] J. Revuelta Herrero *et al.*, "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*, 2018, pp. 125–138.

[57] S. S. Hosseini, K. Agbossou, S. Kelouwani, and A. Cardenas, "Non-intrusive load monitoring through home energy management systems: A comprehensive review," *Renewable and*

Sustainable Energy Reviews, vol. 79, no. April 2016, pp. 1266–1274, 2017, doi: 10.1016/j.rser.2017.05.096.

[58] H. Liu, *Non-intrusive load monitoring: Theory, technologies and applications*. 2019. doi: 10.1007/978-981-15-1860-7.

[59] C. Shin, S. Rho, H. Lee, and W. Rhee, "Data Requirements for Applying Machine Learning to Energy Disaggregation", doi: 10.3390/en12091696.

[60] W. Kong, Z. Y. Dong, B. Wang, J. Zhao, and J. Huang, "A practical solution for non-intrusive type II load monitoring based on deep learning and post-processing," *IEEE Trans Smart Grid*, 2020, doi: 10.1109/TSG.2019.2918330.

[61] J. Huchtkoetter and A. Reinhardt, "On the Impact of Temporal Data Resolution on the Accuracy of Non-Intrusive Load Monitoring," 2020, doi: 10.1145/3408308.

[62] B. Huang, M. Knox, K. Bradbury, L. M. Collins, and R. G. Newell, "Non-intrusive load monitoring system performance over a range of low frequency sampling rates."

[63] H. S. Matthews, L. Soibelman, M. Berges, and E. Goldman, "Automatically disaggregating the total electrical load in residential buildings: a profile of the required solution," *Intelligent Computing in Engineering (ICE08) Proceedings*, pp. 381–389, 2008.

[64] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[65] W. Wichakool, A. Avestruz, R. W. Cox, and S. B. Leeb, "Modeling and Estimating Current Harmonics of Variable Electronic Loads," *IEEE Trans Power Electron*, vol. 24, no. 12, pp. 2803–2811, 2009, doi: 10.1109/TPEL.2009.2029231.

[66] K. D. Lee, S. B. Leeb, L. K. Norford, P. R. Armstrong, J. Holloway, and S. R. Shaw, "Estimation of variable-speed-drive power consumption from harmonic content," *IEEE Transactions on Energy Conversion*, vol. 20, no. 3, pp. 566–574, 2005.

[67] S. R. Shaw, S. B. Leeb, L. K. Norford, and R. W. Cox, "Nonintrusive load monitoring and diagnostics in power systems," *IEEE Trans Instrum Meas*, vol. 57, no. 7, pp. 1445–1454, 2008.

[68] P. Huber, A. Calatroni, A. Rumsch, and A. Paice, “Review on deep neural networks applied to low-frequency nilm,” *Energies*, vol. 14, no. 9. MDPI AG, May 01, 2021. doi: 10.3390/en14092390.

[69] J. Gao, E. C. Kara, S. Giri, and M. Bergés, “A feasibility study of automated plug-load identification from high-frequency measurements,” in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 220–224. doi: 10.1109/GlobalSIP.2015.7418189.

[70] S. Gupta, M. S. Reynolds, and S. N. Patel, “ElectriSense: Single-Point Sensing Using EMI for Electrical Event Detection and Classification in the Home,” in *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, 2010, pp. 139–148. doi: 10.1145/1864349.1864375.

[71] H. Chang, K. Lian, Y. Su, and W. Lee, “Power-Spectrum-Based Wavelet Transform for Nonintrusive Demand Monitoring and Load Identification,” *IEEE Trans Ind Appl*, vol. 50, no. 3, pp. 2081–2089, 2014, doi: 10.1109/TIA.2013.2283318.

[72] Z. Wang and G. Zheng, “Residential appliances identification and monitoring by a nonintrusive method,” *IEEE Trans Smart Grid*, vol. 3, no. 1, pp. 80–92, Mar. 2012, doi: 10.1109/TSG.2011.2163950.

[73] M. L. Marceau and R. Zmeureanu, “Nonintrusive load disaggregation computer program to estimate the energy consumption of major end uses in residential buildings,” *Energy Convers Manag*, vol. 41, no. 13, pp. 1389–1403, 2000.

[74] L. Farinaccio and R. Zmeureanu, “Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses,” *Energy Build*, vol. 30, no. 3, pp. 245–259, 1999.

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

[76] H.-H. Chang, P. W. Wiratha, and N. Chen, “A Non-intrusive Load Monitoring System Using an Embedded System for Applications to Unbalanced Residential Distribution Systems,”

Energy Procedia, vol. 61, pp. 146–150, 2014, doi: https://doi.org/10.1016/j.egypro.2014.11.926.

[77] R. Brito *et al.*, "Instantaneous active and reactive load signature applied in non- intrusive load monitoring systems," *IET Smart Grid*, no. May 2020, pp. 121–133, 2021, doi: 10.1049/stg2.12008.

[78] R. Bonfigli, E. Principi, M. Fagiani, M. Severini, S. Squartini, and F. Piazza, "Non-intrusive load monitoring by using active and reactive power in additive Factorial Hidden Markov Models," *Appl Energy*, 2017, doi: 10.1016/j.apenergy.2017.08.203.

[79] M. B. Figueiredo, A. de Almeida, and B. Ribeiro, "An experimental study on electrical signature identification of non-intrusive load monitoring (nilm) systems," in *International Conference on Adaptive and Natural Computing Algorithms*, 2011, pp. 31–40.

[80] H. Najmeddine *et al.*, "State of art on load monitoring methods," in *2008 IEEE 2nd International power and Energy Conference*, 2008, pp. 1256–1258.

[81] A. G. Ruzzelli, C. Nicolas, A. Schoofs, and G. M. P. O'Hare, "Real-time recognition and profiling of appliances through a single electricity sensor," in *2010 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2010, pp. 1–9.

[82] T. Kato, H. S. Cho, D. Lee, T. Toyomura, and T. Yamazaki, "Appliance recognition from electric current signals for information-energy integrated network in home environments," in *International Conference on Smart Homes and Health Telematics*, 2009, pp. 150–157.

[83] H.-H. Chang, H.-T. Yang, and C.-L. Lin, "Load identification in neural networks for a non-intrusive monitoring of industrial electrical loads," in *International Conference on Computer Supported Cooperative Work in Design*, 2007, pp. 664–674.

[84] W. K. Lee, G. S. K. Fung, H. Y. Lam, F. H. Y. Chan, and M. Lucente, "Exploration on load signatures," in *International conference on electrical Engineering (ICEE)*, 2004, vol. 152.

[85] X. Huang, B. Yin, R. Zhang, and Z. Wei, "Study of steady-state feature extraction algorithm based on EMD," *IOP Conf Ser Mater Sci Eng*, vol. 490, p. 062036, 2019, doi: 10.1088/1757-899x/490/6/062036.

[86] H. Liu, H. Wu, and C. Yu, “A hybrid model for appliance classification based on time series features,” *Energy Build*, vol. 196, pp. 112–123, 2019.

[87] L. Du, Y. Yang, D. He, R. G. Harley, and T. G. Habetler, “Feature Extraction for Load Identification Using Long-Term Operating Waveforms,” *IEEE Trans Smart Grid*, vol. 6, no. 2, pp. 819–826, 2015, doi: 10.1109/TSG.2014.2373314.

[88] J. Li, S. West, and G. Platt, “Power decomposition based on SVM regression,” in *2012 Proceedings of International Conference on Modelling, Identification and Control*, 2012, pp. 1195–1199.

[89] A. Cole and A. Albicki, “Nonintrusive identification of electrical loads in a three-phase environment based on harmonic content,” in *Proceedings of the 17th IEEE Instrumentation and Measurement Technology Conference [Cat. No. 00CH37066]*, 2000, vol. 1, pp. 24–29.

[90] T. Kato, H. S. Cho, D. Lee, T. Toyomura, and T. Yamazaki, “Appliance recognition from electric current signals for information-energy integrated network in home environments,” in *International Conference on Smart Homes and Health Telematics*, 2009, pp. 150–157.

[91] D. Srinivasan, W. S. Ng, and A. C. Liew, “Neural-network-based signature recognition for harmonic source identification,” *IEEE Transactions on Power Delivery*, vol. 21, no. 1, pp. 398–405, 2006, doi: 10.1109/TPWRD.2005.852370.

[92] D. Srinivasan, W. S. Ng, and A. C. Liew, “Neural-network-based signature recognition for harmonic source identification,” *IEEE transactions on power delivery*, vol. 21, no. 1, pp. 398–405, 2005.

[93] A. Faustine and L. Pereira, “Multi-Label Learning for Appliance Recognition in NILM Using Fryze-Current Decomposition and Convolutional Neural Network,” *Energies (Basel)*, vol. 13, no. 16, p. 4154, 2020.

[94] Y. Liu, X. Wang, and W. You, “Non-Intrusive Load Monitoring by Voltage–Current Trajectory Enabled Transfer Learning,” *IEEE Trans Smart Grid*, vol. 10, no. 5, pp. 5609–5619, 2019, doi: 10.1109/TSG.2018.2888581.

[95] L. de Baets, T. Dhaene, D. Deschrijver, C. Develder, and M. Berges, “VI-Based Appliance Classification Using Aggregated Power Consumption Data,” in *2018 IEEE International*

Conference on Smart Computing (SMARTCOMP), 2018, pp. 179–186. doi: 10.1109/SMARTCOMP.2018.00089.

[96] J. Gao, E. C. Kara, S. Giri, and M. Bergés, "A feasibility study of automated plug-load identification from high-frequency measurements," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 220–224. doi: 10.1109/GlobalSIP.2015.7418189.

[97] L. Du, D. He, R. G. Harley, and T. G. Habetler, "Electric Load Classification by Binary Voltage–Current Trajectory Mapping," *IEEE Trans Smart Grid*, vol. 7, no. 1, pp. 358–365, 2016, doi: 10.1109/TSG.2015.2442225.

[98] L. de Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance classification using VI trajectories and convolutional neural networks," *Energy Build*, vol. 158, pp. 32–36, Jan. 2018, doi: 10.1016/j.enbuild.2017.09.087.

[99] A. Faustine and L. Pereira, "Improved appliance classification in non-intrusive load monitoring usingweighted recurrence graph and convolutional neural networks," *Energies (Basel)*, vol. 13, no. 13, Jul. 2020, doi: 10.3390/en13133374.

[100] H. Y. Lam, G. S. K. Fung, and W. K. Lee, "H. Y. Lam et al.: A Novel Method to Construct Taxonomy of Electrical Appliances Based on Load Signatures A Novel Method to Construct Taxonomy of Electrical Appliances Based on Load Signatures," 2007.

[101] A. L. Wang, B. X. Chen, C. G. Wang, and D. Hua, "Non-intrusive load monitoring algorithm based on features of V–I trajectory," *Electric Power Systems Research*, vol. 157, pp. 134–144, 2018.

[102] D. F. Teshome, T. D. Huang, and K. Lian, "Distinctive Load Feature Extraction Based on Fryze's Time-Domain Power Theory," *IEEE Power and Energy Technology Systems Journal*, vol. 3, no. 2, pp. 60–70, 2016, doi: 10.1109/JPETS.2016.2559507.

[103] H.-H. Chang, H.-T. Yang, and C.-L. Lin, "Load identification in neural networks for a non-intrusive monitoring of industrial electrical loads," in *International Conference on Computer Supported Cooperative Work in Design*, 2007, pp. 664–674.

[104] Y. Su, K. Lian, and H. Chang, “Feature Selection of Non-intrusive Load Monitoring System Using STFT and Wavelet Transform,” in *2011 IEEE 8th International Conference on e-Business Engineering*, 2011, pp. 293–298. doi: 10.1109/ICEBE.2011.49.

[105] S. B. Leeb and J. L. Kirtley, “A multiscale transient event detector for nonintrusive load monitoring,” in *Proceedings of IECON ’93 - 19th Annual Conference of IEEE Industrial Electronics*, 1993, pp. 354–359 vol.1. doi: 10.1109/IECON.1993.339053.

[106] H. Wu and H. Liu, “Non-intrusive load transient identification based on multivariate LSTM neural network and time series data augmentation,” *Sustainable Energy, Grids and Networks*, vol. 27, p. 100490, 2021, doi: https://doi.org/10.1016/j.segan.2021.100490.

[107] F. Kupzog, T. Zia, and A. A. Zaidi, “Automatic electric load identification in self-configuring microgrids,” in *AFRICON 2009*, 2009, pp. 1–5. doi: 10.1109/AFRCON.2009.5308129.

[108] B. Liu, W. Luan, and Y. Yu, “Dynamic time warping based non-intrusive load transient identification,” *Appl Energy*, vol. 195, pp. 634–645, 2017, doi: https://doi.org/10.1016/j.apenergy.2017.03.010.

[109] A. Wójcik, P. Bilski, R. Łukaszewski, K. Dowalla, and R. Kowalik, “Identification of the state of electrical appliances with the use of a pulse signal generator,” *Energies (Basel)*, vol. 14, no. 3, Feb. 2021, doi: 10.3390/en14030673.

[110] H. Chang, K. Lian, Y. Su, and W. Lee, “Power-Spectrum-Based Wavelet Transform for Nonintrusive Demand Monitoring and Load Identification,” *IEEE Trans Ind Appl*, vol. 50, no. 3, pp. 2081–2089, 2014, doi: 10.1109/TIA.2013.2283318.

[111] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, “At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential Power Line (Nominated for the Best Paper Award),” in *UbiComp 2007: Ubiquitous Computing*, 2007, pp. 271–288.

[112] H. Chang, K. Chen, Y. Tsai, and W. Lee, “A New Measurement Method for Power Signatures of Nonintrusive Demand Monitoring and Load Identification,” *IEEE Trans Ind Appl*, vol. 48, no. 2, pp. 764–771, 2012, doi: 10.1109/TIA.2011.2180497.

[113] A. I. Cole and A. Albicki, “Data extraction for effective non-intrusive identification of residential power loads,” in *IMTC/98 Conference Proceedings. IEEE Instrumentation and Measurement Technology Conference. Where Instrumentation is Going (Cat. No. 98CH36222)*, 1998, vol. 2, pp. 812–815.

[114] Z. Wang and G. Zheng, “Residential appliances identification and monitoring by a nonintrusive method,” *IEEE Trans Smart Grid*, vol. 3, no. 1, pp. 80–92, 2011.

[115] H.-H. Chang, “Non-intrusive demand monitoring and load identification for energy management systems based on transient feature analyses,” *Energies (Basel)*, vol. 5, no. 11, pp. 4569–4589, 2012.

[116] S. Zhai, Z. Wang, X. Yan, and G. He, “Appliance Flexibility Analysis Considering User Behavior in Home Energy Management System Using Smart Plugs,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp. 1391–1401, 2019, doi: 10.1109/TIE.2018.2815949.

[117] J. Liang, S. Ng, G. Kendall, and J. Cheng, “Load signature study¡ V part I: Basic concept, structure and methodology,” in *IEEE PES General Meeting*, 2010, p. 1.

[118] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, “Load signature study—Part I: Basic concept, structure, and methodology,” *IEEE transactions on power Delivery*, vol. 25, no. 2, pp. 551–560, 2009.

[119] M. Zeifman, “Disaggregation of home energy display data using probabilistic approach,” *IEEE Transactions on Consumer Electronics*, vol. 58, no. 1, pp. 23–31, 2012, doi: 10.1109/TCE.2012.6170051.

[120] A. S. N. Uttama Nambi, A. Reyes Lua, and V. R. Prasad, “LocED: Location-Aware Energy Disaggregation Framework,” in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, 2015, pp. 45–54. doi: 10.1145/2821650.2821659.

[121] L. Pereira and N. Nunes, “An empirical exploration of performance metrics for event detection algorithms in Non-Intrusive Load Monitoring,” *Sustain Cities Soc*, vol. 62, Nov. 2020, doi: 10.1016/j.scs.2020.102399.

[122] K. D. Anderson, M. E. Bergés, A. Ocneanu, D. Benitez, and J. M. F. Moura, “Event detection for Non Intrusive load monitoring,” in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 2012, pp. 3312–3317. doi: 10.1109/IECON.2012.6389367.

[123] M. N. Meziane, P. Ravier, G. Lamarque, J. le Bunetel, and Y. Raingeaud, “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. doi: 10.1109/ICASSP.2017.7952597.

[124] J. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, “Event-Based Energy Disaggregation Algorithm for Activity Monitoring From a Single-Point Sensor,” *IEEE Trans Instrum Meas*, vol. 66, no. 10, pp. 2615–2626, 2017, doi: 10.1109/TIM.2017.2700987.

[125] M. Weiss, A. Helfenstein, F. Mattern, and T. Staake, “Leveraging smart meter data to recognize home appliances,” in *2012 IEEE International Conference on Pervasive Computing and Communications*, 2012, pp. 190–197. doi: 10.1109/PerCom.2012.6199866.

[126] D. Luo, L. K. Norford, S. R. Shaw, and S. B. Leeb, “Monitoring HVAC equipment electrical loads from a centralized location-methods and field test results,” *ASHRAE Trans*, vol. 108, no. 1, pp. 841–857, 2002.

[127] K. Anderson, A. F. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Bergés, “BLUED : A Fully Labeled Public Dataset for Event-Based Non-Intrusive Load Monitoring Research,” 2012.

[128] M. Berges, E. Goldman, H. S. Matthews, L. Soibelman, and K. Anderson, “User-centered nonintrusive electricity load monitoring for residential buildings,” *Journal of computing in civil engineering*, vol. 25, no. 6, pp. 471–480, 2011.

[129] C. C. Yang, C. S. Soh, and V. V. Yap, “Comparative Study of Event Detection Methods for Non-intrusive Appliance Load Monitoring,” *Energy Procedia*, vol. 61, pp. 1840–1843, 2014, doi: https://doi.org/10.1016/j.egypro.2014.12.225.

[130] S. R. Shaw, S. B. Leeb, L. K. Norford, and R. W. Cox, “Nonintrusive load monitoring and diagnostics in power systems,” *IEEE Trans Instrum Meas*, vol. 57, no. 7, pp. 1445–1454, 2008.

[131] L. K. Norford and S. B. _ Leeb, “Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms,” 1996.

[132] J. M. Alcalá, J. Ureña, and Á. Hernández, “Event-based detector for non-intrusive load monitoring based on the Hilbert Transform,” in *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 2014, pp. 1–4.

[133] S. R. Shaw and C. R. Laughman, “A Kalman-Filter Spectral Envelope Preprocessor,” *IEEE Trans Instrum Meas*, vol. 56, no. 5, pp. 2010–2017, 2007, doi: 10.1109/TIM.2007.904475.

[134] B. Wild, K. S. Barsim, and B. Yang, “A new unsupervised event detector for non-intrusive load monitoring,” in *2015 IEEE global conference on signal and information processing (GlobalSIP)*, 2015, pp. 73–77.

[135] B. Zhao, L. Stankovic, and V. Stankovic, “On a Training-Less Solution for Non-Intrusive Appliance Load Monitoring Using Graph Signal Processing,” *IEEE Access*, vol. 4, pp. 1784–1799, 2016, doi: 10.1109/ACCESS.2016.2557460.

[136] L. de Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, “On the Bayesian optimization and robustness of event detection methods in NILM,” *Energy Build*, vol. 145, pp. 57–66, Jun. 2017, doi: 10.1016/j.enbuild.2017.03.061.

[137] Z. Zheng, H. Chen, and X. Luo, “A supervised event-based non-intrusive load monitoring for non-linear appliances,” *Sustainability (Switzerland)*, vol. 10, no. 4, Mar. 2018, doi: 10.3390/su10041001.

[138] M. Afzalan, F. Jazizadeh, and J. Wang, “Self-configuring event detection in electricity monitoring for human-building interaction,” *Energy Build*, vol. 187, pp. 95–109, Mar. 2019, doi: 10.1016/j.enbuild.2019.01.036.

[139] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, “Effective non-intrusive load monitoring of buildings based on a novel multi-descriptor fusion with dimensionality reduction,” *Appl Energy*, vol. 279, p. 115872, 2020, doi: https://doi.org/10.1016/j.apenergy.2020.115872.

[140] Q. v Le, “Building high-level features using large scale unsupervised learning,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8595–8598. doi: 10.1109/ICASSP.2013.6639343.

[141] K. Basu, V. Debusschere, S. Bacha, A. Hably, D. van Delft, and G. Jan Dirven, “A generic data driven approach for low sampling load disaggregation,” *Sustainable Energy, Grids and Networks*, vol. 9, pp. 118–127, 2017, doi: https://doi.org/10.1016/j.segan.2016.12.006.

[142] V. M. Salerno and G. Rabbeni, “An Extreme Learning Machine Approach to Effective Energy Disaggregation,” *Electronics (Basel)*, vol. 7, no. 10, 2018, doi: 10.3390/electronics7100235.

[143] T.-T.-H. Le and H. Kim, “Non-Intrusive Load Monitoring Based on Novel Transient Signal in Household Appliances with Low Sampling Rate,” 2018, doi: 10.3390/en11123409.

[144] T.-T.-H. Le, H. Kang, and H. Kim, “Household Appliance Classification Using Lower Odd-Numbered Harmonics and the Bagging Decision Tree,” *IEEE Access*, vol. 8, pp. 55937–55952, 2020, doi: 10.1109/ACCESS.2020.2981969.

[145] D. Egarter, A. Sobe, and W. Elmenreich, “Evolving non-intrusive load monitoring,” in *European Conference on the Applications of Evolutionary Computation*, 2013, pp. 182–191.

[146] W. Kong, Z. Y. Dong, J. Ma, D. J. Hill, J. Zhao, and F. Luo, “An extensible approach for non-intrusive load disaggregation with smart meter data,” *IEEE Trans Smart Grid*, vol. 9, no. 4, pp. 3362–3372, 2016.

[147] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, “Improving nonintrusive load monitoring efficiency via a hybrid programing method,” *IEEE Trans Industr Inform*, vol. 12, no. 6, pp. 2148–2157, 2016.

[148] M. B. Figueiredo, A. De Almeida, and B. Ribeiro, “An experimental study on electrical signature identification of non-intrusive load monitoring (NILM) systems,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6594 LNCS, no. PART 2, pp. 31–40, 2011, doi: 10.1007/978-3-642-20267-4_4.

[149] S. Giri, M. Bergés, and A. Rowe, “Towards automated appliance recognition using an EMF sensor in NILM platforms,” *Advanced Engineering Informatics*, vol. 27, pp. 477–485, 2013, doi: 10.1016/j.aei.2013.03.004.

[150] J. Gao, E. C. Kara, S. Giri, and M. Bergés, “A feasibility study of automated plug-load identification from high-frequency measurements,” in *2015 IEEE global conference on signal and information processing (GlobalSIP)*, 2015, pp. 220–224.

[151] A. Moradzadeh, S. Zeinal-Kheiri, B. Mohammadi-Ivatloo, M. Abapour, and A. Anvari-Moghaddam, “Support vector machine-assisted improvement residential load disaggregation,” *2020 28th Iranian Conference on Electrical Engineering, ICEE 2020*, pp. 1–6, 2020, doi: 10.1109/ICEE50131.2020.9260869.

[152] Y. H. Lin and M. S. Tsai, “Applications of hierarchical support vector machines for identifying load operation in nonintrusive load monitoring systems,” *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, pp. 688–693, 2011, doi: 10.1109/WCICA.2011.5970603.

[153] X. Zhou, S. Li, C. Liu, H. Zhu, N. Dong, and T. Xiao, “Non-Intrusive Load Monitoring Using a CNN-LSTM-RF Model Considering Label Correlation and Class-Imbalance,” *IEEE Access*, 2021.

[154] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14, 2015.

[155] H. Chen, Y.-H. Wang, C.-H. Fan, and H. Chen, “A convolutional autoencoder-based approach with batch normalization for energy disaggregation,” vol. 77, pp. 2961–2978, 2021, doi: 10.1007/s11227-020-03375-y.

[156] A. Moradzadeh, B. Mohammadi-Ivatloo, M. Abapour, A. Anvari-Moghaddam, S. Gholami Farkoush, and S. B. Rhee, “A practical solution based on convolutional neural network for non-intrusive load monitoring,” *J Ambient Intell Humaniz Comput*, 2021, doi: 10.1007/s12652-020-02720-6.

[157] S. Barker, M. Musthag, D. Irwin, and P. Shenoy, “Non-intrusive load identification for smart outlets,” in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014, pp. 548–553.

[158] P. Meehan and S. Daniels, “An Efficient, Scalable Time-Frequency Method for Tracking Energy Usage of Domestic Appliances Using a Two-Step Classification Algorithm,” vol. 7, pp. 7041–7066, 2014, doi: 10.3390/en7117041.

[159] J. Jiang, Q. Kong, M. D. Plumbley, N. Gilbert, M. Hoogendoorn, and D. Roijers, “Deep learning based energy disaggregation and on/off detection of household appliances,” *arXiv*. 2019.

[160] J. Kim, T.-T.-H. Le, and H. Kim, “Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature,” *Comput Intell Neurosci*, vol. 2017, pp. 1–22, 2017, doi: 10.1155/2017/4216281.

[161] I. Laouali, A. Ruano, M. da G. Ruano, S. D. Bennani, and H. el Fadili, “Non-Intrusive Load Monitoring of Household Devices Using a Hybrid Deep Learning Model through Convex Hull-Based Data Selection,” *Energies (Basel)*, vol. 15, no. 3, 2022, doi: 10.3390/en15031215.

[162] S. Singh and A. Majumdar, “Deep sparse coding for non-intrusive load monitoring,” *IEEE Trans Smart Grid*, vol. 9, no. 5, pp. 4669–4678, 2018, doi: 10.1109/TSG.2017.2666220.

[163] L. Weng, X. Zhang, J. Qian, M. Xia, Y. Xu, and K. Wang, “Non-intrusive load disaggregation based on a multi-scale attention residual network,” *Applied Sciences (Switzerland)*, vol. 10, no. 24, pp. 1–17, 2020, doi: 10.3390/app10249132.

[164] H. He, Z. Liu, R. Jiao, and G. Yan, “A novel nonintrusive load monitoring approach based on linear-chain conditional random fields,” *Energies (Basel)*, vol. 12, no. 9, p. 1797, 2019.

[165] D. Srinivasan, W. S. Ng, and A. C. Liew, “Neural-network-based signature recognition for harmonic source identification,” *IEEE transactions on power delivery*, vol. 21, no. 1, pp. 398–405, 2005.

[166] O. Kramer *et al.*, “On ensemble classifiers for nonintrusive appliance load monitoring,” in *International Conference on Hybrid Artificial Intelligence Systems*, 2012, pp. 322–331.

[167] K. S. Barsim, L. Mauch, and B. Yang, “Neural network ensembles to real-time identification of plug-level appliance measurements,” *arXiv preprint arXiv:1802.06963*, 2018.

[168] T. Hassan, F. Javed, and N. Arshad, "An Empirical Investigation of V-I Trajectory Based Load Signatures for Non-Intrusive Load Monitoring," *IEEE Trans Smart Grid*, vol. 5, no. 2, pp. 870–878, 2014, doi: 10.1109/TSG.2013.2271282.

[169] J. Z. Kolter and M. J. Johnson, "REDD : A Public Data Set for Energy Disaggregation Research," *SustKDD workshop*, 2011.

[170] A. Marchiori, D. Hakkarinen, Q. Han, and L. Earle, "Circuit-level load monitoring for household energy management," *IEEE Pervasive Comput*, vol. 10, no. 1, pp. 40–48, 2010.

[171] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "An unsupervised training method for non-intrusive appliance load monitoring," *Artif Intell*, vol. 217, pp. 1–19, 2014.

[172] K. S. Barsim and B. Yang, "Toward a semi-supervised non-intrusive load monitoring system for event-based energy disaggregation," in *2015 IEEE global conference on signal and information processing (GlobalSIP)*, 2015, pp. 58–62.

[173] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Robust event-based non-intrusive appliance recognition using multi-scale wavelet packet tree and ensemble bagging tree," *Appl Energy*, vol. 267, p. 114877, 2020, doi: https://doi.org/10.1016/j.apenergy.2020.114877.

[174] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "An unsupervised training method for non-intrusive appliance load monitoring," *Artif Intell*, vol. 217, pp. 1–19, 2014.

[175] S. Makonin, F. Popowich, I. v. Bajic, B. Gill, and L. Bartram, "Exploiting HMM Sparsity to Perform Online Real-Time Nonintrusive Load Monitoring," *IEEE Trans Smart Grid*, 2016, doi: 10.1109/TSG.2015.2494592.

[176] K. Basu, V. Debusschere, A. Douzal-Chouakria, and S. Bacha, "Time series distance-based methods for non-intrusive load monitoring in residential buildings," *Energy Build*, vol. 96, pp. 109–117, 2015, doi: https://doi.org/10.1016/j.enbuild.2015.03.021.

[177] A. Zoha, A. Gluhak, M. Nati, and M. A. Imran, "Low-power appliance monitoring using Factorial Hidden Markov Models," 2013. doi: 10.1109/ISSNIP.2013.6529845.

[178] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, "Unsupervised Disaggregation of Low Frequency Power Measurements," in *Proceedings of the 2011 SIAM International Conference on Data Mining*, Apr. 2011, pp. 747–758. doi: 10.1137/1.9781611972818.64.

[179] M. Aiad and P. H. Lee, “Unsupervised approach for load disaggregation with devices interactions,” *Energy Build*, vol. 116, pp. 96–103, 2016, doi: https://doi.org/10.1016/j.enbuild.2015.12.043.

[180] I. Valera, F. J. R. Ruiz, and F. Perez-Cruz, “Infinite Factorial Unbounded-State Hidden Markov Model,” *IEEE Trans Pattern Anal Mach Intell*, vol. 38, no. 9, pp. 1816–1828, 2016, doi: 10.1109/TPAMI.2015.2498931.

[181] Y. Li, Z. Peng, J. Huang, Z. Zhang, and J. H. Son, “Energy disaggregation via hierarchical factorial hmm,” in *Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring, Austin, TX, USA*, 2014, vol. 3.

[182] F. Paradiso, F. Paganelli, D. Giuli, and S. Capobianco, “Context-based energy disaggregation in smart homes,” *Future Internet*, vol. 8, no. 1, pp. 1–22, 2016, doi: 10.3390/fi8010004.

[183] J. Z. Kolter and T. Jaakkola, “Approximate inference in additive factorial HMMs with application to energy disaggregation,” 2012.

[184] H. Shao, M. Marwah, and N. Ramakrishnan, “A Temporal Motif Mining Approach to Unsupervised Energy Disaggregation: Applications to Residential and Commercial Buildings,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 27, no. 1, pp. 1327–1333, Jun. 2013, [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/8485

[185] H. Gonzalves, A. Ocneanu, and M. Bergés, “Unsupervised disaggregation of appliances using aggregated consumption data,” 2011.

[186] J. D. S. Guedes, D. D. Ferreira, and B. H. G. Barbosa, “A non-intrusive approach to classify electrical appliances based on higher-order statistics and genetic algorithm: a smart grid perspective,” *Electric Power Systems Research*, vol. 140, pp. 65–69, 2016, doi: https://doi.org/10.1016/j.epsr.2016.06.042.

[187] E. Elhamifar and S. Sastry, “Energy disaggregation via learning powerlets and sparse coding,” 2015.

[188] Y. H. Lin, M. S. Tsai, and C. S. Chen, “Applications of fuzzy classification with fuzzy c-means clustering and optimization strategies for load identification in NILM systems,” *IEEE*

International Conference on Fuzzy Systems, pp. 859–866, 2011, doi: 10.1109/FUZZY.2011.6007393.

[189] T. T. H. Le and H. Kim, "Non-intrusive load monitoring based on novel transient signal in household appliances with low sampling rate," *Energies (Basel)*, vol. 11, no. 12, 2018, doi: 10.3390/en11123409.

[190] B. Zhang, S. Zhao, Q. Shi, and R. Zhang, "Low-rate non-intrusive appliance load monitoring based on graph signal processing," *2019 International Conference on Security, Pattern Analysis, and Cybernetics, SPAC 2019*, pp. 11–16, 2019, doi: 10.1109/SPAC49953.2019.237866.

[191] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-Intrusive Load Disaggregation Using Graph Signal Processing," *IEEE Trans Smart Grid*, vol. 9, no. 3, pp. 1739–1747, 2018, doi: 10.1109/TSG.2016.2598872.

[192] S. A. Azad, A. B. M. S. Ali, and P. Wolfs, "Identification of typical load profiles using K-means clustering algorithm," *Asia-Pacific World Congress on Computer Science and Engineering, APWC on CSE 2014*, 2014, doi: 10.1109/APWCCSE.2014.7053855.

[193] H. Liu, "Deep Learning Based Appliance Identification," in *Non-intrusive Load Monitoring*, Springer, 2020, pp. 191–214.

[194] L. Mauch and B. Yang, "A new approach for supervised power disaggregation by using a deep recurrent LSTM network," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 63–67.

[195] L. R. Morais and A. R. G. Castro, "Competitive autoassociative neural networks for electrical appliance identification for non-intrusive load monitoring," *IEEE Access*, vol. 7, pp. 111746–111755, 2019.

[196] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 2604–2611, 2018.

[197] X. Zhou, S. Li, C. Liu, H. Zhu, N. Dong, and T. Xiao, “Non-Intrusive Load Monitoring Using a CNN-LSTM-RF Model Considering Label Correlation and Class-Imbalance,” *IEEE Access*, vol. XX, pp. 1–1, 2021, doi: 10.1109/access.2021.3087696.

[198] H. Rafiq, X. Shi, H. Zhang, H. Li, and M. K. Ochani, “A deep recurrent neural network for non-intrusive load monitoring based on multi-feature input space and post-processing,” *Energies (Basel)*, vol. 13, no. 9, 2020, doi: 10.3390/en13092195.

[199] D. Saha, A. Bhattacharjee, D. Chowdhury, E. Hossain, and M. M. Islam, “Comprehensive NILM framework: device type classification and device activity status monitoring using capsule network,” *IEEE Access*, vol. 8, pp. 179995–180009, 2020.

[200] L. Massidda, M. Marrocu, and S. Manca, “Non-Intrusive Load Disaggregation by Convolutional Neural Network and Multilabel Classification”, doi: 10.3390/app10041454.

[201] E. Gomes and L. Pereira, “PB-NILM: Pinball guided deep non-intrusive load monitoring,” *IEEE Access*, vol. 8, pp. 48386–48398, 2020.

[202] T.-T.-H. Le and H. Kim, “Non-Intrusive Load Monitoring Based on Novel Transient Signal in Household Appliances with Low Sampling Rate,” 2018, doi: 10.3390/en11123409.

[203] H. Rafiq, X. Shi, H. Zhang, H. Li, M. K. Ochani, and A. A. Shah, “Generalizability Improvement of Deep System Using Data Augmentation,” *IEEE Trans Smart Grid*, vol. 12, no. 4, pp. 3265–3277, 2021.

[204] L. Guo, S. Wang, H. Chen, and Q. Shi, “A Load Identification Method Based on Active Deep Learning and Discrete Wavelet Transform,” *IEEE Access*, vol. 8, pp. 113932–113942, 2020.

[205] V. Piccialli and A. M. Sudoso, “Improving Non-Intrusive Load Disaggregation through an Attention-Based Deep Neural Network,” 2021, doi: 10.3390/en14040847.

[206] L. De Baets, C. Develder, T. Dhaene, and D. Deschrijver, “Detection of unidentified appliances in non-intrusive load monitoring using siamese neural networks,” *International Journal of Electrical Power and Energy Systems*, vol. 104, no. July 2018, pp. 645–653, 2019, doi: 10.1016/j.ijepes.2018.07.026.

[207] Z. Zhou, Y. Xiang, H. Xu, Z. Yi, D. Shi, and Z. Wang, “A Novel Transfer Learning-Based Intelligent Nonintrusive Load-Monitoring With Limited Measurements,” *IEEE Trans Instrum Meas*, vol. 70, pp. 1–8, 2021, doi: 10.1109/TIM.2020.3011335.

[208] Y. Xiao, Y. Hu, H. He, D. Zhou, Y. Zhao, and W. Hu, “Non-intrusive load identification method based on improved KM algorithm,” *IEEE Access*, vol. 7, pp. 151368–151377, 2019.

[209] J. Z. Kolter and M. J. Johnson, “REDD: A public data set for energy disaggregation research,” in *Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA*, 2011, vol. 25, no. Citeseer, pp. 59–62.

[210] L. Pereira and N. Nunes, “Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—A review,” *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 8, no. 6, pp. 1–17, 2018, doi: 10.1002/widm.1265.

[211] E. T. Mayhorn, G. P. Sullivan, J. M. Petersen, R. S. Butner, and E. M. Johnson, “Load disaggregation technologies: real world and laboratory performance,” *Pacific Northwest National Laboratory (PNNL), Richland, WA (US), Tech. Rep. PNNL-SA-116560*, 2016.

[212] A. Faustine, N. H. Mvungi, S. Kaijage, and K. Michael, “A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem,” *arXiv preprint arXiv:1703.00785*, 2017.

[213] M. Zeifman, “Disaggregation of home energy display data using probabilistic approach,” *IEEE Transactions on Consumer Electronics*, vol. 58, no. 1, pp. 23–31, 2012.

[214] D. Piga, A. Cominola, M. Giuliani, A. Castelletti, and A. E. Rizzoli, “Sparse optimization for automated energy end use disaggregation,” *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 1044–1051, 2015.

[215] C. Klemenjak, A. Reinhardt, L. Pereira, S. Makonin, M. Bergés, and W. Elmenreich, “Electricity Consumption Data Sets: Pitfalls and Opportunities,” in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2019, pp. 159–162. doi: 10.1145/3360322.3360867.

[216] L. Pereira, D. Costa, and M. Ribeiro, “A residential labeled dataset for smart meter data analytics,” *Sci Data*, vol. 9, no. 1, p. 134, 2022, doi: 10.1038/s41597-022-01252-2.

[217] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Building power consumption datasets: Survey, taxonomy and future directions," *Energy Build*, vol. 227, p. 110404, 2020, doi: https://doi.org/10.1016/j.enbuild.2020.110404.

[218] A. Reinhardt *et al.*, "On the accuracy of appliance identification based on distributed load metering data," 2012.

[219] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht, "Smart*: An Open Data Set and Tools for Enabling Research in Sustainable Homes," *SustKDD*, 2012, doi: adf.

[220] G. Johnson and I. Beausoleil-Morrison, "Electrical-end-use data from 23 houses sampled each minute for simulating micro-generation systems," *Appl Therm Eng*, vol. 114, pp. 1449–1456, 2017.

[221] N. Saldanha and I. Beausoleil-Morrison, "Measured end-use electric load profiles for 12 Canadian houses at high temporal resolution," *Energy Build*, vol. 49, pp. 519–530, 2012.

[222] L. Pereira, F. Quintal, R. Gonçalves, and N. J. Nunes, "SustData: A Public Dataset for ICT4S Electric Energy Research.," 2014.

[223] S. Makonin, B. Ellert, I. v Bajić, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014," *Sci Data*, vol. 3, no. 1, pp. 1–12, 2016.

[224] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I. v. Bajić, "AMPds: A public dataset for load disaggregation and eco-feedback research," 2013. doi: 10.1109/EPEC.2013.6802949.

[225] Nipun. Batra, M. Gulati, A. Singh, and M. Srivastava, "It's Different: Insights into home energy consumption in India," *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, no. August, pp. 1–8, 2013, doi: 10.1145/2528282.2528293.

[226] C. Gisler, A. Ridi, D. Zufferey, O. Abou Khaled, and J. Hennebert, "Appliance consumption signature database and recognition test protocols," in *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, 2013, pp. 336–341.

[227] A. Monacchi, D. Egarter, W. Elmenreich, S. D'Alessandro, and A. M. Tonello, "GREEND: An energy consumption dataset of households in Italy and Austria," 2015. doi: 10.1109/SmartGridComm.2014.7007698.

[228] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini, “The ECO data set and the performance of non-intrusive load monitoring algorithms,” 2014. doi: 10.1145/2674061.2674064.

[229] J. Gao, S. Giri, E. C. Kara, and M. Bergés, “PLAID: A public dataset of high-resolution electrical appliance measurements for load identification research,” 2014. doi: 10.1145/2674061.2675032.

[230] O. Parson *et al.*, “Dataport and NILMTK: A building data set designed for non-intrusive load monitoring,” in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 210–214. doi: 10.1109/GlobalSIP.2015.7418187.

[231] D. Murray *et al.*, “A data management platform for personalised real-time energy feedback,” 2015.

[232] J. Kelly and W. Knottenbelt, “The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes,” *Sci Data*, 2015, doi: 10.1038/sdata.2015.7.

[233] S. Grid and M. Data, “RAE : The Rainforest Automation Energy Dataset for Smart Grid Meter Data Analysis,” pp. 1–9, 2018, doi: 10.3390/data3010008.

[234] V. L. Chen, M. A. Delmas, S. L. Locke, and A. Singh, “Information strategies for energy conservation: A field experiment in India,” *Energy Econ*, vol. 68, pp. 215–227, 2017.

[235] T. Kriechbaumer and H.-A. Jacobsen, “BLOND, a building-level office environment dataset of typical electrical appliances,” *Sci Data*, vol. 5, no. 1, pp. 1–14, 2018.

[236] S. Makonin, “HUE: The hourly usage of energy dataset for buildings in British Columbia,” *Data Brief*, vol. 23, 2019.

[237] N. Terry and J. Palmer, “Household electricity survey,” *UK Data Archive Study*, pp. 1–31, 2012, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85090596985&partnerID=40&md5=0d6c0d6e28e510e740367ce2088351b0

[238] K. Bache and M. Lichman, “Individual Household electric power consumption dataset,” *Irvine, CA: University of California, School of Information and Computer Science*, vol. 206, 2013.

[239] C. Holcomb, “Pecan street inc.: A test-bed for NILM,” 2012.

[240] M. Maasoumy, B. Sanandaji, K. Poolla, and A. S. Vincentelli, “Berds-berkeley energy disaggregation data set,” in *Proceedings of the Workshop on Big Learning at the Conference on Neural Information Processing Systems (NIPS)*, 2013, pp. 1–6.

[241] S. Makonin, B. Ellert, I. v Bajić, and F. Popowich, “Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014,” *Sci Data*, vol. 3, no. 1, pp. 1–12, 2016.

[242] M. Gulati, S. S. Ram, and A. Singh, “An in Depth Study into Using EMI Signatures for Appliance Identification,” in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, 2014, pp. 70–79. doi: 10.1145/2674061.2674070.

[243] T. Picon *et al.*, “COOLL: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification,” *arXiv preprint arXiv:1611.05803*, 2016.

[244] M. Kahl, A. Haq, and T. Kriechbaumer, “Whited-a worldwide household and industry transient energy data set,” *3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[245] M. Kahl *et al.*, “Measurement system and dataset for in-depth analysis of appliance energy consumption in industrial environment,” *tm-Technisches Messen*, vol. 86, no. 1, pp. 1–13, 2019.

[246] C. Shin, E. Lee, J. Han, J. Yim, W. Rhee, and H. Lee, “The ENERTALK dataset, 15 Hz electricity consumption data from 22 houses in Korea,” *Sci Data*, vol. 6, no. 1, pp. 1–13, 2019.

[247] D. Jorde, T. Kriechbaumer, T. Berger, S. Zitzlsperger, and H.-A. Jacobsen, “CREAM, a component level coffeemaker electrical activity measurement dataset,” *Sci Data*, vol. 7, no. 1, pp. 1–13, 2020.

[248] J. Kelly and W. Knottenbelt, “Metadata for Energy Disaggregation,” in *2014 IEEE 38th International Computer Software and Applications Conference Workshops*, 2014, pp. 578–583. doi: 10.1109/COMPSACW.2014.97.

[249] N. Batra *et al.*, “NILMTK: An open source toolkit for non-intrusive load monitoring,” 2014. doi: 10.1145/2602044.2602051.

[250] R. Bonfigli and S. Squartini, “Non-intrusive Load Monitoring,” in *Machine Learning Approaches to Non-Intrusive Load Monitoring*, R. Bonfigli and S. Squartini, Eds. Cham: Springer International Publishing, 2020, pp. 3–14. doi: 10.1007/978-3-030-30782-0_2.

[251] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Proc. Adv. Neural Inf. Process. Syst.*, pp. 1097–1105, 2012.

[252] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull Math Biophys*, vol. 5, no. 4, pp. 115–133, 1943.

[253] Bruce Blaus, “Medical gallery of Blausen Medical 2014,” *WikiJournal of Medicine*, vol. 1, no. 2, 2014, doi: 10.15347/wjm/2014.010.

[254] S. Haykin, “Adaptive filter theory 3rd edition Prentice-Hall.” Inc Upper Saddle River New Jersey USA, 1996.

[255] H. R. Khosravani, A. E. Ruano, and P. M. Ferreira, “A convex hull-based data selection method for data driven models,” *Applied Soft Computing Journal*, vol. 47, pp. 515–533, 2016, doi: 10.1016/j.asoc.2016.06.014.

[256] C. Shin, S. Joo, J. Yim, H. Lee, T. Moon, and W. Rhee, “Subtask gated networks for non-intrusive load monitoring,” *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, pp. 1150–1157, 2019, doi: 10.1609/aaai.v33i01.33011150.

[257] A. Ruano, H. R. Khosravani, and P. M. Ferreira, “A Randomized Approximation Convex Hull Algorithm for High Dimensions,” *IFAC-PapersOnLine*, vol. 48, no. 10, pp. 123–128, 2015, doi: https://doi.org/10.1016/j.ifacol.2015.08.119.

[258] D. H. Hubel and T. N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex,” *J Physiol*, vol. 195, no. 1, pp. 215–243, 1968.

[259] K. Fukushima and S. Miyake, “Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition,” in *Competition and cooperation in neural nets*, Springer, 1982, pp. 267–285.

[260] Y. LeCun *et al.*, “Handwritten digit recognition with a back-propagation network,” *Adv Neural Inf Process Syst*, vol. 2, 1989.

[261] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[262] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[263] J. Gu *et al.*, “Recent advances in convolutional neural networks,” *Pattern Recognit*, vol. 77, pp. 354–377, 2018, doi: 10.1016/j.patcog.2017.10.013.

[264] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” 2010.

[265] O. Russakovsky *et al.*, “Imagenet large scale visual recognition challenge,” *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, 2015.

[266] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. icml*, 2013, vol. 30, no. 1, p. 3.

[267] M. D. Zeiler *et al.*, “On rectified linear units for speech processing,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 3517–3521.

[268] S. Hochreiter, “Long Short-Term Memory,” *Neural Comput*, vol. 1780, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[269] J.-G. Kim and B. Lee, “Appliance Classification by Power Signal Analysis Based on Multi-Feature Combination Multi-Layer LSTM,” *Energies (Basel)*, vol. 12, no. 14, 2019, doi: 10.3390/en12142804.

[270] “Sharp NU-AK PV Panels.” https://www.sharp.co.uk/gb (accessed Dec. 14, 2021).

[271] “BYD Battery Box HV.” https://www.bydbatterybox.com/ (accessed Dec. 14, 2021).

[272] “Kostal Plenticore Plus Inverter.” https://www.kostal-solar-electric.com/en-gb/products/hybrid-inverters/ plenticore-plus (accessed Dec. 14, 2021).

[273] G. Mestre *et al.*, "An intelligent weather station," *Sensors (Switzerland)*, vol. 15, no. 12, pp. 31005–31022, 2015, doi: 10.3390/s151229841.

[274] "Tp-link: Wi-Fi Smart Plug." https://www.tp-link.com/pt/home-networking/smart-plug/hs100/ (accessed Jun. 24, 2022).

[275] C. Gavazzi, "Carlo Gavazzi Automation Company: EM340 utilises touchscreen technology." https://www.carlogavazzi.co.uk/blog/carlo-gavazzi-energy-solutions/em340-utilises-touchscreen-technology (accessed Aug. 25, 2021).

[276] "Circutor: Consumption analyzers." http://circutor.com/en/products/measurement-and-control/fixed-power-analyzers/consumption-analyzers/wibeee-series-detail (accessed Aug. 20, 2021).

[277] "Kostal: Kostal Smart Energy Meter." https://shop.kostal-solar-electric.com/en/kostal-smart-energy-meter.html (accessed Jun. 25, 2022).

[278] A. E. Ruano *et al.*, "The IMBPC HVAC system: A complete MBPC solution for existing HVAC systems," *Energy Build*, vol. 120, pp. 145–158, 2016.

[279] A. Ruano, S. Silva, H. Duarte, and P. M. Ferreira, "Wireless sensors and IoT platform for intelligent HVAC control," *Applied Sciences*, vol. 8, no. 3, p. 370, 2018.

[280] A. Ruano, K. Bot, and M. G. Ruano, "Home Energy Management System in an Algarve Residence. First Results," in *CONTROLO 2020*, 2021, pp. 332–341.

[281] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15, 2015.

[282] R. Caruana, S. Lawrence, and L. Giles, "Over Fitting in Neural Nets: Back-propagation, Conjugate Gradient, and Early Stopping Neural Information Processing Systems," *Denver, Colorado*, 2000.

[283] M. A. Mengistu, A. A. Girmay, C. Camarda, A. Acquaviva, and E. Patti, "A Cloud-Based On-Line Disaggregation Algorithm for Home Appliance Loads," *IEEE Trans Smart Grid*, vol. 10, no. 3, pp. 3430–3439, 2019, doi: 10.1109/TSG.2018.2826844.

[284] C. Athanasiadis, D. Doukas, T. Papadopoulos, and A. Chrysopoulos, "A scalable real-time non-intrusive load monitoring system for the estimation of household appliance power consumption," *Energies (Basel)*, vol. 14, no. 3, 2021, doi: 10.3390/en14030767.

[285] C. Cortes and V. Vapnik, "Support-vector networks," *Mach Learn*, vol. 20, no. 3, pp. 273–297, 1995.

[286] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN Model-Based Approach in Classification," in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, 2003, pp. 986–996.

[287] C. C. Yang, C. S. Soh, and V. V. Yap, "A systematic approach in appliance disaggregation using k-nearest neighbours and naive Bayes classifiers for energy efficiency," *Energy Effic*, 2018, doi: 10.1007/s12053-017-9561-0.

[288] H. Liu, "Machine Learning Based Appliance Identification," in *Non-intrusive Load Monitoring*, Springer, 2020, pp. 141–162.

[289] J. R. Quinlan, "Induction of decision trees," *Mach Learn*, vol. 1, no. 1, pp. 81–106, 1986, doi: 10.1007/BF00116251.

[290] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," Royal Signals and Radar Establishment Malvern (United Kingdom), 1988.

[291] A. E. Ruano, E. M. Crispim, and P. M. Frazão, "MOGA design of neural network predictors of inside temperature in public buildings," in *Soft Computing Based Modeling in Intelligent Systems*, Springer, 2009, pp. 35–61.

[292] K. Levenberg and F. Arsenal, "A method for the solution of certain non-linear problems in least squares," *Quart. Appl. Math*, vol. 2, pp. 164–168, 1944, doi: https://doi.org/10.1090/qam/10666.

[293] C. Chinrungrueng and C. H. Sequin, "Optimal adaptive k-means algorithm with dynamic adjustment of learning rate," *IEEE Trans Neural Netw*, vol. 6, no. 1, pp. 157–169, 1995.

[294] S. Haykin and N. Network, "A comprehensive foundation," *Neural networks*, vol. 2, no. 2004, p. 41, 2004.

[295] H. REZA KHOSRAVANI, “Artificial neural network models: Data selection and online adaptation,” University of Algarve, Faro, 2017. Accessed: Jul. 20, 2022. [Online]. Available: https://sapientia.ualg.pt/bitstream/10400.1/10807/1/PhD_thesis_final_print_after_corrections.pdf

[296] P. M. Ferreira and A. E. Ruano, “Evolutionary Multiobjective Neural Network Models Identification: Evolving Task-Optimised Models,” *Studies in Computational Intelligence*, vol. 372, pp. 21–53, 2011, doi: 10.1007/978-3-642-11739-8_2.

[297] K. Bot, A. Ruano, M. Da, and G. Ruano, “Short-Term Forecasting Photovoltaic Solar Power for Home Energy Management Systems,” 2021, doi: 10.3390/inventions6010012.

[298] H. Harkat, A. E. Ruano, M. G. Ruano, and S. D. Bennani, “GPR target detection using a neural network classifier designed by a multi-objective genetic algorithm,” *Applied Soft Computing Journal*, vol. 79, pp. 310–325, 2019, doi: 10.1016/j.asoc.2019.03.030.

[299] R. Machlev, J. Belikov, Y. Beck, and Y. Levron, “MO-NILM: A multi-objective evolutionary algorithm for NILM classification,” *Energy Build*, 2019, doi: 10.1016/j.enbuild.2019.06.046.

[300] K. Deb, “Multi-Objective Optimization,” in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, E. K. Burke and G. Kendall, Eds. Boston, MA: Springer US, 2005, pp. 273–316. doi: 10.1007/0-387-28356-0_10.

[301] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[302] D. Whitley, “A genetic algorithm tutorial,” *Stat Comput*, vol. 4, no. 2, pp. 65–85, 1994.

[303] M. Safe, J. Carballido, I. Ponzoni, and N. Brignole, “On stopping criteria for genetic algorithms,” in *Brazilian Symposium on Artificial Intelligence*, 2004, pp. 405–413.

[304] C. M. Fonseca and P. J. Fleming, “An overview of evolutionary algorithms in multiobjective optimization,” *Evol Comput*, vol. 3, no. 1, pp. 1–16, 1995.

[305] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, “Multiobjective evolutionary algorithms: A survey of the state of the art,” *Swarm Evol Comput*, vol. 1, no. 1, pp. 32–49, 2011.

[306] E. Hajimani, “Intelligent support system for CVA diagnosis by cerebral computerized tomography,” University of Algarve, Faro, 2017. Accessed: Jul. 20, 2022. [Online]. Available: http://hdl.handle.net/10400.1/10799

[307] C. M. Fonseca and P. J. Fleming, “Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation,” *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 28, no. 1, pp. 26–37, 1998.

[308] C. M. Fonseca and P. J. Fleming, “Multiobjective genetic algorithms made easy: selection sharing and mating restriction,” in *First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, 1995, pp. 45–52.

[309] D. W. Marquardt, “An Algorithm for Least-Squares Estimation of Nonlinear Parameters,” *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, Jun. 1963, doi: 10.1137/0111030.

[310] C. A. Teixeira, M. G. Ruano, A. E. Ruano, and W. C. A. Pereira, “A soft-computing methodology for noninvasive time-spatial temperature estimation,” *IEEE Trans Biomed Eng*, vol. 55, no. 2, pp. 572–580, 2008, doi: 10.1109/TBME.2007.901029.

[311] E. Hajimani, M. G. Ruano, and A. E. Ruano, “MOGA design for neural networks based system for automatic diagnosis of Cerebral Vascular Accidents,” *WISP 2015 - IEEE International Symposium on Intelligent Signal Processing, Proceedings*, 2015, doi: 10.1109/WISP.2015.7139170.

[312] UALG, “NILMFORIHEM _DATA,” 2022. https://csi.ualg.pt/nilmforihem/en/acesso-a-dados-en/ (accessed Jul. 26, 2022).

[313] L. Liu *et al.*, “Non-Intrusive Load Monitoring Method Considering the Time-Segmented State Probability,” *IEEE Access*, vol. 10, pp. 39627–39637, 2022.

[314] R. Machlev, Y. Levron, and Y. Beck, “Modified cross-entropy method for classification of events in NILM systems,” *IEEE Trans Smart Grid*, vol. 10, no. 5, pp. 4962–4973, 2018.

[315] K. Bot, I. Laouali, A. Ruano, and M. da G. Ruano, “Home Energy Management Systems with Branch-and-Bound Model-Based Predictive Control Techniques,” *Energies (Basel)*, vol. 14, no. 18, p. 5852, 2021.