Laboratoire Systèmes et Applications des Technologies de l'Information et de l'Energie

Dipartimento di Ingegneria dell'Informazione ed Elettrica e Matematica Applicata

Ecole Doctorale Sciences et Ingénierie nº 417: Génie électrique et électronique

Dottorato di Ricerca, di Interesse Nazionale in "Photovoltaics", Ciclo 38

Ph.D. Thesis

On-line diagnosis and optimization of Energy Management Systems for Smart Buildings

Luis Enrique Garcia Marrero

Supervisors

Prof. Eric Monmasson

Prof. Giovanni Petrone

Jury President: Prof. Mickael Hilairet

Academic Year: 2024-2025

Abstract

This thesis presents an integrated framework for demand- and supply-side intelligence in smart-building Energy Management Systems (EMS) designed for deployment on resource-constrained edge-computing platforms. The demand-side research focuses on appliance load monitoring using the cost-efficient Non-Intrusive Load Monitoring (NILM) approach. Current state-of-the-art NILM methods face limitations, including poor generalization to domain shifts and high computational requirements. Initially, a theoretical study of convolutional neural network-based NILM architectures is conducted, quantifying performance degradation under domain shifts through a first-order Taylor expansion and identifying the primary error sources that affect generalization. To solve this practical limitation, a novel training-less NILM framework is developed, combining a probabilistic appliance state model, dynamic programming for sequential updates, a lightweight base-load estimation module, and a population-based incremental learning algorithm. The proposed method operates in real-time, is robust to domain shifts, and eliminates the need for abundant appliance-specific training data.

The supply-side research focuses on model-based PV diagnostics, with a particular emphasis on enhancing parameter identification under real-world operating conditions. Current model-based methods for monitoring PV modules typically rely on the single-diode model (SDM) or its variants, assuming uniform operating conditions that are rarely achieved in real-world applications. When a PV module operates under mismatching conditions, estimating SDM parameters under the assumption of uniformity introduces errors that render the parameters unreliable for diagnostic purposes. As a first step, a multi-objective optimization framework is developed for the joint identification of parameters in static and dynamic PV models, integrating current-voltage (I-V) curve data with impedance spectroscopy (IS) measurements to produce physically meaningful parameter estimates. However, this approach is currently limited by its reliance on IS hardware, which is not yet sufficiently mature for online PV applications. To overcome this limitation, a self-adapting seven-parameter Double Single-Diode Model (D-SDM) is introduced,

which estimates parameters using only I-V data. A robust error function is proposed to isolate valid curve segments, and evolutionary algorithms are employed for parameter fitting, ensuring stable and accurate estimation under real-world operating conditions, while reliably detecting degradation phenomena such as increases in series resistance.

Finally, to demonstrate the practical feasibility of the technical advancements developed in this thesis, the NILM and PV diagnostic frameworks are deployed into a unified edge-computing platform capable of executing both tasks concurrently. The system is tested on a Raspberry Pi 4 running Home Assistant, an open-source platform widely used for home automation. All processing is performed locally on the edge device, and results are delivered through a single, integrated interface that ensures user privacy and ease of use. Experimental validation confirms that the platform maintains low latency, stable performance, and continuous operation while providing real-time demand- and supply-side monitoring. Furthermore, to support large-scale deployment of edge-based energy monitoring solutions, a commercialization strategy is proposed that combines local computation with optional cloud-based services such as accurate appliance modeling, forecasting, and automated insights. This strategy targets residential users, PV system owners, and professional installers, offering a scalable and cost-effective solution for real-time energy monitoring and management in smart buildings.

This thesis is conducted within the framework of the SMARTGYsum (SMART Green energY Systems and bUsiness Models) research and training program. The main objective of this Innovative Training Network (ITN) is to establish a multidisciplinary and innovative framework that integrates academic and industrial expertise to train a new generation of Early Stage Researchers (ESRs) capable of advancing the European Green Energy Economy. SMARTGYsum specifically focuses on equipping ESRs with essential knowledge, methodologies, and practical skills across various disciplines related to the energy ecosystem. The program emphasizes the Renewable Electric Energy Systems (REES), aiming to support the development and consolidation of business models that facilitate the integration of renewable technologies into the energy system.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 955614.

Contents

Abstract iii

Acknowledgements v

List of Publications xiii

List of Symbols and Acronyms xviii

List of Figures xix

List of Tables xxiii

List of Algorithms xxv

I Research Context & Contributions 1

1 Introduction 3

1.1 Smart-Building Energy Management in Context 3

1.2 NILM and Demand-Side Intelligence 5

1.3 PV diagnostic and Supply-Side Intelligence 7

1.4 Thesis Contributions: Guided by the SmartGySum Project Objectives 8

1.5 Structure of the Dissertation 11

II Demand-Side Intelligence Through Non-Intrusive Load Monitoring 15

2 NILM Fundamentals 17

2.1 Problem formulation 17

2.2 Critical Overview of state-of-the-art 18

2.3 Design Constraints for Widespread NILM Adoption 21

2.4 Publicy Available Datasets 22

2.5 Performance Metrics 23
2.6 Conclusion of the Chapter 24

3 Transfer Capabilities of Seq2Seq and Seq2Point CNN architectures in Non-Intrusive Load Monitoring with Unseen Appliances 27
3.1 Introduction 27
3.2 Seq2Seq and Seq2Point models 29
3.3 Mathematical Analysis in the Presence of Unknown Appliances . . . 29
3.3.1 Single-Appliance Formulation 30
3.3.2 Sensitivity Analysis via First-Order Taylor Expansion 32
3.3.3 Influence of Appliance Characteristics 32
3.4 Experiments detail 34
3.4.1 Performance Metrics 34
3.4.2 Datasets and selected appliances 34
3.4.3 Settings 35
3.5 Experimental Results 38
3.6 Proposed Directions Towards Practical NILM 40
3.7 Conclusion of the Chapter 43

4 Online Real-Time Robust Framework for Non-Intrusive Load Monitoring in Constrained Edge Devices 45
4.1 Introduction 45
4.1.1 Related research 46
4.1.2 Our contribution 48
4.2 Preliminaries 50
4.3 Proposed framework: High-level architecture 52
4.4 Low level design 54
4.4.1 Module 1: Edge Detection and Base Load Detection 54
4.4.2 Module 2: State Probabilities Update 57
4.4.3 Module 3: State Probabilities Tuning 61
4.4.4 Module 4: States Prediction 62
4.5 Experimental Results 64
4.5.1 Performance Metrics 66
4.5.2 Experimental Setup 66

4.5.3 Performance Results 68
4.5.4 Robustness to Noise 69
4.5.5 Computational complexity and real-time capabilities 70
4.6 Conclusion of the chapter 71

III Supply-Side Intelligence Through Robust PV Characterization 73

5 PV Characterization Fundamentals 75
5.1 SDM of a PV cell and module 75
5.2 PV degradation impact on the SDM parameters 77
5.3 Overview of SDM parameter estimation methods 78
5.4 Limits of the Uniformity Assumption in SDM Parameter Estimation 80
5.5 Frequency Domain Analysis as a Complement 82
5.5.1 IS Technique 82
5.5.2 Frequency domain modeling with a CPE 83
5.5.3 Practical Limitations of IS Technique 84
5.6 Conclusion of the Chapter 86

6 Multi-Objective Optimization for Identifying SDM and CPE Model Parameters 87
6.1 Introduction 87
6.2 Multi-objective formulation and validation 89
6.2.1 SDM fitting error calculation 89
6.2.2 CPE fitting error calculation 90
6.2.3 Decision variables analysis 90
6.2.4 Normalization method 90
6.2.5 Proposed Algorithm 91
6.2.6 Temperature dependence of reverse saturation current for parameters validation 92
6.3 Results on experimental data 93
6.3.1 Experimental data 93
6.3.2 Fitting models independently 94
6.3.3 Multi-objective optimization results 96

6.3.4 Impact of tradeoff solutions in the model's parameters 98
6.4 Conclusion of the Chapter 98

7 Self-Adaptive Single-Diode Model Parameter Identification Under Real-World Operating Conditions 101
7.1 Introduction 101
7.2 Limitations of the Uniformity Assumption Revisited 104
7.3 Adaptive Double-SDM of a PV module 106
7.4 Proposed method to identify the D-SDM parameters 112
7.4.1 Error calculation 115
7.4.2 Optimization method 116
7.5 Validation Results 116
7.5.1 Experimental campaign 117
7.5.2 Overview of D-SDM parameter estimates 118
7.5.3 Assessing the Stability of Parameter Estimates 121
7.5.4 Validation through simulated I–V curves 122
7.6 Detection of series resistance degradation 126
7.6.1 Parameter estimation of the SDM 126
7.6.2 Parameter estimation of the CPE model 126
7.6.3 Experimental Campaign 127
7.6.4 Overview of the fitting for each analyzed model 127
7.6.5 Estimation of the degradation 129
7.7 Conclusion of the Chapter 130

IV Integrated Demand-Supply Energy Intelligence and Business Model 133

8 Practical deployment of O2RE-NILM and D-SDM PV parameter identification in a Unified Platform 135
8.1 Introduction 135
8.2 Home Assistant Installation and Initial Setup 137
8.3 O2RE-NILM: practical deployment 139
8.3.1 Architecture of the prototype 139
8.3.2 Configuration and Visualization of the Results 141

8.4 D-SDM PV parameter identification: practical deployment 144
8.4.1 Architecture of the prototype 144
8.4.2 Configuration and Visualization of the Results 145
8.5 Performance Evaluation 147
8.5.1 O2RE-NILM performance 148
8.5.2 D-SDM PV performance 149
8.6 Conclusions of the Chapter 150

9 Business Model for Integrated Real-Time Energy Intelligence 153
9.1 Introduction 153
9.2 Architectural Overview of the IREI Model 154
9.3 Integrated Analytical Services 157
9.4 Market Positioning and Growth Potential 158
9.5 Conclusion of the Chapter 159

V Concluding Remarks and Future Directions 161

10 General Conclusions & Future Work 163
10.1 General Conclusions 163
10.2 Future Work 165
10.2.1 NILM 165
10.2.2 PV diagnostic 166
10.2.3 Integration of NILM, PV Diagnostics, and EMS 166

Bibliography 190

List of Publications

The research in this thesis has been previously disseminated through a series of journal and conference publications. Publications marked with an asterisk **(*)** correspond to the main contributions, as they summarize the essential results of this thesis. The remaining publications apply and extend the methods, models, and approaches developed here to complementary problems. These publications are:

Journals

***J1.** **L.E. Garcia-Marrero**, E. Monmasson, G. Petrone, "Online real-time robust framework for non-intrusive load monitoring in constrained edge devices," *Applied Energy*, vol. 378, Jan. 2025, 124814. DOI: `https://doi.org/10.1016/j.apenergy.2024.124814`

***J2.** **L.E. Garcia-Marrero**, C.I. Pavón-Vargas, J.D. Bastidas-Rodríguez, E. Monmasson, G. Petrone, "Self-adaptive single-diode model parameter identification under small mismatching conditions," *Renewable Energy*, vol. 245, Jun. 2025, 122735. DOI: `https://doi.org/10.1016/j.renene.2025.122735`

***J3.** **L.E. Garcia-Marrero**, G. Petrone, E. Monmasson, "Transfer capabilities of Seq2Seq and Seq2Point CNN architectures in Non-intrusive Load Monitoring with unseen appliances," *Mathematics and Computers in Simulation*, vol. 239, Jan. 2026, pp. 211–222. DOI: `https://doi.org/10.1016/j.matcom.2025.05.021`

J4. M. Piliougine, **L.E. Garcia-Marrero**, K. Lappalainen, G. Spagnuolo, "Influence of the temperature on the intrinsic parameters of thin-film photovoltaic modules," *Renewable Energy*, vol. 240, Feb. 2025, 122068. DOI: `https://doi.org/10.1016/j.renene.2024.122068`

J5. C. Pavón-Vargas, **L.E. Garcia-Marrero**, J.D. Bastidas-Rodríguez, R.A. Guejia-Burbano, G. Petrone, "Experimental Assessment of Partial Shading

Detection in PV Panels Using Impedance Spectroscopy," *IEEE Transactions on Industry Applications*, 2025, pp. 1–14. DOI: https://doi.org/10.1109/TIA.2025.3561708

J6. C. Pavón-Vargas, **L.E. Garcia-Marrero**, G. Petrone, "Dynamic Photovoltaic Modeling for Circuit-Emphasis Simulation," *Submitted to Renewable Energy*, 2025

Conferences

***C1.** **L.E. Garcia-Marrero**, R.A. Guejia-Burbano, G. Petrone, M. Piliougine, E. Monmasson, "Identification of static and dynamic parameters of PV models through multi-objective optimization," *2023 IEEE CPE-POWERENG*, Jun. 2023, pp. 1–6. DOI: https://doi.org/10.1109/CPE-POWERENG58103.2023.10227400

***C2.** **L.E. Garcia-Marrero**, M. Piliougine, G. Petrone, M. De Riso, P. Guerriero, E. Monmasson, "Challenges in photovoltaic parameter identification under mismatching conditions," *2023 ICCEP*, Jun. 2023, pp. 436–444. DOI: https://doi.org/10.1109/ICCEP57914.2023.10247445

***C3.** **L.E. Garcia-Marrero**, G. Petrone, E. Monmasson, "Detection of Series Resistance Degradation in PV Modules Using Measured Current-Voltage and Frequency-Domain Impedance," *2025 ICCEP*, Jun. 2025, pp. 250–255. DOI: http://dx.doi.org/10.1109/ICCEP65222.2025.11143628

C4. C. Pavón-Vargas, **L.E. Garcia-Marrero**, G. Petrone, S. Curcio, "An enhanced single-diode model of photovoltaic panels for SPICE simulations," presented at *ELECTRIMACS 2024*, 27$^{\text{th}}$–30$^{\text{th}}$ May 2024, Castelló, Spain. Accepted as a book chapter for publication in the Springer series *Lecture Notes in Electrical Engineering*. URL: https://www.springer.com/series/7818

C5. **L.E. Garcia-Marrero**, C. Pavón-Vargas, G. Petrone, E. Monmasson, "Advancements in Photovoltaic Research: Electro-Impedance Spectroscopy for PV Cell Characterization," *2024 SPEEDAM*, Jun. 2024, pp. 776–781. DOI: https://doi.org/10.1109/SPEEDAM61530.2024.10609231

C6. M. De Riso, I. Matacena, P. Guerriero, S. Daliento, **L.E. Garcia-Marrero**, G. Petrone, "Dynamic Modeling of Si-based Photovoltaic Modules using Impedance Spectroscopy Technique," *2023 ICCEP*, Jun. 2023, pp. 430–435. DOI: https://doi.org/10.1109/ICCEP57914.2023.10247369

C7. C. Pavón-Vargas, R.A. Guejia-Burbano, **L.E. Garcia-Marrero**, G. Petrone, "Impedance Spectroscopy for partial shading detection on series-connected PV panels," *2024 SPEEDAM*, Jun. 2024, pp. 762–767. DOI: https://doi.org/10.1109/SPEEDAM61530.2024.10609152

List of Symbols and Acronyms

%-NM — Percent-Noisy Measure

$I_{\mathbf{ph}}$ — photoinduced current

$I_{\mathbf{sc}}$ — short-circuit current

$I_{\mathbf{s}}$ — saturation current

$R_{\mathbf{sh}}$ — shunt resistance

$R_{\mathbf{s}}$ — series resistance

$V_{\mathbf{oc}}$ — open-circuit voltage

η — ideality factor

AFHMM — Additive Factorial Hidden Markov Model

AILP — Aided Linear Integer Programming

BW — Balanced Window

CNN — Convolutional Neural Network

CPE — Constant Phase Element

D-SDM — Double Single-Diode Model

DDM — Double-Diode Model

DW — Dishwasher

EA — Estimation Accuracy

EIS — Electrochemical Impedance Spectroscopy

EMS — Energy Management System

EO — Electric Oven

ESR — Early Stage Researcher

EVA — ethylene-vinyl acetate

FF — fill factor

FHMM — Factorial Hidden Markov Model

FM — F-measure

FSM — Finite State Machine

GA — Genetic Algorithm

GSP — Graph Signal Processing

HA — Home Assistant

HMM — Hidden Markov Model

I–V — current–voltage

ICT — Information and Communication Technologies

IEA — International Energy Agency

IoT — Internet of Things

IP — Integer Programming

IREI — Integrated Real-Time Energy Intelligence

IS — Impedance Spectroscopy

ITN — Innovative Training Network

K — Kettle

MILP — Mixed Integer Linear Programming

MLMoCo — Multilayer Momentum Contrast

MLPE — Module-Level Power Electronics

MOEA — Multi-Objective Evolutionary Algorithm

MOP — Multi-Objective Optimization Problem

MPP — Maximum power point

MW — Microwave

NILM — Non-Intrusive Load Monitoring

NSGA-II — Nondominated Sorting Genetic Algorithm II

O2RE — Online, Real-time, Robust, and Edge-driven NILM framework

OEA — Overall Estimation Accuracy

OFM — Overall F-measure

PBIL — Population-Based Incremental-Learning

PID — potential-induced degradation

PSO — Particle Swarm Optimization

PV — Photovoltaic

PVPS — Photovoltaic Power Systems Programme

RC — Rice Cooker

REDD — Reference Energy Disaggregation Dataset

REES — Renewable Electric Energy Systems

SDM — Single-Diode Model

Seq2Point — Sequence-to-Point

Seq2Seq — Sequence-to-Sequence

SMARTGYsum — SMART Green energY Systems and bUsiness Models

SSL — Self-Supervised Learning

TDM — Triple-Diode Model

UDA — Unsupervised Domain Adaptation

UK-DALE — UK Domestic Appliance-Level Electricity

WD — Washer Dryer

WM — Washing Machine

WP4 — Work Package 4

List of Figures

1.1 Load Monitoring Approaches 6
1.2 Residential PV Monitoring Architecture 8
1.3 Considered Integrated Architecture 9

2.1 Non-intrusive load monitoring decomposes the aggregate mains power trace (left) into its appliance-level loads (right). 18

3.1 Architectures for sequence-to-point and sequence-to-sequence neural networks. 29
3.2 Breakdown of household appliance contributions to total energy levels for UKDALE-5, UKDALE-2, and REDD-1 datasets. 'Other' accounts for non-monitored appliances. 36
3.3 Timeline of activation for all submetered appliances in the three datasets. 37
3.4 Robustness to noise for the UKDALE-5 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases. 39
3.5 Robustness to noise for the UKDALE-2 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases. 39
3.6 Robustness to noise for the REDD-1 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases. 39
3.7 Appliance metrics as percent-noisy increases for UKDALE-5 dataset. 41
3.8 Appliance metrics as percent-noisy increases for UKDALE-2 dataset. 41
3.9 Appliance metrics as percent-noisy increases for REDD-1 dataset. . . 42

4.1 Finite State Machine Model. The diagram shows the power consumption for each state (P) and the probability (S_p) of being in that state at a given time t. Transitions between states are labeled with power differences (ΔP). 51

4.2 Example of the likelihood for four possible state transitions. For the edge $E = 250W$, the highest likelihood is associated with the appliance transition from 0 to $230W$ that, among the other possible state transitions and the unknown transition reported in the figure, is the most likely to represent the power variation E. 59

4.3 Performance results Scenario 1 (REDD-1, 96h) 69

4.4 Performance results Scenario 2 (UKDALE-2, 720h) 69

4.5 Performance results Scenario 3 (UKDALE-5, 72h) 69

4.6 Robustness as the percent-noisy increases 70

5.1 Single diode model for a single solar cell. 76

5.2 Impact of Single-Diode Model (SDM) parameter variations on the $I-V$ characteristics of a PV module. The legend shows the effect of varying one parameter while keeping the others at nominal values. It indicates the specific parameter change and the corresponding relative loss in maximum power point ΔP_{mpp}. 79

5.3 Effect of progressively shading a single solar cell in a 32-cell series string (2×16-cell sub-modules). The left panel shows simulated IV-curves for five shading levels (0–20 % in 5 % steps). The right panel shows curves obtained by fitting a SDM to each simulation. Legend entries on the right combine the shading percentage with the relative change of key SDM parameters with respect to the unshaded baseline: I_s and R_{sh} as multiplicative factors ($\times$ or /), and I_{ph}, R_s and η as signed percentage changes. 82

5.4 Dynamic equivalent circuit including a constant phase element (CPE). 84

5.5 Impedance response in the complex plane based on the CPE model. The arc reflects the non-ideal capacitive behavior of the junction. . . 85

6.1 Independently optimization 95

6.2 Effect of the difference between R_s and R_0 97

6.3 Normalized Pareto front 97

6.4 Fitting for tradeoff solution 99

7.1 Uniform assumption limitations. A. Experimental data; B. SDM fitting using the full experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc} 105
7.2 Two-irradiance levels assumption for a PV module with N_s cells in series 106
7.3 D-SDM model representing groups a and b in series. 107
7.4 Simulink block of the schema shown in Figure 7.2 108
7.5 Operating voltage of Cell versus PV module voltage in different mismatched scenarios for a 2% difference in irradiance conditions ($G_a = 0.98G_b$). 108
7.6 Mismatching cases analyzed. (I). Clean panel without shadows but with non-uniform radiation produced by the reflexes on the surface; (II). Partial shading by surrounding objects; (III). Partial shading by bird droppings; (IV). Partial shading by soiling. 119
7.7 Example of fitting results for experimental I–V curves. I. Uniform Conditions; II. Partial shading by surrounding objects; III. Partial shading by bird droppings; I–V. Partial shading by soiling 119
7.8 Uniform assumption limitations. A. Experimental data; B. SDM fitting using all the experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc}, D. D-SDM fitting using all the experimental points. 120
7.9 Comparison of coefficient of variation for R_s and R_{sh} 122
7.10 Comparison of coefficient of variation for I_s and η 123
7.11 True value versus estimation for $R_{s\text{-}c}$ (left) and $R_{sh\text{-}c}$ (right) 125
7.12 True value versus estimation for $N_{s\text{-}a}/N_s$ (left) and $I_{ph\text{-}a}/I_{ph\text{-}b}$ (right) 125
7.13 Fitting results for the Single Diode Model (SDM) 128
7.14 Fitting results for the Double Single Diode Model (D-SDM) 129
7.15 Fitting results for the Constant Phase Element (CPE) model 129
7.16 Estimation of series resistance degradation by the SDM, D-SDM, and CPE models as a function of irradiance 130
7.17 Relative error in the estimation of ΔR_s for the SDM, D-SDM, and CPE models as a function of irradiance 131

8.1 Installation of Home Assistant OS for a Raspberry Pi 138

8.2 Raspberry Pi unit connected to a router, serving as a central hub for smart home automation and energy monitoring tasks. 138
8.3 Non-Intrusive Load Monitoring (NILM) Process: From Aggregated Power to Appliance-Level Disaggregation 139
8.4 Configuration of the NILM Add-on 142
8.5 Real-time and history visualization in Home Assistant 143
8.6 Visualization of total energy and individual appliances' energy in Home Assistant 143
8.7 Online PV parameter identification process in an edge device 145
8.8 D-SDM PV add-on configuration 146
8.9 D-SDM PV add-on results visualization 147
8.10 Concurrent operation of NILM and PV add-ons in Home Assistant . 148

List of Tables

2.1 Functional requirements for the NILM algorithm. 22

4.1 Appliance states used in simulation 67
4.2 Test Scenarios Description 67
4.3 Test Scenarios Settings (O2RE framework) 68

6.1 Specifications of the photovoltaic module 94
6.2 Parameters for independent optimization 96
6.3 Parameters for tradeoff solutions 98

7.1 Uniform assumption limitations. B. SDM fitting using the full experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc} 106
7.2 Best fitting parameters of a single cell for D-SDM, D-SDM-I, and SDM 121
7.3 Used ranges for generating simulated curves 124
7.4 Operating Conditions for Each Selected Case 127
7.5 Series Resistance and Fitting Error for Each Model 128

8.1 Performance Analysis of O2RE deployment Across Average and Critical Events 149
8.2 Runtime percentiles and implied safe launch rates 150

9.1 Lean Business Model Canvas for IREI 156

List of Algorithms

1 O2RE NILM Framework: High-level Architecture 55

2 Base Load Estimation: **UpdateBaseLoad**() function in Alg. 1 . . . 56

3 PBIL for appliance states prediction 65

4 Proposed algorithm for jointly SDM and CPE optimization 92

5 D-SDM: Binary search for finding $I_m(i)$ 114

Part I

Research Context & Contributions

1

Introduction

1.1 Smart-Building Energy Management in Context

Buildings are responsible for more than one-third of total electricity consumption and a comparable share of greenhouse gas emissions in developed countries [1]. This share continues to grow due to the increasing electrification of heating, cooling, and mobility. To address this trend, energy policies are moving toward greater efficiency, decarbonization, and local generation. These goals require Energy Management System (EMS) capable of supervising energy flows with fine temporal resolution, allowing for dynamic optimization of demand and supply under changing environmental and operational conditions [2, 3].

Modern EMS relies on data streams made available by smart grid infrastructure. Smart meters, sensors, and communication technologies provide continuous information on building energy consumption. These capabilities enable several energy services, including fault detection [4], short-term load forecasting [5], and automated demand response [6]. However, most existing monitoring systems focus on aggregate signals, which limit visibility of individual behaviors within the building and reduce the precision of optimization strategies. Furthermore, occupant behavior introduces frequent and unpredictable variations that EMS must manage in real-time [7].

On-site renewable energy production, particularly from Photovoltaic (PV)

systems, introduces additional challenges. PV systems are affected by external and dynamic environmental conditions such as irradiance, temperature, and shading. These factors directly impact the electrical characteristics of PV modules and result in variations in energy production throughout the day. Phenomena like partial shading, soiling, and ageing can alter the internal behavior of the modules in non-uniform ways, degrading performance and making it difficult to estimate the system's state accurately [8]. Inconsistent or delayed detection of these conditions can reduce yield, compromise system reliability, and lead to suboptimal control decisions by the EMS [9].

To operate effectively, EMS must access detailed and reliable information about both consumption and production. Cloud-based solutions have traditionally been used to process such data, but they involve high latency, increased communication overhead, and raise privacy concerns [10, 11]. Edge computing has emerged as a viable alternative by enabling local data processing directly within the building environment [12]. This architecture reduces reliance on external servers and enhances data privacy and responsiveness. However, the computational resources of edge devices are limited, particularly in residential contexts where cost constraints favor low-power platforms. These constraints require the design of efficient algorithms that can operate with minimal memory, processing power, and energy consumption while still maintaining accurate and timely EMS operation [13].

The integration of real-time energy monitoring with edge-based EMS introduces a complex set of requirements. On the demand side, systems must deliver detailed and timely information on appliance-level consumption without relying on intrusive sub-metering. Non-Intrusive Load Monitoring (NILM) offers a viable solution but must operate efficiently on low-power edge devices while maintaining robustness under noisy and dynamic household conditions [14]. Real-time feedback from NILM is essential for enabling demand-response actions, detecting abnormal consumption patterns, and optimizing control decisions at the appliance level.

On the supply side, EMS requires continuous and accurate diagnosis of PV system performance to ensure reliable generation forecasts, fault detection, and long-term system integrity. To maintain effective EMS operation, PV diagnosis must identify changes in key electrical parameters, such as series resistance, with high sensitivity and stability across varying irradiance levels. These estimations

must be performed frequently to reflect real-time conditions and integrated into the EMS without imposing excessive computational loads. This requirement becomes even more stringent when diagnosis is implemented on edge devices. Consequently, PV diagnostic algorithms must balance accuracy and robustness with efficiency, ensuring they remain functional under hardware limitations while supporting critical EMS tasks [15].

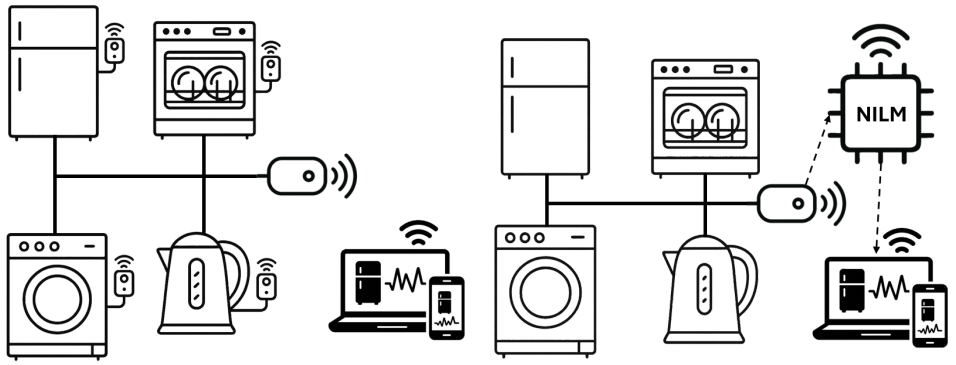
Combining these functions within the constraints of edge computing involves balancing accuracy, speed, and hardware limitations. EMS must integrate robust NILM and PV diagnostic tools that function with minimal resources, while still providing the precision and responsiveness needed for effective optimization. These challenges define the operational context for future EMS, which must deliver reliable, autonomous energy management within realistic smart-building environments.

1.2 NILM and Demand-Side Intelligence

EMS requires detailed knowledge of consumption patterns to perform fine-grained control, detect inefficiencies, and respond to user behavior in real time. However, most buildings are equipped only with a single electricity meter that provides the total active power consumption of all devices combined. This aggregated signal hides the operation of individual appliances, making it difficult to optimize energy use, detect faults, or provide feedback to occupants. While the installation of dedicated sub-meters can resolve this issue, it remains economically and logistically impractical in most residential and small-commercial settings [16].

NILM addresses this problem by estimating the power consumption of individual appliances from the aggregated signal, using algorithms that disaggregate the total load into its constituent components [17]. Figure 1.1 presents a diagram of load monitoring approaches, with the intrusive approach depicted on the left and the non-intrusive approach on the right. Since its original formulation, NILM has evolved from simple thresholding and rule-based techniques to advanced machine learning approaches that model the temporal dynamics of appliances with increasing accuracy [18]. In particular, deep learning architectures such as sequence-to-point and sequence-to-sequence convolutional networks have demonstrated state-

of-the-art performance on benchmark datasets [19]. These methods learn appliance signatures directly from data and can infer the state and power consumption of each device with high accuracy [20].



(a) Intrusive Load Monitoring (b) Non-Intrusive Load Monitoring

Figure 1.1: Load Monitoring Approaches

Despite this progress, several challenges limit the deployment of NILM in real-world EMS. First, most high-accuracy models require large volumes of labeled data for training, which is rarely available in domestic settings. The need for appliance-specific annotations complicates the use of supervised methods, especially when the appliance inventory changes over time or differs across households [16]. Second, these models tend to overfit to specific usage patterns and struggle with domain transferability, showing significant performance degradation when applied to new environments with unseen loads [18]. Third, deep learning methods are computationally intensive and memory demanding, which hinders their implementation on edge devices that operate under strict resource constraints [14].

In summary, existing NILM approaches achieve high accuracy under controlled conditions but fail to generalize in the presence of unseen appliances, limited training data, and constrained hardware. There is a growing need for NILM methods that are robust, adaptive, and compatible with real-time edge processing. These limitations define the research gap for practical NILM deployment and motivate the development of algorithms that can be integrated into edge-based EMS without sacrificing performance or usability.

1.3 PV diagnostic and Supply-Side Intelligence

EMS in smart buildings must monitor not only consumption but also local energy generation. PV systems, now widely adopted in residential and commercial buildings, are subject to environmental variability, ageing, and physical mismatches that affect their performance over time. Reliable supply-side intelligence, therefore, requires continuous diagnostic capabilities to detect faults, assess degradation, and maintain efficient operation under real-world conditions [21].

Model-based diagnostic methods are commonly employed to monitor the state of PV systems by fitting electrical models to measured data [22]. The SDM and its extensions are widely used to characterize the current-voltage (I-V) behavior of PV modules. By estimating parameters such as series resistance, shunt resistance, photocurrent, and ideality factor, these methods allow for the detection of performance issues and the tracking of degradation trends over time [23]. Because these parameters reflect internal physical changes in the module, model-based diagnostics provide a deeper understanding than simple energy output comparisons or power ratio metrics.

The practical application of this approach has become increasingly feasible due to the evolution of inverter technology. Many commercial inverters now include integrated I-V curve tracing capabilities that allow the system to perform voltage sweeps on demand or at scheduled intervals [24]. These measurements, originally intended for commissioning and performance verification, can now be used to support continuous monitoring. Furthermore, advanced inverter architectures such as Module-Level Power Electronics (MLPE) and microinverters enable I-V curve tracing at the level of individual PV modules [25]. This increased resolution improves fault localization and allows for a more granular and accurate diagnosis of system health. Figure 1.2 illustrates the considered PV monitoring architecture. The inverter sent I-V curve data to a local edge-processing device. This on-site processor analyzes the I-V curve in real-time and wirelessly transmits the processed data to user dashboards.

Despite these advancements, significant challenges remain. Traditional parameter extraction techniques are based on the assumption of uniform irradiance and temperature across the PV array. In real installations, this assumption is often

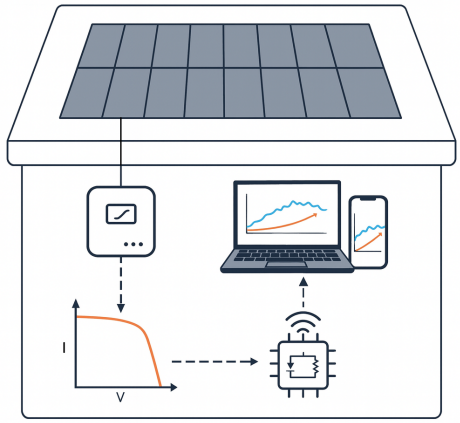


Figure 1.2: Residential PV Monitoring Architecture

violated due to partial shading, dirt accumulation, and mismatched components, resulting in distorted or non-standard I-V curves that are difficult to interpret. These conditions degrade the reliability of the parameter estimates and reduce their diagnostic value. Additionally, to be suitable for real-time EMS integration, diagnostic algorithms must operate with low computational overhead and high robustness, particularly when deployed on resource-constrained edge devices.

In summary, the ability of modern inverters to trace I-V curves, both at the string and module level, makes model-based PV diagnostic a practical and powerful tool for supply-side monitoring. However, accurate diagnosis under non-uniform conditions, consistent parameter estimation over time, and computational efficiency remain open challenges. Addressing these issues is essential to provide EMS with reliable real-time insights into PV performance and to ensure optimal energy management in smart buildings.

1.4 Thesis Contributions: Guided by the SmartGySum Project Objectives

This thesis is developed within the framework of the SmartGySum project, a European initiative aimed at supporting decentralized, intelligent, and sustainable energy systems. The work contributes directly to the goals of Work Package 4 (WP4), titled End User of Energy and Prosumers, which promotes the integration

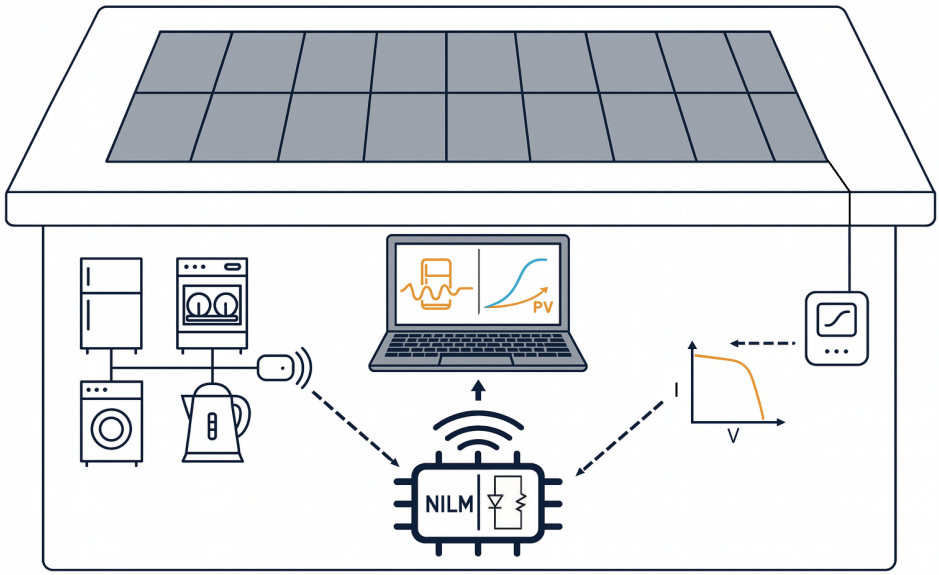


Figure 1.3: Considered Integrated Architecture

of advanced monitoring tools, behavioral engagement mechanisms, and flexible energy strategies to optimize local energy consumption and production. WP4 specifically encourages the use of power converters and Information and Communication Technologies (ICT) to reduce energy demand, improve user awareness, and support the practical deployment of intelligent control solutions in real environments [26].

In this context, the thesis addresses the technical and operational challenges of online diagnosis and optimization of renewable energy sources and electrical loads in smart buildings, with the aim of improving the EMS. Figure 1.3 provides an overview of the architecture considered in this work. It illustrates the integrated monitoring platform developed in the thesis, where both NILM and PV parameter identification tasks are performed by the same edge computing device. The processed information is made available to users through an external interface, enabling real-time, decentralized energy monitoring. Besides increasing user awareness and enabling timely maintenance, this information can then be used by the EMS to perform optimization tasks, such as scheduling controllable loads, improving self-consumption, identifying abnormal behaviors, and adapting energy usage strategies based on current and predicted conditions.

Based on the considered architecture, the contributions of the thesis are

structured around three main axes:

The first contribution focuses on demand-side monitoring through the development and evaluation of NILM algorithms suitable for deployment in edge devices. The thesis presents a theoretical and experimental analysis of convolutional neural network-based NILM architectures (Seq2Seq and Seq2Point), quantifying their sensitivity to domain shifts caused by the presence of unknown appliances in the aggregated signal. Using a first-order Taylor expansion, the analysis identifies the sources of error that limit transferability, providing a foundation for designing future architectures with improved robustness under zero-shot learning scenarios. To address the limitation of existing solutions, a novel training-less NILM framework is proposed, combining a probabilistic model for appliance state transitions, a dynamic programming algorithm for sequential state updates, and a lightweight base load estimation module. These components are integrated using a Population-Based Incremental Learning algorithm, resulting in a computationally efficient NILM method suitable for real-time operation on constrained hardware. The framework's robustness and accuracy are demonstrated using public datasets.

The second contribution focuses on supply-side intelligence through the development of improved PV parameter identification methods for robust diagnostics under real operating conditions. The work builds on a preliminary result where a multi-objective optimization framework jointly identified parameters for static and dynamic PV models using Impedance Spectroscopy (IS) and current–voltage (I–V) data, producing coherent and physically meaningful estimates. However, its deployment is limited by the need for IS hardware, the sensitivity of Pareto front solution selection, and the reliance on the uniformity assumption. To address these limitations, a self-adapting seven-parameter Double Single-Diode Model (D-SDM) is proposed, capable of accurately representing PV modules under both uniform and mismatched conditions. The methodology estimates parameters directly from I–V curves using evolutionary algorithms. A robust error function is proposed to isolate the portion of the I–V curve in which all cells operate at a positive voltage. The approach is validated using experimental and simulated I–V curves under various mismatch patterns. Its capability for detecting degradation phenomena such as series resistance increases is demonstrated through comparisons with the standard SDM and the dynamic Constant Phase Element (CPE) model. Results confirm the

stability, adaptability, and practicality of the method, which operates using only I–V data, enabling cost-effective deployment in field monitoring systems.

The third contribution is the practical integration of both NILM and PV parameter identification frameworks into a single edge-based platform. The same hardware unit executes both monitoring tasks concurrently, processes data locally, and exposes the results through a unified user interface. This implementation demonstrates the feasibility of combining demand- and supply-side intelligence into a decentralized EMS that functions autonomously in real environments. It confirms that high-resolution monitoring and optimization can be achieved using cost-effective, low-power embedded systems without compromising privacy or performance. In addition, a commercialization framework is proposed to support the deployment of the developed methods as a viable service, highlighting the importance of aligning technical innovation with practical pathways for adoption and large-scale impact.

1.5 Structure of the Dissertation

The remaining parts of the dissertation are structured to address the key research contributions developed throughout this work:

Part II presents the contributions related to demand-side monitoring through NILM.

Chapter 2 introduces NILM, defining its problem formulation and the challenges of estimating appliance-level consumption from aggregated measurements. It outlines practical design constraints such as low-frequency active power sampling and computational efficiency for edge devices. The chapter presents REDD and UK-DALE as standard datasets for reproducible evaluation and describes key performance metrics. It also provides a critical overview of training-based and training-less approaches, summarizing their strengths and limitations for practical NILM adoption.

Chapter 3 analyses the performance degradation of convolutional neural network architectures under domain shifts. The study focuses on Seq2Seq and Seq2Point models and explores how the presence of unknown appliances in the aggregated power data affects disaggregation accuracy. A theoretical analysis based

on a first-order Taylor expansion is developed to identify the sources of additional error introduced by unseen appliance signatures. Experimental tests are performed using real-world datasets with increasing levels of noise to simulate domain mismatches. The results demonstrate a significant dependency on the structure of the training domain and suggest that future methods should focus on appliance-specific representations. The content of this chapter has been previously divulged in **Publication J3.** [**27**].

Chapter 4 introduces a training-less NILM framework suitable for real-time deployment on edge computing devices. The algorithm models appliances as finite state machines and uses a probabilistic model to track state transitions, which are updated only upon detection of significant power edges. A separate module estimates the base load to enhance accuracy. These components are combined using a Population-Based Incremental Learning algorithm that determines the most likely appliance states at each time step. The system is evaluated against state-of-the-art methods and shows strong performance in accuracy, robustness to unknown appliances, and computational efficiency. The content of this chapter has been previously divulged in **Publication J1.** [**28**].

Part III addresses the contributions related to supply-side intelligence through improved PV parameter identification methods.

Chapter 5 introduces the fundamentals of PV characterization, focusing on the SDM as a compact representation of module behavior for condition monitoring and degradation analysis. It explains how common failure mechanisms affect SDM parameters and reviews analytical, numerical, and hybrid estimation methods, noting their reliance on the uniformity assumption, which is often violated in real conditions due to shading, soiling, or aging. The chapter demonstrates how such mismatches can lead to inaccurate and non-physical parameter estimates and presents IS as a complementary frequency-domain technique to enhance parameter identification. Although IS provides valuable diagnostic insights, current hardware limitations restrict its practical use to controlled environments.

Chapter 6 investigates the joint identification of parameters for static and dynamic PV models using a multi-objective optimization framework. It integrates frequency-domain IS data with static I–V characteristics, and exploits the shared series resistance in the models to obtain coherent and physically meaningful pa-

rameter sets. By selecting a tradeoff solution from the Pareto front, the approach enhances the reliability, supporting the unification of static and dynamic PV diagnostics. However, its practical deployment is limited by the need for additional IS hardware, the sensitivity of Pareto front solution selection to threshold tuning, and the reliance on a uniformity assumption, which can limit the practical real-world application. The content of this chapter has been previously divulged in **Publication C1.** [**29**].

Chapter 7 presents a self-adapting seven-parameter PV model, the D-SDM, which enhances the reliability of parameter estimation under both uniform and mismatched conditions. The methodology includes a robust fitting error function that automatically selects the valid region of the I–V curve and applies evolutionary algorithms for parameter estimation. The approach is validated using experimental and simulated curves obtained under real outdoor conditions with diverse mismatching patterns. The capability of D-SDM for detecting series resistance degradation is evaluated and compared with SDM and the dynamic CPE model. Measured I–V and impedance data from a commercial PV module under normal and degraded conditions are employed in this evaluation. The results confirm the stability and adaptability of the model, making it suitable for real-world PV monitoring. The content of this chapter has been previously divulged in **Publications J2., C2. and C3.** [**30–32**].

Part IV presents the integration and validation of the proposed NILM and PV methods on a shared edge-based platform, together with an accompanying business model.

Chapter 8 describes the practical deployment of the O2RE-NILM and D-SDM PV parameter identification methods within a unified edge-computing platform based on a Raspberry Pi 4 running Home Assistant, which is an open-source software widely used for home automation. The system enables concurrent real-time appliance-level load disaggregation and PV parameter estimation, integrating both functions into a user-friendly smart home environment with local processing, privacy, and minimal installation effort. The NILM add-on was validated with simulated and real sensors, providing responsive, real-time monitoring and historical visualization of appliance consumption. The D-SDM PV add-on, tested with simulated inverter data, performed parameter estimation reliably within typical

PV monitoring intervals. Performance evaluation over extended operation demonstrated that both add-ons ran concurrently without overloading the device, confirming the feasibility of combining NILM and PV diagnostics in a single, low-cost platform suitable for residential energy management.

Chapter 9 outlines a commercialization framework that enables the NILM and PV diagnostic methods proposed in this thesis to be deployed as a viable service. The approach uses a hybrid edge-cloud architecture, running the primary NILM algorithm locally on user-owned hardware for privacy and low-latency performance, while offering optional cloud services such as appliance embeddings, load forecasting, AI-driven insights, and PV diagnostics. A free tier with unlimited manually configured appliance models provides immediate entry, with pay-per-appliance and subscription-based upgrades allowing scalable accuracy and functionality. Market analysis identifies home automation users, PV system owners, and installers as key segments, with the Lean Business Model Canvas illustrating the alignment between architecture, value proposition, and revenue streams. Performance, modularity, and privacy-oriented design position the system to close the gap between underutilized smart meter data and practical, real-time residential energy intelligence.

Part V concludes the dissertation and defines research lines for future work.

Chapter 10 provides the general conclusions of the thesis and outlines future research directions. The general conclusions summarize the main findings of the thesis, highlighting the contributions to demand-side NILM and supply-side PV diagnostics, as well as their integration into a unified, edge-deployable platform. The proposed future work includes advancing NILM through the integration of auxiliary appliance parameters, incremental learning, and hybrid deep learning approaches with the possible extension of these approaches to industrial applications; extending D-SDM-based PV diagnostics for automated fault detection and validation across different PV technologies; and unifying demand- and supply-side analysis within EMS to enable AI-driven insights, accurate forecasting, and optimized energy flow for objectives such as cost reduction, self-consumption maximization, and carbon footprint minimization.

Part II

Demand-Side Intelligence Through Non-Intrusive Load Monitoring

2

NILM Fundamentals

Using data collected from smart meters brings substantial benefits to grid operation and maintenance [33]. These advantages include, but are not limited to, fault detection [4], improved load forecasting [5], and more efficient demand response [6]. Consequently, the availability of detailed consumption power data is determined to play an important role in ensuring the success of smart grid implementations [34].

In the field of energy monitoring, NILM empowers end-users and building owners by allowing them to track the power consumption of a group of appliances. This is achieved without the need for individual smart meters or sensors for each device. Instead, NILM analyzes the combined consumption data from a single smart meter, intending to provide an accurate estimation of power consumption at the appliance level. This approach, first introduced by Hart [17], is also referred to as load disaggregation or energy disaggregation.

2.1 Problem formulation

The goal of NILM is to produce accurate estimates $[\hat{p_1}(t), \ldots, \hat{p_N}(t)]$ to the actual power consumption $[p_1(t), \ldots, p_N(t)]$ of N monitored appliances at each time step t, using only the information of the power reading $P(t)$ that comes from the main meter in a household. The temporal interval between two successive time steps, denoted as t and $t+1$, defines the frequency at which data points are recorded or observed. The reading $P(t)$ is expected to aggregate the contribution

of monitored appliances, additional not-monitored appliances that could exist in a household, and measurement noise (Eq. (2.1)).

$$P(t) = \sum_{i=1}^{N} p_i(t) + \eta(t) \tag{2.1}$$

In (2.1), the residual term $\eta(t)$ includes measurement noise and the contribution of other not-monitored appliances. A diagram illustrating the disaggregation process is provided in Fig. 2.1. It is shown how, from one composite signal, the distinct usage patterns of appliances can be extracted.

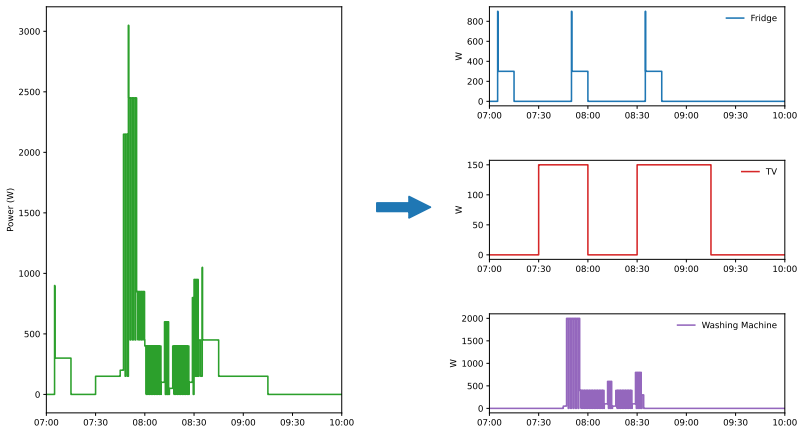


Figure 2.1: Non-intrusive load monitoring decomposes the aggregate mains power trace (left) into its appliance-level loads (right).

2.2 Critical Overview of state-of-the-art

Several approaches have been proposed to solve the NILM problem, with training-based algorithms being popular. Hidden Markov Models (HMM) and their extensions are training-based algorithms that build a probabilistic model of appliance states over time and then use this model to infer the operating states of the appliances [35, 36]. A practical limitation of this approach is its high computational complexity. To overcome this to some extent, researchers have introduced procedures exploiting sparsity [35], employing Factorial Hidden Markov Model

(FHMM) [36], additive FHMM (AFHMM) [37], Adaptive Density Peak Clustering FHMM [38], and event-based classification combined with AFHMM [39]. However, these HMM techniques require either abundant submetered data from the test domain or long periods of unsupervised training to build the probabilistic models, thus limiting seamless practical integration.

Lastly, various deep learning approaches, based on Convolutional Neural Networks (CNNs) [20, 40], Multilayer Momentum Contrast (MLMoCo) [41], self-supervised learning (SSL) [42] and unsupervised domain adaptation (UDA) [43], have been proposed in the literature. CNN-based architectures, in particular, have demonstrated state-of-the-art performance due to the meaningful latent feature extraction for appliances [44]. Since the introduction of sequence-to-sequence (Seq2Seq) and sequence-to-point (Seq2Point) CNN-based architectures [20], researchers have highlighted their potential for advancing NILM [19, 45].

However, three main drawbacks limit the seamless practical applicability of these methods for real-time predictions. The first limitation concerns generalizability. It is reasonable to assume that submetered data may not be available in real-world test environments. Therefore, improving their ability to generalize is crucial for successful deployment. Recent studies [18, 46] have highlighted that generalization to unseen sites remains a significant hurdle for state-of-the-art deep learning-based NILM models. This challenge arises from the diverse load characteristics and operating patterns across different locations, making it difficult for these models to adapt and perform accurately in new environments.

The second limitation is related to the explainability of NILM models [18]. Deep learning algorithms are inherently complex, often making their operations and decision-making processes difficult to interpret or replicate. Enhancing the transparency of NILM approaches could lead to a better understanding of their functionality, thus aiding in the refinement of models and identifying significant data features that influence decisions [18].

The last drawback is associated with specific resource constraints. As the variety of IoT services grows, multiple services running on the same hardware through a common software framework may become usual. Hence, there is a need to create lightweight and accurate models that can be seamlessly integrated into edge-constrained devices. Although several studies propose reduced models by

compression or reduced architectures of neural networks [14, 40, 47, 48], without significantly impacting accuracy, the achieved model size can still be prohibitive for constrained edge devices. Furthermore, the impact on the generalization capabilities of these lightweight models has not yet been thoroughly analyzed and validated.

In this context, a training-less solution holds practical significance for deploying NILM algorithms on edge devices. In this category, methods based on Integer Programming (IP) [49, 50], Graph Signal Processing (GSP) [51, 52], subtractive clustering [53], and the balanced window technique (BW) [54] have been proposed. Training-Less NILM approaches are commonly divided into two main groups: state-based algorithms and event-based algorithms [18]. State-based methods represent each appliance operation using a finite state machine (FSM) with distinct state transitions. Inside this group, optimization-based algorithms search for the global optimal state combination of appliances, either considering a time window or a unique time step. The main practical limitation of window-based optimization methods is their high computational complexity, especially at resolutions of a few seconds, which restricts their use in low-power embedded systems. They are also sensitive to the presence of unknown appliances, which can degrade accuracy.

On the other hand, event-based methods work by detecting edges in the aggregated power consumption and assigning each extracted feature to one appliance state transition. Unlike optimization-based methods, event-based approaches are less susceptible to the presence of unknown appliances. However, these methods are vulnerable to measurement noise and outliers. Typically, these approaches disaggregate appliances one at a time and often fail to verify whether the sum of the disaggregated loads approaches the actual measured total [53, 55]. This oversight can lead to significant load over/underestimation [56]. In the following chapters, a more targeted review of the relevant literature will be presented, aligned with the specific content of each chapter.

2.3 Design Constraints for Widespread NILM Adoption

Widespread adoption of NILM systems depends on designing algorithms that align with the technical limitations of existing infrastructure and the practical needs of real-world deployment. One key constraint is the sampling frequency. High-frequency data capture detailed features such as transients, harmonics, and voltage–current (V–I) trajectories, which can improve disaggregation accuracy [18, 46]. However, collecting such data requires advanced hardware, increasing cost, and reducing scalability [46]. Standard smart meters typically operate at frequencies below 100 Hz. To support near-real-time feedback and maintain compatibility with widely deployed devices, we consider appropriate, a sampling interval of no more than a few seconds.

We further restrict the monitored feature to active power. Smart meters deployed in most residential settings typically provide active power readings, as these are sufficient for billing purposes. Including additional features such as reactive power, voltage, or current can improve classification in specific contexts, but it also introduces some disadvantages. First, these features are often not accessible in low-cost metering infrastructure and require more complex hardware installations or privileged access to grid-level data. Second, using multiple features increases the dimensionality of the input space, which complicates algorithm design.

Moreover, incorporating additional electrical features generally increases the complexity of model configuration and fine-tuning. In real-world deployments, users or installers may be required to adjust hyperparameters or interpret the influence of various signals on appliance detection. This is less feasible when multiple correlated or noisy inputs are used. By focusing solely on active power, we reduce both hardware dependencies and algorithmic complexity, which in turn simplifies deployment, maintenance, and scalability. This design decision aligns with previous work that demonstrated the feasibility of active-power-only approaches under practical constraints [41, 42, 57].

In line with widely accepted development guidelines for NILM algorithms [58], Table 2.1 summarizes the key assumptions and functional requirements that guide the design of our framework.

Table 2.1: Functional requirements for the NILM algorithm.

#	Requirement	Key details
1	**Monitored feature & sampling rate**	Active power measurements with a sampling interval ranging from 0.125 to 1 Hz.
2	**Minimal training effort**	Requires unsupervised or semi-supervised learning, with minimal to no occupant involvement.
3	**Near-real-time feedback**	The algorithm should be computationally efficient and robust, providing feedback as close as possible to real events.
4	**Edge-device readiness**	Designed for online operation; handles a small stream of power measurements efficiently. Can be deployed on devices with limited processing power and memory.
5	**Robustness to appliance changes**	Performance must remain stable even when the set of monitored or unmonitored appliances changes within the building.

2.4 Publicy Available Datasets

Publicly available datasets are essential for evaluating NILM methods under reproducible conditions. Two widely used datasets in the NILM community are REDD [59] and UK-DALE [60]. REDD provides data from 10 houses in the United States. It includes whole-home voltage and current waveforms sampled at 15 kHz, up to 24 circuit-level measurements recorded at 0.5 Hz, and up to 20 plug-level appliance channels recorded at 1 Hz. Each channel is labeled by appliance category, supporting both supervised training and detailed disaggregation analysis.

UK-DALE contains data from five houses in the United Kingdom. It records the active power of individual appliances and the whole-house apparent power every six seconds. For three houses, it also provides whole-house voltage and current waveforms sampled at 44.1 kHz and downsampled to 16 kHz. House 1 includes 54 appliance channels and was recorded over 655 days. The dataset also includes active power, apparent power, and RMS voltage at 1 Hz, making it one of the most detailed long-term datasets available for residential energy research.

These datasets are particularly well-suited for low-frequency NILM studies because of their compatible sampling rates. They continue to serve as standard

benchmarks in recent research, with several low-frequency NILM studies relying on them for algorithm evaluation and comparison [41, 42, 61, 62]. Importantly, REDD and UK-DALE also broaden the evaluation scope geographically, supporting the assessment of NILM algorithms under different household conditions and energy usage behaviors in the United States and the United Kingdom.

2.5 Performance Metrics

In assessing the accuracy of NILM methods for individual appliances, we employ two widely used performance metrics: Estimation Accuracy (EA) and F-measure (FM). These metrics are defined by equations (2.2) and (2.3), respectively.

$$EA_i = 1 - \frac{\sum_{t=1}^{T} |p_i(t) - \hat{p}_i(t)|}{2 \sum_{t=1}^{T} p_i(t)} \tag{2.2}$$

$$\mathrm{FM}_i = \frac{2 \cdot \mathrm{P}_i \cdot \mathrm{R}_i}{\mathrm{P}_i + \mathrm{R}_i} \tag{2.3}$$

Where:

$$\mathrm{P}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i}, \quad \mathrm{R}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \tag{2.4}$$

Here, TP (true positive) represents the instances where an appliance is correctly detected as ON; FP (false positive) signifies instances where an appliance is wrongly detected as ON; and FN (false negative) indicates instances where an appliance is incorrectly identified as OFF.

The EA metric measures the proximity between actual energy consumption and the predicted consumption in each time step. Meanwhile, the F-measure evaluates the precision of algorithms in predicting the ON or OFF state of each appliance at each time step. The optimal value for all these metrics is 1. Additionally, we consider the Overall Estimation Accuracy (OEA) for all N monitored appliances, defined as:

$$OEA = 1 - \frac{\sum_{i=1}^{N} \sum_{t=1}^{T} |p_i(t) - \hat{p}_i(t)|}{2 \sum_{i=1}^{N} \sum_{t=1}^{T} p_i(t)} \tag{2.5}$$

Furthermore, we include the Overall F-measure (OFM) by aggregating values over all appliances for TP, FP, and FN:

$$\mathrm{OFM} = \frac{2 \cdot \mathrm{P}_{\mathrm{total}} \cdot \mathrm{R}_{\mathrm{total}}}{\mathrm{P}_{\mathrm{total}} + \mathrm{R}_{\mathrm{total}}} \tag{2.6}$$

Where:

$$\mathrm{P}_{\mathrm{total}} = \frac{\sum_{i=1}^{N} \mathrm{TP}_i}{\sum_{i=1}^{N} (\mathrm{TP}_i + \mathrm{FP}_i)}, \quad \mathrm{R}_{\mathrm{total}} = \frac{\sum_{i=1}^{N} \mathrm{TP}_i}{\sum_{i=1}^{N} (\mathrm{TP}_i + \mathrm{FN}_i)} \tag{2.7}$$

In addition to accuracy metrics, it is important to quantify the noise present in the aggregate signal. Here, noise refers to the proportion of aggregate energy that cannot be attributed to the ground truth consumption of the appliances selected for disaggregation. Noise arises from unmonitored appliances, random power fluctuations, etc. To capture this, we report the percent-noisy measure [63]:

$$\% - NM = \frac{\sum_{t=1}^{T} \left| P(t) - \sum_{i=1}^{N} p_i(t) \right|}{\sum_{t=1}^{T} P(t)} \tag{2.8}$$

A value of 0% noise indicates a fully explained aggregate signal, which is rarely achievable in practical settings. Using such artificially clean data can lead to inflated performance statistics that do not reflect real-world disaggregation challenges. Reporting this measure ensures that algorithm evaluation accounts for dataset imperfections and allows meaningful comparison across different studies.

2.6 Conclusion of the Chapter

This chapter establishes the fundamentals of NILM, introducing its formal problem formulation, analyzing representative approaches, defining practical design constraints to guarantee widespread deployment, presenting publicly available datasets for benchmarking, and outlining the key performance metrics used to evaluate the accuracy of disaggregation methods.

We outlined representative NILM approaches, including training-based models such as probabilistic methods and deep neural networks, as well as training-less techniques based on optimization and event detection. Each category offers different trade-offs in terms of accuracy, interpretability, and deployment complexity. These foundations serve as a reference for developing methods that guarantee widespread deployment.

We defined practical design constraints to ensure that NILM solutions are compatible with existing infrastructure. These constraints include limiting the input to active power, using low-frequency sampling rates, and enabling execution on devices with restricted processing and memory resources. This configuration supports cost-effective deployment and minimizes the need for manual setup or supervision.

REDD and UK-DALE were described as standard benchmarking datasets due to their appliance-level detail, sampling compatibility, and diversity of household environments. They continue to provide a basis for performance comparisons under consistent and reproducible conditions.

To evaluate disaggregation quality, the chapter introduced Estimation Accuracy, F-measure, and their aggregate forms. The percent-noisy measure was also included to represent unexplained components in the aggregate signal, allowing a more realistic assessment of algorithm performance.

3

Transfer Capabilities of Seq2Seq and Seq2Point CNN architectures in Non-Intrusive Load Monitoring with Unseen Appliances

3.1 Introduction

Several methods have been proposed to address the NILM problem. HMM-based approaches and extensions have been a common choice among researchers [35, 36]. However, the requirement of abundant submetered data from the test domain or long periods of unsupervised training limits the practical application of these HMM techniques. In contrast, training-less solutions have been proposed to overcome the lack of abundant data in the test domain. However, these approaches often require parameter tuning in the test domain and struggle with accurately modeling complex load patterns. The reader is referred to Chapter 2, Section 2.2 for examples of training-less methods.

Lastly, several deep learning approaches have been proposed, gaining attention for their higher disaggregation accuracy compared to traditional methods. The reader is referred again to Chapter 2, Section 2.2 for examples of deep learning methods. CNN-based architectures, in particular, have demonstrated state-of-the-art performance due to the meaningful latent feature extraction for appliances [44].

Since the introduction of Seq2Seq and Seq2Point CNN-based architectures [20], researchers have highlighted their potential for advancing NILM [19, 45].

However, the main drawback of these deep-learning-based methods is their generalization across different domains [18], which is essential for real-world implementation. In practical applications, the assumption of independent and identically distributed data between the training dataset from the source domain and the test dataset is often unrealistic. Differences in grid conditions, appliance brands, consumer electricity usage patterns, and the addition of new appliances create significant variability. These differences result in poor performance across different domains [64].

Therefore, analyzing performance drops due to domain shifts is essential for developing future approaches with improved transfer capabilities. Here, transfer capabilities refer to an algorithm's ability to excel in a domain distinct from its training domain. Given its practical significance, this chapter considers zero-shot transfer learning, where the model is expected to generalize on the test domain without additional training. Although some works evaluate the transferability of some deep learning architectures [44,65], analyzing performance degradation under an increasing noise domain is essential. This analysis is expected to provide insights for developing future methods.

This chapter aims to analyze the performance drop of Seq2Seq and Seq2Point CNN-based architectures when additional appliances, not present in the training mains readings, are introduced. A theoretical analysis based on a first-order Taylor expansion examines the variables contributing to the additional disaggregation error. Furthermore, experimental tests are designed to simulate an increasingly noisy domain by incorporating unknown appliances, using publicly available real-world datasets. Finally, proposed directions toward practical deep-learning-based NILM are stated. The content of this chapter has been previously divulged in the **Journal Publication J3.** [**27**].

The rest of the chapter is organized as follows. Section 3.2 introduces the architecture of the algorithms. Section 3.3 exposes a mathematical analysis of the performance drop in the presence of unknown appliances. Section 3.4 details the testing methodology, performance metrics, used datasets, selected appliances, and algorithm settings. Section 3.5 presents and discusses the experimental results.

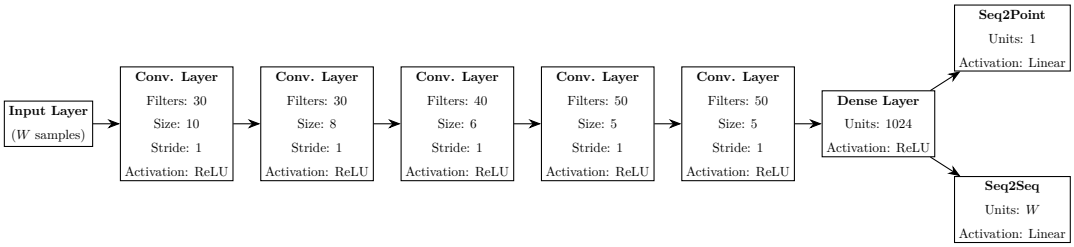


Figure 3.1: Architectures for sequence-to-point and sequence-to-sequence neural networks.

Section 3.6 states the proposed directions toward practical NILM and section 3.7 concludes the chapter.

3.2 Seq2Seq and Seq2Point models

In the NILM context, Seq2Seq and Seq2Point deep learning architectures receive as input continuous and equal-length windows of mains aggregate readings $[P(t), P(t+1), ..., P(t+W-1)]$, where W is the length of the window. In the case of Seq2Seq, the output is the predicted power at the same time steps $[\hat{p}_i(t), \hat{p}_i(t+1), ..., \hat{p}_i(t+W-1)]$ for each target appliance i. Since there are several predictions for p_i, one for each sliding window that contains time step t, the average of all the predicted values is reported as the final result. In contrast, the output of the Seq2Point algorithm is only the prediction for the midpoint element of each window $p_i(t+\lfloor W/2 \rfloor)$. This method assumes that the midpoint of the target appliance has a strong correlation with the aggregated information before and after that time step. The advantage of Seq2Point is that there is a single prediction for every t, and thus, no average operation is required. The architectures for both algorithms are shown in Fig. 3.1 as originally proposed in [20]. Typically, a separate model is trained for each monitored appliance in the training domain, expecting that it will generalize properly in the test domain.

3.3 Mathematical Analysis in the Presence of Unknown Appliances

We formally defined the NILM problem through Eq. (2.1), which states that the main meter reading aggregates the consumption of N monitored appliances

$\{p_1, \ldots, p_N\}$ and a residual term $\eta(t)$ that may include noise and not-monitored loads. As mentioned before, one neural network is trained *per appliance*, aiming to predict $p_i(t)$ from $P(t)$ for each i. This section explores theoretically how introducing additional non-monitored appliances at test time can degrade the performance of the analyzed models.

3.3.1 Single-Appliance Formulation

Let $\mathcal{D}_{\text{train}}$ be the training dataset comprising pairs of input windows and corresponding appliance power values. For each integer time index t where sufficient data exists, an input window

$$\mathbf{P}_{[t,\, t+W-1]} \;=\; \big[P(t),\, P(t+1),\, \ldots,\, P(t+W-1)\big]^{\mathsf{T}} \;\in\; \mathbb{R}^{W}. \tag{3.1}$$

is extracted from the mains reading. We associate it with the ground-truth power consumption of appliance i for either **all** time steps in the window (*Seq2Seq*) or **a single** time step in the window (*Seq2Point*).

Formally, the training set for a single appliance i can thus be expressed as:

$$\mathcal{D}_{\text{train}} \;=\; \begin{cases} \Big\{\big(\mathbf{P}_{[t,\, t+W-1]},\; [\,p_i(t),\ldots,\, p_i(t+W-1)]\big)\Big\}_{t\in\mathcal{T}_{\text{train}}} & \text{(Seq2Seq)} \\[2ex] \Big\{\big(\mathbf{P}_{[t,\, t+W-1]},\; p_i(t+\lfloor W/2\rfloor)\big)\Big\}_{t\in\mathcal{T}_{\text{train}}} & \text{(Seq2Point)}, \end{cases} \tag{3.2}$$

where $\mathcal{T}_{\text{train}}$ is the set of valid time indices in the training data.

Let $u_j(t)$ be the consumption of M *new* non-monitored appliances not present in $\mathcal{D}_{\text{train}}$. We define

$$U(t) \;=\; \sum_{j=1}^{M} u_j(t). \tag{3.3}$$

Then, at test time, the effective mains reading observed becomes:

$$P'(t) \;=\; P(t) \;+\; U(t), \tag{3.4}$$

Analogously to (3.1), the additional unknown load appearing at test time can be used to form a W-dimensional vector over the same window,

$$\mathbf{U}_{[t,\,t+W-1]} \;=\; \big[U(t),\,U(t+1),\,\ldots,\,U(t+W-1)\big]^{\mathsf{T}} \;\in\; \mathbb{R}^{W}. \tag{3.5}$$

Then, the input to the model at test time becomes:

$$\mathbf{P}'_{[t,\,t+W-1]} \;=\; \mathbf{P}_{[t,\,t+W-1]} \;+\; \mathbf{U}_{[t,\,t+W-1]}. \tag{3.6}$$

Analogously to (3.2), the test set $\mathcal{D}_{\text{test}}$ contains input windows from times $\mathcal{T}_{\text{test}}$ and their corresponding ground-truth appliance power values:

$$\mathcal{D}_{\text{test}} \;=\; \begin{cases} \Big\{\big(\mathbf{P}'_{[t,\,t+W-1]},\; [\,p_i(t),\ldots,\,p_i(t+W-1)]\big)\Big\}_{t\in\mathcal{T}_{\text{test}}} & \text{(Seq2Seq)} \\ \Big\{\big(\mathbf{P}'_{[t,\,t+W-1]},\; p_i(t+\lfloor W/2\rfloor)\big)\Big\}_{t\in\mathcal{T}_{\text{test}}} & \text{(Seq2Point),} \end{cases} \tag{3.7}$$

For clarity of exposition, we focus on the *Seq2Point* setting in the subsequent analysis. Concretely, each sample $(\mathbf{P}, p_i)$ in the training set $\mathcal{D}_{\text{train}}$ refers to an input window of length W and a single target value $p_i(t+\lfloor W/2\rfloor)$.

The deep learning model $f_{\boldsymbol{\theta}_i}$, in the training phase, is optimized to minimize:

$$\boldsymbol{\theta}_i^{*} \;=\; \arg\min_{\boldsymbol{\theta}_i} \sum_{(\mathbf{P},\,p_i)\,\in\,\mathcal{D}_{\text{train}}} \Big(f_{\boldsymbol{\theta}_i}(\mathbf{P}) \;-\; p_i\Big)^2, \tag{3.8}$$

where $f_{\boldsymbol{\theta}_i}(\mathbf{P})$ returns the predicted midpoint power estimate of appliance i given the W-length aggregate input $\mathbf{P}$.

At test time, the trained model $f_{\boldsymbol{\theta}_i^{*}}$ sees $\mathbf{P}'$ instead of $\mathbf{P}$, and thus, may produce a different prediction $\hat{p}'_i$:

$$\hat{p}'_i \;=\; f_{\boldsymbol{\theta}_i^{*}}\Big(\mathbf{P}'\Big). \tag{3.9}$$

Since $f_{\boldsymbol{\theta}_i^{*}}$ was trained without exposure to $\mathbf{U}$, the presence of such previously unseen loads alters the distribution of inputs and can degrade the predictive accuracy for p_i. This may highlight the need for domain adaptation strategies, more robust modeling, or extended training sets that simulate the influence of unknown appliances.

3.3.2 Sensitivity Analysis via First-Order Taylor Expansion

Let $f_{\boldsymbol{\theta}_i} : \mathbb{R}^W \to \mathbb{R}$ denote the single-appliance model, which maps an input window $\mathbf{P} \in \mathbb{R}^W$ to a predicted appliance power consumption. If an additional unknown load introduces a perturbation $\mathbf{U} \in \mathbb{R}^W$ to the input, the perturbed input becomes $\mathbf{P} + \mathbf{U}$. Assuming that $\mathbf{U}$ is sufficiently small, we can approximate the new output via a first-order Taylor expansion around $\mathbf{P}$:

$$f_{\boldsymbol{\theta}_i}\big(\mathbf{P} + \mathbf{U}\big) \;\approx\; f_{\boldsymbol{\theta}_i}\big(\mathbf{P}\big) \;+\; \nabla f_{\boldsymbol{\theta}_i}\big(\mathbf{P}\big)^{\mathsf{T}}\,\mathbf{U}, \tag{3.10}$$

where $\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P}) \in \mathbb{R}^W$ is the gradient evaluated at $\mathbf{P}$.

Assuming that in the training domain, the model approximates the true appliance power, i.e.,

$$f_{\boldsymbol{\theta}_i}\big(\mathbf{P}\big) \approx p_i,$$

the prediction under the perturbed input is given by

$$\hat{p}_i' = f_{\boldsymbol{\theta}_i}\big(\mathbf{P} + \mathbf{U}\big) \;\approx\; p_i \;+\; \nabla f_{\boldsymbol{\theta}_i}\big(\mathbf{P}\big)^{\mathsf{T}}\,\mathbf{U}.$$

Thus, we can define the additional error introduced by the unknown load as:

$$\Delta p_i \;=\; \hat{p}_i' - p_i \approx \nabla f_{\boldsymbol{\theta}_i}\big(\mathbf{P}\big)^{\mathsf{T}}\,\mathbf{U}, \tag{3.11}$$

so that the corresponding additional squared error is:

$$\epsilon^2 \;=\; \Big(\Delta p_i\Big)^2 \approx \Big(\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})^{\mathsf{T}}\,\mathbf{U}\Big)^2. \tag{3.12}$$

Taking the expectation over the joint distribution of $\mathbf{P}$ and $\mathbf{U}$ at test time, the expected additional squared error scales as

$$\mathbb{E}_{\mathbf{P},\,\mathbf{U}}\Big[\Big(\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})^{\mathsf{T}}\,\mathbf{U}\Big)^2\Big].$$

This expression shows that the error increase depends both on the magnitude and variability of the unknown load $\mathbf{U}$ and on the model's local sensitivity (as measured by $\|\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})\|$).

3.3.3 Influence of Appliance Characteristics

As mentioned before, each appliance i is modeled with its own Seq2Point network $f_{\boldsymbol{\theta}_i}$, and its gradient $\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})$ captures the unique temporal signature and

power contribution of the appliance within a window of length W. Based on the previous sensitivity analysis, this characteristic behavior is expected to influence how the model reacts to unknown loads $\mathbf{U}$:

- **High-Power Appliances:** For appliances that dominate the aggregate reading, the model is expected to quickly recognize their presence, resulting in a relatively flat mapping. This is reflected in a lower gradient $\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})$, so that even if additional loads $\mathbf{U}$ are present, the term $\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})^{\mathsf{T}}\mathbf{U}$ remains moderate unless $\mathbf{U}$ is very large.

- **Multi-State Appliances:** Appliances with multiple operating states generate subtle, transient signatures. Capturing these nuances requires higher model sensitivity, resulting in a steeper gradient. As a result, even moderate perturbations $\mathbf{U}$ may cause significant errors.

- **Lower-Power Appliances:** For low-power appliances, the target signal is weak and more prone to interference from background noise or additional loads. Detecting these small signals requires higher model sensitivity, resulting in a larger gradient. Consequently, even small or moderate unknown loads $\mathbf{U}$ can impact the prediction, increasing the squared error.

This analysis shows that unknown loads can shift the input $\mathbf{P}$ into regions outside the training distribution, with the resulting error determined by both the magnitude of $\mathbf{U}$ and the local sensitivity $\nabla f_{\boldsymbol{\theta}_i}(\mathbf{P})$. Reducing the local sensitivity of these networks (i.e., lowering $\|\nabla f_{\boldsymbol{\theta}_i}(\cdot)\|$) or augmenting training data to simulate potential unknown load scenarios may help mitigate this domain-shift effect. As real-world households frequently change their set of appliances or usage patterns, developing NILM models with stronger *domain adaptation* or *transfer* capabilities is essential. In the following, we evaluate the performance of Seq2Seq and Seq2Point using real-world, publicly available datasets. This analysis provides further insights into how unknown additional loads affect the performance of these methods for common appliances.

3.4 Experiments detail

To test the performance of Seq2Seq and Seq2Point CNN-based architectures in the presence of additional appliances not included in the mains reading that was used for training the algorithms, we created artificial aggregates as follows. First, aggregating only monitored appliances $P_{A,0}(t) = \sum_{i=1}^{N} p_i(t)$ we created the artificial mains that together with the ground truth $p_i(t)$ for each monitored appliance were used to train all the algorithms.

Once trained, we created several test scenarios by adding other appliances of the same dataset, once at a time, selected in decreasing order of their energy consumption ($P_{A,r}(t) = P_{A,0}(t) + \sum_{j=1}^{r} p_j(t), 1 \leq r \leq K$), where K is the number of appliances from the same dataset not considered in the monitored appliances. In this way, we created $K + 1$ different scenarios with an increasing level of noise (Including the zero-noise scenario $P_{A,0}$). Then we assessed the performance of each method for each test scenario with the input $P_{A,r}(t);\ 0 \leq r \leq K$ that represents the mains reading. For each scenario, the percent-noisy measure, defined in Eq. 2.8, is used to estimate the induced level of noise in the artificial aggregate.

3.4.1 Performance Metrics

In assessing the accuracy of the considered methods for individual appliances, we employ the Estimation Accuracy and F-measure. These metrics were defined by equations (2.2) and (2.3), respectively. Additionally, we consider the Overall Estimation Accuracy, defined in Equation (2.5), and the Overall F-measure defined in Equation (2.6).

3.4.2 Datasets and selected appliances

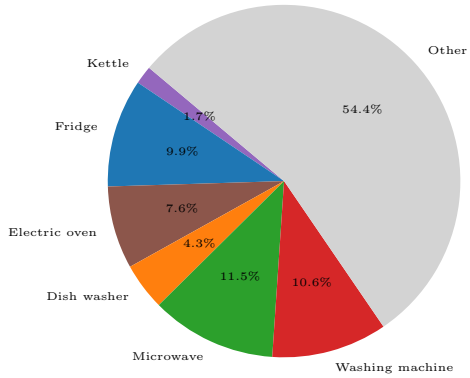
We tested the algorithms on REDD [59] and UK-DALE [60] datasets. Both datasets provide a large number of submetered appliances, which allows us to create complex scenarios with increasing levels of noise. We use house 1 from REDD (REDD-1) and houses 2 (UKDALE-2) and 5 (UKDALE-5) from UK-DALE. We selected commonly used appliances from each dataset, trying to choose the same across the different datasets when possible. From UKDALE-2 we selected the Fridge (F), Dishwasher (DW), Microwave (MW), Rice cooker (RC), Kettle (K),

and Washing machine (WM). From UKDALE-5 we selected Fridge (F), Dishwasher (DW), Microwave (MW), Electric Oven (EO), Kettle (K), and Washing machine (WM). From REDD-1 we use the Fridge (F), Dishwasher (DW), microwave (MW), Electric Oven (EO), and Washer Dryer (WD).

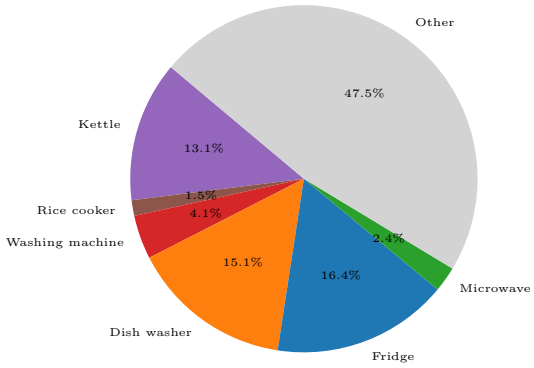
To facilitate the interpretation of the results, the pie charts in Figures 3.2a, 3.2b, and 3.2c illustrate the contribution of each household appliance to the total energy consumption recorded in the UKDALE-5, UKDALE-2, and REDD-1 datasets, respectively. Although a substantial portion is categorized as 'Other', corresponding to non-monitored appliances, the monitored appliances account for a significant share of the total energy consumption: approximately 45.6% in UKDALE-5, 52.5% in UKDALE-2, and 52.1% in REDD-1. Figures 3.3a, 3.3b, and 3.3c present the activation timelines of all submetered appliances in each dataset. A considerable number of devices are individually monitored, accurately reflecting the complexity and diversity of real-world households. The 'Other' category (Figures 3.2a, 3.2b, and 3.2c) mainly arises from the aggregation of several low-consumption appliances or from always-on appliances, such as the server computer in Figure 3.3a, which significantly contribute to the base load even though they do not produce visible events in the mains signal.

3.4.3 Settings

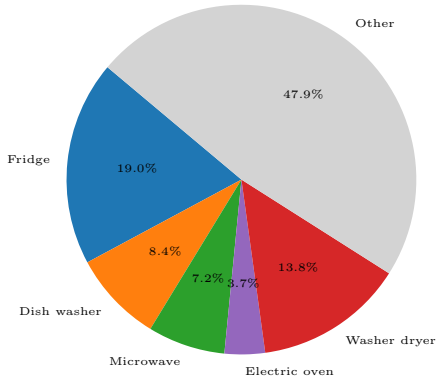
For the UKDALE dataset, the sample period was set to 6 seconds, and for the REDD dataset 3 seconds. These are the original sampling rates for the appliance's specific meters. Training and testing periods were selected based on measurement continuity and consistent appliance-level activity. We used only uninterrupted segments where most submetered appliances were active and fully metered. For UKDALE-5 and UKDALE-2, which provide approximately two months of continuous data that meet these criteria, we selected a five-week training period followed by a fixed 15-day testing window, with a short gap between them. For the shorter REDD-1 dataset, which is split into two uninterrupted segments, we used the first segment for training and the second for testing. Specifically, we trained the algorithms for UKDALE-5, UKDALE-2, and REDD-1 datasets respectively between 20/06/2014 and 27/07/2014, 20/06/2013 and 27/07/2013, and 18/04/2011 and 03/05/2011. Testing periods ranged from 10/08/2014 to 25/08/2014 for UKDALE-



(a) UKDALE-5

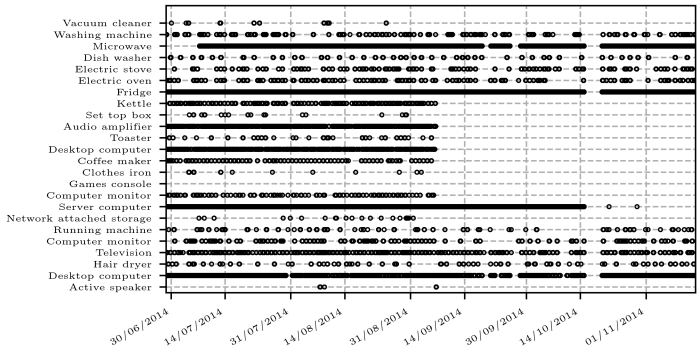


(b) UKDALE-2

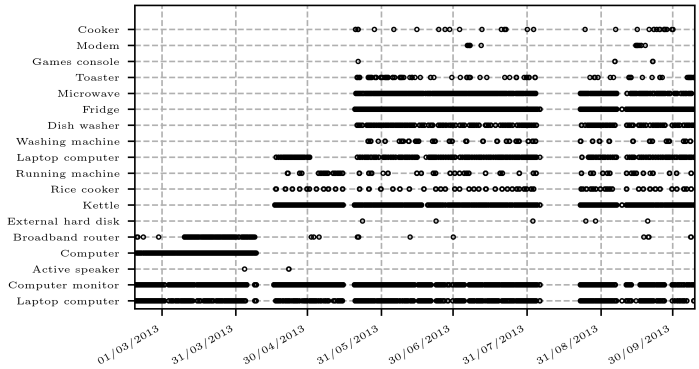


(c) REDD-1

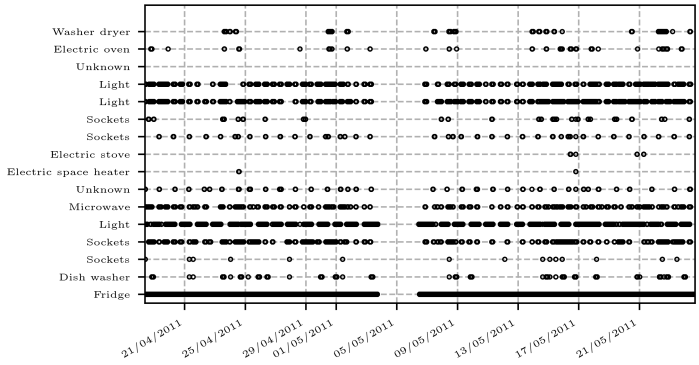
Figure 3.2: Breakdown of household appliance contributions to total energy levels for UKDALE-5, UKDALE-2, and REDD-1 datasets. 'Other' accounts for non-monitored appliances.



(a) UKDALE-5



(b) UKDALE-2



(c) REDD-1

Figure 3.3: Timeline of activation for all submetered appliances in the three datasets.

5; 28/07/2013 to 12/08/2013 for UKDALE-2; and 10/05/2011 to 25/05/2011 for REDD-1.

Based on previous literature [19, 44], both algorithms were trained for a maximum of 50 epochs using a batch size of 1024. A random 15% of the training data was set aside as a validation set, and the model achieving the lowest validation loss was selected as the final model. Early stopping with a patience of 5 epochs was employed to improve training efficiency and prevent overfitting, as in [44]. The window length W was set to 99. The choice of the window length is important as pointed out in [66], we decided to use 99 because it is the default value in the NILMTK-CONTRIB implementation [19], and this value is also tested in [66] with good results, in similar data frequencies. The implementation of the methods is based on the NILMTK toolkit [19], where the ADAM optimizer algorithm [67] is used for training.

3.5 Experimental Results

The Overall Estimation Accuracy and Overall F-measure are reported for each method in Figures 3.4a and 3.4b for UKDALE-5 dataset, in Figures 3.5a and 3.5b for UKDALE-2 dataset, and in Figures 3.6a and 3.6b for REDD-1 dataset. From the results, it can be concluded that in general, the performance of the methods tends to decrease when the noise increases. For UKDALE-5 and REDD-1 datasets, the drop in performance is significant, starting with almost a perfect prediction in a zero noise domain, the accuracy goes down progressively as the noise increases, ending with a mediocre prediction when the noise is greater than 55% for UKDALE-5 or 45% for REDD-1. For the UKDALE-2 dataset, the drop in performance seems to be not significant, this is expected by the fact this dataset is the least noisy from the considered, and the not-monitored appliances present in this dataset either are rarely used or consume a low amount of power.

To analyze in detail the drop in performance, the Estimation Accuracy and F-measure are reported for each monitored appliance and each algorithm in Figures 3.7, 3.8 and 3.9, corresponding to UKDALE-5, UKDALE-2 and REDD-1 datasets respectively. Appliances with low-power states or complex patterns, such as the Fridge (F), Dishwasher (DW), and Washing Machine (WM), exhibit significant

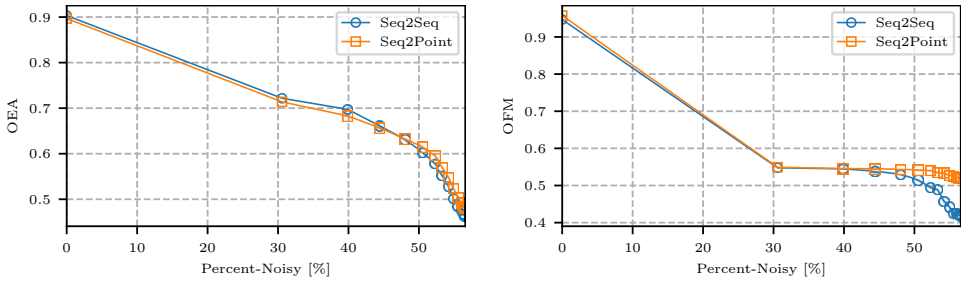


Figure 3.4: Robustness to noise for the UKDALE-5 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases.

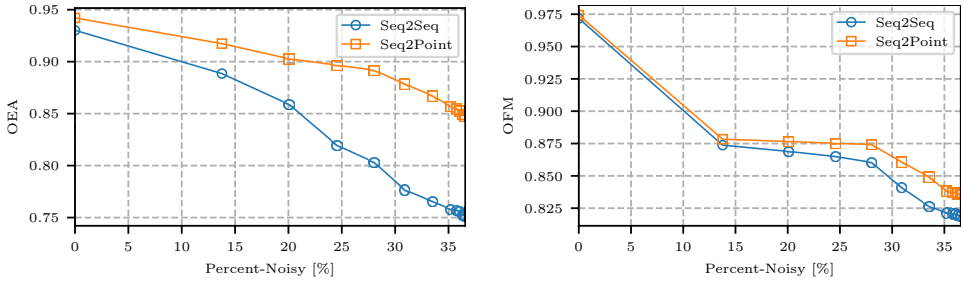


Figure 3.5: Robustness to noise for the UKDALE-2 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases.

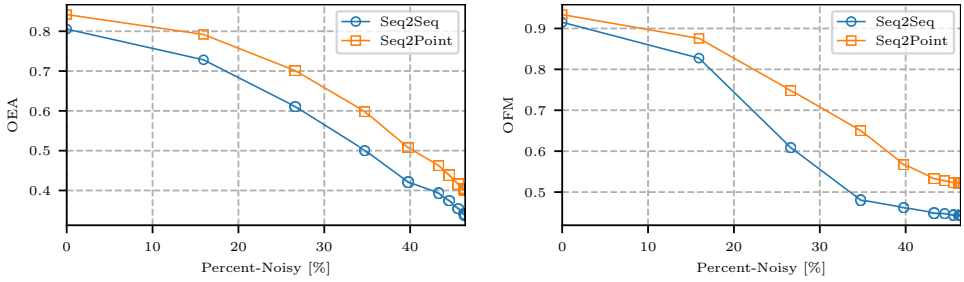


Figure 3.6: Robustness to noise for the REDD-1 dataset: (a) OEA as percent-noisy increases; (b) OFM as percent-noisy increases.

performance deterioration, making the results unreliable. This is more evident in the UKDALE-5 dataset Fig. 3.7, where there exist some appliances (not in the monitored set) with low but consistent power consumption, that seem to affect in particular, the performance of monitored appliances with low-power states or complex patterns. Appliances with high power consumption and with predictable

patterns like the Kettle (K) are less affected by increasing noise. This suggests that both methods are highly sensitive to domain shifts, making it difficult to transfer them across domains with varying consumption patterns without a domain adaptation phase. This behavior also aligns with the theoretical results presented in Section 3.3.

Although the performance drop in UKDALE-2 (Figure 3.8) appears small, this is due to UKDALE-2 being the least noisy dataset among those considered. However, the most affected appliance types are the same as in UKDALE-5: those with low-power states or complex usage patterns (Fridge (F), Dishwasher (DW), Rice cooker (RC)). Regarding the REDD-1 dataset (Figure 3.9), the most affected appliances are the Fridge (F), Dishwasher (DW), and Microwave (MW), while the most robust to the noise are the Electric Oven (EO) and Washer Dryer (WD). This dataset originates from the USA, whereas UKDALE comes from the UK, leading to differences in appliance signatures between the two. However, the overall behavior is similar: appliances with low-power states and complex patterns are the most affected, while those with high power consumption and predictable patterns are more resilient. Although training in multiple domains can partially mitigate this drawback, the main limitation is the difficulty of obtaining extensive training data that captures most user patterns. Efforts have to be made to realize an effective training strategy in such a way that the accuracy of the algorithm depends as little as possible on consumption patterns. In that sense, methods focusing on appliance signatures should be preferred over those relying heavily on aggregated power patterns, as they are expected to improve transferability.

3.6 Proposed Directions Towards Practical NILM

Although multi-appliance models, domain adaptation, and data augmentation represent interesting solutions toward robust and practical NILM, they still face practical limitations. Multi-appliance methods, for instance, demand higher computational resources and continuous re-training to accommodate new devices; domain adaptation techniques typically rely on unlabeled data from the new environment, which might raise privacy concerns or be unavailable; and synthetic data augmentation can lead to unrealistic load patterns.

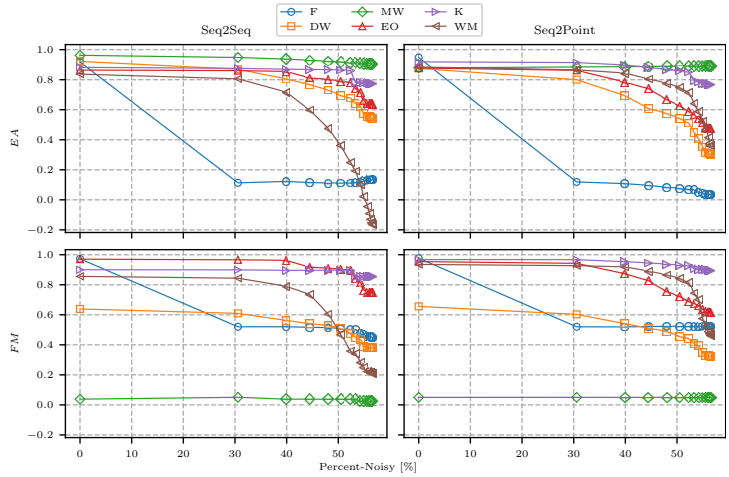


Figure 3.7: Appliance metrics as percent-noisy increases for UKDALE-5 dataset.

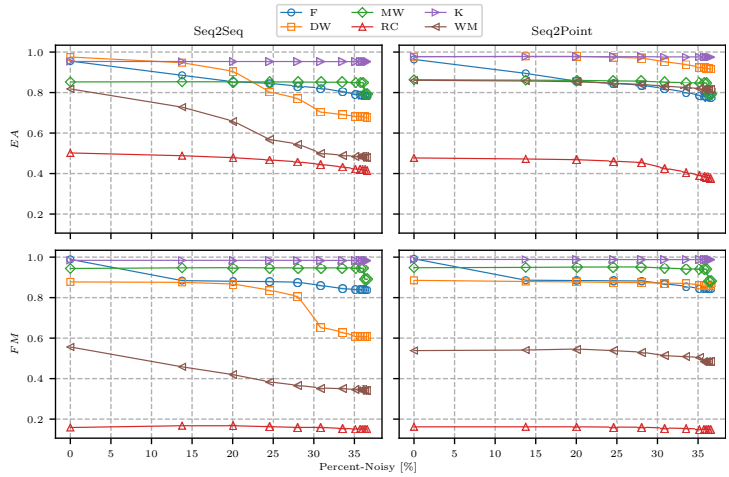


Figure 3.8: Appliance metrics as percent-noisy increases for UKDALE-2 dataset.

Recent work has explored alternative approaches to learning robust, transferable representations for appliances. For instance, in [68], a contrastive learning method combined with transfer learning is proposed to extract appliance-specific features from power sequences. Similarly, in [69], a self-supervised contrastive model based on temporal convolutional networks is developed to learn generalizable load signatures from unlabeled data. These approaches show potential in reducing the need for labeled data and in capturing useful representations across domains. However, they are generally developed for a fixed set of appliance classes and may require additional modules or retraining when used in new environments.

A promising alternative is to learn robust, appliance-specific signatures that

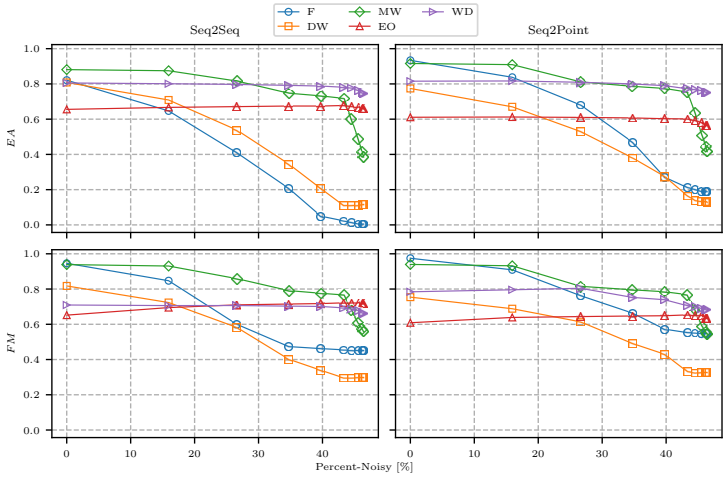


Figure 3.9: Appliance metrics as percent-noisy increases for REDD-1 dataset.

remain identifiable even when additional unknown loads appear. Rather than regressing directly from the aggregate signal to a power estimate, such methods focus on extracting an invariant representation of the target appliance that captures its unique power cycle or spectral fingerprint. In practice, an unseen signal is matched against this signature via a suitable distance metric, allowing the algorithm to flag the presence of that appliance even when novel loads perturb the overall mains reading. It offers a more naturally portable strategy: once a robust signature is learned, it can be deployed in new environments without exhaustive retraining. Exploring these signature-based approaches represents a valuable future research direction to bridge the gap between current NILM solutions and the constantly changing appliance sets.

This perspective also suggests a more fundamental limitation in current deep learning NILM architectures: both single-appliance and fixed multi-appliance models are based on static mappings from aggregated signals to appliance-level outputs, which limits their capacity to generalize across domains and adapt to unseen devices. A more flexible alternative is to structure the model around a shared feature extractor that learns general-purpose, appliance-agnostic representations. These representations can be trained using contrastive learning to encode invariant patterns of appliance activity across different aggregate contexts.

To improve further adaptability, lightweight appliance-specific components could be attached to the shared backbone and fine-tuned with limited supervision.

Meta-learning or few-shot learning strategies can enable these modules to generalize from only a few labeled examples, making it feasible to extend the system to new appliance types without retraining the entire model. This shift from direct regression to modular, representation-based learning supports better scalability and generalization, and aligns more naturally with real-world NILM scenarios where appliance sets and usage conditions vary across households.

3.7 Conclusion of the Chapter

This chapter critically analyzes and evaluates the performance of Seq2Seq and Seq2Point NILM CNN-based architectures in the presence of additional appliances not included in the mains reading used for training. Initially, a theoretical analysis based on a first-order Taylor expansion suggests that the additional error incurred depends on the magnitude of the unknown loads and the local sensitivity, which is highly dependent on the disaggregated appliance. In the second step, the methods are tested with real-world datasets by creating artificial mains, first aggregating only monitored appliances to train the algorithms, and then testing them in several noisy scenarios by adding other appliances of the same dataset to the mains reading. The results showed a decrease in the performance of the methods when the noise increases, especially for scenarios with considerable levels of noise and for appliances with low-power states or complex patterns. The results show agreement between theoretical and experimental findings and highlight a strong dependence on the aggregated power pattern observed during training. To enhance transferability, particularly in zero-shot learning, further research should focus on methods that rely primarily on monitored appliance signatures rather than aggregated power patterns.

4

Online Real-Time Robust Framework for Non-Intrusive Load Monitoring in Constrained Edge Devices

4.1 Introduction

Real-time, appliance-level feedback on energy consumption improves energy efficiency by allowing users to monitor individual devices and make immediate adjustments. This feedback helps users identify inefficient devices and optimize usage strategies, which can lead to significant energy savings [46]. Appliance-specific data not only enables more effective load scheduling but also increases user engagement, as users find this level of feedback more relevant and actionable compared to household-level data [70]. Additionally, this feedback enhances grid operations by providing granular data that improves load forecasting and demand-side management [71].

As discussed in Chapter 2, Section 2.2, despite its recent popularity, deep learning techniques have limitations related to generalization, interpretability, and deployment complexity. Even when trained within the same environment used for testing, these models fail to maintain robustness when the set of appliances in the mains reading changes, as demonstrated in Chapter 3. This behavior contradicts Requirement 5 in Table 2.1, which specifies that a practical and widely deployable NILM algorithm must remain stable under variations in appliance composition.

In this context, a training-less solution holds practical significance for deploying NILM algorithms on edge devices.

4.1.1 Related research

Several approaches have been proposed for real-time NILM, with some focusing on high-frequency sampling [71–74], while others consider low-frequency sampling [35, 40, 75–78]. High-frequency sampling methods typically require an offline training phase, needing transient responses for each target appliance [71, 72]. Similarly, many low-frequency approaches necessitate extensive training data and a pre-deployment training phase [35, 40, 75, 76]. In contrast, some low-frequency, training-less methods assume prior knowledge of the average active power associated with each appliance state [77, 78].

Training-Less NILM approaches are commonly divided into two main groups: state-based algorithms and event-based algorithms [18]. State-based methods represent each appliance operation using a finite state machine (FSM) with distinct state transitions. Inside this group, optimization-based algorithms, search for the global optimal state combination of appliances, either considering a time window or a unique time step.

Window-based optimization methods are often more accurate at the cost of increasing the computational complexity. In this category, integer programming (IP) solvers are a popular choice, and studies based on Aided Linear Integer Programming (AILP) [49,79], Mixed Integer Linear Programming (MILP) [80] and Mixed-Integer Nonlinear Programming [50,81] have been proposed in the literature. These approaches formulate NILM as a constrained optimization problem where the temporal linkage is preserved by imposing state transition constraints [50, 81] or by a correction phase of the output of the IP solver [79]. The main practical limitation of these methods is their high computational complexity, especially if a resolution in the order of a few seconds is required, making its deployment difficult in an embedded system with low capabilities. Regarding unknown appliance's presence, a couple of papers [50, 79] in this research group attempt to consider this issue by minimizing the number of active appliances in each time step. Although useful, this heuristic does not always hold [79], and a complement with an event detection phase could improve its effectiveness in the presence of unknown loads.

Those optimization methods considering a unique time step are often better suited for real-time applications because of the lower-size optimization problem. In [77], the authors propose a modified cross-entropy method to solve a penalty-based optimization problem that considers dependency between measurements. In [78] a multiobjective formulation is proposed and solved with the NSGA-II algorithm that penalizes changes between consecutive events. Although these algorithms are simple, online, and can be used in real-time, not relying on an event detection phase makes them susceptible to unknown appliances. Moreover, the optimization problem at hand is dynamic by nature. However, existing algorithms typically restart the search from scratch for each timestep or time window, which may not be optimal for real-time predictions. To the best of the authors' knowledge, no optimization-based algorithm has yet proposed a method to effectively address the intrinsic dynamics of the problem by leveraging previous solutions. This approach would allow the search to begin from a partially converged population, potentially enhancing computational efficiency.

On the other hand, event-based methods work by detecting edges on the aggregated power consumption and assigning each extracted feature to one appliance state transition. In this research group, several methods have been proposed, including those based on Graph Signal Processing (GSP) [51, 52], subtractive clustering [53], non-negative tensor factorization [56], and the balanced window technique (BW) [54]. Unlike optimization-based methods, event-based approaches are less susceptible to the presence of unknown appliances. However, these methods are vulnerable to measurement noise and outliers. Typically, these approaches disaggregate appliances one at a time and often fail to verify whether the sum of the disaggregated loads approaches the actual measured total [53, 55]. This oversight can lead to significant load over/underestimation [56]. To overcome this, paper [55] proposes a general post-processing optimization phase based on a regularization term. However, the regularization term, used as a weight for the confidence in the original output, relies heuristically on the mean power consumption of each appliance [55]. This heuristically-based regularization term has the drawback that for high consumption loads that are easy to disaggregate for an event-based method (ex. Kettle), the algorithm will put more weight in the optimization-based term, which can result in a wrong final prediction. In this sense, event-based algorithms

that provide insights into the confidence of their output, are of interest, especially for real-time applications.

4.1.2 Our contribution

This chapter introduces the O2RE framework: a novel Online, Real-time, Robust, and Edge-driven NILM framework. The framework's main focus is the practical deployment of a real-time NILM algorithm at frequencies around 1Hz in a constrained edge device. In this sense, we also consider the possible contribution of unknown loads given in the form of FSM or constantly-on appliances. The content of this chapter has been previously divulged in the **Journal Publication J1.** [**28**].

The proposed framework evolves a dynamic structure S_P, where $S_{P_{i,j}}$ denotes the probability that appliance i is in state j. S_P is sequentially updated upon detecting a significant edge (event-based) by a novel dynamic programming algorithm. This way, S_P efficiently preserves historical information and temporal linkages in a compact storage format. Unlike existing algorithms [77, 78], which assume the previous sample state is precisely known, our method does not rely on this assumption, providing more accurate handling of temporal dependencies.

In addition, this architecture benefits from S_P functioning as a discrete probability distribution, which, when sampled, is likely to yield high-quality solutions with high probability. This characteristic enhances NILM state-based optimization algorithms by initiating the search process from a partially converged population, instead of restarting the search for each time step. Furthermore, as S_P is the result of evolving an event-based NILM method over time, by initiating the search in S_P, the drawback of state-based algorithms associated with the possible presence of unknown appliances is expected to be eliminated.

To determine the operating state of each appliance, S_P is integrated into a state-based optimization algorithm, specifically the Population-Based Incremental Learning (PBIL) algorithm. This algorithm improves the efficiency of the optimization process compared to traditional metaheuristics because it evolves a single probability vector. The probability vector is updated depending on the performance of the sampled solutions, thus, eliminating the computational burden associated with managing multiple individual solutions usually found in other metaheuristics like

Genetic Algorithms.

To summarize, the specific contributions of this work are detailed in the following:

1. A training-less probabilistic model for power state transitions in FSM appliances is proposed. The model only depends on the average power of each appliance state, and two setting parameters ($\sigma^2_{\Delta\tilde{P}}$ and δ) common to all the appliances. Parameter $\sigma^2_{\Delta\tilde{P}}$ is related to the variance of state transitions and δ is a robust parameter to account for the presence of unknown appliances

2. Building on the probabilistic model, we have developed a dynamic programming algorithm that updates the state probability of each appliance sequentially. This algorithm is designed to be lightweight, running only when a significant edge is detected, enhancing efficiency in both storage and computation time

3. A lightweight online algorithm is proposed to estimate the total consumption of always-on appliances (base load). By accurately characterizing the base load, algorithms can effectively differentiate it from monitored appliance-specific states, improving the overall performance of NILM systems. Moreover, the user must be aware of the global consumption of these constantly-on devices

4. State probabilities and base load are integrated into a Population-Based Incremental Learning algorithm to determine the most likely state of each appliance at each time step. The idea is to profit from the advantages of combinatorial optimization-based approaches while accounting for the output of event-based algorithms (in the form of state probabilities). Moreover, this architecture enhances optimization algorithms by improving performance when initiating searches from a partially converged population

5. Using publicly available datasets, the framework's accuracy and robustness are demonstrated by comparison with state-of-the-art training-less algorithms

To the best of our knowledge, no solution exists in NILM literature integrating into a unique framework the advantages of event-based and optimization-based

approaches in a storage, time-efficient, and robust solution capable of handling real-time frequencies of the order of 1Hz in constrained edge devices. The rest of the chapter is organized as follows. In section 4.2 important definitions are established. Section 4.3 presents the high-level architecture of the O2RE framework following a modular structure. Section 4.4 outlines the theory, methods, and implementation details for each framework module. Section 4.5 shows the performance evaluation and comparison with other state-of-the-art algorithms, and section 4.6 concludes the paper.

4.2 Preliminaries

In this chapter, we introduce a slight modification to the original problem definition introduced in Equation 2.1 by explicitly including the base load in (4.1). This adjustment will be useful for the analyses and methods presented in the remainder of the chapter.

$$P(t) = \sum_{i=1}^{N} p_i(t) + B(t) + \eta(t) \tag{4.1}$$

In (4.1), the value of $P(t)$ is composed of N appliance-level signals $p_i(t)$, the base load $B(t)$ (consistent power consumption contributed by unknown appliances operating in a steady state), and a residual term $\eta(t)$. The residual term comprises (measurement) noise and other loads that are not considered in the N monitored appliances. Here, we are going to consider appliances represented with a Finite State Machine (FSM) model (Figure 4.1). In this formulation, each appliance is modeled based on a set of discrete states, and a set of allowed state transitions. Obtaining a complete FSM model (including statistical information) often requires a previous training phase, but the minimum necessary information, that is the average active power of each appliance state, only requires a few full operation cycles of each appliance or information in the appliance user manual when available. It is worth to note that a large number of appliances operate as FSM systems, thus the proposed approach is widely exploitable.

If we define $P_{i,j}$ as the nominal power of appliance i when it is working in the state j ($0 \leq j \leq M_i$) (M_i is the number of "On" states in appliance i), then the estimated power $\hat{p_i}(t)$ of appliance i at any moment takes one of the discrete values

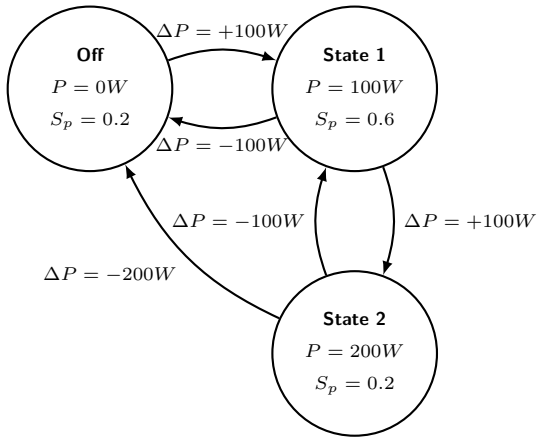


Figure 4.1: Finite State Machine Model. The diagram shows the power consumption for each state (P) and the probability (S_p) of being in that state at a given time t. Transitions between states are labeled with power differences (ΔP).

$\{P_{i0}, P_{i1}, ..., P_{iM_i}\}$. In this way, the problem can be defined as finding the value of the operating state indicator vector S that better represents the real behavior of the appliances. Here $S_{i,j}$ is defined as follows:

$$S_{i,j} = \begin{cases} 1 & \text{if appliance } i \text{ is operating in state } j \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

S.T:

$$\sum_{j=0}^{M_i} S_{i,j} = 1 \quad \text{for each appliance } i \tag{4.3}$$

Constraint (4.3) implies that any appliance always operates exactly in one of the given states. Equation (4.1) can now be rewritten in terms of $S_{i,j}$ as follows:

$$P(t) = \sum_{i=1}^{N} \sum_{j=0}^{M_i} S_{i,j}(t) P_{i,j} + B(t) + e(t) \tag{4.4}$$

Here $e(t)$ is the error term that includes $\eta(t)$ (Equation 4.1) plus an additional error due to the estimation of $p_i(t)$. A direct criterion for estimating $S_{i,j}$ is the minimization of the absolute value of the error term $|e(t)|$:

$$|e(t)| = |P(t) - B(t) - \sum_{i=1}^{N} \sum_{j=0}^{M_i} S_{i,j} P_{i,j}| \tag{4.5}$$

As stated in [17], this criterion for estimating S_{ij} has several difficulties. The first drawback is associated with the fact that if (4.5) is used in the presence of unknown loads, it would attempt to describe their behavior as a combination of the known appliance loads. Another problem is that the solution that (4.5) provides can be very inappropriate since a small change in the measured $P(t)$ would often be associated with the change of state of many appliances at the same time [17]. This behavior is against the Switch Continuity Principle [17], which states that in a very small time interval, we expect the number of appliances that change state to be usually zero or one, and very rarely more than one in a typical load. Researchers commonly use minimization of (4.5) in combination with the Switch Continuity principle to design their state-based algorithms [77, 78].

An important discrimination feature for NILM algorithms using sampling frequencies around 1Hz is the power difference between continuous steady states. Let $\overline{P}_{\text{curr}}$ represents the power value of the current steady state and $\overline{P}_{\text{prev}}$ represents the previous steady-state value. Then we define the edge E as:

$$E = \overline{P}_{\text{curr}} - \overline{P}_{\text{prev}} \tag{4.6}$$

The frequency considered allows us to reasonably assume that the edge E is caused by the change in the state of exactly one appliance. Based on this assumption, E can be used as a discrimination feature. Due to transient spikes and signal fluctuations, the reliability of this feature depends heavily on the steady-state detection method employed.

4.3 Proposed framework: High-level architecture

The proposed framework evolves a dynamic structure S_P, where $S_{P_{i,j}}$ denotes the probability that appliance i is in state j. Unlike a fixed-dimension matrix, S_P supports a variable number of states per appliance. S_P is sequentially updated upon detecting a significant edge (event-based) or inconsistencies with the total aggregated power. After updating S_P an appliance might still be associated with multiple states with varying probabilities. To define a unique state for each appliance, S_P is integrated into a state-based optimization algorithm by employing the total matching criterion and possibly other additional criteria.

S_P efficiently preserves historical information and temporal linkages in a compact storage format. This also represents an advantage over existing algorithms [77, 78], where the temporal linkage is based on the assumption that the previous sample state is exactly known, and an incorrect prediction negatively impacts the future predictions. In addition, this architecture benefits from S_P functioning as a discrete probability distribution, which, when sampled, is likely to yield high-quality solutions with high probability. This characteristic enhances NILM state-based optimization algorithms by initiating the search process from a partially converged population, instead of restarting the search for each time step.

The framework optionally leverages information about the transition probabilities. Transition probability is the probability of observing a transition from state j to state k in appliance i at a given time t. Although the availability of transition probabilities is not required for the algorithm's functioning, it is expected to enhance its accuracy, especially for appliances with similar power levels. However, this information requires a training phase or a general model of the monitored appliances. In addition, the framework is robust to the presence of unknown appliances. Each time an edge is detected it is considered the possibility of an unknown load causing the edge. Based on its characteristics, Online, Real-time, Robust, and Edge driven, we will refer to it as the O2RE framework. The proposed O2RE framework (Alg. 1) comprises four principal modules:

Module 1: Edge Detection and Base Load Detection. This module detects events and steady states from the aggregated power profile. It also includes a base load detection procedure that identifies consistent power consumption from appliances operating in a steady state. The detection of steady states isolates periods with minimal power variation. Edge detection identifies significant changes (edges) in the aggregated power profile, potentially indicating appliance state changes (Section 4.4.1).

Module 2: State Probabilities Update. If a significant edge is detected, the state probabilities S_P are updated based on the edge's value, and, if available, additional information such as transition probabilities. However, it is also important to consider that the detected edge could be caused by an unknown load, which refers to a non-monitored appliance included in the aggregated power (Section 4.4.2).

Module 3: State Probabilities Tuning. In the tuning phase, the framework ensures that the state probabilities are coherent with additional criteria like the current aggregated power in a steady state. In this phase, the state probabilities S_P are updated to fulfill some constraints while minimizing the distance between the initial and updated state probabilities (Section 4.4.3).

Module 4: States Prediction. In this phase, the framework combines information on the state probabilities S_P with the total matching criterion and possibly other additional criteria, to predict the most likely state for each appliance. This step involves a state-based optimization algorithm (Section 4.4.4).

4.4 Low level design

In this section, the architecture exposed above is explained in detail through the framework modules.

4.4.1 Module 1: Edge Detection and Base Load Detection

Edge Detection

In practical scenarios, load meter measurements are prone to the presence of sensor noise, transient spikes, and signal fluctuations around the mean operating power of an appliance. The edge detection algorithm starts by identifying if the system is likely to be in a steady state considering the power grid noise σ_g^2 variance. For this task, we use the three-point method [53] which is simple and effective. Consider the current aggregated sample $P(t)$. The three-point method calculates the local average power $\mu_P(t)$, and the local deviation $\sigma_P^2(t)$ around $\mu_P(t)$.

$$\mu_P(t) = \frac{1}{3}\sum_{i=1}^{3} P(t - i\Delta t) \tag{4.7}$$

$$\sigma_P^2(t) = \frac{1}{3}\sum_{i=1}^{3} \left(P(t - i\Delta t) - \mu_P(t)\right)^2. \tag{4.8}$$

If $\sigma_P(t) < \sigma_g$, then the power is likely to be in a steady state, and we update the current and previous steady states: $\overline{P}_{\text{prev}} = \overline{P}_{\text{curr}}; \overline{P}_{\text{curr}} = \mu_P(t)$. This is the basis of the function **UpdateSteadyStates** in Algorithm 1. Then, we identify

Algorithm 1 O2RE NILM Framework: High-level Architecture

//Initialization
$S_P \leftarrow$ InitializeStateProb()
$S \leftarrow None$ ▷ Operating state indicator vector (Eq. 4.2)
$B \leftarrow \infty$ ▷ B represents the base load
for each t **do**
    //Edge and Base load detection (Module 1)
    $\overline{P}_{\text{curr}}\ \overline{P}_{\text{prev}} \leftarrow$ **UpdateSteadyStates**() ▷ Eq. 4.7 and 4.8
    $E \leftarrow \overline{P}_{\text{curr}} - \overline{P}_{\text{prev}}$
    $B \leftarrow$ **UpdateBaseLoad**()
    //State Probabilities Update (Module 2)
    $state\_prob\_updated \leftarrow False$
    if $|E| > \theta$ **then**
        $S_P \leftarrow$ **UpdateStateProb**()
        $state\_prob\_updated \leftarrow True$
    end if
    $\mathbb{E}[P] \leftarrow \sum_{i=1}^{N} \sum_{j=0}^{M_i} S_{P_{i,j}} \cdot P_{i,j}$ ▷ Expected Value of predicted power
    $P_{\text{fluc}} \leftarrow max(\overline{P}_{\text{curr}} - B, 0)$
    //State Probabilities Tune (Module 3)
    if $\mathbb{E}[P] > P_{\text{fluc}}$ **then**
        $S_P \leftarrow$ **TuneStateProb**()
        $state\_prob\_updated \leftarrow True$
    end if
    //States Prediction (Module 4)
    if $state\_prob\_updated == True$ **or** S **is** $None$ **then**
        $S \leftarrow$ **PredictStates**()
    end if
    output S as the prediction for time t
end for

← : Assignation, //: Section comments, ▷ : Line Comments

significant edges when Eq.(4.6) provides $|E| > \theta$, where θ is a significance threshold holding the condition $\theta \geq \sigma_g$.

Base load detection

Accurate estimation of the base load is also important for enhancing the accuracy of NILM algorithms, particularly those employing state-based approaches. The base load represents the consistent electrical consumption of appliances operating in a steady state. By accurately characterizing the base load, algorithms can effectively differentiate it from monitored appliance-specific states, improving the overall performance of NILM systems. In this work, the base load is estimated sequentially as shown in Algorithm 2.

Algorithm 2 Base Load Estimation: **UpdateBaseLoad**() function in Alg. 1

$\alpha \leftarrow$ *User-defined value* ▷ Set a value in the range $[0, 1]$
if $P(t) \leq B$ **then**
    $B \leftarrow P(t)$
else
    $\lambda \leftarrow \alpha \cdot \left(\frac{B}{P(t)}\right)^{(1-\alpha)}$
    $B \leftarrow B \cdot (1 - \lambda) + P(t) \cdot \lambda$
end if
return B

Algorithm 2 estimates the minimum value in a sliding window by using only the current measurement and stored past information. Classical algorithms for calculating exactly the minimum in a sliding window require space complexity proportional to the window size in the worst case. We propose an $O(1)$ space complexity approximation algorithm. The algorithm receives a tunable parameter α, which could be interpreted as $\alpha = \frac{1}{window\_size}$, it lies in the range [0,1]. By iteratively updating the base load estimation, the algorithm implicitly tracks the minimum power consumption within the sliding window. The estimated base load adapts to the fluctuations in power consumption and provides a dynamic estimate of the minimum value observed within the recent history of measurements.

In each step, the algorithm updates the value of the base load depending on the current power measurement $P(t)$. When $P(t)$ is less than or equal to the base

load it is trivial to equal it to the current value of $P(t)$. Otherwise, the value of the base load is increased taking into account how far is the current measurement from the base load, and the adjustable parameter α. Intuitively, if $P(t)$ is much larger than the base load, it is less probable that it becomes the minimum value in a window. On the other hand, when the window size is approaching 1 (α approaching 1) the probability of $P(t)$ becoming the next minimum increases. On the contrary, if the window size is approaching infinity (α approaching 0), the algorithm provides the minimum value found in all previous measurements. Setting the tunable parameter α involves balancing responsiveness to recent data against the smoothness of the estimation over a longer period. Assuming that the constantly-on devices remain the same for a long time and based on the experience of experimental tests, we consider a good tradeoff the interval, $1 \times 10^{-6} \leq \alpha \leq 1 \times 10^{-5}$, for sampling frequencies around 1Hz.

4.4.2 Module 2: State Probabilities Update

In this section we present the basis of the function **UpdateStateProb** in algorithm 1.

Probabilistic model for power state transitions in FSM appliances

The active power in each state of FSM appliances can not be considered constant. Active power can fluctuate due to changes in supply voltage and variations in load impedance [82]. Due to the large number of uncertainties, and provided that the sources of uncertainty are independent with finite variance it is reasonable to assume a Gaussian distribution for samples with large statistics. In [82], by assuming a Gaussian distribution for the grid voltage and conductance, is found a dependency of the active power variance with its expected value, and it is proposed a log-transformation in order to represent the transformed active power with a Gaussian distribution and remove this dependency. In this paper, we will consider this transformation as follows. If the random fluctuations of the power transition in load i from state j to state k is represented with the random variable $\Delta P_{i,j,k}$, we define the transformed random variable $\Delta \tilde{P}_{i,j,k}$ as:

$$\Delta\tilde{P}_{i,j,k} = \text{sgn}(\Delta P_{i,j,k}) \cdot \ln(|\Delta P_{i,j,k}|) \tag{4.9}$$

$$\mu_{\Delta\tilde{P}_{i,j,k}} = \text{sgn}(P_{i,k} - P_{i,j}) \cdot \ln(|P_{i,k} - P_{i,j}|) \tag{4.10}$$

$$\Delta\tilde{P}_{i,j,k} \sim \mathcal{N}(\mu_{\Delta\tilde{P}_{i,j,k}},\ \sigma^2_{\Delta\tilde{P}}) \tag{4.11}$$

Where sgn(.) is the sign of the given argument. Here we are assuming $\Delta\tilde{P}_{i,j,k}$ to be normal distributed with mean $\mu_{\Delta\tilde{P}_{i,j,k}}$ and variance $\sigma^2_{\Delta\tilde{P}}$. Please note that the value of $\mu_{\Delta\tilde{P}_{i,j,k}}$ only depends on the average power of the appliance states. On the other hand, we consider a fixed value of the variance $\sigma^2_{\Delta\tilde{P}}$ and this is a tunable parameter of the algorithm. Is important to note that a variance depending on the appliance may increase the accuracy of the method, but in order to retain the simplicity of the algorithm we consider using a single fixed value in our model. Details concerning the value of this parameter are provided in the experimental section of this paper. Equations (4.9)-(4.11) are important to estimate the likelihood of observing a detected edge E given a power transition in load i from state j to state k.

Likelihood calculation given an appliance state transition

Once an edge E is detected, we are interested in assigning a likelihood value $L(E|T_{ijk})$ to each appliance state transition T_{ijk}. Considering the model explained before (4.9)-(4.11) and normalizing each distribution to the standard normal distribution $N(0,1)$ ($\widetilde{x} = \frac{x-\mu}{\sigma}$) we obtain:

$$x_{ijk} = \frac{\tilde{E} - \mu_{\Delta\tilde{P}_{ijk}}}{\sigma_{\Delta\tilde{P}}} \tag{4.12}$$

Where:

$$\tilde{E} = \text{sgn}\ (E) \ln(|E|) \tag{4.13}$$

In this way, by evaluating x_{ijk} in the pdf function of the standard normal distribution, we calculate the likelihoods:

$$L(E|T_{ijk}) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2_{ijk}}{2}} \tag{4.14}$$

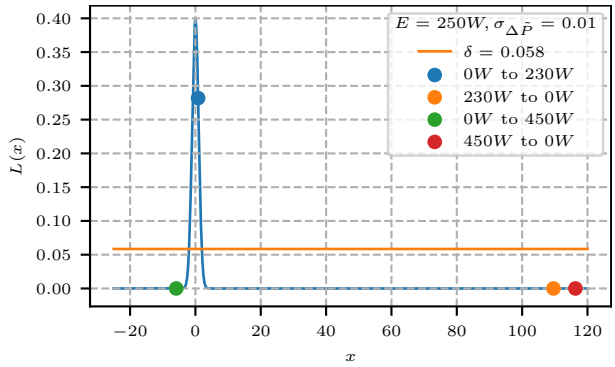


Figure 4.2: Example of the likelihood for four possible state transitions. For the edge $E = 250W$, the highest likelihood is associated with the appliance transition from 0 to $230W$ that, among the other possible state transitions and the unknown transition reported in the figure, is the most likely to represent the power variation E.

Likelihood of unknown state transition

To be realistic, it is considered that unknown loads could exist in measured aggregated power, and therefore, it could be the case that the change E is due to an unknown state transition and not due to a transition in the state of any of the N monitored appliances. To assign a likelihood of the unknown transition, this work exploits the idea of an improper constant density over the whole Euclidean (Eq. 4.15) which is often used in outliers clustering [83, 84].

$$L(E|UnkTrans) = \delta \tag{4.15}$$

The constant $\delta > 0$ is the improper constant density, and it is one of the settable parameters of the algorithm. The idea of the method is that transitions that lie in low-density areas ($L(E|T_{ijk}) < \delta$) are unlikely to explain the edge E. In other words, if $L(E|T_{ijk}) < \delta$, is more likely that edge E be due to a different transition. The reader is referred to Figure 4.2 for a visual example of the likelihood calculation.

Conditional probability calculation

Once the likelihoods are calculated, we assign a probability to each possible event conditioned on the value of the edge E. The events here are all the possible state transitions plus the unknown transition event. Defining then, $Pr(T_{ijk}|E)$

as the probability that the state transition in the appliance i from the state j to the state k is causing the edge E, $Pr(UnkTrans|E)$ as the probability that an unknown load causes the change E, and employing Bayes' theorem we can write:

$$Pr(T_{ijk}|E) = \frac{L(E|T_{ijk})Pr(T_{ijk})}{\sum\limits_{\substack{\forall i'j'k' \\ j' \neq k'}} L(E|T_{i'j'k'})Pr(T_{i'j'k'}) + UTT} \tag{4.16}$$

Where $UTT = L(E|UnkTrans) \cdot Pr(UnkTrans)$ is the unknown transition term and $Pr(T_{ijk})$ and $Pr(UnkTrans)$ are the prior probability terms. In (4.16) the subscripts i', j', k' move on all appliances, and for all the states transitions. It can be noted that the denominator in (4.16) is a direct application of the Law of Total Probability, since the state transitions and the unknown transition events are jointly exhaustive and mutually exclusive. In sections 4.4.2 and 4.4.2 it is explained how to calculate the likelihoods, equations (4.14) and (4.15), the terms $L(E|T_{ijk})$ and $L(E|UnkTrans)$ in (4.16) respectively. The prior probability terms $Pr(T_{ijk})$ and $Pr(UnkTrans)$, which represent the unconditioned probabilities for each state transition and the unknown state transition respectively, are important to distinguish similar loads. In general, accurately estimating these terms requires a prior training phase. Information about these terms is important to enhance the accuracy of the algorithm for similar loads, but in case this information is not available, the terms $Pr(T_{ijk})$ and $Pr(UnkTrans)$ can be fixed to the central probability vector. As a consequence, the starting values for $Pr(T_{ijk})$ and $Pr(UnkTrans)$ can be set to $\frac{1}{M+1}$, where M is the number of possible state transitions in all the appliances. That means that initially, all the events are equally probable. From now we will refer to this as the minimum information setting.

Updating sequentially the appliance state probabilities

We propose a recursive algorithm to update the probability of one appliance being in one certain state. Before, we defined $S_{P_{i,j}}$ as the probability of appliance i operating in state j, then, after an edge is detected, these probabilities are updated by using the conditional probabilities $Pr(T_{ijk}|E)$ previously calculated, assuming that the edge E can only be caused by exactly one change in the appliance state transition (Switch Continuity Principle) or it is caused by other unknown transition. The last cause implies that the appliance does not change state in the

current time step. In (4.17) is shown the equation to calculate the updated state probabilities.

$$\begin{aligned} S'_{P_{i,j}} &= [\sum_{j \neq k} Pr(T_{ikj}|E) \cdot S_{P_{i,k}}] \\ &+[(1 - \sum_{j \neq k} Pr(T_{ijk}|E)) \cdot S_{P_{i,j}}] \end{aligned} \tag{4.17}$$

The first addend in equation (4.17) calculates the probability of a state change in the appliance i to the state j from all possible state k. The second addend of (4.17) calculates the probability of no change in the appliance i from the state j. Then by simply setting $S_{P_{i,j}}$ to $S'_{P_{i,j}}$ we store in $S_{P_{i,j}}$ the updated state probabilities.

4.4.3 Module 3: State Probabilities Tuning

Up to now, the state probabilities variable relies only on the edge detection phase, thus, a considerable difference in a detected edge from its expected value can negatively impact the accuracy of the method. As an example, consider that the rising edge of an appliance is detected correctly but the corresponding falling edge is detected with a significantly different value than expected. In that case, the edge could be treated as coming from an unknown load, and the appliance would wrongly remain in an on-state.

The tuning phase aims to maintain S_P robust to anomalous cases as previously exposed. In this paper, we consider that S_P needs to be tuned when the expected value of power $\mathbb{E}[P]$ is greater than the fluctuating power P_{fluc}. Here, we define P_{fluc}, as the maximum between zero and the difference between the current steady state $\overline{P}_{\text{curr}}$ and the base load B. In other words, P_{fluc} is the part of the total power that is not constant (fluctuating) and is expected to integrate all FSM appliances.

As we are assuming that the appliances are independent, the expected value of power can be defined as

$$\mathbb{E}[P] = \sum_{i=1}^{N} \sum_{j=0}^{M_i} S_{P_{i,j}} \cdot P_{i,j} \tag{4.18}$$

In that sense, if $\mathbb{E}[P] > P_{\text{fluc}}$, is an indication that we are overestimating the power and some appliances can result wrongly in an on-state. Then we adjust S_P to ensure that $\mathbb{E}[P] \leq P_{\text{fluc}}$ while minimizing the distance to the original S_P. As a measure of distance, we consider the relative entropy [85] between the original and updated S_P. To solve the resultant convex-constrained optimization problem, the CVXPY package is used [86,87]. Analyzing the opposite case, if a rising edge is not correctly detected, the appliance would remain in an incorrect off state and we may underestimate the power consumption. It could be reasonable to set a minimum power P_{min} and impose another constraint in the way $\mathbb{E}[P] > P_{min}$. Considering that the on-time of most consuming FSM appliances is rather short, and setting a meaningful P_{min} could be difficult, we decided not to use a constraint of this type in this work. Another constraint considering a maximum on-time for each appliance can also be used as a complement.

4.4.4 Module 4: States Prediction

After updating the probabilities of different appliance states, uncertainty often persists, indicating that an appliance could still be associated with multiple states, each with varying probabilities. This results in a higher entropy scenario, where an exact operating state for some appliances remains less predictable. This step aims to converge toward lower entropy state probabilities where each appliance adopts a singular state with a probability near 1. It is also an opportunity to use criteria not employed in the steps before, like the Total Matching Criterion. We propose a modified Population-Based Incremental Learning algorithm (PBIL) [88] for this step.

PBIL algorithm

PBIL algorithms, first introduced by Baluja [88] as an abstraction of genetic algorithms (GAs), explicitly maintain the statistics contained in the population [89]. Traditional GAs mimic natural evolution through selection, crossover, and mutation operators applied to individual candidates. In contrast, PBIL evolves a probability vector $\vec{Pr} = [Pr_1, ..., Pr_N]$ (N is the length of the vector) to represent the likelihood of different traits contributing to high-quality solutions. The standard PBIL algorithm was designed for binary vector encoding but can be ex-

tended to consider integer vector encoding. In this case, the probability vector is defined as: $\vec{Pr} = [\vec{Pr}_1, ..., \vec{Pr}_N]$, where $\vec{Pr}_i = [p_1, ..., p_{M_i}]$ (M_i is the length of the vector $\vec{Pr}_i$), and $\sum_{j=1}^{M_i} p_j = 1$. PBIL is an iterative algorithm in which a set *pop* of S_z solutions is sampled from the probability vector $\vec{Pr}$ each iteration. Then, the probability vector $\vec{Pr}$ is learned toward the best sample $\vec{B}$ in *pop*, according to the problem-specific fitness function. To maintain diversity, a mutation process is carried out, learning the vector $\vec{Pr}_i$ towards a neutral vector $\vec{M}_i$ with a certain probability $\vec{p_m}$, and the cycle is repeated. As the iterations increase, each element in $\vec{Pr}_i$ moves from their initial values towards either 0.0 or 1.0 (vector $\vec{Pr}_i$ and as a consequence $\vec{Pr}$ converge to a lower entropy vector). The search process concludes when the fulfillment of a certain condition, such as reaching the predefined maximum iteration count or achieving convergence of the probability vector is reached.

PBIL for appliance states prediction

Here we consider the Total Matching Criterion and the state probabilities S_P as the bases for the fitness function. The first criterion considered is the minimization of $|e(t)|$ in (4.5). The second criterion is the maximization of the probability of a given solution S. As we are assuming that appliances are independent, this probability can be defined as $Pr = \prod_{i=1}^{N} [\sum_{j=0}^{M_i} S_{i,j} \cdot S_{P_{i,j}}]$. An equivalent computationally robust formulation of this criterion is the minimization of minus the logarithm of the probability (4.19):

$$-\ln(Pr) = -[\sum_{i=1}^{N} \ln(\sum_{j=0}^{M_i} S_{i,j} \cdot S_{P_{i,j}})] \tag{4.19}$$

To choose the best sample from the population *pop*, we define two ranking arrays based on separate criteria: *rank_e* and *rank_p*. Specifically,

- *rank_e*[*i*] denotes the rank of the *i*-th sample *pop*[*i*] when the population is sorted by the first criterion in ascending order.
- *rank_p*[*i*] denotes the rank of *pop*[*i*] when sorted by the second criterion in ascending order.

The best sample $\vec{B}$ from the population is identified by selecting the sample

$pop[i]$ that minimizes the maximum rank value between the two criteria for each sample. Formally, this can be expressed as:

$$\vec{B} = pop[i] \quad \text{where} \quad i = \arg\min_{i} \max(rank\_e[i], rank\_p[i]) \tag{4.20}$$

Algorithm 3 presents the pseudocode for the state prediction module, which represents the base of the function **PredictStates** in Algorithm 1. One significant advantage of Algorithm 3 is initiating the search process from a partially converged probability vector S_P (Algorithm 3 line 2). This initialization strategy allows the algorithm to bypass the initial exploratory phase required in random initialization. Consequently, this approach reduces the number of generations required to achieve convergence, enhancing the computational efficiency of state-based optimization NILM algorithms. Moreover, as S_P is the result of evolving an event-based NILM method over time, by initiating the search in S_P, the drawback of state-based algorithms associated with the possible presence of unknown appliances is expected to be eliminated.

The choice of PBIL improves the efficiency of the optimization process compared to traditional metaheuristics because PBIL evolves a single probability vector. The probability vector is updated depending on the performance of the sampled solutions, thus, eliminating the computational burden associated with managing multiple individual solutions usually found in other metaheuristics like Genetic Algorithms. This makes PBIL particularly advantageous for scenarios requiring efficient computation and scalability.

4.5 Experimental Results

This section presents the performance evaluation of our method in real-world publicly-available datasets. We compare the proposed framework with state-of-the-art approaches available in the literature and evaluate the robustness of our approach in the presence of unmonitored appliances. We compare our method with NILM algorithms proposed for real-time low-frequency data that only require information on the average power for each appliance state.

Algorithm 3 PBIL for appliance states prediction

Initialize: Sample size S_z, learning rate λ, mutation rate p_m, maximum generations G
$\vec{Pr}_i \leftarrow [S_{P_{i,0}}; ...; S_{P_{i,M_i}}]$, for each appliance i ▷ Initialization
for $g \leftarrow 1$ to G **do**
    Generate a set *pop* of S_z samples from $\vec{Pr}$
    Set $\vec{B}$ to the best sample in *pop* (See section 4.4.4)
    for $i \leftarrow 1$ to N **do** ▷ Learn $\vec{Pr}$ towards $\vec{B}$
        $\vec{Pr}_i \leftarrow (1 - \lambda) \cdot \vec{Pr}_i + \lambda \cdot \vec{B}_i$
    end for
    for $i \leftarrow 1$ to N **do** ▷ Mutate $\vec{Pr}$
        if $rand([0, 1]) < p_m$ **then**
            $\vec{Pr}_i \leftarrow (1 - \lambda) \cdot \vec{Pr}_i + \lambda \cdot \vec{M}_i$
        end if
    end for
end for
$S_{i,j} \leftarrow 0$ for each appliance i and state j
$selected\_state_i \leftarrow argmax(\vec{Pr}_i)$ for each appliance i
$S_{i,selected\_state_i} \leftarrow 1$ for each appliance i
return S

4.5.1 Performance Metrics

In assessing the accuracy of the considered methods for individual appliances, we employ the Estimation Accuracy and F-measure. These metrics were defined by equations (2.2) and (2.3), respectively. Additionally, we consider the Overall Estimation Accuracy, defined in Equation (2.5), and the Overall F-measure defined in Equation (2.6).

4.5.2 Experimental Setup

The framework with the minimum information setting O2RE-MIN (assuming only information on the average power for each appliance state), is tested on REDD [59] and UK-DALE [60] datasets. We chose house 1 from REDD and houses 2 and 5 from UK-DALE. From each dataset, we selected five common FSM appliances. Appliances and their average power per on-state can be found in Table 4.1.

Test scenarios ranging from 72 to 720 hours are selected to test the methods on different noise levels. The scenarios were designed to ensure activations across all monitored appliances in the selected intervals. As a measure of noise, we consider the percent-noisy measure defined in Eq. 2.8. We report the percent-noisy measure and the test period for each scenario. The sampling rate was set to 3s for all considered scenarios. Details of each scenario can be found in Table 4.2.

For benchmarking we compare with the MCE algorithm [77], MO-NILM [78], and the Hart combinatorial optimization (CO) baseline algorithm [17]. We use two versions of the CO algorithm, the traditional version and a custom version (CO-B) that subtracts the estimated base load (Algorithm 2) from the aggregated power. This custom version is introduced to evaluate the impact of the base load detection on the performance of state-based algorithms. All algorithms are implemented to be compatible with the widely adopted NILMTK toolkit [90]. These algorithms are selected because they are designed to be real-time, can handle frequencies on the order of 1Hz, use only the active power, disaggregate one reading at a time, and only require the average power in each appliance state. This makes them ideal for comparison with our proposed framework in a minimum information setting (O2RE-MIN). All algorithms compared are implemented with the recommended

Table 4.1: Appliance states used in simulation

Dataset	Appliance **(Acr.)**	On-States $[W]$
REDD-1	fridge **(F)**	190, 230
	dish washer **(DW)**	225, 1115
	microwave **(MW)**	1520
	electric oven **(EO)**	3350, 4090
	washer dryer **(WD)**	450, 2700
UKDALE-2	fridge **(F)**	90
	dish washer **(DW)**	100, 2000
	microwave **(MW)**	1308
	kettle **(K)**	2940
	rice cooker **(RC)**	400
UKDALE-5	fridge **(F)**	108
	dish washer **(DW)**	94, 1665
	microwave **(MW)**	1505
	electric oven **(EO)**	2120
	kettle **(K)**	2900

Table 4.2: Test Scenarios Description

Scenario	Dataset	Test start	Test hours	Percent-Noisy
1	REDD-1	2011-05-14	96	44
2	UKDALE-2	2013-06-01	720	59
3	UKDALE-5	2014-08-20	72	72

Table 4.3: Test Scenarios Settings (O2RE framework)

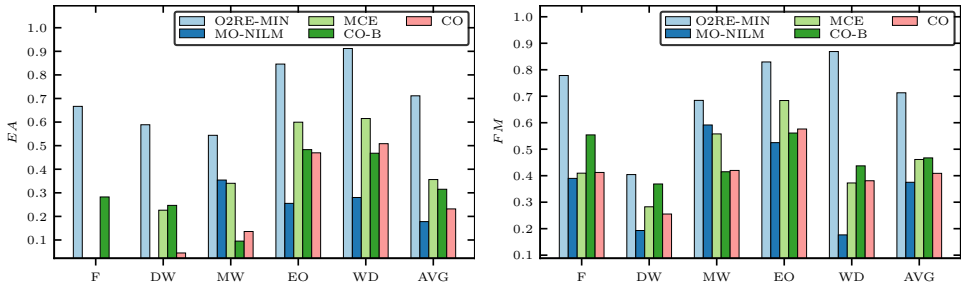
$\sigma_g[W]$	$\theta[W]$	$\sigma^2_{\Delta\tilde{P}}$	δ	α	λ	p_m	S_z	G
15	50	0.01	0.058	1×10^{-6}	0.025	0.01	30	50

settings as explained in their original papers. For the proposed O2RE framework, we use the same settings for each scenario and each appliance. These settings were chosen according to the number of appliances and by trial and error for calibration of the most noisy scenario. The value of each required parameter is presented in Table 4.3. Please refer to sections 4.4.1 and 4.4.4 for the definition of them.

4.5.3 Performance Results

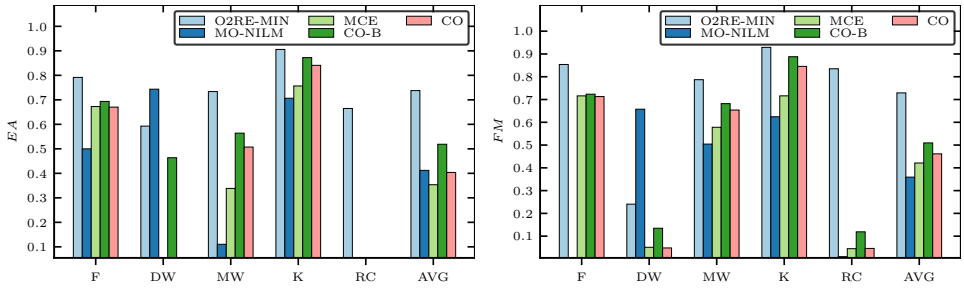
Performance results for each scenario defined in Table 4.2 for each monitored appliance are shown in Figures 4.3-4.5. We report the Estimation Accuracy and the F-measure for each appliance, as well as its average value for each scenario. From the figures is visible the superiority of the O2RE framework in all scenarios. When considering the appliance-specific metrics, the dishwasher shows the lowest accuracy. This is due to the fact that we are using only information on the average power consumption and the minimum dishwasher on-state is similar to the fridge on-state, causing some false positives in the dishwasher. For better discrimination in these cases, additional information like the on-probability, transition probability, etc; has to be fed to the framework. Is important to remark, that the low accuracy of the microwave in UKDALE-5 is related to a continuous power consumption near 50W in this appliance, which only is present in this dataset.

In general, the O2RE framework is stable for each considered scenario, even for the most noisy scenario 4, and reports high reliability for each considered appliance. Consider also, that we are using the main readings, and besides the contribution of unmonitored appliances, data can contain Gaussian noise due to fluctuations in sensor/ADC (analog-to-digital converter) precision [63]. Competitors' algorithms show a lack of robustness, especially for the noisiest scenarios. Also is interesting to note how the CO-B algorithm in some cases outperforms other more sophisticated ones, especially when the base load is significant. This emphasizes the importance of this estimation, especially for state-based approaches.



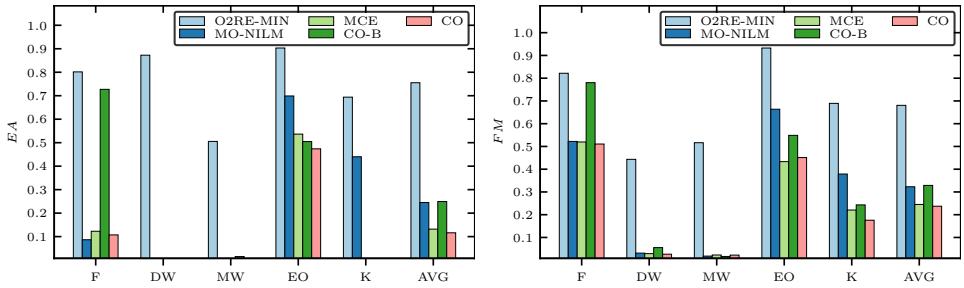
(a) Estimation accuracy (b) F-measure

Figure 4.3: Performance results Scenario 1 (REDD-1, 96h)



(a) Estimation accuracy (b) F-measure

Figure 4.4: Performance results Scenario 2 (UKDALE-2, 720h)



(a) Estimation accuracy (b) F-measure

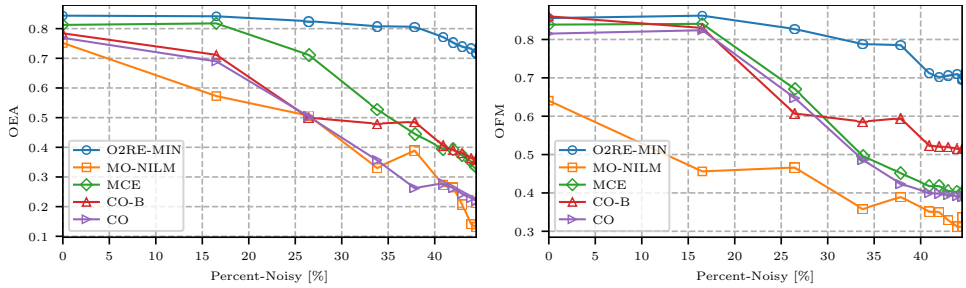
Figure 4.5: Performance results Scenario 3 (UKDALE-5, 72h)

4.5.4 Robustness to Noise

Intending to test how resilient our approach is to the presence of unmonitored appliances, we created artificial aggregates in the REDD-1 dataset. Starting by summing monitored appliances ($P_{A,0}(t) = \sum_{i=1}^{N} p_i(t)$) we add other appliances once at the time, selected in decreasing order of their energy consumption ($P_{A,r}(t) = P_{A,0}(t) + \sum_{j=1}^{r} p_j(t), 1 \leq r \leq K$), K is the number of appliances from

the same dataset not considered in the monitored appliances. In this way, we created $K+1$ different scenarios with an increasing level of noise. Then, we compare the Overall Estimation Accuracy (OEA) and the Overall F-measure (OFM) for each considered algorithm.

Results are shown in Figures 4.6a and 4.6b. It can be appreciated how our proposal is resilient to the presence of unmonitored appliances, reporting OEA and OFM greater than 0.7 for all scenarios, even when the noise is greater than 40%. The opposite trend is appreciated for competing algorithms, where there is a tendency to decrease performance as the noise increases. Also is interesting to note how the CO-B algorithm performs better than other more sophisticated algorithms for some noisy scenarios. This is because some always-on appliances are present in the dataset, and by subtracting the base load from the aggregated signal, the negative effect of these unknown appliances is alleviated.



(a) Overall Estimation Accuracy (b) Overall F-measure

Figure 4.6: Robustness as the percent-noisy increases

4.5.5 Computational complexity and real-time capabilities

The dominant parts in the O2RE framework (Algorithm 1) are the complexity associated with the dynamic programming algorithm implemented in **UpdateStateProb**() function and the complexity associated with the PBIL optimization algorithm. The time and storage complexity of the dynamic programming algorithm is $O(N \times M^2)$ since the function updates the state probabilities for each appliance considering each possible state transition. Here, N represents the number of appliances, and M is the maximum number of states in the appliances.

The time complexity of the PBIL algorithm is $O(G \times N \times S_z \times M)$. This complexity arises from operations across G generations, drawing populations of

size S_z, evaluating fitness, and updating state probabilities. Instead, the storage complexity of this part is $O(S_z \times N)$, which comes from storing the states of all appliances for each individual in the population.

As an example, if we assume 32-bit floating-point numbers for all calculations and consider a scenario involving 10 appliances (each with 3 states), a population size of 50, and 100 generations in the PBIL algorithm. The RAM usage includes approximately 360 bytes for the dynamic programming method (10×3^2 transitions $\times$ 4 bytes each), 2000 bytes for the PBIL algorithm (50 population size $\times$ 10 appliances $\times$ 4 bytes each), and additional 10 KB if we consider this value to account for typical overhead in embedded systems programming. This brings the total RAM requirement to about 12.36 KB, which is well within the memory capacity of constrained devices.

In terms of CPU workload, the dynamic programming method requires 90 operations (10 appliances $\times 3^2$ transitions), while the PBIL algorithm involves 150,000 operations (100 generations $\times$ 10 appliances $\times$ 50 population size $\times$ 3 states). These requirements are modest and fall well within the capabilities of constrained edge devices. For instance, the Itron Riva meter [91], equipped with an ARM Cortex-A8 MPU and 128 MB of memory [91], easily meets these demands. Our framework, which requires 12.36 KB of RAM and manageable computational operations, is suitable for deployment on such devices.

4.6 Conclusion of the chapter

This chapter presents an online real-time NILM framework for frequencies in the order of 1 Hz. By sequentially detecting edges and base load from the measured aggregated power, the probability of an appliance being in a given state is updated, taking into account the possibility of the presence of unknown loads. Using the information of the state probabilities, a modified PBIL algorithm is proposed to estimate the power consumption of each appliance. Experimentation with real-world publicly available datasets demonstrated its accuracy and robustness in the presence of unknown appliances. Benchmarking with online state-of-the-art algorithms using only information on the average power for each state results in higher accuracy for overall and appliance-specific metrics. The framework can also

be enhanced with additional information like the on-probabilities, the expected on-time, tuned algorithm parameters, etc; that are important to increase the accuracy for appliances with very similar power consumption. Investigation into how to obtain this information without involving significant occupant efforts for algorithm training is a research topic for future work.

Part III

Supply-Side Intelligence Through Robust PV Characterization

5

PV Characterization Fundamentals

In real-world applications, monitoring the condition of PV modules, quantifying degradation, and diagnosis are often evaluated using model-based parameter identification methods [22, 92]. These approaches involve fitting the electrical behavior of the module to an equivalent electric circuit, or model, which requires identifying a set of parameters. By analyzing the behavior of these parameters over time, it is possible to detect performance degradation or faults [93]. In addition to PV modules, PV submodules can be used for a more detailed diagnosis, representing a set of PV cells connected in series with a bypass diode in antiparallel [94].

Different approaches are used to describe the characteristics of PV panels in terms of their electrical and material characteristics. PVs are modeled using time-domain methods for describing their electrical features through static models such as the single-diode model (SDM). On the other hand, inside the frequency domain, solar cells are fundamentally considered as p–n junctions modeled through resistances, capacitances, and inductances using equivalent small-signal dynamic circuits.

5.1 SDM of a PV cell and module

The basic constructive element of a PV module is a cell, which can be represented by the SDM shown in Fig. 5.1. The relation between the cell's current (I_c) and voltage (V_c) is introduced in (5.1), where $I_{ph\text{-}c}$ is the photo-induced cur-

rent, $I_{s\text{-}c}$ is the inverse saturation current, η_c is the diode ideality factor, $R_{s\text{-}c}$ is the series resistance, and $R_{sh\text{-}c}$ is the parallel resistance. Moreover, $V_{th\text{-}c}$ is the thermal voltage, which can be defined as a function of the cell temperature (T_c), the Boltzmann constant k, and the electron charge q as follows: $V_{th\text{-}c} = kT/q$.

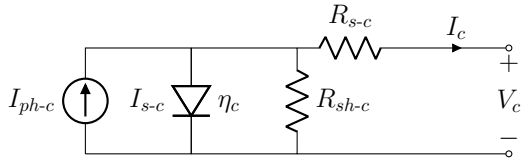


Figure 5.1: Single diode model for a single solar cell.

$$I_c = I_{ph\text{-}c} - I_{s\text{-}c}\left(\exp\left(\frac{V_c + I_c R_{s\text{-}c}}{\eta_c V_{th\text{-}c}}\right) - 1\right) - \frac{V_c + I_c R_{s\text{-}c}}{R_{sh\text{-}c}} \tag{5.1}$$

The SDM parameters depend on the cell's irradiance (G_c) and temperature (T_c); therefore, they cannot be assumed to be constants. Instead, the typical procedure proposed to identify the parameters for a given G_c and T_c is to calculate a set of reference SDM parameters ($I^*_{ph\text{-}c}$, $I^*_{s\text{-}c}$, η^*_c, $R^*_{s\text{-}c}$, $R^*_{sh\text{-}c}$) for a reference irradiance (G^*) and temperature (T^*), e.g. standard test conditions. Then, the reference parameters are used to correct their values for a given G_c and T_c by using explicit equations [95–98]. Often, these explicit equations include additional intrinsic coefficients that are provided by the manufacturer, fitted from experimental data, or generic values for each PV technology can be taken from the literature.

Although experimental studies in the literature do not provide a consensus on the behavior of each model parameter under varying temperature or irradiance levels [99], some assumptions are widely accepted. The saturation current is typically assumed to depend only on temperature [100, 101], while the parallel resistance is generally considered to depend only on irradiance [101, 102]. The ideality factor is assumed to be independent of irradiance, while some studies suggest it may depend on temperature [100, 102]. The series resistance is often treated as independent of both irradiance and temperature [101], although some research reports a linear variation of R_s with temperature [103].

The SDM can be scaled to represent a PV module consisting of N_s cells connected in series, provided that all N_s cells are identical and operate under uniform conditions. In that case, it is possible to use the same structure of (5.1) but scaling the parameters as shown in (5.2)-(5.7), where subindex m is used to

indicate the parameters of the entire module.

$$V_{th\text{-}m} = N_s \cdot V_{th\text{-}c} \tag{5.2}$$

$$I_{ph\text{-}m} = I_{ph\text{-}c} \tag{5.3}$$

$$I_{s\text{-}m} = I_{s\text{-}c} \tag{5.4}$$

$$\eta_m = \eta_c \tag{5.5}$$

$$R_{s\text{-}m} = N_s \cdot R_{s\text{-}c} \tag{5.6}$$

$$R_{sh\text{-}m} = N_s \cdot R_{sh\text{-}c} \tag{5.7}$$

5.2 PV degradation impact on the SDM parameters

This section examines how common degradation mechanisms in PV modules are reflected in variations of the SDM parameters, using the simulated I–V curves in Figure 5.2 as reference, and relates these variations to field-observed failures reported by the International Energy Agency (IEA) Photovoltaic Power Systems Programme (PVPS) Task 13 group [104].

Photoinduced current (I_{ph}) decreases when light transmission is reduced due to optical degradation. In the simulation, a 40% reduction in I_{ph} leads to a 41.4% drop in Maximum power point (MPP). This pattern reflects losses caused by encapsulant browning, ethylene-vinyl acetate (EVA) yellowing, or glass corrosion, which reduce photon access to the cell surface.

Saturation current (I_{s}) increases when junction leakage or enhanced recombination occurs. A 50-fold increase in I_{s} results in a 26.3% reduction in MPP, mainly due to a decrease in open-circuit voltage (V_{oc}) while short-circuit current (I_{sc}) remains largely unchanged. This behavior is typically associated with degradation mechanisms such as potential-induced degradation (PID) and environmental stress.

Ideality factor (η) decreases when the number of electrically active cells in the module is reduced due to disconnection or physical damage. A lower η shifts the I–V curve to the left without altering the slope near V_{oc}. Common causes include broken interconnects and cell cracks induced by thermal cycling or mechanical load.

Series resistance (R_{s}) increases when internal current paths deteriorate due

to corrosion, delamination, or contact degradation. In the simulation, a 200% increase in R_s results in a 6.8% loss in MPP. This parameter directly influences the slope of the I–V curve near V_{oc} and strongly affects the fill factor (FF). Among all SDM parameters, R_s is one of the most sensitive to degradation. It represents losses along metallic pathways and interfaces, and its increase is commonly linked to failed solder joints, broken ribbons, and other conductive failures documented in field-exposed modules.

Shunt resistance (R_{sh}) decreases when parasitic paths form across the p-n junction. In the simulation, a 50-fold reduction in R_{sh} causes a 21.0% drop in MPP, primarily due to increased leakage currents that lower the module voltage. Moisture ingress, encapsulant corrosion, or localized damage can lead to such behavior, particularly in modules operating in humid or coastal environments.

5.3 Overview of SDM parameter estimation methods

The problem of estimating the parameters of the SDM can be defined as follows: Given an experimental current–voltage (I–V) characteristic of a photovoltaic (PV) system, determine the set of parameters that best fit the measured data. This task differs from the forward simulation problem, where the objective is to compute the I–V curve from a known set of parameters. The simulation problem is typically less demanding, as it involves solving a system of non-linear equations and can be completed with moderate computational resources and high accuracy. In contrast, parameter estimation requires iterative optimization procedures to converge to an accurate solution. As a result, detailed modeling aspects, such as cell-level behavior and the effect of bypass diodes, are often omitted to reduce computational complexity.

Extensive research has been conducted on methods for identifying the parameters of the SDM. These methods can generally be classified into two categories: analytical and numerical approaches [105, 106]. Analytical methods simplify the PV model by focusing on key points of its current vs. voltage (I–V) characteristic [105]. In contrast, numerical methods utilize the entire I–V characteristic, or a portion of it, to determine a set of parameters that provides an optimal fitting [107].

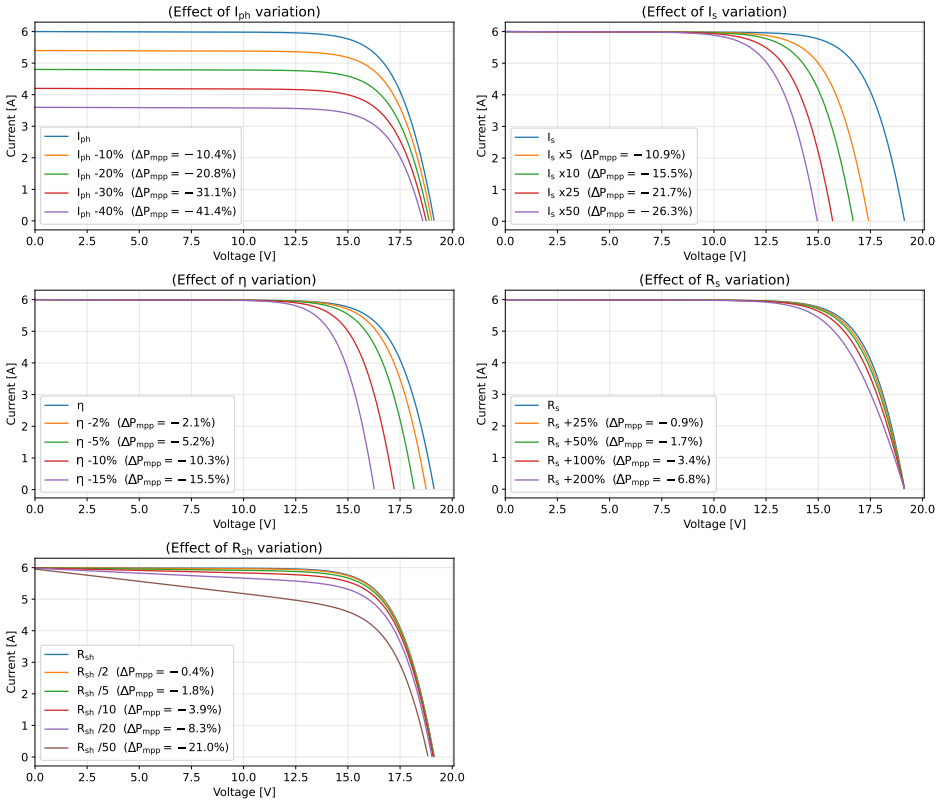


Figure 5.2: Impact of SDM parameter variations on the $I-V$ characteristics of a PV module. The legend shows the effect of varying one parameter while keeping the others at nominal values. It indicates the specific parameter change and the corresponding relative loss in maximum power point ΔP_{mpp}.

The study in [108] proposes an iterative method based on the Levenberg-Marquardt algorithm and simultaneous determination of PV module temperature and irradiance, eliminating explicit equations. A numerical method is introduced in [109], employing a genetic algorithm (GA) that relies on only five I-V curve points. The work in [110] presents an analytical method using explicit equations based on the Lambert W function for parameter identification from datasheet values. Similarly, [111] describes an integral-based analytical approach that employs linear least squares to enhance accuracy. In [112], a metaheuristic numerical method is presented, using Particle Swarm Optimization (PSO) and incorporating temperature-dependent series resistance.

A hybrid analytical-numerical technique is detailed in [113], combining PSO

with temperature modeling for improved results. In [114], a method combining analytical and iterative techniques is described through explicit equations and step-based refinement. The work in [29] proposes a multi-objective evolutionary algorithm designed to optimize static and dynamic model parameters simultaneously to achieve consistency across environmental conditions. The study in [115] introduces a deterministic numerical approach using derivative-based optimization, improving accuracy with minimal computation. Lastly, [116] presents a metaheuristic method based on Diversity Improvement-Oriented Differential Evolution (DIODE), which enhances population diversity to prevent premature convergence.

However, all the previous approaches assume that the cells within a PV module are identical and operate under uniform irradiance and temperature conditions. As a result, the model produces I–V curves with the same shape and curvature as those of a single cell, scaled according to the number of series-connected (N_s) and parallel-connected (N_{sh}) cells. This strong assumption is difficult to fulfill in real applications since the cells in a PV module or string, mismatch due to different causes. Some of the most common causes are partial shading due to surrounding objects (temporary or permanent), soiling, uneven aging of the PV cells, early degradation, and differences in the manufacturing process [31, 117]. The same assumption is adopted when using other models in the literature, such as the double-diode model (DDM) and triple-diode (TDM) model, which are applied to model PV generators with cells from various technologies [96].

5.4 Limits of the Uniformity Assumption in SDM Parameter Estimation

In realistic scenarios, each PV cell operates under unique irradiance and temperature conditions, influenced by partial shading, soiling, or manufacturing variability. These mismatches lead to electrical imbalances among cells in a series string, causing deviations from the behavior expected under the uniformity assumption commonly applied when estimating SDM parameters. Even when the global I–V curve does not show visible distortion, the underlying non-uniform conditions can severely compromise the accuracy and physical validity of the fitted parameters.

SDM fitting under mismatched conditions tends to redistribute the error across parameters in non-physical ways. Figure 5.3 illustrates this effect through simulations where one cell in a 32-cell module (two 16-cell submodules) is progressively shaded. Each cell is modeled using the following set of parameters: $I^*_{ph\text{-}c} = 6$ A, $I^*_{s\text{-}c} = 1 \times 10^{-7}$ A, $\eta^*_c = 1.3$, $R^*_{s\text{-}c} = 0.003\ \Omega$, and $R^*_{sh\text{-}c} = 20\ \Omega$. To simulate the shading, the parameter $I_{ph\text{-}c}$ of the shaded cell is reduced proportionally to the shading level, while the other parameters are considered constants. The bypass diode effect in each 16-cell submodule is also modeled in this exercise.

The left panel of Figure 5.3 displays the simulated I–V curves under five shading levels, ranging from 0% to 20%. The right panel presents the corresponding fitted curve and the relative error in each parameter, assuming uniform conditions across all cells. Although the visual changes in the I–V curves are subtle, particularly at lower shading levels, the estimated parameters in the right panel deviate significantly from the reference values.

Already at 5% shading, the saturation current (I_s) is reduced by a factor of 13.7, the shunt resistance (R_sh) drops by a factor of 7.3, and the series resistance (R_s) increases by 25.3%. These changes do not reflect physical alterations in the module but are artifacts of the fitting process, attempting to replicate a non-uniform behavior using a uniform model. As the shading level increases to 10%, 15%, and 20%, the deviations become even more pronounced. For example, at 10% shading, I_s drops by a factor of 231.3, while R_s increases by 38.1% and R_sh decreases by a factor of 15.2. These parameter shifts are disproportionate to the actual physical disturbance and illustrate the inadequacy of assuming homogeneity under mismatching.

It is important to recognize that some parameters, such as the saturation current (I_s), have well-established dependencies on environmental conditions, particularly temperature. Therefore, care must be taken to distinguish between variations caused by normal operating conditions and those resulting from degradation or poor model assumptions. The deviations in fitted parameters shown in Figure 5.3 are not related to changes in the operating conditions. Instead, they reflect the model's attempt to compensate for cell-to-cell mismatches using a uniform parameter set, making it difficult to separate actual aging or fault effects from errors introduced by oversimplified modeling.

These results highlight that the uniformity assumption is fragile. Even low levels of mismatch that do not produce obvious visual distortions in the I–V curve can lead to unphysical SDM parameter estimates. Such discrepancies reduce the reliability of degradation diagnostics, fault detection, and performance modeling when using SDM-based monitoring tools under field conditions.

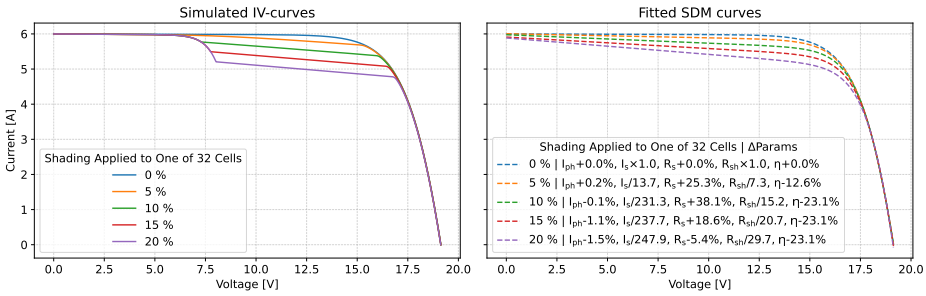


Figure 5.3: Effect of progressively shading a single solar cell in a 32-cell series string (2×16-cell sub-modules). The left panel shows simulated IV-curves for five shading levels (0–20 % in 5 % steps). The right panel shows curves obtained by fitting a SDM to each simulation. Legend entries on the right combine the shading percentage with the relative change of key SDM parameters with respect to the unshaded baseline: I_s and R_{sh} as multiplicative factors ($\times$ or /), and I_{ph}, R_s and η as signed percentage changes.

5.5 Frequency Domain Analysis as a Complement

One way to improve the robustness of SDM parameter estimation under mismatching conditions is to incorporate complementary diagnostic methods that capture different physical properties of the photovoltaic device. Frequency-domain analysis, particularly through small-signal perturbation techniques, offers access to internal dynamic behavior that static current-voltage characterization cannot resolve. This additional layer of information can constrain the parameter estimation process and reduce sensitivity to external distortions.

5.5.1 IS Technique

IS is a non-invasive technique originally developed for characterizing electrochemical systems, including batteries and fuel cells. It operates by applying a small

sinusoidal perturbation to the voltage or current of a system and measuring the resulting response over a range of frequencies. The method assumes a linear response due to the low amplitude of the excitation, allowing the impedance spectrum to be interpreted in terms of distinct physical processes.

In photovoltaic applications, IS enables the identification of resistive, capacitive, and diffusion-related effects within the device. High-frequency behavior is generally associated with fast processes such as contact resistance and carrier collection, while low-frequency behavior reflects slower mechanisms such as recombination, leakage paths, or charge accumulation at interfaces. These dynamic signatures provide insight into the internal structure of the device, including junction quality, interfacial integrity, and material degradation.

By integrating frequency-domain information into the estimation framework, it becomes possible to decouple phenomena that are otherwise indistinguishable in a static curve. The data obtained through IS can serve as physical constraints or validation references during model fitting, especially under conditions where mismatches or local defects distort the global device response. Since the technique operates with minimal perturbation, it is suitable for online diagnostics and long-term monitoring in real operating environments.

5.5.2 Frequency domain modeling with a CPE

The frequency-dependent information provided by IS can be interpreted using equivalent circuit models that represent the internal structure and dynamic processes of PV devices. These models translate the impedance spectrum into electrical elements such as resistors, capacitors, inductors, and more general components. A well-defined model helps quantify how different physical mechanisms contribute across frequency ranges and supports accurate fitting of measured data.

Solar cells often show deviations from ideal capacitive behavior, especially in the presence of imperfections at junctions or interfaces. These deviations result in impedance spectra that do not form perfect semicircles in the complex plane. To capture this non-ideal behavior, the constant phase element (CPE) is introduced. The CPE extends the standard capacitor model and enables a better fit of experimental data, particularly for silicon-based technologies.

The dynamic model that includes the CPE is presented in Fig. 5.4, and its

corresponding impedance expression is given in Eq. (5.8) [118]. This formulation captures both the resistive and capacitive effects across the junction, as well as inductive behavior at high frequencies.

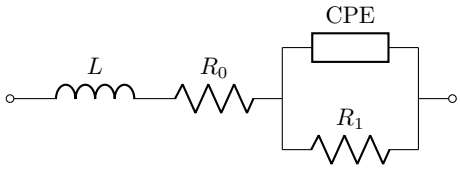


Figure 5.4: Dynamic equivalent circuit including a constant phase element (CPE).

$$Z_{eq}(\omega) = j\omega L + R_0 + \frac{R_1}{[(j\omega)^{\beta} Q R_1 + 1]} \tag{5.8}$$

In Eq. (5.8), L represents the parasitic inductance, R_0 the series resistance, and R_1 the resistance across the junction. The CPE models the non-ideal capacitive behavior through the parameters β and Q [119].

Fig. 5.5 shows the expected impedance response in the complex plane based on the CPE model. The high-frequency region appears on the left side of the plot, where the inductive effect from L is visible as a deviation into the negative imaginary axis, labeled as "L effect." As the frequency decreases, the impedance response traces a single arc, which results from the combined influence of the junction resistance R_1 and the CPE. At the lowest frequencies, the curve flattens and approaches the total resistance $R_0 + R_1$ along the real axis. The figure includes visual markers for R_0, R_1, and the inductive effect, clearly illustrating the contribution of each element to the overall impedance shape. This impedance profile resembles typical experimental data of PV modules.

5.5.3 Practical Limitations of IS Technique

Implementing IS on PV panels under real-world conditions presents substantial hardware challenges. The technique requires precise injection of small AC perturbations and synchronized measurement of the voltage and current response over a broad frequency range. In outdoor applications, fluctuations in irradiance and temperature during the measurement window can alter the operating point, introducing instability and distortion in the impedance data. To address this, the

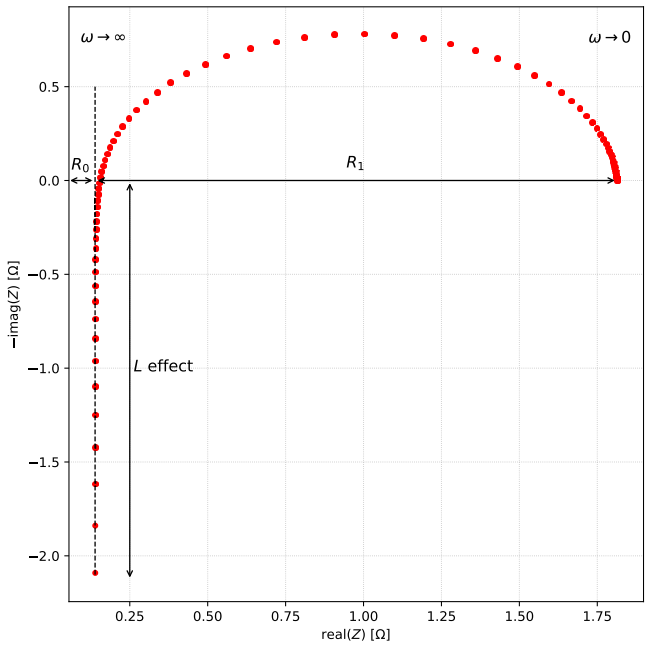


Figure 5.5: Impedance response in the complex plane based on the CPE model. The arc reflects the non-ideal capacitive behavior of the junction.

hardware must acquire complete spectra rapidly while maintaining tight control over the system's electrical state, which significantly increases circuit complexity.

In addition, the wide impedance variation across frequencies demands highly sensitive and low-noise analog front ends capable of maintaining resolution at both high and low impedance values. Long cable runs in PV installations introduce parasitic effects that further complicate accurate measurement, particularly at high frequencies where inductive and capacitive coupling become significant.

At present, these technical requirements, combined with the need for robustness, safety, and low cost, limit the practical deployment of IS to laboratory or highly controlled environments. The technique is not yet mature for routine field application. However, the richness of the physical information accessible through IS justifies continued study of its models and measurement strategies, to enable reliable integration into PV diagnostics in the future.

5.6 Conclusion of the Chapter

This chapter presented the fundamentals of electrical characterization in PV systems, focusing on the use of the SDM to describe the electrical behavior of modules through a compact set of physically meaningful parameters. These parameters are commonly used for condition monitoring and degradation analysis. However, the assumption of uniform operating conditions across all cells within a module is difficult to satisfy in practice. Mismatches caused by shading, soiling, or aging introduce distortions that can lead to inaccurate and non-physical parameter estimates, even when the overall I–V curve appears undistorted.

The analysis of parameter behavior under mismatch conditions highlights the limitations of standard SDM fitting procedures. Errors introduced by the uniformity assumption are distributed across parameters in unpredictable ways, reducing the diagnostic value of the fitted model. To complement the SDM analysis, this chapter introduced IS as a technique capable of revealing additional information about the internal dynamics of PV devices. Although still not mature as a practical solution for PV systems, IS provides additional insights that can support the interpretation of I–V-based models. Despite current practical limitations in measurement hardware, its integration with time-domain analysis represents a promising direction for improving model robustness.

The following chapter examines how IS measurements can be integrated with the SDM to improve the accuracy and stability of parameter identification by incorporating complementary frequency-domain information. This integration demonstrates the added value of using dynamic electrical behavior to reinforce static model fitting. In contrast, the subsequent chapter proposes a more practical approach based solely on I–V curve data, without requiring additional hardware. By increasing the flexibility of the modeling structure and refining the parameter estimation strategy, this method offers a solution better suited for widespread deployment in real PV systems using existing measurement capabilities.

6

Multi-Objective Optimization for Identifying SDM and CPE Model Parameters

6.1 Introduction

Building on the limitations identified in Chapter 5, this chapter investigates how frequency-domain data obtained through IS can be integrated into the identification process of SDM parameters. While conventional methods rely solely on fitting the static I–V characteristics, incorporating complementary dynamic measurements offers an opportunity to improve the stability and consistency of the parameter estimates. The approach developed here formulates a multi-objective optimization problem that leverages both the static and dynamic representations of photovoltaic modules to obtain more coherent and physically meaningful parameter sets. The content of this chapter has been previously divulged in **Publication C1.** [**29**].

Several techniques have been developed to estimate optimal parameter sets that align photovoltaic models with experimental measurements. Analytical, iterative, and metaheuristic methods are commonly used to approximate solutions to this estimation problem [120]. However, beyond the choice of optimization method, the accuracy and reliability of the results are strongly influenced by factors such as measurement noise, the distribution of observation points, initialization strategies,

and the structure of the objective function. More importantly, as highlighted in Chapter 5, the presence of non-uniform operating conditions across cells in a module can distort the fitting process and lead to non-physical parameter estimates. These limitations highlight the need to incorporate additional physical information to enhance the identification of model parameters.

Static and dynamic models have some common points corresponding to the inherent material and electrical PV solar cell characteristics. In [121], it is shown that the R_{mpp} (that is the differential resistance of the PV panel at its Maximum Power Point) and the PV series resistance, appearing in both SDM and dynamic model, could be calculated using each model independently. Moreover, [121] compares the performance of the two models for estimating the series resistance when PV panels are affected by the series resistance degradation phenomenon. By assuming the series resistance as a common parameter for the two models, the parameter identification task can be formulated as a multi-objective problem. The joint parameters estimation seeks more reliable and stable parameters across different environmental conditions, as shown in the following.

This approach relies on two complementary representations of PV behavior. The static characteristics are modeled using the SDM, previously defined in Chapter 5 (Eq. 5.1), which relates current and voltage through five physical parameters. The dynamic behavior is captured by the impedance model introduced in Chapter 5 (Eq. 5.8), which incorporates a CPE to model non-ideal capacitive dispersion observed in practical IS measurements. Together, these models form the basis for the joint parameter identification process proposed in this work.

The rest of the Chapter is organized as follows: section 6.2 introduces some concepts related to multi-objective optimization and provides the complete mathematical formulation and methodology; section 6.3 presents the results achieved and the discussion, and section 6.4 presents the conclusions of the investigation.

6.2 Multi-objective formulation and validation

A multi-objective optimization problem (MOP) is a mathematical optimization problem that can be formulated as follows:

$$\begin{aligned} &\text{minimize } \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T \\ &\text{subject to } \mathbf{x} \in \Omega \subseteq \mathbb{R}^n \end{aligned} \tag{6.1}$$

where Ω is the decision space or feasible region, $\mathbf{F} : \Omega \to \mathbb{R}^m$ consists of m real-valued objective functions and $\mathbb{R}^m$ is the objective space. In a MOP the objectives often conflict with each other, making it impossible to find a single solution that optimizes all objectives simultaneously. As a result, the search for a solution involves finding a set of optimal trade-off solutions, where improvement in one objective cannot be achieved without negatively impacting at least one of the other objectives. The best tradeoffs among the objectives can be defined in terms of Pareto optimality.

Given two solutions $x, y \in \Omega$, x is said to dominate y ($x \prec y$) if and only if $f_i(x) \leq f_i(y)$ for all $i = 1, 2, ..., m$, and $f(x) \neq f(y)$. A point $x^* \in \Omega$ is Pareto optimal if and only if there is no point $x \in \Omega$ such that $F(x) \prec F(x^*)$. The set of all Pareto optimal points forms the so-called Pareto set (PS), while the set of all corresponding Pareto optimal objective vectors constitutes the Pareto Front (PF).

In many practical situations, a decision-maker needs an approximation of the PF to choose the preferred solution from. Often, computing the complete PF can be impractical due to the large number of Pareto optimal solutions that the MOP may have. To address this issue, multiobjective optimization algorithms aim to find a smaller set of Pareto optimal solutions, that are evenly distributed along the PF and thus provide a good representative of the entire PF.

6.2.1 SDM fitting error calculation

The first objective function is defined as the root mean squared error between the experimental ($I_{i,exp}$) and calculated ($I_{i,cal}$) current for each of the N experimental voltage values. $I_{i,cal}$ is evaluated by using eq. (5.1).

$$f_1 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(I_{i,exp} - I_{i,cal})^2} \tag{6.2}$$

6.2.2 CPE fitting error calculation

The second objective function should measure the difference between the experimental and calculated impedances. Eq.(5.8) is used to evaluate the real ($Z^{'}_{i,cal}$) and the imaginary ($Z^{''}_{i,cal}$) parts of the PV impedance. In this case, the model parameters are selected to minimize the sum of weighted square differences between the experimental and calculated impedances.

$$f_2 = \sum_{i=1}^{M} \{ \frac{\left(Z^{'}_{i,exp} - Z^{'}_{i,cal} \right)^2}{\sqrt{Z^{'2}_{i,cal} + Z^{''2}_{i,cal}}} + \frac{\left(Z^{''}_{i,exp} - Z^{''}_{i,cal} \right)^2}{\sqrt{Z^{'2}_{i,cal} + Z^{''2}_{i,cal}}} \} \tag{6.3}$$

In equation (6.3), the summation runs over all impedance values measured at M frequencies available in experiments. Equation (6.3) implements the modulus weighting approach. In this approach, the small and large impedances contribute in a similar way to the sum of squares [122].

6.2.3 Decision variables analysis

The multi-objective formulation involves 9 decision variables in total (I_{ph}, I_s, η, R_{sh}, R_s, R_1, L, β, Q). Involved in the calculation of (6.2) are I_{ph}, I_s, η, R_{sh}, R_s; and involved in the calculation of (6.3) are $R_0 = R_s$, R_1, L, β and Q. It is important to analyze how the decision variables influence the conflicting objectives, in order to possibly decompose a high-dimensional problem into a set of simpler and low-dimensional subproblems [123].

In our formulation, the variable R_s controls the diversity of solutions as well as the convergence of the population. The rest of the variables only control the convergence of the population. That is to say, for a fixed R_s, it is possible to optimize (6.2) and (6.3) in an independent way. With this in mind, it is possible to decompose the variables into two groups and exploit this to efficiently solve the problem. More details of the decomposition will be given in the next sections.

6.2.4 Normalization method

Objective space normalization is important when the objectives are in different scales. Without normalization, multi-objective evolutionary algorithms (MOEAs) may fail to give equal emphasis to each objective, which can deteriorate their ca-

pabilities to obtain uniformly distributed and well-converged solutions [124]. Objective space normalization can be achieved using the ideal and nadir points. An approximation for the ideal point can be found by minimizing each objective function individually over the decision space. Unfortunately, the nadir point is much more difficult to estimate because it requires the knowledge of the PF [124].

In this work, the estimation of the ideal point is found to minimize f_1 and f_2 independently. The estimation of the nadir point is found as follows. Let us define R_s^{id} and R_0^{id} as the resistances obtained for the ideal solutions. The nadir point is then estimated by minimizing f_1 with R_s fixed to R_0^{id} and minimizing f_2 with R_0 fixed to R_s^{id} (see Algorithm 4 for more details). In (6.4), the normalization formula is shown.

$$\widetilde{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - (ideal_i - \epsilon)}{nadir_i - (ideal_i - \epsilon)} \tag{6.4}$$

The small constant ϵ is introduced to remedy the diversity deterioration caused by inaccurately estimated ideal and nadir points. As in the early generations, the population is far away from the PF, the use of normalization with estimated ideal and nadir points can deteriorate the performance of the algorithm. In this work, this drawback is considered by applying the normalization procedure only when we have a converged population (see Algorithm 4 for more details).

6.2.5 Proposed Algorithm

In MOPs, some decision variables influence the convergence of the obtained solutions, while some decision variables determine the diversity of the solutions [125]. Inspired by the algorithm introduced in [123], we propose a multi-objective evolutionary algorithm that, in the early stages of evolution, fixes the values of the diverse variables (only R_s in our case), and evolve the distance variables only (variables controlling only the convergence, in our case the rest of variables). This is intended to improve the convergence speed of the population.

Different from a single objective optimization problem, MOP needs to optimize all objective functions together. Therefore, in the second part of the algorithm, we optimize all the decision variables as a whole. This is to improve the uniformity of the Pareto front and to take into account the effect of R_s in the convergence of the population. For this optimization task, we use the nondomi-

nated sorting genetic algorithm (NSGA)-II [126]. The MOEA/D algorithm [127] was also tested, but an improvement in the diversity of solutions was found with NSGA-II. Simulated binary crossover ($probability = 0.9$, $\eta = 15$) and polynomial mutation ($probability = 1/chromosome\_length$, $\eta = 20$) were the operators used in NSGA-II. The algorithm was implemented in the pymoo framework [128].

Algorithm 4 Proposed algorithm for jointly SDM and CPE optimization

Require: $V_{i,\exp}, I_{i,\exp}, Z'_{j,\exp}, Z''_{j,\exp}, pop\_size$; $1 \leq i \leq N$; $1 \leq j \leq M$

Ensure: *pop*

// Calculation of ideal and nadir points

$x_1^{id} = I_{ph}^{id}, I_s^{id}, n^{id}, R_{sh}^{id}, R_s^{id} \leftarrow \arg\min(f_1)$

$x_2^{id} = R_0^{id}, R_1^{id}, L^{id}, \beta^{id}, Q^{id} \leftarrow \arg\min(f_2)$

$x_1^{na} = I_{ph}^{na}, I_s^{na}, n^{na}, R_{sh}^{na}, R_0^{id} \leftarrow \arg\min(f_1 \mid R_s = R_0^{id})$

$x_2^{na} = R_s^{id}, R_1^{na}, L^{na}, \beta^{na}, Q^{na} \leftarrow \arg\min(f_2 \mid R_0 = R_s^{id})$

$ideal \leftarrow [f_1(x_1^{id}), f_2(x_2^{id})]$

$nadir \leftarrow [f_1(x_1^{na}), f_2(x_2^{na})]$

// Early stages of evolution

$pop \leftarrow \emptyset$

Set R to pop_size random uniform numbers in the possible R_s interval

for all $r \in R$ **do**

    $x_1 \leftarrow \arg\min(f_1 \mid R_s = r)$

    $x_2 \leftarrow \arg\min(f_2 \mid R_0 = r)$

    $x \leftarrow x_1 \cup x_2$

    Insert x into *pop*

end for

// Optimize all decision variables: Late stages of evolution

Use NSGA-II to evolve *pop*, normalizing with *ideal* and *nadir* using Eq. (6.4)

return *pop*

6.2.6 Temperature dependence of reverse saturation current for parameters validation

The reverse saturation current I_s measures the leakage (or recombination) of minority carriers across the p–n junction in reverse bias; its value can be expressed

in terms of temperature and bandgap energy [129] as follows:

$$I_s = CT^3 \exp\left(-\frac{qE_g(T)}{kT}\right) \tag{6.5}$$

In (6.5), the constant C includes doping and the material parameters of solar cells [129], which implies that its value should remain almost constant for the same solar module in different environmental conditions. The remaining parameters of (6.5) include the elementary charge q, the Boltzmann constant k, the temperature of the cell in Kelvin T, and the bandgap energy $E_g(T)$ at the given temperature. A relation of the variation of E_g with temperature is proposed in [130], the equation is as follows:

$$E_g(T) = E_g(0) - \frac{\alpha T^2}{(T+\beta)} \tag{6.6}$$

Where $E_g(0)$ is the band gap of the semiconductor at $T = 0K$, and α and β are constants for the material. In [130], the values of the constants for different materials are also given. In the case of the Si, the values reported in [130] are $E_g(0) = 1.1557eV$, $\alpha = 7.021 \times 10^{-4} eVK^{-1}$ and $\beta = 1108K$. In this work, C is calculated for each estimated I_s in each experiment considered. The consistency of C is then validated across experiments, as it is expected to remain similar.

6.3 Results on experimental data

6.3.1 Experimental data

The method proposed in this chapter is evaluated using data obtained from the experimental campaign developed within the framework of [121]. All the measurements have been done over a commercial PV module manufactured by *Solbian* with reference *Flex–SP50L* [131] (its specifications under STC are given in Table 6.1). Here, the IS data is acquired in outdoor conditions at the MPP point. The corresponding I-V curve is also measured. The experiments account for operating states at different environmental conditions varying from medium to high irradiance. The reader can refer to [121] for more details on the experimental data.

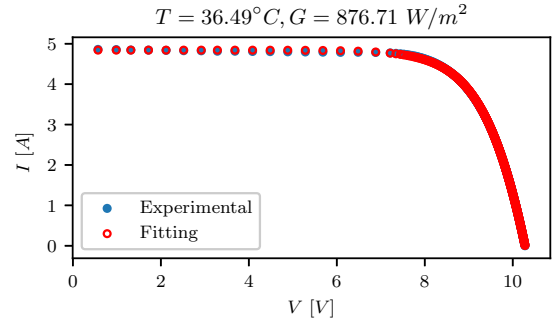
Table 6.1: Specifications of the photovoltaic module

General features	
Manufacturer	Solbian
Model	FLEX SP50–L
Technology	Single–crystalline silicon
	Back–contact
Cell manufacturer	SunPower™
Cells in series	16
Strings in parallel	1
Dimensions	1109 × 292 × 2 mm
Weight	0.8 kg
Main electrical parameters (under STC)	
Maximum power P_{M}	51 W ± 5%
Short–circuit current I_{SC}	6 A
Open–circuit voltage V_{OC}	10.9 V
Current at MPP I_{M}	5.7 A
Voltage at MPP V_{M}	9.0 V
Temperature coefficients (under STC)	
I_{SC} temperature coefficient (α)	+0.05 %/°C
V_{OC} temperature coefficient (β)	−0.27 %/°C
P_{M} temperature coefficient (γ)	−0.38 %/°C

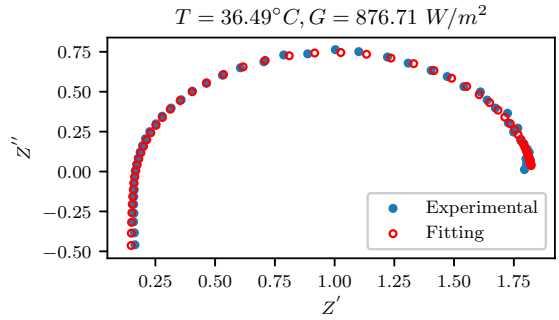
6.3.2 Fitting models independently

In the first step, f_1 and f_2 were optimized independently. That is to say, we search for the values of the parameters $I_{ph}, I_s, \eta, R_{sh}, R_s$, that minimize f_1, thus fitting I-V curve. In the same way, we are looking for the parameters R_0, R_1, L, β, Q that minimize f_2, thus fitting the IS spectrum. The comparison of the experimental with the fitting is shown in Fig. 6.1 for both models.

Table 6.2 shows the values corresponding to the best fitting results. Three important observations can be made. First, the values of R_s differ significantly from those of R_0, indicating that these parameter sets, despite minimizing fitting error, do not accurately represent the PV module behavior. Second, in some experiments,



(a) Experimental vs. Fitting (SDM)



(b) Experimental vs. Fitting (CPE)

Figure 6.1: Independently optimization [29]

Table 6.2: Parameters for independent optimization

Condition		*SDM* (*Mean Fitting Error* : 0.85×10^{-2})						*CPE* (*Mean Fitting Error* : 0.05)				
$T[°C]$	$G[W/m^2]$	$I_{ph}[A]$	$I_s[A]$	η	$R_s[\Omega]$	$R_{sh}[\Omega]$	$C[A/K^3]$	$L_0[H]$	$R_0[\Omega]$	$R_1[\Omega]$	Q_1	β
30.67	535.39	2.83	0.22×10^{-8}	1.17	0.13	1×10^4	199.32	0.22×10^{-5}	0.17	3.04	0.19×10^{-3}	0.90
33.74	600.96	3.21	0.71×10^{-7}	1.37	0.08	1×10^4	3985.18	0.20×10^{-5}	0.16	2.75	0.37×10^{-3}	0.82
32.24	692.70	3.75	0.24×10^{-6}	1.48	0.06	190.02	1.70×10^4	0.19×10^{-5}	0.15	2.36	0.33×10^{-3}	0.86
33.71	758.06	4.16	0.36×10^{-6}	1.50	0.06	274.91	2.02×10^4	0.20×10^{-5}	0.15	2.06	0.37×10^{-3}	0.85
36.49	876.71	4.84	0.11×10^{-5}	1.57	0.06	1×10^4	3.92×10^4	0.19×10^{-5}	0.14	1.68	0.25×10^{-3}	0.93

R_{sh} approaches the upper bound of the search interval, suggesting inconsistency when compared across different environmental conditions. Third, the constant C varies considerably between experiments, even though it should remain relatively stable since it refers to the same PV panel.

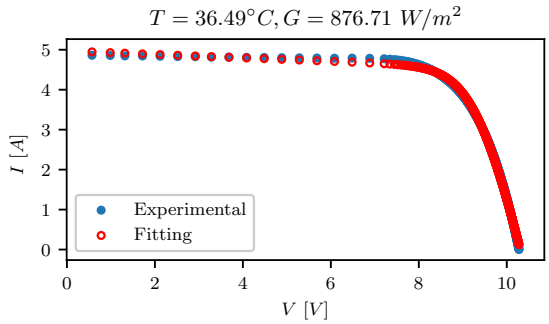
To understand the effect of the variation between R_s and R_0, Fig. 6.2a shows the results of the optimization of f_1 with R_s fixed to the R_0 obtained in the previous optimization of f_2 (see the value in Table 6.2). Also, Fig. 6.2b shows the results of the optimization of f_2 with R_0 fixed to the R_s obtained in the previous optimization of f_1 (see the value in Table 6.2).

Figure 6.2 shows that the fitting quality of the SDM and CPE models is highly sensitive to the values of R_s and R_0, respectively. This suggests that obtaining reliable parameters for evaluating the state of the photovoltaic module requires finding a tradeoff between the f_1 and f_2 fitting errors.

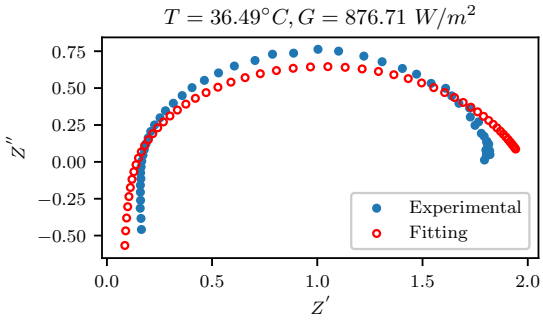
6.3.3 Multi-objective optimization results

To provide the decision-maker with a set of optimal tradeoff solutions, from which to choose the preferred solution, the functions f_1 and f_2 are optimized in a multi-objective fashion subject to the physical constraint of $R_s = R_0$. It is important to note that the aim of this work is not to obtain an ideal algorithm for this task, but to evidence the impact of tradeoff solutions on reliable parameter identification. The resultant Pareto front is shown in Fig. 6.3 for one experiment.

It can be appreciated from Fig. 6.3 the uniformity of solutions obtained along the objective domain. This behavior is very important for the decision-maker because it provides a visual representation of the trade-offs between different objectives. Knowing the real shape of the PF can also guide an intensification of the search in the preferred regions.



(a) Experimental vs. Fitting (SDM)



(b) Experimental vs. Fitting (CPE)

Figure 6.2: Effect of the difference between R_s and R_0 [29]

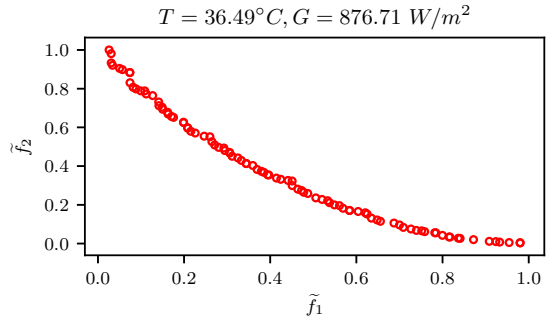


Figure 6.3: Normalized Pareto front [29]

Table 6.3: Parameters for tradeoff solutions

Condition		*SDM* (*Mean Fitting Error* : 0.03)					*CPE* (*Mean Fitting Error* : 0.06)					
$T[°C]$	$G[W/m^2]$	$I_{ph}[A]$	$I_s[A]$	η	$R_s[\Omega]$	$R_{sh}[\Omega]$	$C[A/K^3]$	$L_0[H]$	$R_0[\Omega]$	$R_1[\Omega]$	Q_1	β
30.67	535.39	2.83	0.17×10^{-9}	1.04	0.16	993.53	15.79	0.22×10^{-5}	0.16	3.04	0.19×10^{-3}	0.90
33.74	600.96	3.24	0.35×10^{-9}	1.06	0.15	99.92	19.97	0.20×10^{-5}	0.15	2.77	0.39×10^{-3}	0.82
32.24	692.70	3.83	0.19×10^{-9}	1.04	0.14	35.74	13.65	0.19×10^{-5}	0.14	2.38	0.36×10^{-3}	0.84
33.71	758.06	4.23	0.38×10^{-9}	1.06	0.14	40.53	21.45	0.21×10^{-5}	0.14	2.10	0.44×10^{-3}	0.83
36.49	876.71	4.96	0.50×10^{-9}	1.05	0.13	31.32	18.21	0.19×10^{-5}	0.13	1.74	0.35×10^{-3}	0.89

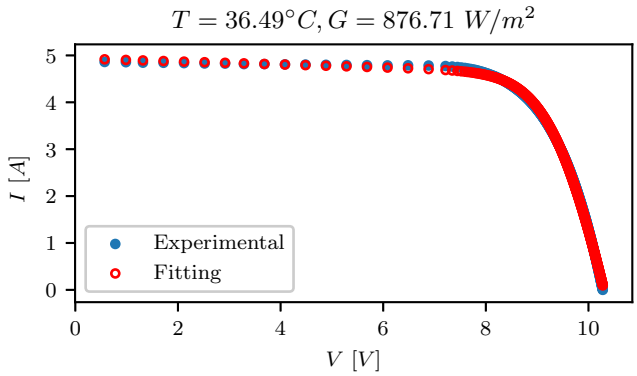
6.3.4 Impact of tradeoff solutions in the model's parameters

Obtaining a well-defined Pareto front is only part of the problem; selecting a single preferred Pareto-optimal solution is equally important and challenging. Although this paper does not focus on decision-making, we implement a straightforward approach to demonstrate how tradeoff solutions influence the model parameters. To select a single solution from the Pareto front, we use the goal programming method [132]. This method requires the decision-maker to define target values for each objective and then identifies the solution that either meets these targets or minimizes deviations from them. In this study, we set the aspiration values for f_1 and f_2 to 0.018 and 0.07, respectively, to reflect acceptable levels of fitting quality. Aspiration levels in this context represent the maximum acceptable error values for the fitting objectives, meaning that solutions closer to or below these values are considered satisfactory. Table 6.3 reports the model parameters corresponding to each selected solution for the analyzed experimental cases.

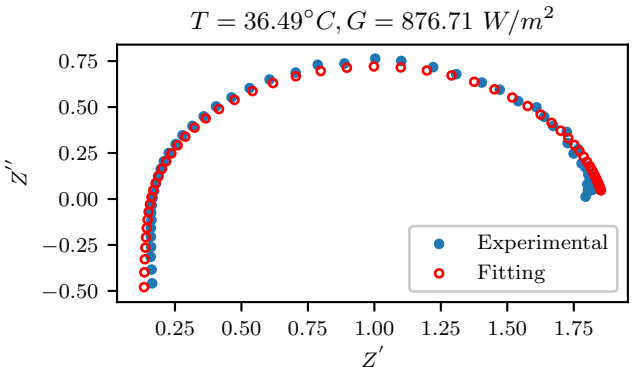
Table 6.3 shows that the limitations observed in the independent fitting across different environmental conditions are significantly reduced. The values of R_{sh} are no longer constrained to the boundary of the interval, and the parameters C and η exhibit greater stability across experiments, as expected. Figure 6.4 presents the visual fitting for the experiment analyzed in this study. Although the selected solution does not provide the best fit for either model individually, it achieves a satisfactory compromise, yielding good agreement for both models simultaneously.

6.4 Conclusion of the Chapter

This chapter presents and evaluates a multi-objective formulation for parameter identification in photovoltaic modules. Independent fitting of parameters for



(a) Experimental vs. Fitting (SDM)



(b) Experimental vs. Fitting (CPE)

Figure 6.4: Fitting for tradeoff solution [29]

the static (SDM) and dynamic (CPE) models revealed inconsistencies across different environmental conditions. To address this, the fitting problem is reformulated as a multi-objective optimization task that exploits the shared series resistance between the models and is solved using evolutionary algorithms. A decision variable analysis guided the design of an algorithm to approximate the Pareto front. Selecting a tradeoff solution from the Pareto front using a goal programming approach yields more coherent parameter sets for each experimental condition.

7

Self-Adaptive Single-Diode Model Parameter Identification Under Real-World Operating Conditions

7.1 Introduction

While the integration of static and dynamic models through multi-objective optimization improves the coherence of parameter identification, the approach introduced in the previous chapter (Chapter 6) presents practical limitations for deployment in real-world systems. The use of Electrochemical Impedance Spectroscopy (EIS) requires additional hardware that is not yet mature for routine field applications, and selecting a tradeoff solution from the Pareto front involves threshold tuning that may not generalize across operating conditions. Moreover, the underlying uniformity assumption remains a critical source of inaccuracy. This chapter proposes an alternative methodology that operates solely on I–V curve data, aiming to increase the flexibility of the modeling structure and refine the parameter identification procedure. The resulting approach offers a more practical and scalable solution for integration into standard PV monitoring systems. The content of this chapter has been previously divulged in **Publications J2., C2. and C3.** [**30**–**32**].

When a PV module or string is under non-uniform conditions, estimating the SDM parameters from an experimental I-V curve will lead to unrealistic values. The

reader is referred to Chapter 5, where some simulated examples showing this issue are provided. Non-uniform conditions produce distorted I-V curves that generate unrealistic parameters even if the distortion is difficult to identify with the naked eye [31, 117]. For example, some authors have shown that mismatching conditions generate an I-V curve with a sharper curvature around the maximum power point [133]. In that case, the uniformity assumption provides unrealistic parameters for the SDM due to the lack of the necessary degrees of freedom to account for the underlying physical structure. Moreover, the unrealistic SDM parameters may lead to wrong diagnostics or degradation quantification of the monitored system.

One approach to consider realistic mismatching conditions of the cells forming a PV module or string is to increase the model granularity by representing each cell with the Bishop equivalent circuit [134], which requires eight parameters. This approach has been used to simulate different PV generators operating under mismatching conditions, assuming as known the parameters of the cells [135]. Nevertheless, the parameter identification of a model with this granularity significantly increases the complexity and computational burden of the model, since in a module or string with N_s cells, the number of parameters to identify would be $8N_s$. Additionally, in a PV module, measuring the voltage of each cell is not possible, making it even more difficult to identify the parameters of the Bishop model.

Submodule-level simulations, where each submodule includes a group of cells protected by a single bypass diode, reduce computational complexity. In [136], a submodule-based modeling approach is proposed for series-parallel PV arrays under mismatching conditions, using the bisection method instead of Jacobian-based methods. In [137], a model is introduced to predict the performance of large PV systems, focusing on submodule-level mismatch calculations and explicit I–V curve evaluation. In [138], submodule-level simulations are compared to finer cell granularity methods, such as the cell-by-cell method, achieving similar accuracy, especially when the submodule is uniformly shaded. Although these approaches are sufficiently accurate for simulation studies, they all assume uniform operating conditions across the submodule. This assumption limits their applicability for reliable parameter identification from experimental I–V curves, as demonstrated in [31].

Another approach to model PV strings operating under mismatching con-

ditions is to represent each module with the SDM [105, 139], DDM [140], or TDM [141]. Then, the modules are connected in series to form an equivalent circuit that represents the string. A methodology proposed in [139] focuses on detecting voltage inflection points and knees in the I–V curve. In this approach, the SDM is first fitted under various uniform operating conditions to derive a functional relationship between the parameters and the operating conditions. However, these approaches also assume that all the cells in each module are equal and operate under uniform conditions, which is not realistic, as explained before.

A model-based online diagnostic tool may rely on mismatch detection algorithms to run parameter identification procedures when operating conditions are near uniform. However, in the presence of small mismatches, such as minor soiling or manufacturing differences, uniformity may be incorrectly assumed, resulting in significant errors in parameter estimation. These errors can obscure the SDM parameters trends, complicating the identification of early signs of malfunctions or degradation. Relying solely on mismatch detection to decide when to perform parameter identification, assumes that uniform conditions are always necessary, which is unrealistic in real-world PV systems. Many small mismatches occur frequently that can distort critical parameters if not accounted for. To fill this gap, this chapter introduces a novel parameter identification procedure that remains robust to non-ideal operating conditions. Consequently, this work seeks to improve the practical applicability of parameter identification methods, enabling their use in online diagnostic tasks under real operating conditions.

This chapter proposes a procedure to estimate the SDM parameters that is robust to mismatched conditions. The self-adaptive nature of the procedure allows it to adjust its parameters in response to varying shading patterns, ensuring consistent performance across different scenarios, and making it applicable in both uniform and mismatched conditions.

In summary, the main contributions of this work are:

- A 7-parameter self-adapting SDM-based model, denominated Double Single-Diode Model (D-SDM), is proposed to enhance the accuracy and reliability of parameter identification in PV modules under mismatched conditions
- A robust methodology based on evolutionary algorithms is introduced to estimate the parameters of the proposed D-SDM model

- The advantages of the proposed model and methodology are demonstrated using experimental and simulated I–V curves across various real partial shading patterns
- The capability of the D-SDM procedure to detect series resistance degradation is evaluated and compared with the SDM model and the dynamic CPE model to identify the advantages and disadvantages of each parameter identification method

The rest of the chapter is organized as follows. Section 7.2 revisits the limitations of the SDM model when uniform conditions are assumed. Section 7.3 outlines the foundations of the proposed D-SDM model, including the circuit design, underlying assumptions, and relevant equations. Section 7.4 describes the methodology to extract the parameters of the D-SDM model. Section 7.5 validates the model and methods by using experimental and simulated I–V curves, Section 7.6 evaluates the capability of the D-SDM to detect series resistance degradation in comparison with the SDM and the dynamic CPE model, and section 7.7 closes the chapter with the conclusions.

7.2 Limitations of the Uniformity Assumption Revisited

The SDM is widely used to represent a PV module consisting of N_s series-connected cells and protected by bypass diodes connected in antiparallel [94]. However, this approach assumes that all the cells are the same and operate under the same irradiance and temperature conditions, which is difficult to guarantee in real applications. This section highlights the limitations of the uniformity assumption when the PV module is under mismatching conditions.

In realistic scenarios, each PV cell operates under unique irradiance and temperature conditions, influenced by factors such as shading or manufacturing differences, which can result in mismatches among the cells. Even when the mismatch level is small enough that the deformation of the I-V curve is not visually noticeable, fitting the SDM to such a curve will fail to identify the parameters accurately. The five parameters adjusted during the curve-fitting process often reach

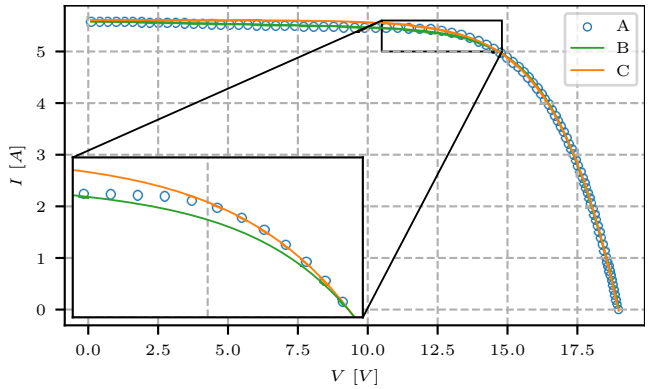


Figure 7.1: Uniform assumption limitations. A. Experimental data; B. SDM fitting using the full experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc}

unrealistic values. In Chapter 5, some simulated examples showing this issue are provided. In this section, we will provide an additional example that highlights the limitation of SDM under the uniformity assumption, coming from a commercial PV module in outdoor conditions.

Consider the experimental curve shown in Fig. 7.1. It refers to the PV string shown in Fig. 7.6, Case I, where two modules connected in series seem under the same irradiance conditions, but there is a small mismatch possibly caused by the reflexes on the surface of the module. In these conditions, bypass diodes are not active and the I-V curve is expected to be well fitted with the SDM. Nevertheless, when attempting to fit the SDM to the experimental curve from 0 to V_{OC}, the optimization algorithm tends to prioritize solutions that fit a larger number of points, which compromises the accuracy near the MPP (as seen in Fig. 7.1, curve B). One potential solution to improve the fitting around the MPP is to focus on a portion of the curve near this point. Figure 7.1, curve C illustrates the fitting when using data points from voltages slightly below the MPP to V_{OC}. As seen in Table 7.1, this modification results in significantly different estimated parameters. The series resistance R_s decreases notably, while the parallel resistance R_{sh} increases dramatically, converging to the upper bound of the search interval. These inconsistencies between the two fittings, when similar values might be expected, highlight the limitations of the uniformity assumption, even when the mismatching conditions are not immediately visible.

Table 7.1: Uniform assumption limitations. B. SDM fitting using the full experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc}

Fitting	$I_{ph}[A]$	$I_s[A]$	η	$R_s[\Omega]$	$R_{sh}[\Omega]$
B	5.6029	4.8474e-06	1.4934	0.17562	93.121
C	5.6089	9.5054e-05	1.8901	0.090265	∞

7.3 Adaptive Double-SDM of a PV module

To consider the mismatching conditions in a PV module, it is proposed to divide the N_s module cells into two groups, i.e. group a and group b, considering that the module cells operate under two main irradiance levels: shaded and unshaded. This approach increments the freedom degrees without significantly increasing the model complexity, the number of parameters to identify, and the computational burden, as occurs in the case of using a dedicated model for each cell. In realistic scenarios, each PV cell operates under unique irradiance and temperature conditions. However, in cases of small mismatches, where the difference in irradiance is not significant, the conditions of the affected cells are often similar. As a result, the two-irradiance levels assumption becomes a reasonable approximation. Figure 7.2 presents a schema of the two-irradiance levels assumption for a PV module with N_s cells in series, where $N_{s\text{-}a}$ and $N_{s\text{-}b}$ are the number of cells in each group and $N_s = N_{s\text{-}a} + N_{s\text{-}b}$.

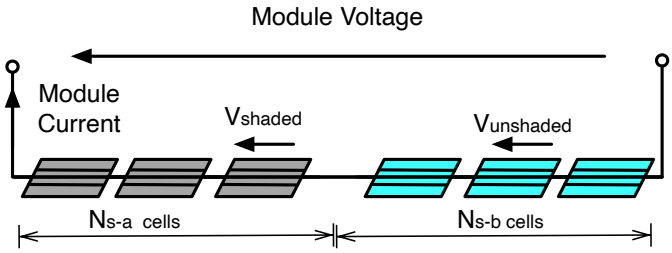


Figure 7.2: Two-irradiance levels assumption for a PV module with N_s cells in series

Small mismatch scenarios are of particular interest, as these are more challenging to detect from I-V curve analysis and are highly relevant for robust parameter identification. In this case, the operating conditions differ only slightly, making the deviations in the I-V curve less pronounced and more difficult to identify. Significant mismatch scenarios, in contrast, typically result in clearly notice-

able deviations in the I-V curve. In such cases, running a parameter identification method may not be meaningful, as these mismatches are often transient conditions caused by factors such as clouds.

In the proposed approach, each group is represented by an SDM as illustrated in Fig. 7.3. It is important to note that the modeling strategy focuses on estimating the number of affected cells within the module rather than identifying their exact locations. By categorizing cells into shaded and unshaded groups, the model achieves a more accurate representation of the electrical behavior of the module under non-uniform conditions, without the complexity of modeling each cell individually. The parameters $N_{s\text{-}a}$ and $N_{s\text{-}b}$ dynamically adjust themselves to the operating conditions, allowing the model to adapt itself to various scenarios. This adaptability makes the model more general, as it can handle uniform and non-uniform operating conditions. Hence, the circuit model shown in Fig. 7.3 is denominated as Double SDM (D-SDM).

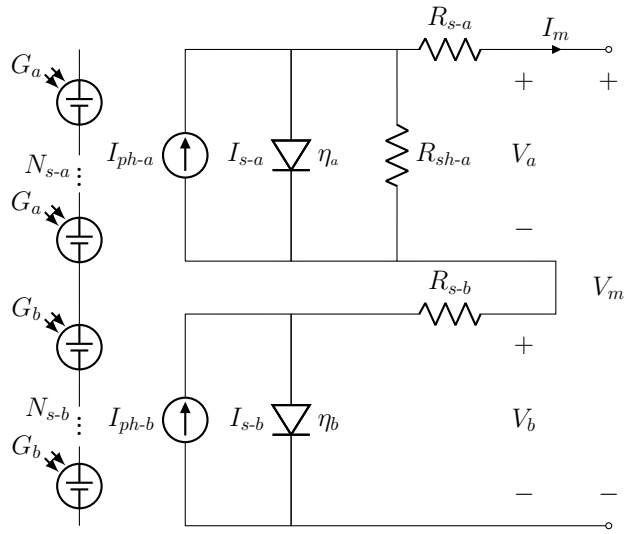


Figure 7.3: D-SDM model representing groups a and b in series.

This modeling strategy aims to maintain the critical parameters of the SDM ($I_{s\text{-}c}$, η_c, $R_{s\text{-}c}$, $R_{sh\text{-}c}$) independent from mismatching patterns. This approach prevents distortions in the I-V curve caused by mismatch issues from resulting in incorrect parameter estimation intended to fit the curve. Instead, any distortion in the I-V curve is expected to be captured by the introduced parameters $N_{s\text{-}a}$, $N_{s\text{-}b}$, $I_{ph\text{-}a}$, and $I_{ph\text{-}b}$.

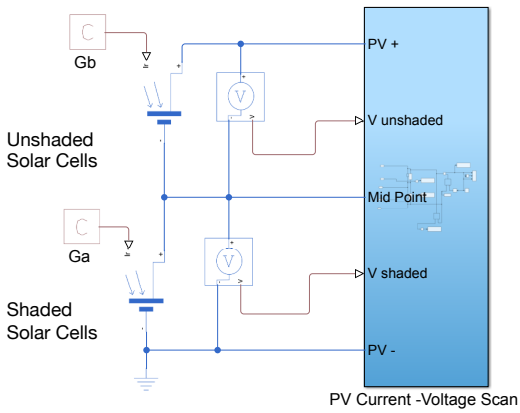


Figure 7.4: Simulink block of the schema shown in Figure 7.2

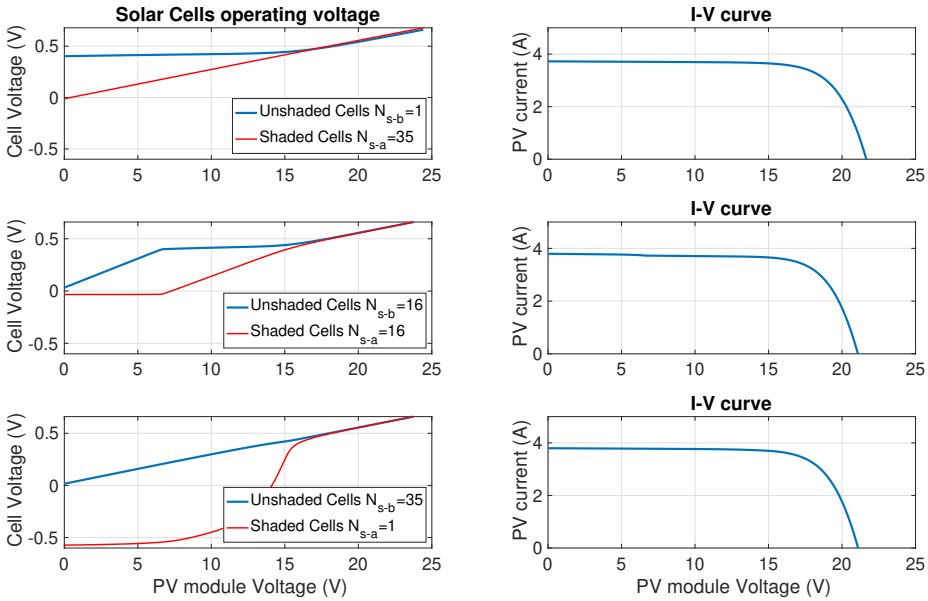


Figure 7.5: Operating voltage of Cell versus PV module voltage in different mismatched scenarios for a 2% difference in irradiance conditions ($G_a = 0.98G_b$).

Assuming the cells of groups a and b operate under irradiances G_a and G_b, respectively, and $G_b > G_a$ without loss of generality, it is possible to analyze the section of the I–V curve where all the cells operate at positive voltages. The validity of the SDM is limited to positive voltages, so it is important to consider only the part of the I–V curve where all cells are operating within this range. As widely discussed in the literature, shaded cells in a PV string may operate in

reversed mode depending on the number of shaded cells, the shading effect level, and the PV module operating voltage. In the presence of a significant mismatch level, as for the example discussed in [142], the shading effect introduces a visible deformation of the I-V curve by reducing the delivered power and with the risk of triggering hot-spots in the cells operating with negative voltage. On the other side, if the level of mismatch is small, the effect is not visible on the global I-V curve, nevertheless it may introduce a difference in the operating voltage of each cell.

To explain this effect, a Simulink block reproducing the scheme of Figure 7.2 has been simulated to evaluate the voltage across the shaded and unshaded cells as a function of the PV output voltage for three cases: 1 shaded cell, 18 shaded cells, and 35 shaded cells in a PV module composed of 36 series connected cells. A 2% difference in irradiance conditions between the cells in each block ($G_a = 0.98G_b$) is analyzed for each simulation. The *Solar Cell* Simulink embedded block [143] is used with its default parameter values, as shown in Figure 7.4. Details on the *Solar Cell* block and its integration with a variable electric load for I-V curve tracing are available on the MATLAB webpage [143]. As it is evident from the plots on the left side of Figure 7.5 the voltage of each cell changes significantly as a function of the number of shaded cells even if the I-V curve of the PV module remains almost unchanged (right plots of Figure 7.5). Hence, the key idea of having a self-adapting number of shaded/unshaded cells helps to tune the D-SDM also in the presence of small mismatching conditions, when the I-V curves seem almost uniform; moreover, this model automatically converges to the SDM in the presence of really uniform conditions.

It is worth noting that $R_{sh-b} \approx \infty\ \Omega$ in the model presented in Fig. 7.3 for the group of unshaded cells. This approximation is performed considering that R_{sh-b} has a negligible impact on the output current of group b (i.e. in the I-V curve of group b) for high voltages; since in this section the output current principally depends on the current of the diode, which increases exponentially and is much larger than R_{sh-b} current. Moreover, the section of the module I-V curve where shaded and unshaded cells operate at positive voltages corresponds to high voltages for the I-V curve of group b (unshaded cells). Numerical validation of this simplification is provided in section 7.5.4.

It is worth noting that the cell temperature is assumed to be the same

across the module $T_a = T_b = T_c$. In steady-state conditions, the cell temperature T_c can be estimated as a function of the ambient temperature T_{amb} and the incident irradiance G_T through (7.1) [144], where T_{NOCT} is the Nominal Operating Cell Temperature (NOCT), typically 45°C; $T_{amb,\mathrm{NOCT}}$ is the ambient temperature under NOCT conditions, typically 20°C; and G_{NOCT} is the solar irradiance under NOCT conditions, typically $800\,\mathrm{W/m^2}$.

$$T_c = T_{amb} + \left(\frac{T_{\mathrm{NOCT}} - T_{amb,\mathrm{NOCT}}}{G_{\mathrm{NOCT}}}\right) G_T \tag{7.1}$$

The difference in cell temperature $\Delta T_c = T_b - T_a$ due to a difference in irradiance $\Delta G = G_b - G_a$ in a PV module can be derived by manipulating (7.1) as:

$$\Delta T_c = \left(\frac{T_{\mathrm{NOCT}} - T_{amb,\mathrm{NOCT}}}{G_{\mathrm{NOCT}}}\right) \Delta G \tag{7.2}$$

For typical values ($T_{\mathrm{NOCT}} = 45°\mathrm{C}$, $T_{amb,\mathrm{NOCT}} = 20°\mathrm{C}$, $G_{\mathrm{NOCT}} = 800\,\mathrm{W/m^2}$), the proportionality constant is approximately 0.03. Thus, a difference in irradiance of $100\,\mathrm{W/m^2}$ corresponds to a cell temperature variation of approximately 3°C. Given that the proposed approach focuses on small mismatches and the minimal effect of irradiance variation on temperature changes, temperature variations are not included in the model proposed in this paper to improve the computational efficiency. However, it is important to note that such variations can be included in the D-SDM model by increasing the number of parameters to identify, which would consequently raise the computational complexity.

In real applications, measuring the temperatures of all the cells is not practical. Therefore, it is assumed that the temperature can be approximated either as the average of the available temperature sensors in the module [145] or, as discussed earlier, estimated using ambient temperature and irradiance data [144, 146].

From the circuit shown in Fig. 7.3, it is possible to write (7.3) and (7.4), where the voltage and the current of the module are represented by V_m and I_m respectively. In those equations, the SDM parameters of each group are identified by subscripts a and b: $I_{ph\text{-}a}$, $I_{s\text{-}a}$, η_a, $R_{s\text{-}a}$, $R_{sh\text{-}a}$ for group a, and $I_{ph\text{-}b}$, $I_{s\text{-}b}$, η_b, and $R_{s\text{-}b}$ for group b. It can be observed that the currents of groups a and b are equal to the module current, i.e. $I_m = I_a = I_b$; while the relation between V_m and the voltages of groups a (V_a) and b (V_b) is given in (7.5).

$$I_a = I_{ph\text{-}a} - I_{s\text{-}a}\left(\exp\left(\frac{V_a + I_a R_{s\text{-}a}}{\eta_a V_{th\text{-}a}}\right) - 1\right) - \frac{V_a + I_a R_{s\text{-}a}}{R_{sh\text{-}a}} \tag{7.3}$$

$$I_b = I_{ph\text{-}b} - I_{s\text{-}b}\left(\exp\left(\frac{V_b + I_b R_{s\text{-}b}}{\eta_b V_{th\text{-}b}}\right) - 1\right) \tag{7.4}$$

$$V_m = V_a + V_b \tag{7.5}$$

The inverse saturation current and ideality factor do not depend on the number of cells in each group, according to (5.4) and (5.5); hence, they have the same values for groups a and b (see (7.6) and (7.7)). The inverse saturation current is well known to depend on the temperature, but the assumption of the same temperature for all the cells justifies the use of (7.6). Furthermore, the series resistances, parallel resistance, and the thermal voltages of groups a and b depend on the number of cells in each group as introduced in (7.8), (7.9), and (7.10) since those parameters are directly proportional to the number of cells connected in series in each group.

$$I_{s\text{-}c} = I_{s\text{-}a} = I_{s\text{-}b} \tag{7.6}$$

$$\eta_c = \eta_a = \eta_b \tag{7.7}$$

$$R_{s\text{-}c} = \frac{R_{s\text{-}a}}{N_{s\text{-}a}} = \frac{R_{s\text{-}b}}{N_{s\text{-}b}} \tag{7.8}$$

$$R_{sh\text{-}c} = \frac{R_{sh\text{-}a}}{N_{s\text{-}a}} \tag{7.9}$$

$$V_{th\text{-}a} = N_{s\text{-}a} \cdot \frac{\mathrm{k}\, T_c}{\mathrm{q}} \qquad V_{th\text{-}b} = N_{s\text{-}b} \cdot \frac{\mathrm{k}\, T_c}{\mathrm{q}} \tag{7.10}$$

Equations (7.6)–(7.9) incorporate certain assumptions regarding the dependence of parameters on irradiance and temperature. Although experimental studies in the literature do not provide a consensus on the behavior of each model parameter under varying temperature or irradiance levels [99], some assumptions are widely accepted. The saturation current is typically assumed to depend only on temperature [100, 101], while the parallel resistance is generally considered to depend only on irradiance [101,102]. The ideality factor is assumed to be independent of irradiance, while some studies suggest it may depend on temperature [100, 102].

The series resistance is often treated as independent of both irradiance and temperature [101], although some research reports a linear variation of R_s with temperature [103].

Consistent with these prior studies, this work assumes that the saturation current, the ideality factor, and the series resistance are independent of irradiance, a widely supported assumption in the literature [147]. Although these assumptions are adopted here with good results, the proposed D-SDM model could incorporate alternative dependency relationships between the two blocks, which, depending on the PV technology, may lead to improved accuracy. Finally, the parameters $I_{ph\text{-}a}$ and $I_{ph\text{-}b}$ depend on G_a and G_b, respectively. In summary, the D-SDM is represented in Fig. 7.3 and described by (7.3)-(7.5). Considering equations (7.6)-(7.10), the parameters to identify can be deduced:

- Replacing $N_{s\text{-}b} = N_s - N_{s\text{-}a}$ in (7.10)
- Replacing (7.6)-(7.10) in (7.3) and (7.4)

Performing these steps, the D-SDM has seven (7) parameters to be identified that can be defined as follows:

$$\vec{P} = [N_{s\text{-}a}, I_{ph\text{-}a}, I_{ph\text{-}b}, I_{s\text{-}c}, \eta_c, R_{s\text{-}c}, R_{sh\text{-}c}] \tag{7.11}$$

It is important to note that the identified parameters correspond to one cell; therefore, they can be scaled to represent groups a and b by using (5.2)-(5.7) but replacing N_s with $N_{s\text{-}a}$ and $N_{s\text{-}b}$, respectively.

7.4 Proposed method to identify the D-SDM parameters

The identification of $\vec{P}$ can be formulated as an optimization problem by fitting the equations (7.3)-(7.5) to the experimental I-V curves. As widely discussed in the literature, different approaches can be considered for fitting the SDM parameters to experimental data [31,116]. One way is to use the experimental voltage values ($V_m = V_{exp}$) in the model and determine the parameters $\vec{P}$ that best fit the calculated current (I_m) with the experimental current values (I_{exp}), thus requiring

the current to be expressed as a function of voltage. A complementary approach is given if the experimental current values are used in the model ($I_m = I_{exp}$) and parameters are optimized to fit the calculated voltage (V_m) with the corresponding experimental values (V_{exp}). It should be noted that equations (7.3)-(7.4) have an implicit form; therefore, they cannot be directly used to determine the module current (I_m) for a given voltage (V_m) or vice versa, which is a key step in calculating the error between the model and the experimental I-V curve.

As a consequence, the first step is to express the implicit equation (7.3) as an explicit equation by using the *Lambert-W* function [148]. That explicit equation is introduced in (7.12), where $\mathcal{W}(\cdot)$ indicates the principal branch of the *Lambert-W* function. In contrast, V_b can be expressed explicitly from (7.4), as shown in (7.15), thanks to the parallel resistance simplification of group b.

$$\begin{aligned} V_a &= -I_m\left(R_{s\text{-}a} + R_{sh\text{-}a}\right) \\ &\quad + R_{sh\text{-}a}\left(I_{ph\text{-}a} + I_{s\text{-}a}\right) - \eta_a V_{th\text{-}a}\, L_a \end{aligned} \tag{7.12}$$

$$L_a = \mathcal{W}\left\{\frac{I_{s\text{-}a}\ R_{sh\text{-}a}\ \exp\left(\mathcal{S}_a\right)}{\eta_a\ V_{th\text{-}a}}\right\} \tag{7.13}$$

$$\mathcal{S}_a = \frac{R_{sh\text{-}a}(-I_m + I_{ph\text{-}a} + I_{s\text{-}a})}{\eta_a V_{th\text{-}a}} \tag{7.14}$$

$$V_b = \eta_b V_{th\text{-}b} \ln\left(\frac{I_{ph\text{-}b} - I_m}{I_{s\text{-}b}} + 1\right) - I_m R_{s\text{-}b} \tag{7.15}$$

Using Equations (7.12) and (7.15), V_a and V_b can be calculated for a given I_m, followed by the calculation of V_m using (7.5). Hence, it could be possible to fit an I-V curve by defining the fitness function of the optimization problem as a function of the module voltage; in other words, using I_m as the independent variable and V_m as the dependent one. Nevertheless, this fitting approach tends to produce more errors near the short-circuit current of any subblock (i.e., at low voltages) because small errors or variations in I_m lead to significant variations in V_m.

To increase the robustness of the fitting process, a binary search procedure is proposed to determine I_m for a given V_m, with a defined accuracy ϵ. For each experimental voltage point, Algorithm 5 estimates the corresponding current using an iterative approach that leverages the monotonicity of the I-V curve. The inputs

of this algorithm are $V_{exp}(i)$, $I_{exp}(i)$, ϵ, τ, and the parameters of the D-SDM model ($\vec{P}$), where $V_{exp}(i)$ and $I_{exp}(i)$ describe the i-th experimental point of the I-V curve, and τ is a constant greater than 1 that defines the maximum value of I_m for the initial search range. The output of Algorithm 5 is $I_m(i)$, which is the calculated current of the PV module corresponding to $V_m(i) = V_{exp}(i)$; a brief description of how it works is reported below.

The algorithm begins by defining an initial search range, i.e. $I_{min} < I_m < I_{max}$, where $I_{max} = \tau I_{exp}$. Then, it calculates the midpoint between I_{min} and I_{max} (I_{mid}), which is used to determine V_a and V_b for $I_m = I_{mid}$ by using (7.12) and (7.15) as well as V_m by evaluating (7.5). After, $V_m > V_{exp}(i)$ means that $I_{mid} > I_{exp}(i)$; then, the search range of $I_m(i)$ can be reduced by setting $I_{min} = I_{mid}$; otherwise, the search range can be restricted by making $I_{max} = I_{mid}$. It is important to note that this algorithm is valid because the I–V curve is always decreasing.

Algorithm 5 D-SDM: Binary search for finding $I_m(i)$

Inputs: $V_{exp}(i)$, $I_{exp}(i)$, ϵ, τ, and $\vec{P}$
Output: $I_m(i)$
for i=1:N **do**
    $I_{min} = 0, I_{max} = \tau I_{exp}[i]$
    while $I_{max} - I_{min} > \epsilon$ **do**
        $I_{mid} = \frac{I_{max}+I_{min}}{2}$
        Calculate V_a and V_b for I_{mid} using (7.12) and (7.15)
        $V_m = V_a + V_b$
        if $V_m > V_{exp}(i)$ **then**
            $I_{min} = I_{mid}$
        else
            $I_{max} = I_{mid}$
        end if
    end while
    $I_m(i) = \frac{I_{max}+I_{min}}{2}$
end for

7.4.1 Error calculation

At this point, it is possible to calculate the currents of an experimental I-V curve with N points by using D-SDM for a given set of parameters $\vec{P}$. The experimental I-V curve is described by $V_{exp}(i)$ and $I_{exp}(i)$ with $i \in [1, \cdots, N]$; then, the D-SDM currents ($I_m(i)$) are calculated by executing Algorithm 5 for each value of $V_{exp}(i)$.

To enhance the robustness of the error calculation for a given set of parameters $\vec{P}$, only experimental points where the validity of the D-SDM is satisfied are considered; i.e. all the cells operate at positive voltages. In other words, the error calculation only considers the section of the I-V curve where V_a and V_b are positive. More formally, if the experimental I-V points are sorted in decreasing order by their voltage values:

$$V_{\text{exp}}(i) > V_{\text{exp}}(i+1) \quad \text{for} \quad i \in [1, \ldots, N-1],$$

then, the error calculation considers M points, where M is the maximum index such that the point $V_{\text{exp}}(M+1)$ fails to meet the validity criteria $V_a > 0\ V$ and $V_b > 0\ V$.

Given this rule for selecting valid points on the I-V curve, the error function must be carefully defined. It should prioritize including as many points as possible from the I-V curve while minimizing the discrepancies between the experimental and calculated curves. Then, the error function is defined based on the relative area between these two curves. The error is computed using the trapezoidal rule, where the voltage difference between consecutive points is defined as:

$$\Delta V(i) = V_{\text{exp}}(i+1) - V_{\text{exp}}(i)$$

The error, denoted by $\mathcal{E}$, is then calculated as the ratio of the area between the experimental and estimated curves to the area under the experimental curve:

$$\mathcal{E} = \frac{\sum_{i=1}^{M-1} \left[\frac{|I_{\text{exp}}(i) - I_m(i)| + |I_{\text{exp}}(i+1) - I_m(i+1)|}{2} \right] \Delta V(i)}{\sum_{i=1}^{M-1} \left[\frac{I_{\text{exp}}(i) + I_{\text{exp}}(i+1)}{2} \right] \Delta V(i)} \tag{7.16}$$

It is worth noting that M must be identified each time (7.16) is evaluated since it depends on the parameters defined in $\vec{P}$ (7.11).

7.4.2 Optimization method

Differential evolution algorithm (DE) [149] is selected to find the values of $\vec{P}$ that minimizes $\mathcal{E}$. Particularly, the implemented variant is *DE/best/1/bin* using the Python framework Pymoo [128]. Moreover, the algorithm parameters are selected by attending the guidelines given in [150]: scale factor $F = 0.8$, and crossover rate $Cr = 0.9$, which helps to optimize non-separable functions.

Given the integer nature of the parameter $N_{s\text{-}a}$, this paper evaluates two variants of the optimization method. The first, named **D-SDM**, treats $N_{s\text{-}a}$ as a real parameter, while the second, named **D-SDM-I** considers it as an integer value. DE is inherently a global optimization algorithm for real-encoded variables [149]; therefore, additional customization is required to handle integer variables. In the first methodology, where $N_{s\text{-}a}$ is treated as a real variable, the standard DE is employed with a population size of $NP = 40$, running the algorithm until convergence.

In the second methodology, where $N_{s\text{-}a}$ is treated as an integer parameter, a population of $NP = 40$ individuals evolves until convergence, with $N_{s\text{-}a}$ considered as a real parameter, as described previously. After convergence, the final population is expanded by adding 5 random individuals, resulting in a new population of $NP = 45$. This expanded population then evolves again until convergence, but this time an integer repair operator is applied to each individual in every generation before evaluation. The idea behind this approach is to enhance exploration in the early stages of evolution by allowing $N_{s\text{-}a}$ to vary without constraints. Once the initial population has converged, diversity is increased by introducing random solutions. The subsequent evolution with the integer constraint on $N_{s\text{-}a}$ ensures that the final optimal solution includes $N_{s\text{-}a}$ as an integer.

7.5 Validation Results

The validation of the proposed identification procedure considers the two variants introduced before **D-SDM** and **D-SDM-I**. The traditional **SDM** under uniform assumption is used as a baseline comparison model (see (5.1)-(5.7)). From this point forward, this parameter identification approach will be referred to as **SDM**. For a fair comparison with **SDM**, it is necessary to calculate the minimum

voltage V_{low}, where $V_{low} < V_{mpp}$, in such a way that the portion of the experimental curve from V_{low} to V_{oc} does not contain any activation of the bypass diode. Such activation is reflected in a drastic change in the first derivative of the I-V curve, which would lead to larger errors in the **SDM** parameters estimation. Then, the experimental points are considered from V_{low} to V_{oc} used for the fittings.

In summary, for each experimental I-V curve, the best-fitting parameters are determined using the compared procedures (**D-SDM**, **D-SDM-I**, and **SDM**). Section 7.4 provides details on the implementation of the **D-SDM** and **D-SDM-I** parameter identification procedures. The traditional **SDM** procedure follows the model equations (5.1)-(5.7) and employs the same optimization algorithm to compute the best-fitting parameters. The estimated parameters are then compared based on metrics such as stability under similar operating conditions to assess the practical relevance of each procedure. Additionally, the **D-SDM** procedure is evaluated for its accuracy in identifying the true model parameters using artificially generated I–V curves designed to represent a wide range of PV technologies.

7.5.1 Experimental campaign

The experimental campaign was designed to test the identification procedures under common mismatching phenomena that may occur in real applications in outdoor conditions. All experimental I–V curves were measured on a commercial PV module manufactured by *Solbian* with reference *Flex–SP50L* [131] (its specifications under STC are given in Table 6.1). It comprises 16 pseudo square back–contact mono–crystalline Silicon cells, all connected in series, and a bypass diode. A configuration of two *Flex–SP50L* series-connected modules is considered in our experimental campaign with three Pt100 RTD sensors attached to each module to measure the average temperature. A solarimeter LM1-10V PRO [151] was used for measuring the irradiance $G[W/m^2]$ on the plane of each solar module. The temperature and irradiance sensors were connected to ADAM 6015 and ADAM 6017 [152] to save the data in a computer.

Four common mismatch cases are analyzed as illustrated in Fig. 7.6, which are described below:

- Case (I) closely resembles a scenario with uniform conditions. In this case, the primary cause of mismatch is the non-uniform radiation caused by the

reflexes on the surface of the module. While the I–V curve shows only minor deviations from the uniform condition, these deviations impact the accuracy of parameter identification in models that assume fully uniform conditions (see Fig. 7.1 and Fig. 7.7 (Case I)).

- Case (II) represents partial shading caused by surrounding objects. In this scenario, the deformation of the I–V curve is noticeable, as shown in Fig. 7.7 (Case II). Fitting only the rightmost portion of the curve from V_{low} to V_{oc} (see definition of V_{low} at the beginning of Section 7.5), using a model that assumes uniform conditions will not be accurate. This is due to the increased curvature around V_{mpp}, which requires the model to have more degrees of freedom to capture the shape effectively [133].

- Case (III) simulates the mismatch effect caused by bird droppings. A small deviation from the uniform condition is evident in the I–V curve, as shown in Fig. 7.7 (Case III), which is sufficient to challenge traditional parameter identification procedures.

- Case (IV) represents scenarios where dust accumulates and rain redistributes it across the module surface, creating a non-uniform shading pattern. The deformation in the I–V curve is evident, as shown in Fig. 7.7 (Case IV), but may not be easily noticeable if only the rightmost portion of the curve is considered.

Several I–V curves were measured for each representative mismatch scenario under outdoor conditions. The objective is to assess the capabilities of the proposed identification procedures to obtain consistent parameter estimates under various mismatching conditions, which are introduced in Table 7.2. This table also includes the fitting error for each case obtained with (7.16).

7.5.2 Overview of D-SDM parameter estimates

Section 7.2 highlighted the limitations of the uniformity assumption to fit an I-V curve of a PV module under mismatching conditions. Figure 7.8 extends Fig. 7.1 by including the fitting of the **D-SDM** for the I-V curve under study and considering two cases for the SDM (under uniformity assumptions), the first one

(I) (II) (III) (IV)

Figure 7.6: Mismatching cases analyzed. (I). Clean panel without shadows but with non-uniform radiation produced by the reflexes on the surface; (II). Partial shading by surrounding objects; (III). Partial shading by bird droppings; (IV). Partial shading by soiling.

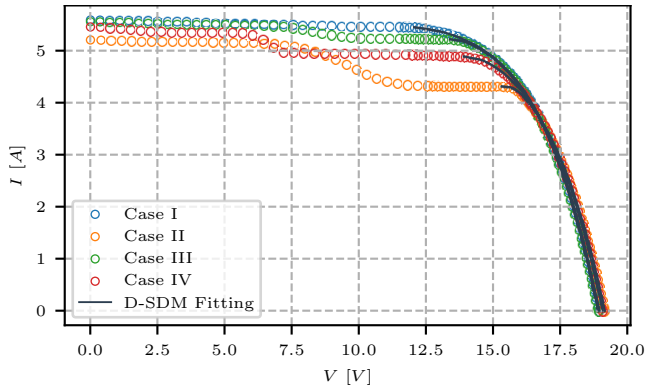


Figure 7.7: Example of fitting results for experimental I–V curves. I. Uniform Conditions; II. Partial shading by surrounding objects; III. Partial shading by bird droppings; I–V. Partial shading by soiling

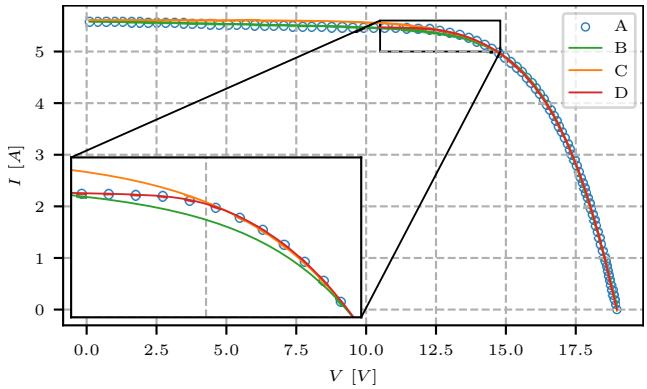


Figure 7.8: Uniform assumption limitations. A. Experimental data; B. SDM fitting using all the experimental points, C. SDM fitting using only the points with voltages between $V_{mpp} - 0.2V_{mpp}$ to V_{oc}, D. D-SDM fitting using all the experimental points.

using all the experimental data (line B in Fig. 7.8) and the second one considering the experimental points from $V_{low} = V_{mpp} - 0.2V_{mpp}$ to V_{oc} (line C in Fig. 7.8). In the first case, the SDM parameters fail to fit the I-V curve around the MPP, while in the second case, the parameters fail to fit the I-V curve for low voltages. In contrast, the **D-SDM** effectively follows the I-V curve around the MPP and for high voltages.

Regarding the results shown in Fig. 7.7 and Table 7.2, it can be observed that the two variants proposed fit significantly better than the **SDM**. The fitting quality is further supported by the data in Table 7.2, where the **D-SDM**-based identification procedures consistently exhibit errors with one or two orders of magnitude lower than the ones of the **SDM** for all the mismatching cases. Additionally, the proposed identification procedure automatically identifies the range where all the cells operate at positive voltages for the fitting process.

Table 7.2 also confirms the stability of the estimated series resistance across different mismatching patterns. It is well-known that the series resistance is a crucial parameter in determining aging and potential faults in the modules; thus, a reliable parameter estimation method should be able to detect stable and accurate, variations in this parameter. In contrast, **SDM** predictions vary significantly from Case I to Case II, III, and IV. Another key parameter representing losses is the parallel resistance. The **SDM** was unable to estimate this parameter in the mismatching cases analyzed, as $R_{sh\text{-}c}$ consistently converged to the upper bound

Table 7.2: Best fitting parameters of a single cell for D-SDM, D-SDM-I, and SDM

Parameter	Case I			Case II			Case III			Case IV		
	D-SDM	D-SDM-I	SDM	D-SDM	D-SDM-I	SDM	D-SDM	D-SDM-I	SDM	D-SDM	D-SDM-I	SDM
$I_{ph\text{-}a}[A]$	5.44	5.48	5.61	4.31	4.31	4.39	5.22	5.22	5.35	4.88	4.87	4.98
$I_{s\text{-}c}[A]$	1.41×10^{-4}	1.56×10^{-4}	9.51×10^{-5}	8.88×10^{-5}	5.64×10^{-5}	2.69×10^{-9}	1.30×10^{-4}	1.25×10^{-4}	3.16×10^{-6}	5.84×10^{-5}	5.40×10^{-5}	1.59×10^{-7}
$\eta_c[-]$	1.96	1.98	1.89	1.93	1.85	1.00	1.92	1.92	1.43	1.84	1.83	1.22
$R_{s\text{-}c}[\Omega]$	0.0024	0.0023	0.0028	0.0025	0.0028	0.0068	0.0025	0.0026	0.0055	0.0029	0.0029	0.0069
$R_{sh\text{-}c}[\Omega]$	6.66	11.40	∞	27.81	30.34	∞	16.11	17.55	∞	6.53	7.06	∞
$I_{ph\text{-}b}[A]$	5.65	5.69	-	5.31	5.26	-	5.65	5.64	-	5.36	5.34	-
$N_{s\text{-}a}[-]$	1.00	4	-	1.00	1	-	1.00	1	-	1.18	1	-
Error $\mathcal{E}$	2.12×10^{-4}	2.21×10^{-4}	1.01×10^{-3}	2.96×10^{-4}	2.85×10^{-4}	1.36×10^{-2}	2.48×10^{-4}	2.48×10^{-4}	6.18×10^{-3}	1.88×10^{-4}	2.04×10^{-4}	5.99×10^{-3}

of the search interval. In contrast, both **D-SDM** and **D-SDM-I** provided reasonable estimations for this parameter across all mismatching cases. It is important to note that the resistance values in the results are scaled to a single cell. It is also important to note, that the results show minimal differences in general between the parameters estimated by **D-SDM** and **D-SDM-I**.

7.5.3 Assessing the Stability of Parameter Estimates

This section evaluates the ability to obtain stable and reliable parameters for different partial shading patterns across the compared procedures. Since the SDM parameters are known to fluctuate based on operating conditions, I-V curves measured under similar conditions for each shading pattern are selected (See Fig. 7.6). In this study, similar conditions are defined as those where the irradiance varies by no more than $15W/m^2$ and the temperature by no more than 5 K across all curves. It is important to note that the I-V curves were measured in outdoor conditions, making this approximation necessary.

Figure 7.9 compares the coefficient of variation (CV) for the estimated series resistance R_s and the parallel resistance R_{sh} across the compared models and shading patterns. The coefficient of variation is a standardized measure of dispersion, defined as the ratio of the standard deviation σ to the mean value μ, that is $\text{CV} = \frac{\sigma}{\mu}$. This dimensionless metric allows for evaluating the relative variability of model parameters regardless of their scale or units. A consistent and reliable identification procedure should yield similar parameter estimates under similar operating conditions, resulting in a low CV.

Figure 7.9 highlights the advantages of the proposed identification procedures over **SDM**. For the series resistance R_s, **SDM** achieves a lower coefficient of variation (CV) only in the first mismatching pattern, which closely resembles

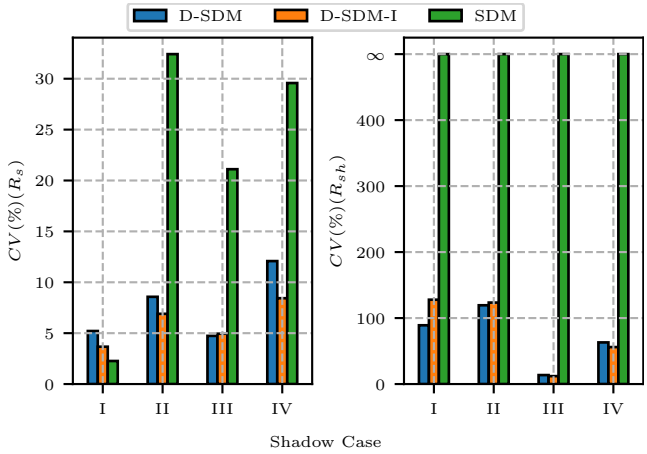


Figure 7.9: Comparison of coefficient of variation for R_s and R_{sh}

uniform conditions. However, **SDM** struggles to estimate the parallel resistance R_{sh} accurately. In all mismatching cases, **SDM** estimation of R_{sh} converges to the boundary of the search interval, which is consistent with the limitations previously illustrated in Fig. 7.1.

Figure 7.10 presents the CV for the saturation current I_s and the ideality factor η across various mismatching cases. The proposed identification procedures consistently yield a significantly lower CV for both parameters, particularly in shading scenarios from cases II to IV. Notably, the accuracy of the **SDM** estimations deteriorates for the more complex mismatching patterns, which is evident in cases II through IV. A comparison of the two procedures proposed in this paper shows that **D-SDM-I** generally achieves a slightly lower CV than **D-SDM** (see Figures 7.9 and 7.10). However, the computational cost for obtaining the parameters with **D-SDM-I** is nearly twice higher than for **D-SDM** (see Section 7.4.2). Therefore, the choice of a procedure for practical applications will depend on the computational resources available on the hardware in use.

7.5.4 Validation through simulated *I*–*V* curves

This section aims to validate the identification procedure by using artificially created *I*–*V* curves that are expected to cover a wide range of PV technologies under a wide range of partial shading conditions. To numerically validate the proposed simplification of neglecting the parallel resistance of the group at higher irradiance

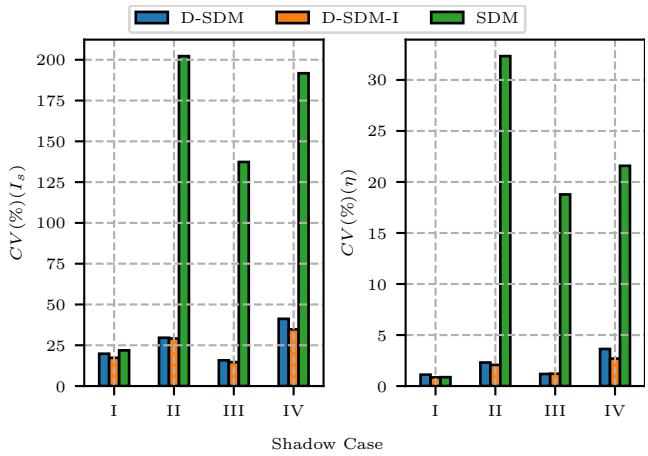


Figure 7.10: Comparison of coefficient of variation for I_s and η

(group b), the effect of this parallel resistance is incorporated into the simulated curves. This allowed us to evaluate the ability of the model to estimate the parallel resistance for the low irradiance block (group a).

To simulate the effect of partial shading conditions, the Bishop model [134] is used to replicate the behavior of the cells when working in the first and second quadrants of the I-V curve. According to this model, I_c is calculated by using (7.17), where a and m_{Bi} are fitting parameters and V_{br} is the reverse breakdown voltage of the cell.

$$\begin{aligned} I_c &= I_{ph\text{-}c} - I_d - I_{sh} \\ I_d &= I_{s\text{-}c}\left(\exp\left(\frac{V_c + I_c R_{s\text{-}c}}{\eta_c V_{th\text{-}c}}\right) - 1\right) \\ I_{sh} &= \frac{V_c + I_c R_{s\text{-}c}}{R_{sh}}\left[1 + a\left(1 - \frac{V_c + I_c R_{s\text{-}c}}{V_{br}}\right)^{-m_{Bi}}\right] \end{aligned} \tag{7.17}$$

In the simulation, the I-V curve of the cell at low irradiance is calculated from (7.17), then it is scaled by $N_{s\text{-}a}$ to obtain the I-V curve of group a and a similar procedure is performed for group b. Then, the module I-V curve is obtained by calculating V_a and V_b for the same values of I_m, and finally, V_m is just the sum of V_a and V_b as defined in (7.5).

The parallel resistance behavior exhibits an exponential trend as described in (7.18) [31], which shows that $R_{sh\text{-}b}$ can be defined in terms of $R_{sh\text{-}a}$ and ν. In a previous work [97], the authors showed that ν varies from 0.7 to 1.31 for different PV technologies.

$$R_{sh\text{-}b} = R_{sh\text{-}a} \cdot \left(\frac{N_{s\text{-}b}}{N_{s\text{-}a}}\right) \cdot \left(\frac{I_{ph\text{-}b}}{I_{ph\text{-}a}}\right)^{-\nu} \tag{7.18}$$

Considering this, a summary of the range defined for each parameter to generate the simulated curves is shown in Table 7.3. The set of parameters for each curve is generated randomly from a uniform distribution. For each simulated curve, the parameters of the **D-SDM** procedure are estimated and compared with the true parameters used for its generation.

Table 7.3: Used ranges for generating simulated curves

Parameter	**Lower Limit**	**Upper Limit**
$N_s[-]$	16	120
$N_{s\text{-}a}[-]$	1	N_s
$I_{ph\text{-}b}$[A]	2	12
$I_{ph\text{-}a}$[A]	2	$I_{ph\text{-}b}$
$I_{s\text{-}c}$[A]	10^{-12}	10^{-5}
$\eta_c[-]$	1	2
$R_{s\text{-}c}[\Omega]$	0.001	0.01
$R_{sh\text{-}c}[\Omega]$	10	50
$\nu[-]$	0.7	1.31
$T_c[K]$	273	343
$a[-]$	$2 \times 10^{-}3$	$15 \times 10^{-}3$
$m_{Bi}[-]$	2	6
$V_{br}[V]$	-40	-20

Figure 7.11 compares the true series resistance with the estimated values from the proposed procedure on the left and the true parallel resistance with the calculated values on the right. In both comparisons, the high R^2 coefficients demonstrate the method's effectiveness in accurately determining the series and parallel resistances. Additionally, these results confirm that simplifying the model by neglecting the parallel resistance at high irradiance does not compromise the accuracy of the estimated parallel resistance at low irradiance. This numerical evidence supports the model simplification, which reduces the number of parameters while maintaining high accuracy.

Figure 7.12 assesses the proposed method's ability to detect mismatch levels.

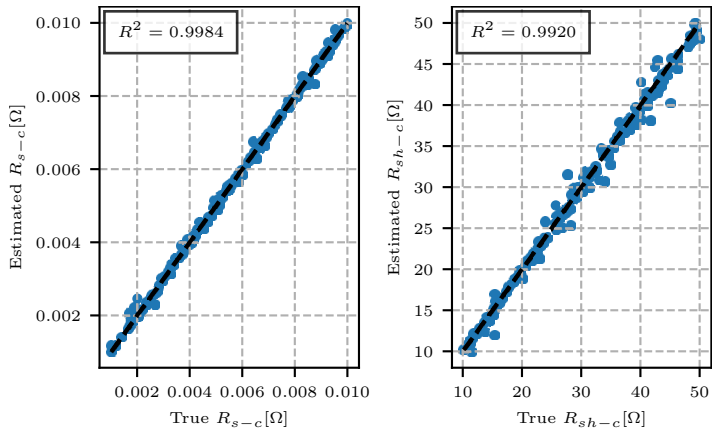


Figure 7.11: True value versus estimation for $R_{s\text{-}c}$ (left) and $R_{sh\text{-}c}$ (right)

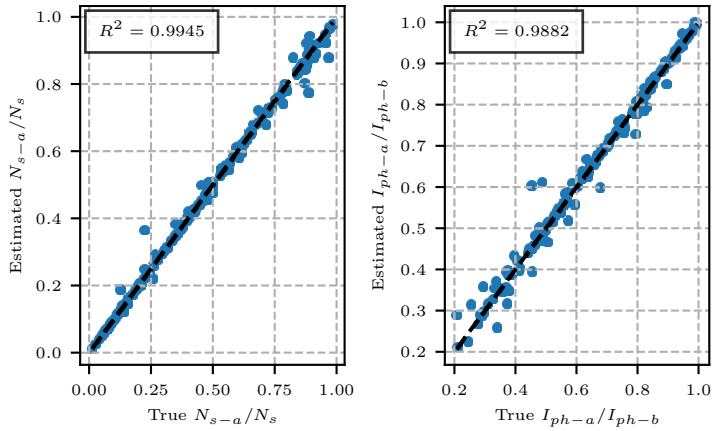


Figure 7.12: True value versus estimation for $N_{s\text{-}a}/N_s$ (left) and $I_{ph\text{-}a}/I_{ph\text{-}b}$ (right)

The left panel compares the ratio of shaded cells to the total number of cells. In contrast, the right panel compares the ratio of the photogenerated current at low irradiance to that at high irradiance. In this case, the high R^2 coefficient further demonstrates the effectiveness of the model in detecting mismatch levels.

7.6 Detection of series resistance degradation

The aging of PV modules is typically reflected in an increase in series resistance [153], making it a crucial parameter for PV diagnosis [154]. This section aims to evaluate the capability of the D-SDM parameter identification procedure to detect series resistance degradation. We compare the results with the traditional SDM model ((5.2)-(5.7)) and the dynamic CPE model (Eq. (5.8)) to identify the advantages and disadvantages of each parameter identification method. Although the meaning of the series resistance appearing in the static and dynamic models is not exactly equivalent, both refer to ohmic losses and therefore can be used to detect degradation phenomena [121].

7.6.1 Parameter estimation of the SDM

To estimate the parameters of the SDM, the error function is defined as the root mean squared error between the experimental ($I_{exp}(i)$) and calculated ($I_c(i)$) current for each of the N experimental voltage values (Eq. 6.2). An optimization problem is formulated to minimize the error in (6.2). The parameters are estimated using the DE optimization algorithm [149]. Specifically, the implemented variant is *DE/best/1/bin*, executed using the Python framework Pymoo [128]. The algorithm parameters are chosen based on the guidelines in [150], with a scale factor of $F = 0.8$ and a crossover rate of $Cr = 0.9$, which enhance the optimization of non-separable functions.

7.6.2 Parameter estimation of the CPE model

In this case, the error function measures the difference between the experimental and calculated impedances. Equation (5.8) is used to evaluate the real ($Z'_{i,cal}$) and the imaginary ($Z''_{i,cal}$) parts of the PV impedance. Then the model parameters are selected to minimize the sum of weighted square differences between

the experimental and calculated impedances (Eq. 6.3). In equation (6.3), the summation includes all impedance values measured at L frequencies in the experiments. This equation follows the modulus weighting approach, ensuring that both small and large impedance values contribute equally to the sum of squares [122]. The optimization method is specified as *DE/best/1/bin* with $F = 0.8$ and $Cr = 0.9$ to maintain a fair comparison.

7.6.3 Experimental Campaign

All experimental I–V curves were measured on a commercial PV module, the *Flex–SP50L*, manufactured by *Solbian* [131]. Its specifications under STC are provided in [31]. The module consists of 16 pseudo-square back-contact monocrystalline silicon cells connected in series, along with a bypass diode.

A series-connected resistor of nominal value $\Delta R_{s\text{-}m} = 0.175$ is used to simulate series resistance degradation. The experimental data spans a range of irradiance levels, from low to high, under uniform irradiance and temperature conditions. Two datasets with similar operating conditions are analyzed: the first corresponds to the non-degraded state without the additional resistor, while the second represents the degraded state with the additional resistor.

7.6.4 Overview of the fitting for each analyzed model

The fitting overview analyzes the performance of each model in estimating the series resistance ($R_{s\text{-}m}$) under 3 selected cases at low, medium, and high irradiance. Table 7.4 presents the temperature (T) and irradiance (G) for each analyzed case, while Table 7.5 provides the obtained resistance values and the associated fitting error ($\mathcal{E}$) for the compared methods.

Table 7.4: Operating Conditions for Each Selected Case

Case	$T[K]$	$G[W/m^2]$
I	302.08	482.05
II	305.39	692.70
III	309.64	876.71

The results indicate differences in the estimated series resistance values among the models. The SDM consistently provides higher $R_{s\text{-}m}$ values compared

Table 7.5: Series Resistance and Fitting Error for Each Model

Case	SDM		D-SDM		CPE	
	$R_{s\text{-}m}$	$\mathcal{E}$	$R_{s\text{-}m}$	$\mathcal{E}$	$R_{s\text{-}m}$	$\mathcal{E}$
I	0.157	0.0107	0.0394	0.0002	0.1654	0.0173
II	0.0607	0.0087	0.0364	0.0006	0.1542	0.0565
III	0.056	0.0082	0.0395	0.0002	0.1441	0.0183

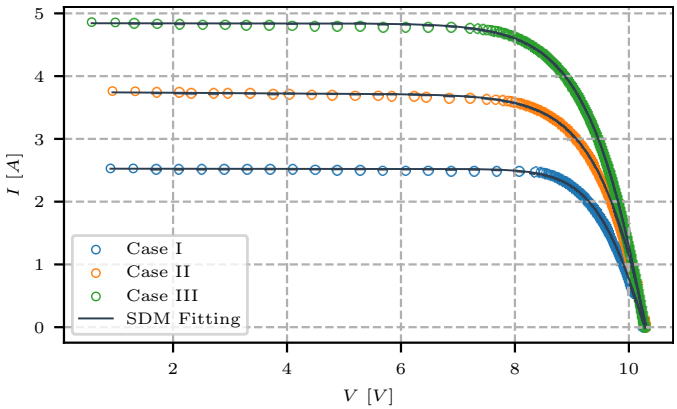


Figure 7.13: Fitting results for the Single Diode Model (SDM) [32].

to the D-SDM, with values decreasing as irradiance increases. This trend suggests that SDM may overestimate resistive losses at lower irradiance levels. The CPE model also yields higher resistance values than D-SDM, with values decreasing as irradiance increases.

The D-SDM estimates the lowest series resistance in all cases, with values ranging from 0.0389 Ω to 0.0391 Ω, which remain relatively stable despite variations in operating conditions. This stability suggests that the D-SDM provides a more consistent estimation of the internal resistance.

Since the fitting error values ($\mathcal{E}$) are obtained from different models and methodologies, they are not directly comparable. However, Figure 7.13 shows the fitting results for the SDM. While the model provides a reasonable approximation of the experimental data, a larger error is observed around the Maximum Power Point (MPP) for each case. This result is consistent with the results presented in [30].

The results for the D-SDM are presented in Figure 7.14. The figure demonstrates a strong agreement between the modeled and measured data, with an excel-

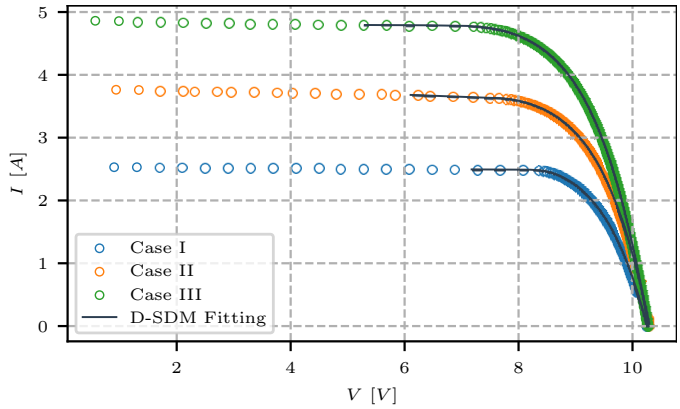


Figure 7.14: Fitting results for the Double Single Diode Model (D-SDM) [32].

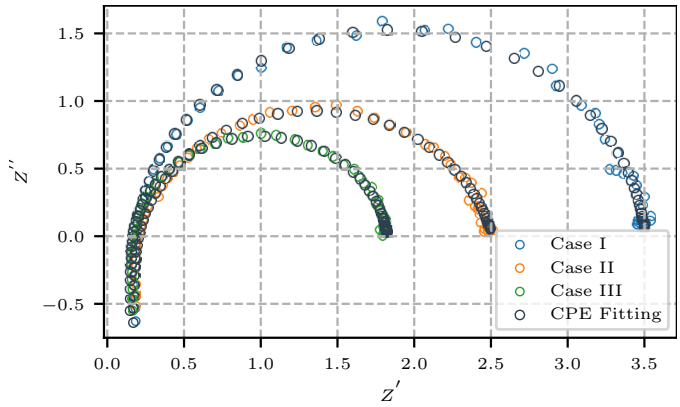


Figure 7.15: Fitting results for the Constant Phase Element (CPE) model [32].

lent match in the region where the model detects that all cells operate at positive voltages [30].

Figure 7.15 illustrates the fitting results for the CPE model. Similar to the D-SDM, the CPE model exhibits a good agreement with the experimental impedance data. This suggests that the CPE model can effectively characterize the system's behavior.

7.6.5 Estimation of the degradation

Figure 7.16 shows the series resistance identified by the SDM, D-SDM, and CPE models across a range of irradiance values. The SDM model starts with the highest series resistance at low irradiance levels but shows a significant decrease as irradiance increases. This suggests a strong dependence of its resistance estimation on irradiance, making it less stable.

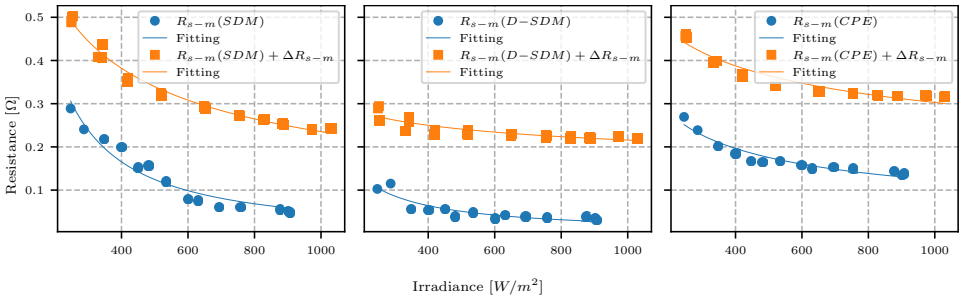


Figure 7.16: Estimation of series resistance degradation by the SDM, D-SDM, and CPE models as a function of irradiance [32].

The D-SDM model estimates the lowest series resistance and exhibits a more stable trend, with resistance values always decreasing with irradiance but at a lower rate. The CPE model follows an intermediate behavior, capturing degradation effects but showing a different dependence on irradiance. Despite these differences, all three models detect an increase in resistance when the additional resistor is introduced. However, the way they interpret this degradation varies, with SDM exhibiting stronger fluctuations, while D-SDM remains the most stable across the irradiance range.

Figure 7.17 presents the relative error in estimating ΔR_s. The SDM model exhibits the highest fluctuations, with errors reaching up to 15%, particularly at lower irradiance levels. It transitions from overestimation at low irradiance to underestimation at high irradiance, making its degradation estimation less reliable.

The D-SDM model maintains lower error values, remaining stable across the entire irradiance range, with a small underestimation at lower irradiance levels and an overestimation at higher irradiance levels. The CPE model also shows low error values and follows a transition from overestimation at low irradiance to underestimation at high irradiance, similar to SDM but with a more stable trend. Unlike SDM, both D-SDM and CPE do not exhibit drastic fluctuations and provide a more balanced estimation of degradation.

7.7 Conclusion of the Chapter

This study presents a 7-parameter self-adapting Double Single-Diode Model (D-SDM) designed to improve the reliability of PV module parameter identification

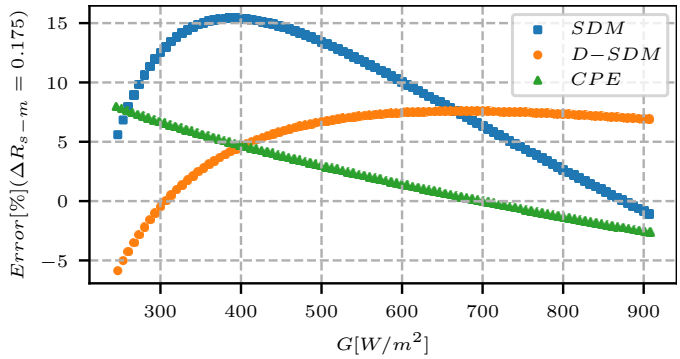


Figure 7.17: Relative error in the estimation of ΔR_s for the SDM, D-SDM, and CPE models as a function of irradiance [32].

under mismatching conditions. A methodology for extracting the parameters of the D-SDM using the Differential Evolution Algorithm and Binary Search is also proposed. This approach reformulates the fitting problem as a function of voltage, enhancing the overall accuracy of the parameter estimation process. A robust error function was defined, considering only the experimental points where all cells operate at positive voltages.

Two variants of the D-SDM were validated, the first treating the number of shaded cells as a real parameter, while the second considering it an integer parameter. Experimental and simulated I–V curves under several mismatching patterns were used to evaluate the two proposed variants. The experimental results revealed that the D-SDM-based methods offer greater stability and reliability in parameter estimation compared to the standard SDM under uniform operating conditions assumption. Additionally, simulated I–V curves highlighted the methodology's effectiveness in accurately determining the true values of the model parameters.

The proposed D-SDM was compared with the standard SDM and with the CPE dynamic model to assess their ability to detect series resistance degradation in PV modules under uniform conditions. To simulate aging, an additional resistor was introduced during the experimental campaign, and each method was evaluated for its ability to detect this change. All three methods identified an increase in resistance, but their performance differed. The SDM showed significant fluctuations in resistance estimation across different irradiance levels, reducing its reliability for long-term monitoring. In contrast, the D-SDM and the CPE model provided more reliable estimations across operating conditions.

Part IV

Integrated Demand-Supply Energy Intelligence and Business Model

8

Practical deployment of O2RE-NILM and D-SDM PV parameter identification in a Unified Platform

8.1 Introduction

This chapter presents the practical implementation of the two novel frameworks developed in this thesis: O2RE-NILM, introduced in Chapter 4, and D-SDM PV parameter identification introduced in Chapter 7. Both methods are integrated into a unified embedded system based on a Raspberry Pi 4 running a home automation framework named "Home Assistant". The implementation aims to validate the real-time operation of the proposed methods in a realistic residential environment, demonstrate their compatibility within a unified architecture, highlight the simplicity of installation, configuration, and use, and evaluate their performance under the computational limitations of low-cost embedded hardware.

A home automation platform is selected as the testing environment to provide a practical and user-oriented interface for deploying and evaluating the proposed NILM and PV parameter identification methods. This environment enables continuous system operation, real-time data processing, and immediate visualization of results, which are crucial for evaluating the feasibility and usability of the developed solutions. Integrating the methods into a platform already familiar to residential users enhances the realism of the testing scenario and aligns the imple-

mentation with real-world conditions.

Home Assistant (`https://www.home-assistant.io/`) is chosen as the software platform due to its open-source nature, wide compatibility with smart home devices, and strong community support. It enables seamless integration of custom modules through its modular architecture, allowing the NILM and PV components developed in this work to operate as part of a cohesive home energy management system. Home Assistant supports structured visualization of outputs through dashboards and virtual sensors, which facilitates monitoring and interpretation of the algorithms' results. Its interface can be accessed through any web browser, providing a flexible and accessible control layer for users.

The decision to use Home Assistant also reflects a broader aim of reducing the barriers to adoption for advanced energy monitoring tools. Users who already have Home Assistant deployed in their homes can integrate the NILM and PV methods without additional hardware. This compatibility is significant, as it minimizes installation effort and cost, making the system more accessible. Moreover, Home Assistant's emphasis on privacy and local processing ensures that all data remains within the household network, avoiding dependence on external cloud services and improving system autonomy.

The hardware platform selected for this implementation is the Raspberry Pi, specifically the 4 model, chosen for its balance of affordability, performance, and compatibility with Home Assistant. These devices offer sufficient processing capacity to support real-time execution of the NILM and PV parameter identification methods while maintaining low energy consumption. Their use reflects the practical constraints of residential settings, where cost and space considerations make high-performance or dedicated hardware impractical.

The Raspberry Pi's built-in networking capabilities, general-purpose input/output interfaces, and compatibility with Linux-based systems make it an ideal choice for embedded energy monitoring applications. Hosting Home Assistant directly on the Raspberry Pi simplifies the system architecture and avoids the need for external servers or infrastructure. This integration ensures that all operations, from data acquisition to result visualization, are performed locally, which enhances system responsiveness and privacy.

Although this work focuses on the Raspberry Pi 4 model, the architecture

is not limited to these specific devices. The modularity and portability of the software components allow deployment on any computing unit supported by Home Assistant, offering flexibility for users with different hardware preferences or performance requirements. This adaptability broadens the relevance of the system across a range of household configurations and user needs.

Running both NILM and PV parameter identification on the same embedded system highlights the potential for a consolidated, decentralized energy management solution. Sharing the hardware and software infrastructure reduces overall system complexity and cost. The unified setup demonstrates how the two methods can operate concurrently within the same home automation environment, making the approach suitable for integration into existing smart home systems. Moreover, the modular design also enables future expansion, allowing additional diagnostic, control, and optimization algorithms to be incorporated within the same system framework.

8.2 Home Assistant Installation and Initial Setup

The initial step to replicate our experimental prototype is selecting and installing the appropriate operating system for your hardware. Figure 8.1 illustrates the Raspberry Pi Imager utility, a user-friendly tool designed to simplify the installation of different operating systems on a Raspberry Pi. Highlighted within this tool is the option to install Home Assistant. Whether you are configuring a Raspberry Pi 4 or a Raspberry Pi 3, the Imager provides a straightforward path to embedding Home Assistant into your hardware. We then connected the Raspberry Pi to a router (Figure 8.2), allowing us to connect devices such as phones, tablets, and laptops to the Wi-Fi created by the router. This enables us to visualize and control the Home Assistant running environment.

Add-ons Installation in Home Assistant

Home Assistant supports a streamlined process for installing add-ons directly from custom or official repositories. To install an add-on, users simply need to add the repository URL to the Home Assistant Supervisor interface. Once the repository is added, available add-ons appear in the interface and can be installed

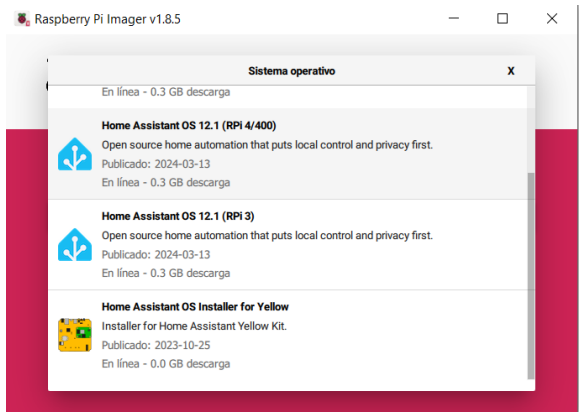


Figure 8.1: Installation of Home Assistant OS for a Raspberry Pi

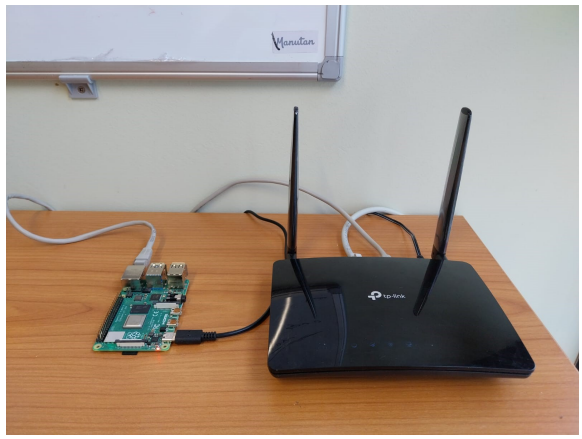


Figure 8.2: Raspberry Pi unit connected to a router, serving as a central hub for smart home automation and energy monitoring tasks.

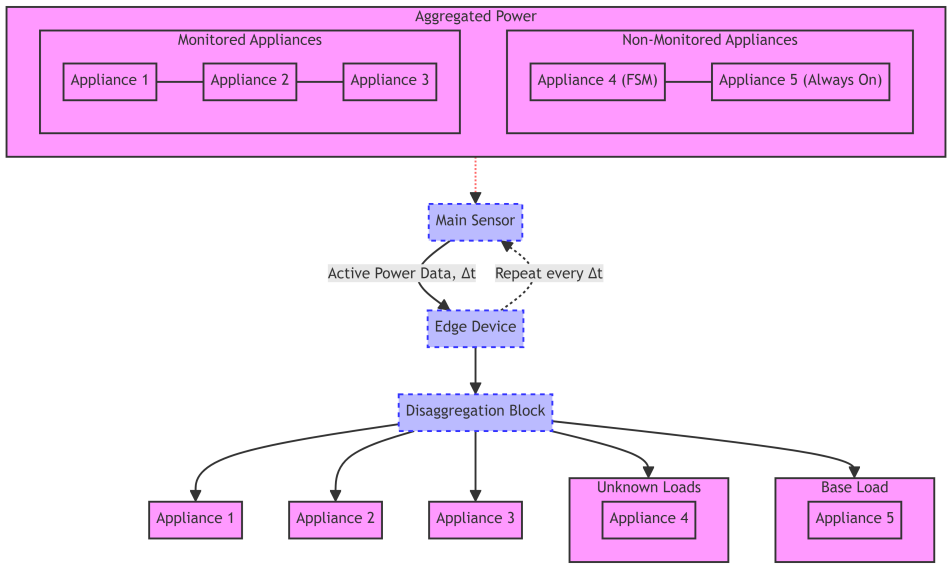


Figure 8.3: Non-Intrusive Load Monitoring (NILM) Process: From Aggregated Power to Appliance-Level Disaggregation

with a single click. This process requires no manual configuration or advanced technical knowledge, making it accessible to most users. After installation, each add-on provides its own configuration panel within the Home Assistant interface. This ease of deployment simplifies the integration of additional functionalities, such as NILM or PV monitoring, into any Home Assistant setup.

8.3 O2RE-NILM: practical deployment

This section presents the practical implementation of the O2RE framework, which provides users with real-time information about selected appliances of interest. The implementation facilitates the use of the NILM technology in real cases and motivates the development of future integrations to accomplish important goals in Energy Management Systems.

8.3.1 Architecture of the prototype

A block diagram of the real-time NILM prototype is presented in Figure 8.3, from Aggregated Power to Appliance-Level Disaggregation:

A detailed explanation of the diagram is provided in the following. The section Monitored Appliances (FSM) represents the set of N monitored appliances, each labeled as "Appliance 1", "Appliance 2", and so on, up to "Appliance 3" in this case. These appliances are finite state machines (FSMs). Each appliance's power consumption contributes to the aggregated power measurement. The block Non-Monitored Appliances is divided into two subcategories: Appliance 4 (FSM), which is an example of an appliance that operates with distinguishable states but is not included in the monitored set. Its consumption is part of the aggregated power, yet is not individually tracked or estimated by the NILM process. Appliance 5 (Always On) represents appliances contributing to the base load B(t). These are typically always-on devices whose consumption is steady and contributes a constant load over time. Main Sensor is the central device capturing the aggregated power consumption P(t), which is the sum of the power from all appliances (monitored and non-monitored), the base load, and any residual noise or unaccounted-for consumption. Edge Device (Disaggregation Block): Serves as the computational core of the NILM system. It continuously monitors changes in the aggregated active power received from the Main Sensor, which provides updates at a frequency of Δt, and applies real-time algorithms to disaggregate this signal into individual appliance-level data. Each monitored appliance's estimated consumption is output separately, allowing for detailed analysis and monitoring of individual appliances. Unknown Loads represents the disaggregated component attributed to FSM appliances not monitored. This block would include the estimated consumption of "Appliance 4" in this context. Base Load reflects the constant power consumption part of the aggregated signal, including "Appliance 5 (Always On)" in this diagram. This load is steady and contributes to the overall consumption without significant variation over short periods.

Main Sensors used to verify the deployment

Two types of sensors were used: The first was simulated using data from the UK-DALE dataset [60], which contains detailed energy usage data from UK households. Instead of relying on physical sensors, we simulated a main sensor by wirelessly transmitting real, publicly available data to the Home Assistant OS. This method allowed us to evaluate the prototype under various conditions and scenar-

ios, closely mimicking real household energy consumption patterns and verifying whether the performance of the add-on matched the theoretical results presented earlier. For this purpose, an auxiliary add-on was implemented in Home Assistant that runs a web server listening for queries. Through a specific endpoint, the server returns the aggregated active power from the UK-DALE dataset, matched as closely as possible to the current time of the server. A REST sensor was then created in Home Assistant to fetch active power values from this endpoint at a defined frequency.

The second was a Shelly EM sensor equipped with a 50A current clamp, installed in the main distribution panel of the author's residence. This sensor provides real-time active power measurements of the entire household load and communicates over Wi-Fi. It integrates seamlessly with Home Assistant via its REST API, allowing frequent data updates at intervals of approximately one to two seconds. The device supports both local and cloud-based operation, offering flexibility in deployment and data access. This setup not only enabled validation of the prototype in a real residential environment but also demonstrated the ease of installation and practical integration using commercially available sensors. Further technical specifications of the sensor can be found in the manufacturer's documentation [155].

8.3.2 Configuration and Visualization of the Results

Once the NILM add-on is installed, configuration can be initiated through a dedicated web-based interface developed specifically for this purpose (Figure 8.4). The "Main Power Sensor" section (Figure 8.4) allows users to select the primary sensor that measures the total power consumption of the household. The "NILM Models (Appliances)" section (Figure 8.4) enables the definition of individual appliances that the algorithm will target for disaggregation.

Virtual sensors are automatically generated within the platform to report the results of NILM disaggregation in real time. This enables a continuous flow of information, ensuring smooth integration of the disaggregated data into the smart home's automation and energy management processes. Once the NILM add-on is running, users can view the real-time power consumption of the configured appliances directly in the Home Assistant dashboard. In addition to appliance-specific

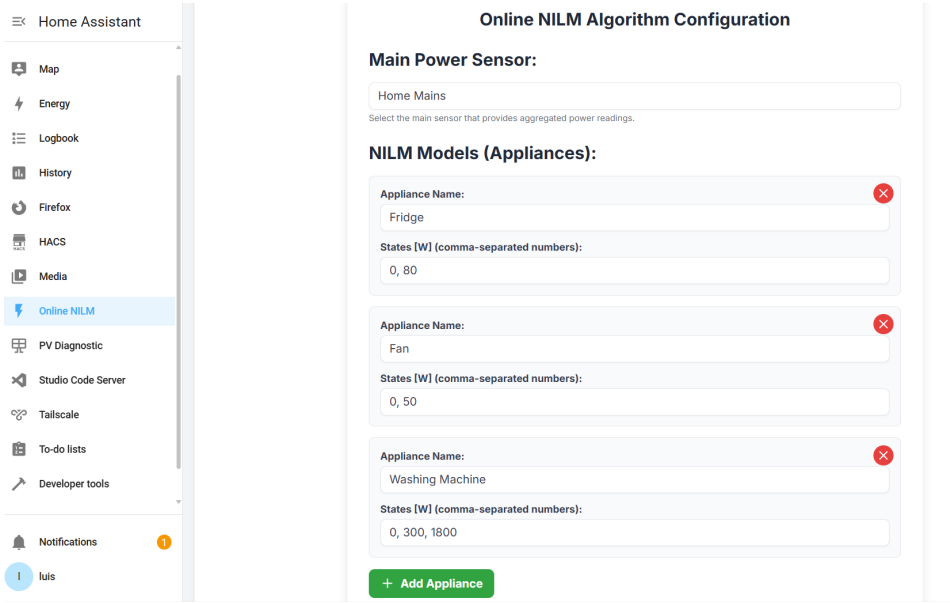
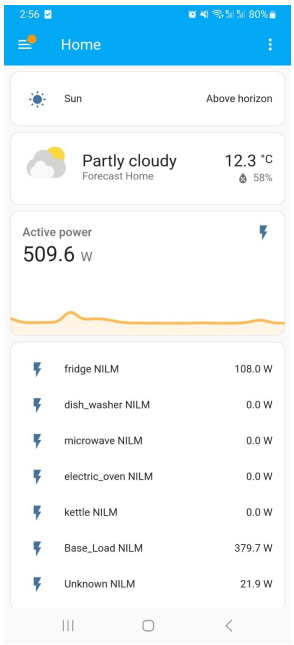


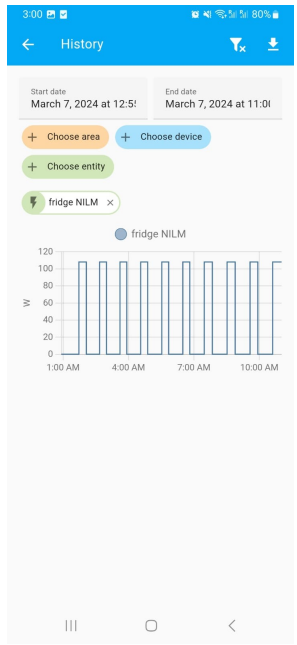
Figure 8.4: Configuration of the NILM Add-on

sensors, the system also provides virtual sensors for the base load and unknown appliances, all of which can be monitored through the interface (Figure 8.5a). The addon further enables users to inspect the power consumption history for each monitored appliance, as illustrated in Figure 8.5b and Figure 8.5c. Access to this historical data helps users understand how their daily habits influence overall energy usage, enabling more informed decisions to reduce consumption and lower energy costs.

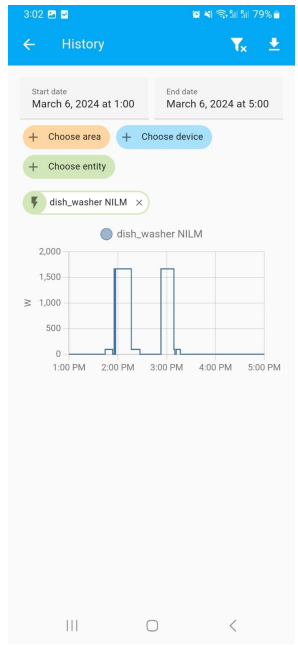
In addition to real-time active power visualization for each monitored appliance, users can also observe the overall energy flow within their homes. The Energy section in Home Assistant displays both total energy consumption and the individual energy usage of each monitored device (Figure 8.6). By presenting this information in a clear and accessible format, the system encourages users to become more aware of their energy habits, often promoting behavior changes that support greater energy efficiency.



(a) Real-Time Information of Power Consumption.

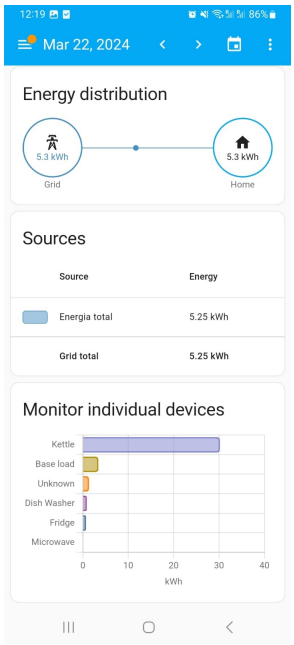


(b) NILM History of the Fridge

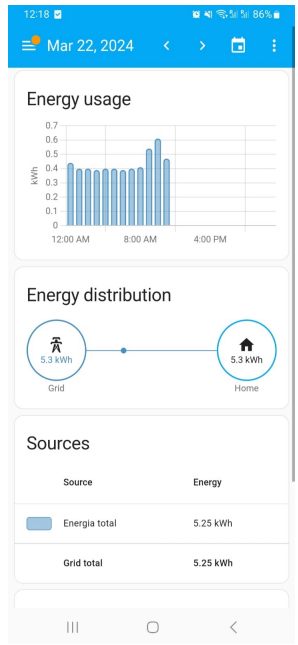


(c) NILM History of the Dish Washer

Figure 8.5: Real-time and history visualization in Home Assistant



(a) Energy Usage and Distribution



(b) Energy Distribution and Sources

Figure 8.6: Visualization of total energy and individual appliances' energy in Home Assistant

8.4 D-SDM PV parameter identification: practical deployment

This section presents the implementation of the PV add-on used to validate the deployment of the parameter estimation method based on the D-SDM, which was introduced earlier in this thesis. The algorithm runs within the same Home Assistant instance as the NILM module and thus operates entirely on the same Raspberry Pi device. Its objective is to demonstrate that robust, real-time PV diagnosis is feasible in a constrained embedded environment using local computation and data visualization.

8.4.1 Architecture of the prototype

Figure 8.7 illustrates the architecture of the prototype, highlighting the complete data and processing flow from the PV system to the visualization of results. The inverter, whether real or simulated, sends IV curve data at a defined interval Δt to the edge device. This edge device, a Raspberry Pi running Home Assistant, hosts the D-SDM Add-on, which performs the parameter estimation process. The estimated parameters are published as sensor entities within Home Assistant, enabling real-time monitoring of the PV system's behavior. The add-on also publishes the fitting error, which serves as an indicator of the model accuracy and can be used to identify and discard parameter estimations that result from anomalous IV curves caused by faults or inconsistencies in the data acquisition workflow.

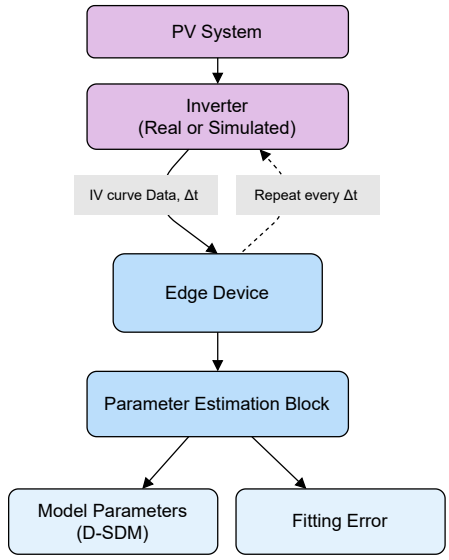


Figure 8.7: Online PV parameter identification process in an edge device

To simulate the behavior of a PV inverter during testing, a custom IV Simulator Add-on has been developed for Home Assistant. This component is not part of the final system architecture but serves as a temporary utility to emulate the IV data that would be provided by a physical inverter. The add-on runs a REST API locally and, upon each request, returns a randomly selected IV curve from a dataset composed of experimental measurements previously used in the development of the D-SDM algorithm [30]. This setup provides a controlled and reproducible data stream for validating the practical deployment of the D-SDM method.

8.4.2 Configuration and Visualization of the Results

Building upon the architecture shown in Figure 8.7, the D-SDM PV parameter identification add-on provides a user-configurable interface and integrated result visualization within Home Assistant. At a user-defined frequency, the add-on retrieves IV curve data from the specified source and applies the D-SDM estimation algorithm. Each cycle produces a set of model parameters, the associated fitting error, and the processing time, which are then published as sensor entities in Home

Assistant.

As shown in Figure 8.8, the configuration panel allows users to define the IV data source URL, set the monitoring interval, and specify the number of series-connected cells. The interface also includes a live plot of the most recent IV curve to support data verification. Configuration settings are retained across system restarts. The estimation results can be visualized through the Home Assistant dashboard or the History dashboard, enabling real-time inspection of parameter trends and model accuracy (Figure 8.9). This setup allows continuous monitoring of the PV system's behavior and supports the identification of PV anomalies or acquisition faults.

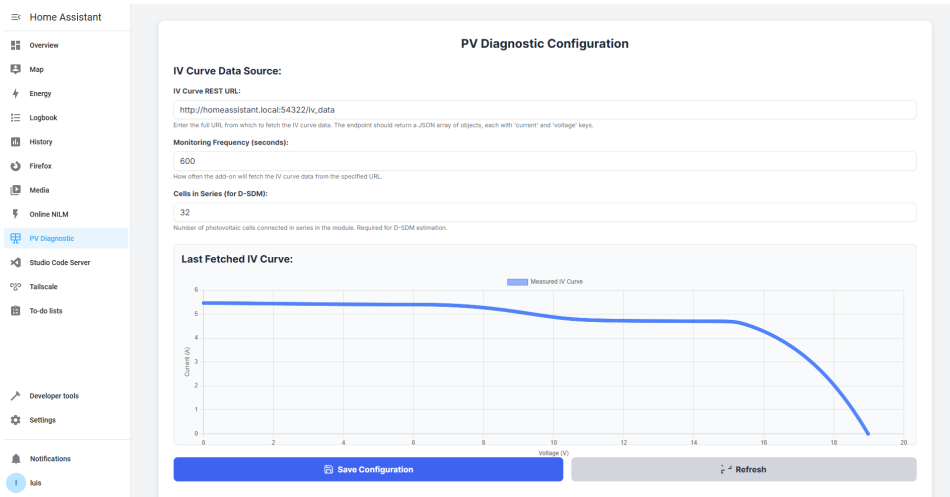


Figure 8.8: D-SDM PV add-on configuration

Figure 8.9: D-SDM PV add-on results visualization

Moreover, the system's modular architecture facilitates future expansion, particularly in the context of advanced energy management and PV system supervision. By publishing model estimation results and fitting errors as standardized Home Assistant sensor entities, the system creates a flexible data infrastructure that can support a wide range of additional functionalities. These include accurate PV forecasting, fault detection, state-of-health estimation, and real-time performance diagnostics. Furthermore, the same framework can be extended to incorporate broader energy management strategies, enabling integration with load forecasting, battery scheduling, and grid interaction optimization. This implementation not only meets immediate monitoring needs but also provides a scalable foundation for more comprehensive energy intelligence solutions.

8.5 Performance Evaluation

This section presents an evaluation of the computational performance of the deployed add-ons: the *O2RE-NILM Add-on* and the *D-SDM PV Parameter Identification Add-on*. The two add-ons were installed and executed concurrently on a single Home Assistant instance over a continuous period of at least one month. The objective is to assess the feasibility of running both components in real-time on a resource-constrained edge device, without compromising the system's responsiveness or stability. The analysis focuses on execution time and the sustained

operation of both add-ons under realistic residential conditions. Figure 8.10 shows the Home Assistant interface with both add-ons running simultaneously, illustrating their operation within the same deployment environment.

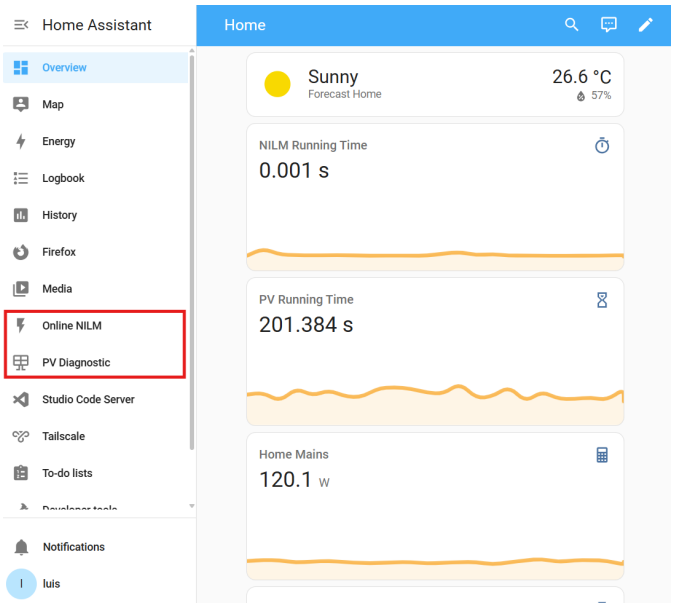


Figure 8.10: Concurrent operation of NILM and PV add-ons in Home Assistant

8.5.1 O2RE-NILM performance

To evaluate the system's ability to perform power disaggregation effectively and to verify its stability under long-term operation, the prototype was tested using two configurations. Initially, a simulated main sensor transmitted data at a fixed interval of 6 seconds. This was followed by testing with a real sensor, the Shelly EM with a 50A clamp [155], which provided measurement updates approximately every 2 seconds, as previously described.

Our original study [28] provided a detailed comparison of the algorithm's performance against existing state-of-the-art approaches. In this evaluation, the focus shifts to its computational efficiency and its ability to operate with real-time responsiveness in a deployed environment. To assess this, the running time required to complete each execution of the algorithm was measured throughout the evaluation period.

Table 8.1 presents the performance analysis of the NILM algorithm across average and critical instances. This analysis focuses on the algorithm's performance

Table 8.1: Performance Analysis of O2RE deployment Across Average and Critical Events

Statistic	Time (seconds)
Mean (All)	0.004
Mean (>0.05s)	0.071
Maximum (>0.05s)	0.132

during crucial situations, which can occur when an edge is detected (triggering the most computationally intensive part) or when there is congestion due to many services running simultaneously in Home Assistant. The table summarizes the mean running time overall and during critical events (running time > 0.05 seconds).

The overall mean running time of approximately 0.004 seconds indicates that, under average conditions, the method operates extremely fast. Moreover, the critical instances, while less frequent, are critically important as they could reflect the algorithm's response to significant changes in energy consumption patterns. The mean running time for these instances is around 0.071 seconds, indicating that while the algorithm takes longer, it still reacts quickly, and the method can seamlessly manage a frequency in the order of 1Hz in the tested edge device. These results suggest that our framework is well-suited for real-time monitoring.

8.5.2 D-SDM PV performance

The computational performance of the *D-SDM PV Parameter Identification Add-on* was evaluated by measuring the execution time required for each estimation cycle. For this assessment, the add-on was tested using a simulated inverter, as previously described, configured to provide an IV curve every 10 minutes. Although this frequency is higher than what would typically be expected in real-world PV monitoring scenarios, it was intentionally selected to place greater computational demand on the system and evaluate its robustness under sustained operation.

Each cycle involved receiving an IV curve, executing the D-SDM parameter estimation algorithm, and publishing the resulting parameters, fitting error, and execution time as Home Assistant sensor entities. A critical factor for real-time deployment of the D-SDM PV add-on is the rate at which the estimation algorithm can be executed without overlap between successive cycles. Table 8.2 presents the

measured runtime percentiles along with the corresponding maximum number of safe executions per hour. These figures assume strictly sequential processing on a single thread.

The median runtime is approximately 234 seconds, which permits around 15 estimation cycles per hour. A more conservative operation, ensuring that 75% of executions complete before the next cycle, would require a spacing of about 390 seconds, limiting the system to 9 runs per hour. For more stringent reliability, targeting 90% or 95% completion before the next trigger, the launch frequency should be reduced to every 523 or 566 seconds, respectively. This corresponds to roughly 7 or 6 runs per hour.

These execution times are consistent with, and in many cases well below, the data acquisition frequencies typically found in real-world PV monitoring systems, which often operate at intervals greater than 10 minutes. Therefore, the algorithm demonstrates sufficient computational performance for deployment in real-time residential PV applications. In cases requiring higher throughput, clustering or parallel processing can be applied to increase execution capacity.

Table 8.2: Runtime percentiles and implied safe launch rates

Percentile	Runtime (s)	Max launches h^{-1}
25^{th} (Q_1)	128.3	28.0
50^{th} (Median)	234.3	15.4
75^{th} (Q_3)	390.4	9.2
90^{th}	523.5	6.9
95^{th}	566.4	6.4

8.6 Conclusions of the Chapter

This chapter presented the design, deployment, and computational performance evaluation of two add-ons developed for Home Assistant: the NILM Disaggregation Add-on and the D-SDM PV Parameter Identification Add-on. Both were executed concurrently on a single Raspberry Pi 4 edge device over an extended period in a real residential environment.

The NILM add-on was tested using both simulated and physical sensors

to assess its responsiveness and real-time processing capabilities. It successfully maintained continuous operation while generating virtual sensors and updating appliance-level consumption estimates at the configured data acquisition rates. Throughout the evaluation, the Home Assistant instance remained responsive, and the author did not observe any delays or performance degradation in the system.

The D-SDM PV add-on was evaluated using a simulated inverter providing IV curves at a frequency of one curve every 10 minutes. The execution time of each estimation cycle remained well below this interval, with typical runtimes supporting safe and reliable operation in real-time conditions. Even when stressed with a higher-than-typical input rate, the system sustained stable performance.

In summary, the results confirm that both add-ons can operate efficiently in a shared Home Assistant environment without overloading the computational resources of a typical edge device, as the Raspberry Pi 4. The system architecture supports real-time energy monitoring and provides a robust foundation for future integration of diagnostic, forecasting, and optimization functionalities within residential energy management systems.

9

Business Model for Integrated Real-Time Energy Intelligence

9.1 Introduction

As mentioned in the introduction, this thesis is developed within the framework of the SMARTGYsum project [26], which integrates research in advanced energy technologies with the development of sustainable market strategies. The thesis develops a dedicated business model to facilitate the translation of its technological innovations into real-world applications. The inclusion of this model is essential, as the project framework recognizes that technical advancements require a clear pathway to commercialization and long-term viability to achieve meaningful adoption. By addressing this requirement, the thesis supports SMARTGYsum's objective of delivering solutions that are technically robust, economically feasible, and accessible to end users, thereby bridging the gap between research outcomes and practical deployment.

The proliferation of home automation ecosystems has driven an increasing number of advanced users to install their own energy meters capable of reporting aggregated household power data at sub-minute intervals, often every few seconds. Unlike utility-provided smart meters, which typically report at fifteen-minute to thirty-minute intervals, these user-installed meters provide sufficiently fine temporal resolution to enable appliance-level disaggregation and performance diagnostics in near real-time. Nevertheless, this potential remains largely untapped; in many

cases, the data is used solely for aggregated consumption displays or basic automation triggers, leaving significant value unrealized.

The underutilization stems from a combination of technical and economic barriers. Existing NILM solutions hosted in the cloud can provide appliance-level estimates but often at really low frequencies, require constant internet connectivity, and pose privacy risks by exporting detailed consumption patterns to third-party servers. Conversely, per-appliance monitoring hardware such as smart plugs or current transformer clamps delivers high accuracy but at a prohibitive cost when scaled to an entire household (several appliances), in addition to introducing complexity in installation and maintenance. As a result, there is a gap in the market for solutions that leverage the rich temporal resolution of user-owned meters while preserving privacy, delivering real-time results, and avoiding the expense of dedicated hardware for each appliance.

9.2 Architectural Overview of the IREI Model

The Integrated Real-Time Energy Intelligence (IREI) model adopts a hybrid edge–cloud architecture in which the primary real-time disaggregation algorithm executes locally on user-owned hardware, typically the same single-board computer, server, or virtualized environment already employed for running home automation platforms such as Home Assistant. This approach ensures that appliance disaggregation operates continuously and with minimal latency, while safeguarding privacy by keeping high-resolution consumption data within the user's premises.

At the core of the system lies a universal NILM processing pipeline capable of handling two categories of appliance representations without requiring separate algorithms. In the **Free tier**, users can configure an unlimited number of manual FSM appliance models, specifying the number of states, their nominal power levels, and allowable transitions. This manual configuration demands some user involvement but requires no financial outlay and offers immediate compatibility with existing aggregated power meters capable of reporting at sub-minute resolution. The disaggregated outputs are published as native virtual sensors within the home automation environment, enabling direct use in automation scripts, dashboards, and optimization routines.

The **Pro features** extend this baseline capability through optional cloud-assisted services. Users may initiate a guided training procedure for a given appliance, involving a short, controlled sequence of on–off activations. The resulting power signature is uploaded to the cloud, where a dedicated embedding-generation model, separate from the inference model, extracts the appliance's characteristic feature vector. This embedding is then downloaded and integrated into the local NILM engine, where it coexists seamlessly with FSM-defined appliances. This one-time upgrade, offered on a pay-per-appliance basis, increases recognition accuracy and user effort, without transmitting continuous operation data to the cloud.

In addition to appliance embeddings, the Pro tier provides advanced analytical modules. A **load forecasting engine** operates on the set of local appliance models, augmented by temporal usage patterns learned on-device, to predict future demand profiles. This supports tariff-aware scheduling, demand response participation, and optimization of on-site renewable utilization. An **AI-driven insights engine** processes disaggregated appliance data to produce targeted recommendations on energy efficiency, fault detection, and behavioral adjustments, with user-defined delivery frequency. Furthermore, a **PV diagnostics module**, grounded in the PV parameter identification methodology developed in this thesis, can be used as an additional feature if the user owns a PV system and the Inverter allows IV curve tracing. This enables detection of partial shading, panel soiling, and progressive degradation, with potential for predictive maintenance through trend analysis.

The architectural modularity ensures that all Pro-tier features integrate into the same local processing framework, minimizing user-side complexity while allowing selective adoption of paid services. Table 9.1 summarizes this structure within the Lean Business Model Canvas, illustrating how the architectural design directly underpins the value proposition, cost structure, and revenue streams of IREI. The architecture thereby enables a tiered adoption pathway: immediate deployment through the free tier, incremental accuracy upgrades via per-appliance embeddings, and full analytical capability through subscription-based Pro services.

Table 9.1: Lean Business Model Canvas for IREI

Problem	**Solution**	**Unique Value Proposition**	**Unfair Advantage**	**Customer Segments**
Energy enthusiasts' smart meters are underused. Cloud NILM providers lack Real-time, pose privacy risks, are not flexible to work with user hardware, while per-appliance sensors are expensive	Local NILM engine for real-time, private disaggregation. **Free tier:** unlimited manual FSM appliances, user sets states manually. **Paid:** cloud-trained appliance embeddings for higher accuracy. AI-driven insights. PV diagnostic.	Appliance-level monitoring without extra hardware, real-time, and privacy-first, cheaper than smart plugs. Full Home Automation Frameworks integration as native virtual sensors for automation and energy optimization.	One universal NILM model handles both manually entered FSMs and cloud-trained appliances. Unique PV diagnostic method. AI-driven insights. Publication in Top Journals. Deep integration into the Home Assistant for immediate adoption.	Home Assistant users, Privacy-aware homeowners, PV owners, and energy service installers seeking accurate, low-cost monitoring.
Existing Alternatives	**Key Metrics**	**High-level Concept**	**Channels**	**Early Adopters**
Smart plugs/CT clamps (costly), cloud NILM (privacy issues, latency, additional hardware)	Free-to-paid model upgrade rate, disaggregation accuracy/latency, and Pro tier retention.	*"Unlock the full potential of your smart meter with real-time appliance disaggregation, AI-driven insights, energy forecasting, and PV diagnostics."*	GitHub distribution, Home Assistant community channels, Conferences, partnerships with hardware vendors and installers.	Home automation enthusiasts (Home Assistant and similar platforms **(+1M users)**), PV owners, and early adopters of smart energy solutions.
Cost Structure		**Revenue Streams**		
R&D Expenses: Developing and refining advanced NILM AI algorithms, solar PV diagnostics, and real-time energy analytics. **Hardware:** Purchasing pre-manufactured edge devices (NILM-PV hubs) from established vendors. **Software Development:** Ongoing expenses for platform maintenance, user interface enhancements, and integration with HAS. **Marketing and Sales:** Promotional campaigns, digital marketing, etc.		**Free tier:** unlimited manual FSM appliances, where users define appliance states manually. **Pay-per-app:** €3-5 one-time fee per downloaded appliance embedding. **Pro subscription:** €5-7/month, includes up to 10 cloud-trained appliances/year plus load forecasting, AI-driven insights, and PV diagnostics. **Additional Revenue Streams:** Sales of pre-configured, plug-and-play NILM-PV energy hub (out-of-the-box solution).		

9.3 Integrated Analytical Services

While the IREI edge component is dedicated to executing real-time disaggregation and publishing appliance-level virtual sensors to the home automation environment, the architecture extends its value proposition through a set of advanced analytical services hosted in the cloud. This division of processing ensures that only latency-sensitive tasks are handled locally, while computationally intensive operations that benefit from large-scale datasets and high-performance computing are executed remotely.

A defining characteristic of this design is that cloud services operate on **compact appliance models** rather than raw high-resolution consumption traces. Each appliance model integrates geometric features (embeddings), FSM state definitions, and locally learned temporal usage patterns. By transmitting only these condensed representations, the system achieves several advantages: (i) *privacy preservation*, since detailed time series containing personal behavioral patterns never leave the user's premises; (ii) *low bandwidth requirements*, enabling operation over limited or intermittent network connections; and (iii) *computational efficiency*, as the cloud can process structured models much faster than raw signal data.

A possible cloud-based service is the **load forecasting module**, which operates on the ensemble of appliance models stored locally. Periodically, aggregated historical usage summaries and embeddings are sent to the cloud, where predictive algorithms generate short- and medium-term demand forecasts. These forecasts are then returned to the user system to support proactive load scheduling, optimization against dynamic tariffs, and better integration with variable renewable energy sources.

Other possible service is based on **AI-driven insights engine**, that leverages cloud resources to transform disaggregation outputs into actionable recommendations. By combining appliance model data with anomaly detection, peer benchmarking, and operational efficiency strategies, it delivers tailored guidance on reducing consumption, optimizing schedules, and detecting performance anomalies. The frequency of these insights is user-configurable, allowing daily, weekly, or other intervals.

Furthermore, a **PV diagnostics module**, grounded in the PV parameter identification methodology developed in this thesis, can be enabled if the user owns a PV system and the inverter supports IV curve tracing. This feature detects partial shading, panel soiling, and progressive degradation, with the potential for predictive maintenance through long-term trend analysis.

By structuring these services as optional and operating exclusively on compact appliance models, IREI preserves the privacy and responsiveness of local real-time operations while offering the flexibility to extend capabilities through the Pro-tier services described in Table 9.1. This architecture enables a staged adoption pathway, transitioning users from basic disaggregation to a complete energy intelligence ecosystem without compromising control over their data.

9.4 Market Positioning and Growth Potential

The IREI model positions itself at the intersection of NILM, home automation, and cloud-augmented energy analytics. Its primary market differentiator lies in combining a real-time, privacy-preserving NILM engine running on user-owned hardware with a scalable set of cloud-based analytical services, all without requiring additional per-appliance hardware. This approach directly addresses the limitations of existing alternatives: smart meters with coarse temporal resolution (typically 15 minutes or more), cloud-only NILM platforms that introduce latency and privacy risks, and hardware-based monitoring solutions that require significant upfront investment.

The target customer base is composed of three primary segments, as defined in Table 9.1: (i) home automation platform users who value integration and customization, (ii) PV system owners seeking advanced diagnostics and optimization, and (iii) installers and integrators who can leverage the system as a low-cost, high-value proposition in energy service portfolios. The integration as native virtual sensors in home automation environments ensures immediate utility, as disaggregated appliance signals can be directly employed for automations, notifications, and optimization strategies.

The free tier functions as both an entry point and a growth driver, allowing unlimited manual FSM appliance configurations. This tier enables users to expe-

rience appliance-level monitoring without additional cost, while highlighting the accuracy and convenience benefits of upgrading to cloud-trained appliance embeddings. The pay-per-appliance model offers a low-cost, low-commitment upgrade path (3–5 EUR per embedding), while the Pro subscription consolidates advanced capabilities including up to 10 trained appliances, load forecasting, AI-driven insights, and PV diagnostics.

Growth potential stems from three complementary dynamics. First, the platform inherently motivates users to expand their set of monitored appliances once the utility of insights and automations becomes evident, creating a natural upsell pathway from manual FSM definitions to cloud-trained embeddings. Second, the addition of Pro-tier analytical services such as forecasting and PV diagnostics increases the perceived value and justifies ongoing engagement. Third, the appliance model architecture enables service diversification without altering the local real-time NILM core: new analytics, predictive maintenance modules, or integration with demand-response programs can be added entirely in the cloud, leveraging the same compact appliance representations. This design also ensures that only highly compact appliance data is transmitted, minimising bandwidth requirements, preserving privacy, and simplifying compliance with data protection regulations.

Strategically, the combination of a frictionless free tier, low-cost per-appliance upgrades, and high-value subscription services positions the system to capture both the enthusiast and professional segments of the smart energy market. By aligning with the growth trajectory of home automation adoption and increasing consumer awareness of energy efficiency, it has the potential to establish itself as a reference solution for real-time, privacy-centric energy intelligence.

9.5 Conclusion of the Chapter

The IREI business model translates the technical innovations of this thesis, alongside anticipated advances in the same research line, into a commercially viable service that targets a specific market gap. It runs the primary NILM algorithm locally on user-owned hardware and offers optional cloud-based enhancements through appliance embeddings, achieving a distinct balance of privacy, performance, and flexibility. Integration with home automation ecosystems, coupled

with extensible analytical tools such as AI-driven energy insights and PV diagnostics, supports both immediate adoption and sustained market relevance. Table 9.1 presents the Lean Canvas summarizing the model's structure, while the discussion in this chapter places it within the wider field of residential energy intelligence, highlighting its capacity to unlock new capabilities of user-installed smart meters.

Part V

Concluding Remarks and Future Directions

10

General Conclusions & Future Work

10.1 General Conclusions

This thesis addresses the challenge of integrated residential energy monitoring by developing methods for both demand-side NILM and supply-side PV diagnostics, and by demonstrating their joint operation in a unified, edge-deployable platform. The work progresses from algorithm design to systematic evaluation and experimental validation in real residential settings, ensuring that the proposed solutions are compatible with low-frequency measurements and limited computational resources, and are therefore suitable for practical deployment.

In the NILM domain, the proposed methods satisfy defined practical constraints while improving robustness over existing approaches. The analysis of deep learning NILM architectures under conditions with unmonitored loads demonstrates that performance degradation scales with the magnitude of unknown consumption and is strongly dependent on appliance-specific patterns. To mitigate these effects, an online framework operating at 1 Hz was developed, combining edge detection, probabilistic state estimation, and a modified Population-Based Incremental-Learning (PBIL) algorithm. Evaluation with public datasets confirms that the framework maintains stable performance in the presence of unknown appliances while meeting the requirements of low-frequency, low-resource residential operation.

For PV characterization, the thesis examines the limitations of the SDM under mismatched conditions and proposes methods to improve the consistency of parameter estimation across operating scenarios. A multi-objective optimization formulation was introduced to jointly fit static SDM and dynamic CPE parameters through a shared series resistance. Considering the practical challenges of implementing EIS in PV systems, the D-SDM was developed as a more deployable alternative that relies solely on the I–V curve. The model incorporates a dedicated fitting procedure and a robust error function that accounts for shaded or mismatched cells, enabling reliable parameter estimation without additional measurement hardware. Experimental and simulated results demonstrate improved stability under mismatch and more consistent detection of series resistance changes compared to the SDM, while preserving a simpler structure than the CPE model.

The deployment stage integrates both methods into add-ons for the open-source Home Assistant platform: one for NILM disaggregation and another for D-SDM PV parameter identification. Home Assistant was selected for its wide adoption, active community, and flexibility, which facilitate integration into existing residential automation systems without proprietary dependencies. Both add-ons operate locally on a Raspberry Pi 4, ensuring low latency, data privacy, and resilience to network outages. Long-term testing in a residential environment confirmed sustained operation without overloading computational resources, with the NILM add-on delivering appliance-level estimates at configured acquisition rates and the PV add-on completing parameter identification within measurement intervals, even under increased input rates. These technical developments are framed within a business model that combines free local deployment with optional cloud-based enhancements, enabling scalable functionality and market adoption.

The principal contribution of this thesis is the development of algorithms and methods for demand- and supply-side energy monitoring that overcome key practical limitations hindering the wider adoption of previous NILM and PV diagnostic approaches. Their implementation in an open-access home automation platform and validation in real residential conditions demonstrate their readiness for direct application. By providing coherent and complementary information on both consumption and generation within a single framework, the work supports the effective integration of monitoring and diagnostics into modern EMS. Coupled

with a commercialization pathway that leverages local processing, modular feature upgrades, and a tiered adoption model, these contributions enable not only coordinated operation, optimization, and decision-making in smart buildings but also a viable route to large-scale deployment in the residential energy market.

10.2 Future Work

10.2.1 NILM

In the NILM research field, future research should focus on enhancing the O2RE framework by incorporating additional appliance-level information, such as on-probabilities, expected on-times, and tuned algorithm parameters. These parameters are essential for improving disaggregation accuracy when appliances have very similar power consumption. The model should be capable of continuously updating these parameters during operation through incremental learning, enabling adaptation to appliance aging, evolving usage patterns, and other changes. Unsupervised techniques for extracting such parameters from historical main data should be developed to minimize the need for occupant involvement in algorithm tuning.

Regarding Deep Learning approaches, future research should focus on improving generalization and portability across domains. A promising direction is to learn appliance-specific signatures that remain identifiable in the aggregate mains signals from the test domain. These signatures should be learned with minimal occupant involvement and limited training effort, ensuring that the approach is deployable on edge devices with restricted computational resources. Instead of directly regressing power estimates, methods should extract invariant representations capturing the unique power cycle or spectral fingerprint of each appliance. Shared feature extractors trained through contrastive learning or its variants could be developed to generate robust appliance-specific representations, which could then be applied directly for accurate disaggregation.

An important future research should focus on integrating deep learning methods with traditional approaches to address the limitations of current one-model-per-appliance deep learning architectures, especially for real-time applications. These architectures are highly flexible and modular, as disaggregating a new

appliance only requires adding a new model or embedding. However, because they process one appliance at a time, they neglect interactions between appliances. Combining these methods with O2RE-style event-based, state-based modeling would enable the inclusion of multi-appliance state dependencies, temporal correlations, and operational priors. This hybrid approach would retain the interpretability and structured temporal modeling of traditional techniques while exploiting the flexibility and accuracy of deep learning. Future work could explore the extension of these approaches to industrial settings, where efficient management of high-consumption machinery is essential for achieving significant energy and cost savings.

10.2.2 PV diagnostic

Regarding PV diagnostic, future research should focus on exploiting the richer information content of D-SDM parameters compared to the standard SDM. The D-SDM can yield additional diagnostic indicators such as the number of shaded cells, the voltage range where certain cells operate in reverse bias, and other mismatch-related signatures. These parameters could serve as input features for a classifier designed to automatically detect specific failures, including mismatching issues, hot spots, and partial shading patterns. Leveraging this extra information could transform the D-SDM into a powerful tool for automated PV fault detection.

Other interesting future research could focus on testing the D-SDM method with different PV cell technologies, such as half-cut cells, bifacial modules, and thin-film panels, to better understand its limitations and applicability across designs. Such an evaluation would reveal how structural differences, interconnection layouts, and intrinsic electrical characteristics affect parameter estimation accuracy. Field trials on varied PV configurations and operating conditions would be essential to validate the method's generalizability and ensure that the diagnostic capabilities remain consistent across technologies.

10.2.3 Integration of NILM, PV Diagnostics, and EMS

The information provided by NILM and PV Diagnostic can be integrated into the EMS. Future research should focus on this integration to jointly address the demand side and supply side of residential energy systems. NILM can provide high-resolution appliance-level consumption data that enhances load forecasting,

while PV diagnostics can deliver accurate generation profiles and system health indicators. Combining these capabilities would enable the EMS to perform AI-driven energy insights, such as identifying optimal load-shifting opportunities, detecting inefficiencies, and recommending behavioral or operational adjustments to improve energy efficiency.

This integration can also enhance the optimization of energy flow within the building, targeting one or more objective functions such as cost reduction, maximization of self-consumption, minimization of carbon footprint, or extension of system longevity. Future research in additional applications could include predictive demand-response participation, self-consumption forecasting, and proactive maintenance scheduling based on both load and generation conditions. Long-term field studies in varied residential contexts should be conducted to validate the interoperability of NILM and PV diagnostic modules within the EMS, evaluate scalability to multi-building or community-level deployments, and quantify the impact on energy efficiency, operational reliability, and sustainability outcomes.

Bibliography

[1] D. Thomas, O. Deblecker, and C. S. Ioakimidis, "Optimal operation of an energy management system for a grid-connected smart building considering photovoltaics' uncertainty and stochastic electric vehicles' driving schedule," *Applied Energy*, vol. 210, pp. 1188–1206, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261917309078

[2] T. Knittel, C. Lowry, M. McPherson, P. Wild, and A. Rowe, "Electrifying end-use demands: A rise in capacity and flexibility requirements," *Energy*, vol. 320, p. 135373, Apr. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.energy.2025.135373

[3] O. Omoyele, S. Matrone, M. Hoffmann, E. Ogliari, J. M. Weinand, S. Leva, and D. Stolten, "Impact of temporal resolution on the design and reliability of residential energy systems," *Energy and Buildings*, vol. 319, p. 114411, Sep. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2024.114411

[4] S. Chakraborty and S. Das, "Application of smart meters in high impedance fault detection on distribution systems," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, p. 3465–3473, May 2019. [Online]. Available: http://dx.doi.org/10.1109/TSG.2018.2828414

[5] K. Brucke, S. Arens, J.-S. Telle, T. Steens, B. Hanke, K. von Maydell, and C. Agert, "A non-intrusive load monitoring approach for very short-term power predictions in commercial buildings," *Applied Energy*, vol. 292, p. 116860, Jun. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2021.116860

[6] Y. Huang, L. Wang, W. Guo, Q. Kang, and Q. Wu, "Chance constrained optimization in a home energy management system," *IEEE Transactions on Smart Grid*, vol. 9, no. 1, p. 252–260, Jan. 2018. [Online]. Available: http://dx.doi.org/10.1109/TSG.2016.2550031

[7] G. Dawes, T. Kirant-Mitić, Z. Jiang, J. Le Dréau, H. Cai, J. Cui, J. Townsend, A. Bampoulas, R. Li, R. A. Lopes, and B. Dong, "Energy flexibility at multi-building scales: A review of the dominant factors and their uncertainties," *Energy and Buildings*, vol. 346, p. 116157, Nov. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2025.116157

[8] L. D. Jathar, S. Ganesan, U. Awasarmol, K. Nikam, K. Shahapurkar, M. E. M. Soudagar, H. Fayaz, A. El-Shafay, M. Kalam, S. Bouadila, S. Baddadi, V. Tirth, A. S. Nizami, S. S. Lam, and M. Rehan, "Comprehensive review of environmental factors influencing the performance of photovoltaic panels: Concern over emissions at various phases throughout the lifecycle," *Environmental Pollution*, vol. 326, p. 121474, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.envpol.2023.121474

[9] B. Meng, R. Loonen, and J. Hensen, "Leveraging dynamic power benchmarks and cusum charts for enhanced fault detection in distributed pv systems," *Energy Conversion and Management*, vol. 314, p. 118692, Aug. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.enconman.2024.118692

[10] C. Feng, Y. Wang, Q. Chen, Y. Ding, G. Strbac, and C. Kang, "Smart grid encounters edge computing: opportunities and applications," *Advances in Applied Energy*, vol. 1, p. 100006, Feb. 2021. [Online]. Available: https://doi.org/10.1016/j.adapen.2020.100006

[11] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE Network*, vol. 33, no. 2, pp. 111–117, Mar. 2019. [Online]. Available: https://doi.org/10.1109/mnet.2019.1800254

[12] P. Kumari, R. Mishra, H. P. Gupta, T. Dutta, and S. K. Das, "An energy efficient smart metering system using edge computing in LoRa network," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 4, pp. 786–798, Oct. 2022. [Online]. Available: https://doi.org/10.1109/tsusc.2021.3049705

[13] L. C. B. C. Ferreira, A. D. R. Borchardt, G. D. S. Cardoso, D. A. Mendes Lemes, G. R. D. R. d. Sousa, F. B. Neto, E. R. de Lima, G. Fraidenraich, P. Cardieri, and L. G. P. Meloni, "Edge computing

and microservices middleware for home energy management systems," *IEEE Access*, vol. 10, p. 109663–109676, 2022. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2022.3214229

[14] S. Sykiotis, S. Athanasoulias, M. Kaselimi, A. Doulamis, N. Doulamis, L. Stankovic, and V. Stankovic, "Performance-aware nilm model optimization for edge deployment," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 3, p. 1434–1446, Sep. 2023. [Online]. Available: http://dx.doi.org/10.1109/TGCN.2023.3244278

[15] S. Sairam, S. Seshadhri, G. Marafioti, S. Srinivasan, G. Mathisen, and K. Bekiroglu, "Edge-based explainable fault detection systems for photovoltaic panels on edge nodes," *Renewable Energy*, vol. 185, p. 1425–1440, Feb. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2021.10.063

[16] T. Todic, V. Stankovic, and L. Stankovic, "An active learning framework for the low-frequency non-intrusive load monitoring problem," *Applied Energy*, vol. 341, p. 121078, Jul. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2023.121078

[17] G. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, p. 1870–1891, 1992. [Online]. Available: http://dx.doi.org/10.1109/5.192069

[18] H. Rafiq, P. Manandhar, E. Rodriguez-Ubinas, O. Ahmed Qureshi, and T. Palpanas, "A review of current methods and challenges of advanced deep learning-based non-intrusive load monitoring (nilm) in residential context," *Energy and Buildings*, vol. 305, p. 113890, Feb. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2024.113890

[19] N. Batra, R. Kukunuri, A. Pandey, R. Malakar, R. Kumar, O. Krystalakos, M. Zhong, P. Meira, and O. Parson, "Towards reproducible state-of-the-art energy disaggregation," in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. New York, NY, USA: ACM, nov 2019, pp. 193–202. [Online]. Available: https://dl.acm.org/doi/10.1145/3360322.3360844

[20] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018. [Online]. Available: http://dx.doi.org/10.1609/aaai.v32i1.11873

[21] I. Ait Abdelmoula, S. Idrissi Kaitouni, N. Lamrini, M. Jbene, A. Ghennioui, A. Mehdary, and M. El Aroussi, "Towards a sustainable edge computing framework for condition monitoring in decentralized photovoltaic systems," *Heliyon*, vol. 9, no. 11, p. e21475, Nov. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.heliyon.2023.e21475

[22] B. Taghezouit, F. Harrou, Y. Sun, and W. Merrouche, "Model-based fault detection in photovoltaic systems: A comprehensive review and avenues for enhancement," *Results in Engineering*, vol. 21, p. 101835, Mar. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.rineng.2024.101835

[23] B. Li, T. Karin, B. E. Meyers, X. Chen, D. C. Jordan, C. W. Hansen, B. H. King, M. G. Deceglie, and A. Jain, "Determining circuit model parameters from operation data for pv system degradation analysis: Pvpro," *Solar Energy*, vol. 254, p. 168–181, Apr. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2023.03.011

[24] Huawei Digital Power Technologies Co., "Smart I-V curve diagnosis-technical datasheet," 2023. [Online]. Available: https://solar.huawei.com/download?p=/media/SolarV4/solar-version2/europe/gr/professionals/all-products/smart-micro-grid/digital-characteristic/datasheet/datasheet-smart-i-v-curve-diagnosis-20230302.pdf

[25] Sungrow Power Supply Co., Ltd., *SP600S Optimizer User Manual*, Mar. 2025, version 111. [Online]. Available: https://info-support.sungrowpower.com/application/pdf/2025/03/07/SP600S-UEN-Ver111-202502.pdf

[26] SmartGYsum, "Smartgysum homepage," https://smartgysum.eu/, 2025, accessed: 2025-07-28.

[27] L. E. Garcia-Marrero, G. Petrone, and E. Monmasson, "Transfer capabilities of seq2seq and seq2point cnn architectures in non-intrusive

load monitoring with unseen appliances," *Mathematics and Computers in Simulation*, vol. 239, p. 211–222, Jan. 2026. [Online]. Available: http://dx.doi.org/10.1016/j.matcom.2025.05.021

[28] L. Garcia-Marrero, E. Monmasson, and G. Petrone, "Online real-time robust framework for non-intrusive load monitoring in constrained edge devices," *Applied Energy*, vol. 378, p. 124814, Jan. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2024.124814

[29] L. E. Garcia-Marrero, R. A. Guejia-Burbano, G. Petrone, M. Piliougine, and E. Monmasson, "Identification of static and dynamic parameters of pv models through multi-objective optimization," in *2023 IEEE 17th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)*. IEEE, Jun. 2023, p. 1–6. [Online]. Available: http://dx.doi.org/10.1109/CPE-POWERENG58103.2023.10227400

[30] L. E. Garcia-Marrero, C. I. Pavon-Vargas, J. D. Bastidas-Rodriguez, E. Monmasson, and G. Petrone, "Self-adaptive single-diode model parameter identification under small mismatching conditions," *Renewable Energy*, p. 122735, Mar. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2025.122735

[31] L. Garcia-Marrero, M. Piliougine, G. Petrone, M. De Riso, P. Guerriero, and E. Monmasson, "Challenges in photovoltaic parameter identification under mismatching conditions," in *2023 International Conference on Clean Electrical Power (ICCEP)*. IEEE, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1109/ICCEP57914.2023.10247445

[32] L. Garcia-Marrero, G. Petrone, and E. Monmasson, "Detection of series resistance degradation in pv modules using measured current-voltage and frequency-domain impedance," in *2025 International Conference on Clean Electrical Power (ICCEP)*. IEEE, Jun. 2025, p. 250–255. [Online]. Available: http://dx.doi.org/10.1109/ICCEP65222.2025.11143628

[33] Y. Wang, Q. Chen, T. Hong, and C. Kang, "Review of smart meter data analytics: Applications, methodologies, and challenges," *IEEE Transactions*

on Smart Grid, vol. 10, no. 3, p. 3125–3148, May 2019. [Online]. Available: http://dx.doi.org/10.1109/TSG.2018.2818167

[34] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, and S. Patel, "Disaggregated end-use energy sensing for the smart grid," *IEEE Pervasive Computing*, vol. 10, no. 1, p. 28–39, Jan. 2011. [Online]. Available: http://dx.doi.org/10.1109/MPRV.2010.74

[35] S. Makonin, F. Popowich, I. V. Bajic, B. Gill, and L. Bartram, "Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, p. 2575–2585, Nov. 2016. [Online]. Available: http://dx.doi.org/10.1109/TSG.2015.2494592

[36] P. Kumar and A. R. Abhyankar, "A time efficient factorial hidden markov model-based approach for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 14, no. 5, p. 3627–3639, Sep. 2023. [Online]. Available: http://dx.doi.org/10.1109/TSG.2023.3245019

[37] J. Z. Kolter and T. Jaakkola, "Approximate inference in additive factorial hmms with application to energy disaggregation," *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, vol. 22, pp. 1472–1482, 2012. [Online]. Available: https://proceedings.mlr.press/v22/zico12.html

[38] Z. Wu, C. Wang, W. Peng, W. Liu, and H. Zhang, "Non-intrusive load monitoring using factorial hidden markov model based on adaptive density peak clustering," *Energy and Buildings*, vol. 244, p. 111025, Aug. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2021.111025

[39] 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 Transactions on Smart Grid*, vol. 10, no. 3, p. 3430–3439, May 2019. [Online]. Available: http://dx.doi.org/10.1109/TSG.2018.2826844

[40] W. Luan, R. Zhang, B. Liu, B. Zhao, and Y. Yu, "Leveraging sequence-to-sequence learning for online non-intrusive load monitoring in edge device," *International Journal of Electrical Power and Energy*

Systems, vol. 148, p. 108910, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.ijepes.2022.108910

[41] Y. Liu, L. Bai, J. Ma, W. Wang, and W. Ouyang, "Self-supervised feature learning for appliance recognition in nonintrusive load monitoring," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, p. 1698–1710, Feb. 2024. [Online]. Available: http://dx.doi.org/10.1109/TII.2023.3280445

[42] S. Chen, B. Zhao, M. Zhong, W. Luan, and Y. Yu, "Nonintrusive load monitoring based on self-supervised learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, p. 1–13, 2023. [Online]. Available: http://dx.doi.org/10.1109/TIM.2023.3246504

[43] Y. Liu, L. Zhong, J. Qiu, J. Lu, and W. Wang, "Unsupervised domain adaptation for nonintrusive load monitoring via adversarial and joint adaptation network," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 1, p. 266–277, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1109/TII.2021.3065934

[44] M. D'Incecco, S. Squartini, and M. Zhong, "Transfer learning for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, p. 1419–1429, Mar. 2020. [Online]. Available: http://dx.doi.org/10.1109/TSG.2019.2938068

[45] S. Mari, G. Bucci, F. Ciancetta, E. Fiorucci, and A. Fioravanti, "An embedded deep learning nilm system: A year-long field study in real houses," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, p. 1–15, 2023. [Online]. Available: http://dx.doi.org/10.1109/TIM.2023.3328085

[46] P. A. Schirmer and I. Mporas, "Non-intrusive load monitoring: A review," *IEEE Transactions on Smart Grid*, vol. 14, no. 1, p. 769–784, Jan. 2023. [Online]. Available: http://dx.doi.org/10.1109/TSG.2022.3189598

[47] R. Kukunuri, A. Aglawe, J. Chauhan, K. Bhagtani, R. Patil, S. Walia, and N. Batra, "Edgenilm: Towards nilm on edge devices," in *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, ser. BuildSys '20. ACM, Nov. 2020. [Online]. Available: http://dx.doi.org/10.1145/3408308.3427977

[48] Z. Lu, Y. Cheng, M. Zhong, W. Luan, Y. Ye, and G. Wang, "Lightnilm: lightweight neural network methods for non-intrusive load monitoring," in *Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, ser. BuildSys '22. ACM, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1145/3563357.3566152

[49] S. Dash, R. Sodhi, and B. Sodhi, "An appliance load disaggregation scheme using automatic state detection enabled enhanced integer programming," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, p. 1176–1185, Feb. 2021. [Online]. Available: http://dx.doi.org/10.1109/TII.2020.2975810

[50] M. Balletti, V. Piccialli, and A. M. Sudoso, "Mixed-integer nonlinear programming for state-based non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 13, no. 4, p. 3301–3314, Jul. 2022. [Online]. Available: http://dx.doi.org/10.1109/TSG.2022.3152147

[51] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-intrusive load disaggregation using graph signal processing," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, p. 1739–1747, May 2018. [Online]. Available: http://dx.doi.org/10.1109/TSG.2016.2598872

[52] B. Zhao, X. Li, W. Luan, and B. Liu, "Apply graph signal processing on nilm: An unsupervised approach featuring power sequences," *Sensors*, vol. 23, no. 8, p. 3939, Apr. 2023. [Online]. Available: http://dx.doi.org/10.3390/s23083939

[53] N. Henao, K. Agbossou, S. Kelouwani, Y. Dube, and M. Fournier, "Approach in nonintrusive type i load monitoring using subtractive clustering," *IEEE Transactions on Smart Grid*, pp. 812–821, 2017. [Online]. Available: http://dx.doi.org/10.1109/TSG.2015.2462719

[54] B. Liu, W. Luan, J. Yang, and Y. Yu, "The balanced window-based load event optimal matching for nilm," *IEEE Transactions on Smart Grid*, vol. 13, no. 6, p. 4690–4703, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1109/TSG.2022.3179440

[55] K. He, D. Jakovetic, B. Zhao, V. Stankovic, L. Stankovic, and S. Cheng, "A generic optimisation-based approach for improving non-intrusive load

monitoring," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, p. 6472–6480, Nov. 2019. [Online]. Available: http://dx.doi.org/10.1109/TSG.2019.2906012

[56] M. Figueiredo, B. Ribeiro, and A. de Almeida, "Electrical signal source separation via nonnegative tensor factorization using on site measurements in a smart home," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 2, p. 364–373, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1109/TIM.2013.2278596

[57] C. Dinesh, S. Welikala, Y. Liyanage, M. P. B. Ekanayake, R. I. Godaliyadda, and J. Ekanayake, "Non-intrusive load monitoring under residential solar power influx," *Applied Energy*, vol. 205, p. 1068–1080, Nov. 2017. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2017.08.094

[58] M. Zeifman, "Disaggregation of home energy display data using probabilistic approach," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 1, p. 23–31, Feb. 2012. [Online]. Available: http://dx.doi.org/10.1109/TCE.2012.6170051

[59] 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*, vol. 25. Citeseer, 2011, pp. 59–62. [Online]. Available: https://zicokolter.com/publications/kolter2011redd.pdf

[60] J. Kelly and W. Knottenbelt, "The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific Data*, vol. 2, no. 1, Mar. 2015. [Online]. Available: http://dx.doi.org/10.1038/sdata.2015.7

[61] K. Zhou, Z. Zhang, and X. Lu, "Non-intrusive load monitoring based on an efficient deep learning model with local feature extraction," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 7, p. 9497–9507, Jul. 2024. [Online]. Available: http://dx.doi.org/10.1109/TII.2024.3383521

[62] J. Xiong, T. Hong, D. Zhao, and Y. Zhang, "Matnilm: Multi-appliance-task non-intrusive load monitoring with limited labeled data," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, p. 3177–3187, Mar. 2024. [Online]. Available: http://dx.doi.org/10.1109/TII.2023.3301026

[63] S. Makonin and F. Popowich, "Nonintrusive load monitoring (nilm) performance evaluation: A unified approach for accuracy reporting," *Energy Efficiency*, vol. 8, no. 4, p. 809–814, Oct. 2014. [Online]. Available: http://dx.doi.org/10.1007/s12053-014-9306-2

[64] Q. Luo, T. Yu, C. Lan, Y. Huang, Z. Wang, and Z. Pan, "A generalizable method for practical non-intrusive load monitoring via metric-based meta-learning," *IEEE Transactions on Smart Grid*, vol. 15, no. 1, p. 1103–1115, Jan. 2024. [Online]. Available: http://dx.doi.org/10.1109/TSG.2023.3263814

[65] D. Murray, L. Stankovic, V. Stankovic, S. Lulic, and S. Sladojevic, "Transferability of neural network approaches for low-rate energy disaggregation," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2019. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2019.8682486

[66] A. Reinhardt and M. Bouchur, "On the impact of the sequence length on sequence-to-sequence and sequence-to-point learning for nilm," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, ser. BuildSys '20. ACM, Nov. 2020. [Online]. Available: http://dx.doi.org/10.1145/3427771.3427857

[67] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[68] A. Gao, J. Zheng, F. Mei, H. Sha, Y. Xie, K. Li, and Y. Liu, "Non-intrusive multi-label load monitoring via transfer and contrastive learning architecture," *International Journal of Electrical Power and Energy Systems*, vol. 154, p. 109443, Dec. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.ijepes.2023.109443

[69] T. Chen, J. Gao, Y. Yuan, S. Guo, and P. Yang, "Non-intrusive load monitoring based on mocov2, time series self-supervised learning," *Energy and Buildings*, vol. 317, p. 114374, Aug. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2024.114374

[70] A. Nilsson, M. Wester, D. Lazarevic, and N. Brandt, "Smart homes, home energy management systems and real-time feedback: Lessons for

influencing household energy consumption from a swedish field study," *Energy and Buildings*, vol. 179, p. 15–25, Nov. 2018. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2018.08.026

[71] 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*, vol. 14, no. 3, p. 767, Feb. 2021. [Online]. Available: http://dx.doi.org/10.3390/en14030767

[72] C. L. Athanasiadis, T. A. Papadopoulos, and D. I. Doukas, "Real-time non-intrusive load monitoring: A light-weight and scalable approach," *Energy and Buildings*, vol. 253, p. 111523, Dec. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2021.111523

[73] P. A. Schirmer and I. Mporas, "Double fourier integral analysis based convolutional neural network regression for high-frequency energy disaggregation," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 3, p. 439–449, Jun. 2022. [Online]. Available: http://dx.doi.org/10.1109/TETCI.2021.3086226

[74] W. Luan, Z. Wei, B. Liu, and Y. Yu, "Non-intrusive power waveform modeling and identification of air conditioning load," *Applied Energy*, vol. 324, p. 119755, Oct. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2022.119755

[75] Y. Liu, W. Liu, Y. Shen, X. Zhao, and S. Gao, "Toward smart energy user: Real time non-intrusive load monitoring with simultaneous switching operations," *Applied Energy*, vol. 287, p. 116616, Apr. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2021.116616

[76] S. Welikala, N. Thelasingha, M. Akram, P. B. Ekanayake, R. I. Godaliyadda, and J. B. Ekanayake, "Implementation of a robust real-time non-intrusive load monitoring solution," *Applied Energy*, vol. 238, p. 1519–1529, Mar. 2019. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2019.01.167

[77] R. Machlev, Y. Levron, and Y. Beck, "Modified cross-entropy method for classification of events in nilm systems," *IEEE Transactions on*

Smart Grid, vol. 10, no. 5, pp. 4962–4973, 2019. [Online]. Available: http://dx.doi.org/10.1109/TSG.2018.2871620

[78] R. Machlev, J. Belikov, Y. Beck, and Y. Levron, "MO-NILM: A multi-objective evolutionary algorithm for NILM classification," *Energy and Buildings*, vol. 199, pp. 134–144, 2019. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2019.06.046

[79] M. Z. A. Bhotto, S. Makonin, and I. V. Bajic, "Load disaggregation based on aided linear integer programming," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 7, p. 792–796, Jul. 2017. [Online]. Available: http://dx.doi.org/10.1109/TCSII.2016.2603479

[80] F. M. Wittmann, J. C. Lopez, and M. J. Rider, "Nonintrusive Load Monitoring Algorithm Using Mixed-Integer Linear Programming," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 2, pp. 180–187, 2018. [Online]. Available: http://dx.doi.org/10.1109/TCE.2018.2843292

[81] S. Zeinal-Kheiri, A. M. Shotorbani, and B. Mohammadi-Ivatloo, "Residential load disaggregation considering state transitions," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, p. 743–753, Feb. 2020. [Online]. Available: http://dx.doi.org/10.1109/TII.2019.2925323

[82] N. Henao, S. Kelouwani, K. Agbossou, and Y. Dubé, "Active power load modeling based on uncertainties for non intrusive load monitoring," in *2016 IEEE 25th International Symposium on Industrial Electronics (ISIE)*, 2016, pp. 684–689. [Online]. Available: http://dx.doi.org/10.1109/ISIE.2016.7744972

[83] P. Coretto and C. Hennig, "Robust Improper Maximum Likelihood: Tuning, Computation, and a Comparison With Other Methods for Robust Gaussian Clustering," *Journal of the American Statistical Association*, vol. 111, no. 516, pp. 1648–1659, 2016. [Online]. Available: http://dx.doi.org/10.1080/01621459.2015.1100996

[84] ——, "Consistency, breakdown robustness, and algorithms for robust improper maximum likelihood clustering," *Journal of Machine Learning*

Research, vol. 18, no. 142, pp. 1–39, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-382.html

[85] D. J. C. MacKay, *Information theory, inference and learning algorithms*. Cambridge, England: Cambridge University Press, Sep. 2003. [Online]. Available: https://www.inference.org.uk/itprnn/book.pdf

[86] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016. [Online]. Available: https://www.jmlr.org/papers/volume17/15-408/15-408.pdf

[87] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, "A rewriting system for convex optimization problems," *Journal of Control and Decision*, vol. 5, no. 1, pp. 42–60, 2018. [Online]. Available: https://web.stanford.edu/~boyd/papers/pdf/cvxpy_rewriting.pdf

[88] S. Baluja, *Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning*. School of Computer Science, Carnegie Mellon University Pittsburgh, PA, 1994. [Online]. Available: https://www.ri.cmu.edu/pub_files/pub1/baluja_shumeet_1994_2/baluja_shumeet_1994_2.pdf

[89] S. Yang and X. Yao, "Population-based incremental learning with associative memory for dynamic environments," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 5, pp. 542–561, 2008. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2007.913070

[90] N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, A. Singh, and M. Srivastava, "NILMTK: An open source toolkit for non-intrusive load monitoring," *e-Energy 2014 - Proceedings of the 5th ACM International Conference on Future Energy Systems*, pp. 265–276, 2014. [Online]. Available: http://dx.doi.org/10.1145/2602044.2602051

[91] R. Sonderegger, "The riva meter and platform," May 2016, presented at the 3rd International Workshop on Non-Intrusive Load Monitoring. [Online]. Available: http://nilmworkshop.org/2016/slides/Itron.pdf

[92] J. D. Bastidas-Rodriguez, E. Franco, G. Petrone, C. A. Ramos-Paja, and G. Spagnuolo, "Model-based degradation analysis of photovoltaic modules through series resistance estimation," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 11, pp. 7256–7265, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1109/TIE.2015.2459380

[93] B. Ando, S. Baglio, A. Pistorio, G. M. Tina, and C. Ventura, "Sentinella: Smart monitoring of photovoltaic systems at panel level," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 8, p. 2188–2199, Aug. 2015. [Online]. Available: http://dx.doi.org/10.1109/TIM.2014.2386931

[94] G. Petrone, C. A. Ramos-Paja, and G. Spagnuolo, *Photovoltaic Sources Modeling*. Chichester, West Sussex, UK: John Wiley & Sons, 2017, ISBN: 978-1-118-75612-6. [Online]. Available: https://doi.org/10.1002/9781118755877

[95] M. G. Villalva, J. R. Gazoli, and E. R. Filho, "Comprehensive approach to modeling and simulation of photovoltaic arrays," *IEEE T Power Electr*, vol. 24, no. 5, pp. 1198–1208, 2009. [Online]. Available: https://doi.org/10.1109/TPEL.2009.2013862

[96] G. Ciulla, V. Lo Brano, V. Di Dio, and G. Cipriani, "A comparison of different one-diode models for the representation of I-–V characteristic of a PV cell," *Renew Sust Energ Rev*, vol. 32, pp. 684–696, 2014. [Online]. Available: https://doi.org/10.1016/j.rser.2014.01.027

[97] C. S. Ruschel, F. P. Gasparin, E. R. Costa, and A. Krenzinger, "Assessment of PV modules shunt resistance dependence on solar irradiance," *Sol Energy*, vol. 133, pp. 35–43, 2016. [Online]. Available: https://doi.org/10.1016/j.solener.2016.03.047

[98] M. Laurino, M. Piliougine, and G. Spagnuolo, "Artificial neural network based photovoltaic module diagnosis by current–voltage curve classification," *Solar Energy*, vol. 236, pp. 383–392, 2022. [Online]. Available: https://doi.org/10.1016/j.solener.2022.02.039

[99] C. S. Ruschel, F. P. Gasparin, and A. Krenzinger, "Experimental analysis of the single diode model parameters dependence on irradiance and

temperature," *Solar Energy*, vol. 217, p. 134–144, Mar. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2021.01.067

[100] W. De Soto, S. Klein, and W. Beckman, "Improvement and validation of a model for photovoltaic array performance," *Solar Energy*, vol. 80, no. 1, p. 78–88, Jan. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2005.06.010

[101] K. J. Sauer, T. Roessler, and C. W. Hansen, "Modeling the irradiance and temperature dependence of photovoltaic modules in pvsyst," *IEEE Journal of Photovoltaics*, vol. 5, no. 1, p. 152–158, Jan. 2015. [Online]. Available: http://dx.doi.org/10.1109/JPHOTOV.2014.2364133

[102] S. P. Aly, S. Ahzi, and N. Barth, "An adaptive modelling technique for parameters extraction of photovoltaic devices under varying sunlight and temperature conditions," *Applied Energy*, vol. 236, p. 728–742, Feb. 2019. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2018.12.036

[103] M. Piliougine, L. E. Garcia-Marrero, K. Lappalainen, and G. Spagnuolo, "Influence of the temperature on the intrinsic parameters of thin-film photovoltaic modules," *Renewable Energy*, vol. 240, p. 122068, Feb. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2024.122068

[104] M. Köntges, S. Kurtz, C. Packard, U. Jahn, K. A. Berger, K. Kato, T. Friesen, H. Liu, and M. V. Iseghem, "Review of failures of photovoltaic modules," International Energy Agency Photovoltaic Power Systems Programme (IEA PVPS), Tech. Rep. IEA-PVPS T13-01:2014, 2014. [Online]. Available: https://iea-pvps.org/wp-content/uploads/2020/01/IEA-PVPS_T13-01_2014_Review_of_Failures_of_Photovoltaic_Modules_Final.pdf

[105] D. Wei, M. Wei, H. Cai, X. Zhang, and L. Chen, "Parameters extraction method of pv model based on key points of i-v curve," *Energy Conversion and Management*, vol. 209, p. 112656, Apr. 2020. [Online]. Available: http://dx.doi.org/10.1016/j.enconman.2020.112656

[106] D. Saadaoui, M. Elyaqouti, K. Assalaou, D. B. Hmamou, S. Lidaighbi, E. H. Arjdal, I. Choulli, and A. Elhammoudy, "Parameters extraction of single

diode and double diode models using analytical and numerical approach: A comparative study," *International Journal of Modelling and Simulation*, p. 1–30, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1080/02286203.2023.2226285

[107] S. Das, A. Hazra, and M. Basu, "Metaheuristic optimization based fault diagnosis strategy for solar photovoltaic systems under non-uniform irradiance," *Renewable Energy*, vol. 118, p. 452–467, Apr. 2018. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2017.10.053

[108] H. Kalliojärvi-Viljakainen, K. Lappalainen, and S. Valkealahti, "A novel procedure for identifying the parameters of the single-diode model and the operating conditions of a photovoltaic module from measured current–voltage curves," *Energy Reports*, vol. 8, p. 4633–4640, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.egyr.2022.03.141

[109] J. Bastidas-Rodriguez, G. Petrone, C. Ramos-Paja, and G. Spagnuolo, "A genetic algorithm for identifying the single diode model parameters of a photovoltaic panel," *Mathematics and Computers in Simulation*, vol. 131, p. 38–54, Jan. 2017. [Online]. Available: http://dx.doi.org/10.1016/j.matcom.2015.10.008

[110] E. I. Batzelis and S. A. Papathanassiou, "A method for the analytical extraction of the single-diode pv model parameters," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 2, p. 504–512, Apr. 2016. [Online]. Available: http://dx.doi.org/10.1109/TSTE.2015.2503435

[111] L. H. I. Lim, Z. Ye, J. Ye, D. Yang, and H. Du, "A linear identification of diode models from single i– v characteristics of pv panels," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 7, p. 4181–4193, Jul. 2015. [Online]. Available: http://dx.doi.org/10.1109/TIE.2015.2390193

[112] M. Nacar, E. Özer, and A. E. Yılmaz, "A six parameter single diode model for photovoltaic modules," *Journal of Solar Energy Engineering*, vol. 143, no. 1, Aug. 2020. [Online]. Available: http://dx.doi.org/10.1115/1.4047853

[113] P. Hao and Y. Zhang, "An improved method for parameter identification and performance estimation of pv modules from manufacturer datasheet

based on temperature-dependent single-diode model," *IEEE Journal of Photovoltaics*, vol. 11, no. 6, p. 1446–1457, Nov. 2021. [Online]. Available: http://dx.doi.org/10.1109/JPHOTOV.2021.3114592

[114] S. Lidaighbi, M. Elyaqouti, D. Ben Hmamou, D. Saadaoui, K. Assalaou, and E. Arjdal, "A new hybrid method to estimate the single-diode model parameters of solar photovoltaic panel," *Energy Conversion and Management: X*, vol. 15, p. 100234, Aug. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.ecmx.2022.100234

[115] J. Xu, C. Zhou, and W. Li, "Photovoltaic single diode model parameter extraction by di/dv-assisted deterministic method," *Solar Energy*, vol. 251, p. 30–38, Feb. 2023. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2023.01.009

[116] C. Ren, Z. Song, and Z. Meng, "Photovoltaic model parameters identification using diversity improvement-oriented differential evolution," *Swarm and Evolutionary Computation*, vol. 90, p. 101689, Oct. 2024. [Online]. Available: http://dx.doi.org/10.1016/j.swevo.2024.101689

[117] E. S. Hasyim, S. Wenham, and M. Green, "Shadow tolerance of modules incorporating integral bypass diode solar cells," *Solar Cells*, vol. 19, no. 2, pp. 109–122, 1986. [Online]. Available: https://doi.org/10.1016/0379-6787(86)90036-0

[118] S. M. Abdulrahim, Z. Ahmad, J. Bahadra, and N. J. Al-Thani, "Electrochemical impedance spectroscopy analysis of hole transporting material free mesoporous and planar perovskite solar cells," *Nanomaterials*, vol. 10, 2020. [Online]. Available: https://doi.org/10.3390/nano10091635

[119] L. Shelembe and P. Barendse, "Online condition monitoring of photovoltaic (PV) cells by implementing electrical impedance spectroscopy using a switch-mode DC-DC converter," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, 2020. [Online]. Available: https://doi.org/10.1109/ecce44975.2020.9235596

[120] A. R. Jordehi, "Parameter estimation of solar photovoltaic (pv) cells: A review," *Renewable and Sustainable Energy Reviews*, vol. 61, pp. 354–371,

2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032116300016

[121] R. A. Guejia-Burbano, G. Petrone, and M. Piliougine, "Impedance spectroscopy for diagnosis of photovoltaic modules under outdoor conditions," *IEEE Journal of Photovoltaics*, vol. 12, no. 6, p. 1503–1512, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1109/JPHOTOV.2022.3195003

[122] A. Lasia, *Electrochemical Impedance Spectroscopy and its Applications*. Springer New York, 8 2014, vol. 9781461489337. [Online]. Available: https://link.springer.com/10.1007/978-1-4614-8933-7

[123] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, M. Yin, and M. Gong, "A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 275–298, April 2016. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2015.2455812

[124] L. He, H. Ishibuchi, A. Trivedi, H. Wang, Y. Nan, and D. Srinivasan, "A survey of normalization methods in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 6, pp. 1028–1048, 2021. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2021.3076514

[125] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, Oct 2006. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2005.861417

[126] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, April 2002. [Online]. Available: http://dx.doi.org/10.1109/4235.996017

[127] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*,

vol. 11, no. 6, pp. 712–731, Dec 2007. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2007.892759

[128] J. Blank and K. Deb, "Pymoo: Multi-objective optimization in python," *IEEE Access*, vol. 8, p. 89497–89509, 2020. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2020.2990567

[129] P. Singh and N. Ravindra, "Temperature dependence of solar cell performance—an analysis," *Solar Energy Materials and Solar Cells*, vol. 101, pp. 36–45, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0927024812000931

[130] Y. Varshni, "Temperature dependence of the energy gap in semiconductors," *Physica*, vol. 34, no. 1, pp. 149–154, 1967. [Online]. Available: https://doi.org/10.1016/0031-8914(67)90062-6

[131] Solbian, *Flex SP50–L Flexible Photovoltaic Panels*. [Online]. Available: https://www.dropbox.com/s/vkj1nidxuoac5nq/Calculationsolar_module_SOLBIAN_SP50L179.pdf

[132] J. P. Ignizio, *Goal Programming and Extensions*. Lexington Books, 1976.

[133] M. Piliougine and G. Spagnuolo, "Mismatching and partial shading identification in photovoltaic arrays by an artificial neural network ensemble," *Solar Energy*, vol. 236, p. 712–723, Apr. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2022.03.026

[134] J. Bishop, "Computer simulation of the effects of electrical mismatches in photovoltaic cell interconnection circuits," *Solar Cells*, vol. 25, no. 1, p. 73–89, Oct. 1988. [Online]. Available: http://dx.doi.org/10.1016/0379-6787(88)90059-2

[135] E. Piccoli, A. Dama, A. Dolara, and S. Leva, "Experimental validation of a model for pv systems under partial shading for building integrated applications," *Solar Energy*, vol. 183, p. 356–370, May 2019. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2019.03.015

[136] X. Qing, H. Sun, X. Feng, and C. Y. Chung, "Submodule-based modeling and simulation of a series-parallel photovoltaic array under mismatch conditions,"

IEEE Journal of Photovoltaics, vol. 7, no. 6, p. 1731–1739, Nov. 2017. [Online]. Available: http://dx.doi.org/10.1109/JPHOTOV.2017.2746265

[137] M. A. Mikofski, M. Lynn, J. Byrne, M. Hamer, A. Neubert, and J. Newmiller, "Accurate performance predictions of large pv systems with shading using submodule mismatch calculation," in *2018 IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC)*. IEEE, Jun. 2018, p. 3635–3639. [Online]. Available: http://dx.doi.org/10.1109/PVSC.2018.8547323

[138] E. D. Chepp, F. P. Gasparin, and A. Krenzinger, "Improvements in methods for analysis of partially shaded pv modules," *Renewable Energy*, vol. 200, p. 900–910, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2022.10.035

[139] Y. Li, K. Ding, J. Zhang, F. Chen, X. Chen, and J. Wu, "A fault diagnosis method for photovoltaic arrays based on fault parameters identification," *Renewable Energy*, vol. 143, p. 52–63, Dec. 2019. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2019.04.147

[140] L. M. Pérez Archila, J. D. Bastidas Rodríguez, and R. Correa, "Implicit modelling of series-parallel photovoltaic arrays using double-diode model and its solution," *Solar Energy*, vol. 214, pp. 131–137, 2021. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2020.11.036

[141] D. Yousri, A. Fathy, H. Rezk, T. S. Babu, and M. R. Berber, "A reliable approach for modeling the photovoltaic system under partial shading conditions using three diode model and hybrid marine predators-slime mould algorithm," *Energy Conversion and Management*, vol. 243, p. 114269, Sep. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.enconman.2021.114269

[142] H. Kawamura, K. Naka, N. Yonekura, S. Yamanaka, H. Kawamura, H. Ohno, and K. Naito, "Simulation of i–v characteristics of a pv module with shaded pv cells," *Solar Energy Materials and Solar Cells*, vol. 75, no. 3–4, p. 613–621, Feb. 2003. [Online]. Available: http://dx.doi.org/10.1016/S0927-0248(02)00134-4

[143] MathWorks, *Solar Cell*, 2025, accessed: 2025-02-17. [Online]. Available: https://www.mathworks.com/help/sps/ref/solarcell.html

[144] E. Skoplaki, A. Boudouvis, and J. Palyvos, "A simple correlation for the operating temperature of photovoltaic modules of arbitrary mounting," *Solar Energy Materials and Solar Cells*, vol. 92, no. 11, p. 1393–1402, Nov. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.solmat.2008.05.016

[145] D. Dirnberger, *Photovoltaic module measurement and characterization in the laboratory.* Elsevier, 2017, p. 23–70. [Online]. Available: http://dx.doi.org/10.1016/B978-1-78242-336-2.00002-1

[146] N. Aoun, "Methodology for predicting the pv module temperature based on actual and estimated weather data," *Energy Conversion and Management: X*, vol. 14, p. 100182, May 2022. [Online]. Available: http://dx.doi.org/10.1016/j.ecmx.2022.100182

[147] H. Ibrahim and N. Anani, "Variations of pv module parameters with irradiance and temperature," *Energy Procedia*, vol. 134, p. 276–285, Oct. 2017. [Online]. Available: http://dx.doi.org/10.1016/j.egypro.2017.09.617

[148] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, "On the LambertW function," *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 329–359, 1996. [Online]. Available: https://doi.org/10.1007/BF02124750

[149] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization.* Springer Berlin, Heidelberg, 2005. [Online]. Available: http://dx.doi.org/10.1007/3-540-31306-0

[150] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, Feb 2011. [Online]. Available: https://doi.org/10.1109/TEVC.2010.2059031

[151] ITSensor, *LITEMETER LM1-10V PRO.* [Online]. Available: https://www.dropbox.com/s/3n7ah2ffkcnj2qb/LiteMeter.pdf

[152] Advantech, "User manual adam-6000 series," 2022. [Online]. Available: https://advdownload.advantech.com/productfile/Downloadfile4/1-2B6FKTG/ADAM-6000_User_Manaul_Ed.12-FINAL.pdf

[153] Y. Sangpongsanont, D. Chenvidhya, S. Chuangchote, and K. Kirtikara, "Corrosion growth of solar cells in modules after 15 years of operation," *Solar Energy*, vol. 205, p. 409–431, Jul. 2020. [Online]. Available: http://dx.doi.org/10.1016/j.solener.2020.05.016

[154] H. Kalliojärvi and K. Lappalainen, "Methods for online identification of photovoltaic module ageing by series resistance from measured current–voltage curves," *Energy Reports*, vol. 13, p. 1558–1570, Jun. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.egyr.2025.01.027

[155] Shelly, "Shelly EM + 50A Clamp," https://www.shelly.com/products/shelly-em-50a-clamp-1, 2025, accessed: 2025-07-26.