Dissertation

An Economic Evaluation of Supervised Learning for Short-term Forecasting and Control of Distributed Energy Resources

in fulfillment of the requirements for the degree of
Doktor der technischen Wissenschaften

submitted at the
Energy Economics Group
Faculty of Electrical Engineering and Information Technology
Technische Universität Wien

by

Nikolaus Houben

Supervisor:

Prof. Dr. Amela Ajanovic
Technische Universität Wien

Reviewer & Examiner:

Prof. Dr. Bert Claessens
Ghent University

Prof. Dr. Angèle Reinders
Eindhoven University of Technology

Vienna, July 2025

Abstract

This thesis investigates the application and economic evaluation of supervised learning methods for short-term energy forecasting and the control of distributed energy resources. The core objective is addressed by answering three interconnected research questions, leading to the following contributions: i) Benchmarking existing supervised learning forecasting methods using both statistical and a novel application-driven economic metric for peak load management; ii) Developing an interpretable and data-efficient model-based deep learning approach for net load forecasting in households with behind-the-meter photovoltaic systems, battery storage, and energy management systems; and iii) Evaluating the economic impact of forecast errors on the dispatch of a multi-energy microgrid using economic Model Predictive Control under various tariff structures. Key findings indicate that while tree-based models often exhibit superior average statistical accuracy, neural networks can be more effective for economic objectives, such as peak shaving. Moreover, by incorporating physical priors into deep learning models net load forecasting accuracy, interpretability, and data efficiency are significantly improved. Furthermore, the analysis of a microgrid reveals that while optimized control offers substantial cost and CO_2 savings, forecast uncertainty diminishes these achievable benefits when compared to a perfect foresight scenario, especially for systems with greater flexibility or those subject to tariffs like demand charges. This work emphasizes the need to co-design and evaluate forecasting methods within their specific application context to unlock their full potential in future energy systems. Finally recommendations to stakeholders are offered: grid operators should use application-specific metrics for tasks like peak shaving; energy retailers need models understanding control logic of prosumers; and end-users must combine energy assets with advanced forecasting and control frameworks.

Kurzfassung

Diese Dissertation untersucht die ökonomische Bewertung und Anwendung von *Supervised Learning* Methoden für die kurzfristige Energieprognose und Steuerung dezentraler Energieressourcen. Die Arbeit adressiert dies durch: i) ein Benchmarking von Prognosemethoden mittels statistischer und ökonomischer Metriken für Spitzenlastmanagement; ii) die Entwicklung interpretierbarer Deep-Learning-Modelle für Prosumer-lasten; iii) die Analyse ökonomischer Auswirkungen von Prognosefehlern in einem Microgrid.

Wichtige Ergebnisse sind: Obwohl *tree-based* Modelle oft statistisch genauer sind, können neuronale Netze für ökonomische Ziele wie Spitzenlastreduktion, besonders bei längeren Horizonten, vorteilhafter sein. Die Integration physikalischen Vorwissens in Deep-Learning-Modelle steigert die Genauigkeit, Interpretierbarkeit und Dateneffizienz von Nettolastprognosen bedeutend. Im Microgrid ermöglicht optimierte Regelung zwar erhebliche Kosten- und CO_2-Einsparungen, Prognoseunsicherheit reduziert diese Vorteile jedoch deutlich, insbesondere bei hoher Systemflexibilität oder komplexen Tarifen wie Leistungspreisen.

Die Arbeit unterstreicht die Notwendigkeit, Prognosemethoden im spezifischen Anwendungskontext zu entwerfen und zu evaluieren. Empfehlungen für Stakeholder umfassen: die Nutzung anwendungsspezifischer Metriken durch Netzbetreiber, die Entwicklung von Modellen mit Verständnis für Prosumer-Steuerlogik für Energiehändler, sowie die Kombination von Energieanlagen mit fortschrittlichen Prognose- und Regelungskonzepten durch Endnutzer.

Contents

Abstract I

1. Introduction 5
- 1.1. Motivation . . . 5
- 1.2. Core Objective and Research Questions . . . 8
- 1.3. Structure of the Thesis . . . 12

2. Review of the State-of-the-Art 14
- 2.1. Time Series Forecasting Approaches . . . 14
- 2.2. Forecasting Horizon and Multi-step Forecasting methods . . . 17
- 2.3. Energy Forecasting Applications and Challenges . . . 18
  - 2.3.1. Temporal Scales . . . 19
  - 2.3.2. Spatial Scales . . . 20
  - 2.3.3. Net Load Forecasting . . . 22
  - 2.3.4. Model Interpretability & Data Efficiency . . . 23
- 2.4. Evaluation of Forecasting Performance . . . 25
  - 2.4.1. Conventional Error Metrics and Their Limitations . . . 25
  - 2.4.2. Economic Evaluation of Energy Forecasts . . . 26
- 2.5. Integration of Forecasts into Control Algorithms . . . 27
  - 2.5.1. Sequential Decision Making under Uncertainty . . . 27
  - 2.5.2. Model Predictive Control . . . 30
  - 2.5.3. Reinforcement Learning . . . 31
- 2.6. Summary of Gaps in the Literature & Contributions . . . 32

3. Methods 36
- 3.1. General Methods & Preliminaries . . . 38
  - 3.1.1. Supervised Learning for Energy Forecasting . . . 38
  - 3.1.2. Economic Model Predictive Control (eMPC) . . . 45

3.2. Methods Per Research Question 51
3.2.1. RQ1: Evaluating Forecasting Errors through the Net Load Error Framework 53
3.2.2. RQ2: Integration of Household Energy Management into Net Load Forecasting Models 60
3.2.3. RQ3: Optimal Operational Dispatch of a Multi-Energy Microgrid under Uncertainty 66
3.2.4. Experimental Setup 71

4. Case Studies and Results **72**
4.1. RQ1: Benchmarking SL Models and Evaluating Economic Impact 73
4.1.1. Case Study Description 73
4.1.2. Results 75
4.2. RQ2: Interpretable Net Load Forecasting 84
4.2.1. Case Study Description 85
4.2.2. Results 89
4.3. RQ3: Optimal Dispatch of a Multi-Energy Microgrid under Uncertainty 93
4.3.1. Case Study Description 93
4.3.2. Results 99

5. Synthesis **107**
5.1. Advancements in Forecasting: From Benchmarking to Application-Specific Models 108
5.2. Integrating Forecasting and Control: Performance, Uncertainty, and Value 111
5.3. Implications for Stakeholders 113
5.4. Limitations and Future Research Directions 115
5.4.1. Limitations of the Presented Research 115
5.4.2. Future Research Directions 116

6. Conclusions and Outlook **119**

7. References **121**
Books 121
Journal Articles 122

Conference Papers 138
Other sources 140

Appendices **142**

A. Mathematical Details of Machine Learning Algorithms **143**
A.1. Linear Regression 143
A.2. Random Forest 144
A.3. XGBoost & LightGBM 145
A.4. Gated Recurrent Unit (GRU) 148
A.5. N-BEATS 151
A.6. Temporal Fusion Transformer (TFT) 153

B. Optimization Input Data, System Parameters, and Net Load Error Details **156**
B.1. Optimization Input Data from Case Study (RQ3) 156
B.1.1. Domestic-Hot-Water Profile Generation (SAX Method) 161
B.1.2. Techno-Economical Parameters for REC Case Study (RQ3) 162
B.2. Hardware and Tariff Information of Households (RQ2) 163
B.2.1. NLE Data Assumptions 164
B.3. Hyperparameters 165
B.3.1. Neural Network Dropout Values (RQ1) 165
B.3.2. Darts and XGBoost Hyperparameters (RQ2) 165
B.4. Weights & Biases Repository Supplementary (RQ1) 166

C. Supplementary Results **167**
C.1. Supplementary Forecasting Results (RQ3) 167
C.2. Detailed Optimization and Economic Analysis Results (RQ3) 168
C.3. Declaration of Authorship and Originality 171
C.4. Use of Artificial Intelligence 171

Glossary

Abbreviations

Abbreviation	Explanation
BESS	Battery Energy Storage System
BTM	Behind-The-Meter
CO2	Carbon Dioxide
DERs	Distributed Energy Resources
DHS	Domestic Heat Storage
DL	Deep Learning
eMPC	Economic Model Predictive Control
EVs	Electric Vehicles
FT	Flat Tariff
FT-DC	Flat Tariff + Demand Charge
GRU	Gated Recurrent Unit
HEMS	Home Energy Management System
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MILP	Mixed-Integer Linear Program
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
MSE	Mean Squared Error
N-BEATS	Neural Basis Expansion Analysis Time Series
NLE	Net Load Error
NN	Neural Network
nRMSE	Normalized Root Mean Squared Error
PF	Perfect Foresight
PtH	Power-to-Heat

Abbreviation	Explanation
PV	Photovoltaic
R2	Coefficient of Determination
RC	Reference Case (Rule-Based Control)
REC	Renewable Energy Community
RF	Random Forest
RMSE	Root Mean Squared Error
RQ	Research Question
SL	Supervised Learning
SoC / SoE	State of Charge / State of Energy
SS	Skill Score
TFT	Temporal Fusion Transformer
TiDE	Time-series Dense Encoder
TOU	Time-of-Use
TOU-DC	Time-of-Use + Demand Charge
XGBoost	Extreme Gradient Boosting

Nomenclature

Symbol	Explanation
Δt	Time step duration
ϵ	Aggregate error score
$\epsilon_{k\|t}$	Forecast error for lead time k made at time t ($\epsilon_{k\|t} = y_{t+k} - \hat{y}_{k\|t}$)
ϵ_t^{net}	Net forecast error at time t ($\epsilon_t^{net} = \epsilon_t^{load} - \epsilon_t^{PV}$)
η	Efficiency (e.g., BESS round-trip efficiency $\eta = 0.86$)
η_{inv}	Inverter efficiency
$\eta^{a,ch}$	Charging efficiency for asset 'a'
$\eta^{a,dis}$	Discharging efficiency for asset 'a'
$\hat{\pi}_{k\|t}^{buy/sell}$	Forecast electricity purchase/sell price for time $t+k$
$\pi_{t+k}^{buy/sell}$	Actual electricity purchase/sell price at time $t+k$
π^{DC}	Demand charge rate
θ	Forecasting model parameters
τ	Step index within the look-back window

Symbol	Explanation
ϕ	Forecasting model hyperparameters
C^{opr}_{actual}	Total operational cost using actual forecasts
C^{opr}_{ideal}	Total operational cost using ideal (perfect) forecasts
C_{rates}	Cost from energy exchange
C_{DC}	Cost from demand charge
C_{total}	Total operational cost
Cap_a	Capacity of asset 'a'
D	Evaluation metric function
$\mathcal{D}$	Training dataset
E_k	Energy stored at time step k
E^{max}	Maximum energy storage capacity
E^{cap}	Battery capacity
E^a_k	Energy stored in asset 'a' at step k
E^{BESS}_k	Energy stored in BESS at step k
E^{DHS}_k	Energy stored in DHS at step k
$\mathcal{F}$	Forecasting model function
$\mathcal{F}_{1step}$	One-step-ahead forecasting model
$\mathcal{F}^{(k)}$	Direct forecasting model for lead time k
$\mathcal{F}_{MIMO}$	Multi-Input Multi-Output forecasting model
f	State transition function
$g(\cdot)$	General path constraints function
H	Control and Forecast horizon
k	Lead time index within the forecast horizon
L	Look-back window length
$l(\cdot)$	Stage cost function (in eMPC)
l^{energy}_k	Energy cost component of stage cost
l^{DC}_k	Demand charge component of stage cost
N	Number of forecast origins in the evaluation dataset
n_u	Dimension of the control input vector
n_x	Dimension of the state vector
p^h	Auxiliary peak power variable in optimization
P^{BESS}_k	BESS charging/discharging power at step k
P^{max}	Maximum BESS power rating
$P^{ch,max}$	Maximum charging power
$P^{dis,max}$	Maximum discharging power

Symbol	Explanation
P_k^{grid}	Net power exchanged with the grid at step k (optimization)
$P_t^{grid,opr}$	Operational net grid power realized at time t
$P_k^{a,ch}$	Charging power for asset 'a' at step k
$P_k^{a,dis}$	Discharging power for asset 'a' at step k
P_{STC}	PV power at Standard Test Conditions
T	Length of time series
t	Time step index (forecast origin / control step)
$\mathcal{U}$	Feasible control input set
u_k	Control input vector at step k
u_0^*	First optimal control action
$V_f(\cdot)$	Terminal cost function
$\mathcal{X}$	Feasible state set
$X^{(t)}$	Input features for time t
x_k	State vector at step k
$x_{current}$	Current system state
x^{past}	Historical input features
x_k^{future}	Known future input features for lead time k
y_{t+k}	Actual target value at time $t+k$
$y^{(t)}$	Sequence of actual target values $\{y_{t+k}\}_{k=1}^{H}$
$\hat{y}_{k\|t}$	Forecast value for lead time k made at time t
$\hat{y}^{(t)}$	Sequence of forecast values $\{\hat{y}_{k\|t}\}_{k=1}^{H}$
$\hat{y}_{k\|t}^{load}$	Forecast electrical load
$\hat{y}_{k\|t}^{PV}$	Forecast PV generation
$\hat{y}_{k\|t}^{DHW}$	Forecast Domestic Hot Water load
$\hat{y}_{k\|t}^{net}$	Forecast net load
$\overline{y}$	Mean of ground truth values in evaluation set
y_t^{load}	Actual non-flexible electrical load at time t
y_t^{prod}	Actual PV production at time t
y_t^{net}	Actual net load at time t
y_t^{PV}	Actual PV generation at time t
y_t^{DHW}	Actual Domestic Hot Water demand at time t
z_k	Binary decision variable vector at step k

1. Introduction

1.1. Motivation

The transition to a decarbonized economy is anchored by several pivotal transformations: the electrification of industry and transport, the deployment of energy efficiency measures, and the integration of renewable energy sources. Central to all three of these transformations is the integration of millions of decentralized energy resources (DERs), such as electric vehicles, batteries, and rooftop photovoltaic (PV) systems into the grid (Pérez-Arriaga and Knittle, 2016). This integration presents not only challenges of matching complex temporal patterns of electricity consumption and intermittent generation but also offers unprecedented opportunities to create economic value for stakeholders of the energy system of the near future.

As this transition unfolds, the roles of various stakeholders are changing, as they adapt to and seek to deal with and leverage the increasing penetration of DERs. This presents a diverse set of challenges and opportunities.

Grid operators are increasingly applying new strategies to maintain system stability as electricity demand patterns evolve (Lindberg et al., 2019) and extreme weather events unfold more frequently (Seneviratne et al., 2021). One of these strategies is the activation of demand-side flexibility (*i.e.,* demand response) to reduce peak loads or generation. Such programs incentivize consumers (*e.g.*, households, commerce, and industry) to reduce or time-shift their electricity demand (and generation), helping balance supply and demand. The U.S. Department of Energy highlights that demand response allows consumers to play a significant role in grid operation by cutting or shifting usage at peak times, and system operators increasingly treat it as

a valuable resource for maintaining the fine balance between supply and demand, and thereby lowering costs (Energy, 2025). The National Grid Electricity System Operator (ESO) in the United Kingdom demonstrated such a program with their Demand Flexibility Service (DFS), in which over 1.6 million households and businesses participated in supporting the ESO balance the grid at peak times (*The ESO's Demand Flexibility Service | ESO* 2024). Furthermore, as shown by the California Independent System Operator (CAISO), the *Flex Alert Program* achieved notable reductions in peak load during heatwaves and cold snaps (Peplinski and Sanders, 2023). Such mechanisms can reduce the need for grid expansion investments (Blokhuis et al., 2011; Asensio et al., 2017), increasing the efficiency of grid operation and simultaneously enhance grid resilience to extreme weather events.

Energy retailers are also adjusting to the rise of distributed energy resources, as these are profoundly shifting demand profiles for which they need to procure energy. A prominent example of novel load patterns is the "Californian Duck Curve," popularized during the roll out of rooftop solar PV in the state. More precisely, in California, midday electricity demand has dropped sharply as abundant solar DERs supply consumers locally, then demand ramps steeply upward in the evening when solar fades; the load curve's shape resembles a duck (Denholm et al., 2015; Hou et al., 2019). Next to rooftop PV, behind-the-meter storage combined with home energy management systems (HEMS) are also exposing retailers to novel net load profiles. Thus, the role of retailers is transitioning from energy procurement, which has historically been based on standard load profiles to actively forecasting electricity demand informed by exogenous factors (Yang et al., 2017), shaping tariffs to resemble spot market prices, and beginning to take active control of behind-the-meter appliances (Kerscher and Arboleya, 2022; Saboori et al., 2011; Nuytten et al., 2013).

Consumers, through collective DER arrangements like energy communities, virtual power plants (VPPs), and microgrids are emerging as important stakeholders to organize and intelligently operate distributed resources. Policy frameworks are actively encouraging this trend. In the US, FERC No. 2222 provides opportunity "for new and potentially increased value, one that is based on intelligently combining many DERs into a single vir-

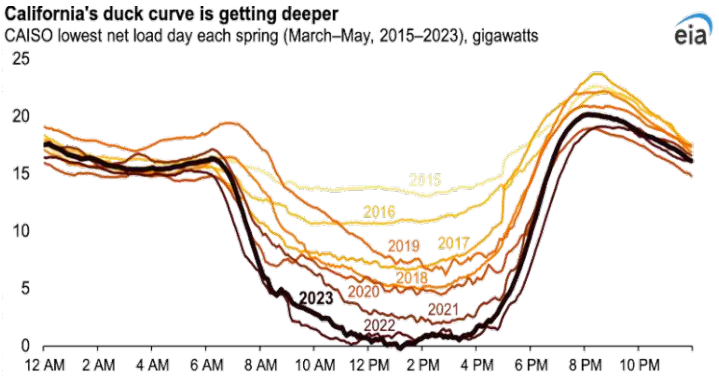


Figure 1.1.: Californian Duck Curve (EIA, 2025)

tual resource that can reduce costs for the bulk power system." (Zhou et al., 2021). In Europe, the Clean Energy for All Europeans Package introduced formal recognition for "Citizen Energy Communities" and "Renewable Energy Communities," establishing an enabling legislative framework for groups of consumers/prosumers to jointly invest in and manage energy resources (European Commission. Joint Research Centre., 2020; Nosratabadi et al., 2017; Gjorgievski et al., 2021). Similarly, regulatory advancements in Great Britain, such as Ofgem's P415 modification to the Balancing and Settlement Code, granting smaller consumers access to Virtual Lead Parties (VLPs)—independent aggregators of consumer flexibility—to participate directly in wholesale electricity markets. This removes barriers for consumers to offer their flexibility and potentially increasing market competition (Ofgem, 2025).

The main enabler of these transitioning roles is advanced information and communication technology (ICT). Specifically, by harnessing the computational capabilities of today's hardware, *i.e.* Graphical Processing Units (GPUs) and the increasing ubiquity of energy time series data, machine learning presents effective tools for actors who want to operate distributed energy resources (DERs) based on their unique objectives. One of the foremost applications of machine learning, and especially its sub-category supervised learning, is energy forecasting, which is central for data-driven control of DERs and forms the topic of this dissertation.

1.2. Core Objective and Research Questions

The core objective of this thesis is to bridge the gap between supervised learning-based very short-term energy forecasting and the control of distributed energy resources. To achieve this objective, the thesis addresses three interconnected research questions, each explored in a peer-reviewed paper that tackles an application of energy forecasting and control at a specific level of aggregation and stakeholder in the transitioning electricity system (see Figure 1.2).

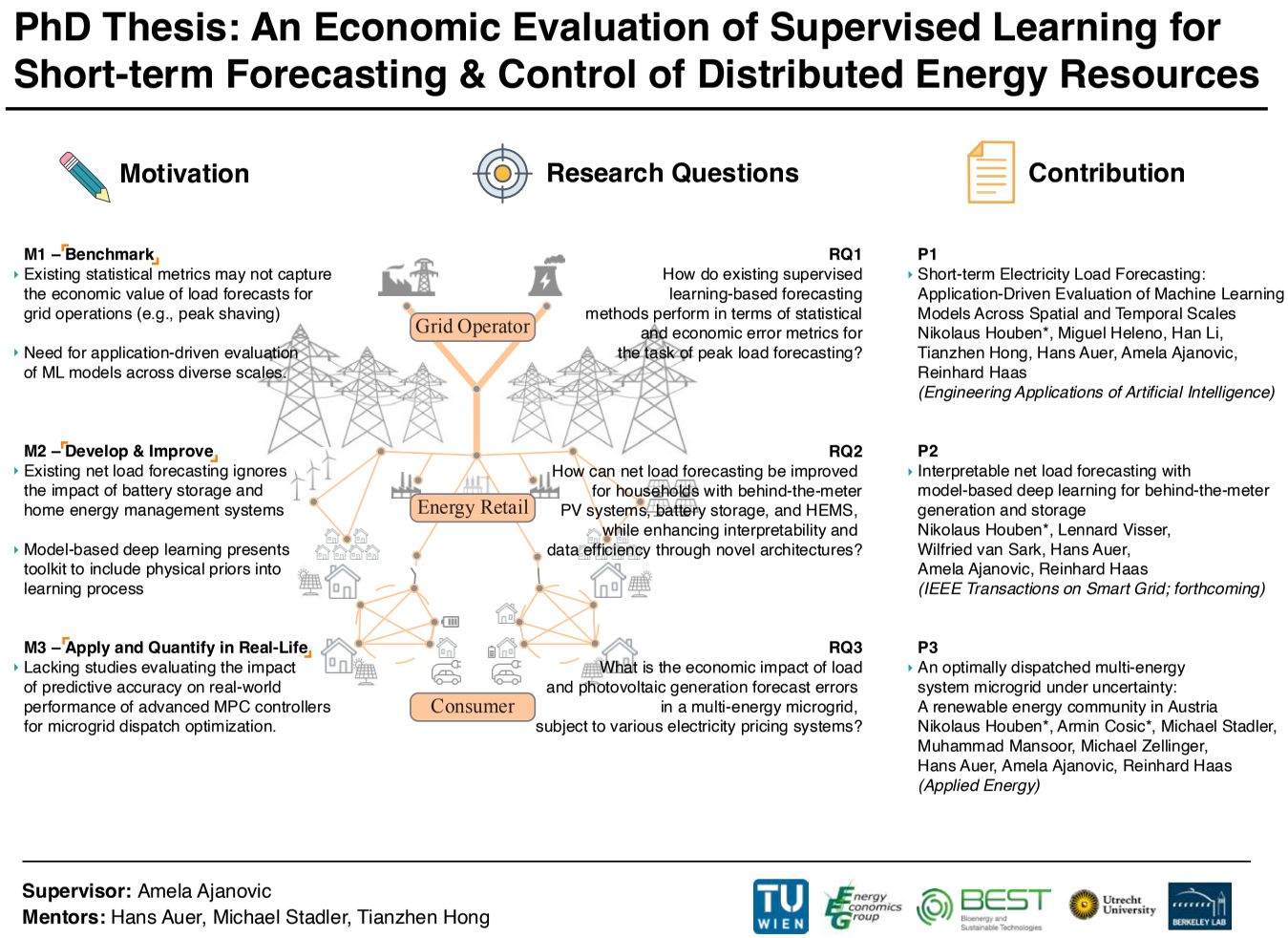


Figure 1.2.: Schematic link between motivation, research questions and research papers

Research Question 1

How do existing supervised learning-based forecasting methods perform in terms of statistical and economic error metrics for the task of peak load forecasting?

This research question motivated a large benchmarking study of supervised learning model architectures for electricity load forecasting, conducted during a research stay at the Lawrence Berkeley National Laboratory (LBNL).

Paper 1: Nikolaus Houben, Miguel Heleno, Han Li, Tianzhen Hong, Hans Auer, Amela Ajanovic, Reinhard Haas (2025). "Short-term Electricity Load Forecasting: Application-Driven Evaluation of Machine Learning Models Across Spatial and Temporal Scales." (Houben et al., 2025a); Under revision in *Engineering Applications of Artificial Intelligence*.

In this work, the two major classes of supervised learning-based methods, *i.e.*, tree-based and neural network-based methods, are compared for the task of load forecasting. Model performance of six architectures is evaluated on twelve datasets at various levels of spatial aggregation and temporal horizons. With a focus on peak load forecasting, a novel error metric is proposed, bridging the gap between statistical accuracy and economic implications of forecasting models. The work concludes with recommendations to grid operators seeking to develop and implement effective data-driven peak load forecasting models.

Research Question 2

How can net load forecasting be improved for households with behind-the-meter photovoltaic systems, battery storage, and home energy management systems, while enhancing interpretability and data efficiency through novel architectures?

The second research question led to a collaboration with researchers at Utrecht University, resulting in a paper on day-ahead net load forecasting from the

perspective of energy retailers.

Paper 2: Nikolaus Houben, Wilfried van Sark, Lennard Visser, Hans Auer, Amela Ajanovic, Reinhard Haas (2025). "Interpretable net load forecasting with model-based deep learning for behind-the-meter generation and storage"(Houben et al., 2025b); Under revision in *IEEE Transactions on Smart Grid*.

Forecasting net load for prosumer households with photovoltaics, batteries, and energy management systems presents a significant challenge due to complex operational strategies. This study addresses this by proposing a model-based deep learning architecture. By embedding a physical PV model and a differentiable HEMS optimization layer, the model incorporates operational priors, leading to higher accuracy, interpretability, and data efficiency over competitive methods in the literature. The findings, furthermore, provide practical guidance for retailers implementing forecasting under real-world data constraints.

Research Question 3

What is the economic impact of load and photovoltaic generation forecast errors in a multi-energy microgrid, subject to various electricity pricing systems?

The third research question, rooted in work performed at the Microgrid Lab in Wieselburg, Austria, is answered in a study comparing the operational dispatch of flexibilities in an energy community based on model predictive control and rule-based control.

Paper 3: Nikolaus Houben, Armin Cosic, Michael Zellinger, Michael Stadler, Hans Auer, Amela Ajanovic, Reinhard Haas (2024). "Optimal Dispatch of a Multi-Energy System Microgrid Under Uncertainty: A Renewable Energy Community in Austria."(Houben et al., 2023) Published in *Applied Energy*.

In this work, a novel multi-step tree-based forecasting method for electricity

load and hot water load, as well as for photovoltaic generation, was developed and integrated into a microgrid controller. By virtue of a real case study of an Energy Community in Austria, the economic potential and limitations of a data-driven control strategy are showcased. The paper concludes with a sensitivity analysis of battery size to inform investment decisions of end-users.

1.3. Structure of the Thesis

This thesis is structured around three interconnected research questions (RQs), each corresponding to a peer-reviewed publication[1] (Paper 1, Paper 2, Paper 3) (see Figure 1.2). To maintain clarity, the abbreviations RQ1, RQ2, and RQ3 are consistently used throughout the main body chapters to link content directly to the relevant question and publication. While drawing significantly from these papers for the State-of-the-Art (Chapter 2), Methods (Chapter 3), and Case Studies and Results (Chapter 4), the thesis integrates these components and is intended to be read as a coherent body of work, building a unified narrative across the research questions. Lastly the commonalities of the three papers are brought together in the Synthesis chapter.

This thesis is structured as follows:

1. **Chapter 2: Review of the State-of-the-Art & Progress Beyond**
   This chapter reviews the relevant literature on time series forecasting techniques, energy forecasting applications (including spatial/temporal scales and net load), model evaluation, interpretability, and the integration of forecasts into DER control strategies. It also identifies research gaps in order to position the contributions of this thesis within the state-of-the-art.

2. **Chapter 3: Methods**
   The chapter introduces the foundational methodologies used throughout the thesis, namely Supervised Learning for forecasting and Eco-

[1] At the time of writing, two papers are still under review.

nomic Model Predictive Control for evaluation and control. Central concepts and a common notation is established here. Furthermore, the specific methods developed or applied for each research question are detailed, including the Net Load Error framework (RQ1), the model-based deep learning architecture (RQ2), and the hybrid forecasting strategy coupled with MILP-based eMPC (RQ3).

3. **Chapter 4: Case Studies and Results**
   This part of the thesis presents the practical application and empirical evaluation of the methods from Chapter 3. For each research question (RQ1, RQ2, RQ3), it describes the specific case study context, data sources, and experimental design, followed by a detailed analysis of the findings and results.

4. **Chapter 5: Synthesis**
   Here, the findings from the individual research questions (Chapter 4) are synthesized. The chapter draws connections across the methodologies and case studies, discusses the advancements in forecasting and control integration, considers implications for stakeholders, and identifies the limitations of this work along with potential directions for future research. Lastly, I provide a reflection on the research trajectory.

5. **Chapter 6: Conclusion and Outlook**
   This chapter provides a concise summary of the thesis's main contributions regarding the economic evaluation and application of supervised learning for DER forecasting and control. It reiterates the key conclusions drawn from the research and offers a final outlook on the significance of these findings for the field.

6. **Appendices**
   Includes supplementary materials, detailed mathematical derivations, additional data, and potentially expanded results relevant to the research presented.

2. Review of the State-of-the-Art

The literature on energy forecasting has evolved significantly over recent years, driven by the need to manage increasingly complex power systems and distributed energy resources (DERs). This chapter presents the state-of-the-art of the literature pertaining to this thesis. The strands of literature covered here include: forecasting methods, their data efficiency and interpretability, their evaluation, and their integration into control algorithms. The chapter concludes with a summary of the identified gaps in the literature and the corresponding contributions of this thesis.

2.1. Time Series Forecasting Approaches

Time series forecasting aims to predict future values of a target variable based on its observed history (De Gooijer and Hyndman, 2006). Let the target time series be denoted by $\{y_1, y_2, \ldots, y_T\}$, where y_t is the observation at time step t. The objective is typically to forecast the next H values, $\{\hat{y}_{T+1}, \ldots, \hat{y}_{T+H}\}$, where H is the forecast horizon. Typically, this involves using not only the past values of the target series (up to a look-back window L) but also potentially incorporating past exogenous features $\{\mathbf{x}_{t-\tau}^{\text{past}}\}_{\tau=0}^{L}$ and known future exogenous features (*e.g.* weather forecasts) $\{\mathbf{x}_{k}^{\text{future}}\}_{k=1}^{H}$ available at the forecast origin T. The forecast for a specific future time step $T+k$ made at origin T is denoted $\hat{y}_{k|T}$.

Classical statistical methods have long been used for time series forecasting (De Gooijer and Hyndman, 2006). Exponential Smoothing (ES) methods, developed since the 1950s, produce forecasts as weighted averages of past observations, with weights decaying exponentially into the past (Gardner Jr.,

1985; De Gooijer and Hyndman, 2006). For instance, Simple Exponential Smoothing (SES) is suitable for series without trend or seasonality, while Holt's method incorporates linear trend, and the Holt-Winters method adds seasonality (Winters, 1960; Hyndman and Khandakar, 2008).

Another foundational statistical approach is the Autoregressive Integrated Moving Average (ARIMA) methodology, established by Box and Jenkins (Box et al., 2015; De Gooijer and Hyndman, 2006). An $ARIMA(p, d, q)$ model explains the current value of a time series based on its own past values (autoregressive component, p), past forecast errors (moving average component, q), and after applying differencing (d) to achieve stationarity. Seasonal ARIMA (SARIMA) models extend this framework to handle seasonal patterns with period m, denoted $ARIMA(p, d, q)(P, D, Q)_m$, incorporating seasonal AR (P), differencing (D), and MA (Q) terms (Hyndman and Khandakar, 2008).

To address the limitations of traditional methods, particularly in handling high-dimensional inputs and nonlinear dynamics, machine learning (ML) approaches have gained prominence (Masini et al., 2023; Lim and Zohren, 2021). Within the ML paradigm, various techniques are employed. Penalized linear regressions, such as LASSO (Least Absolute Shrinkage and Selection Operator) (Tibshirani, 1996), extend linear regression to high-dimensional settings by adding a penalty term to the loss function. This regularization helps prevent overfitting and can simultaneously perform variable selection, shrinking coefficients of irrelevant predictors towards zero (Masini et al., 2023). The Elastic Net (Zou and Hastie, 2005) combines Ridge and LASSO penalties, often providing a good balance between regularization and variable selection, especially when predictors are correlated.

Moving beyond linearity, tree-based methods partition the predictor space recursively to make local predictions. Single regression trees are often unstable, but ensemble methods like Bagging (Breiman, 1996) and Random Forests (Breiman, 2001) improve stability and accuracy by averaging predictions from many trees built on bootstrap samples of the data and random subsets of features (Masini et al., 2023). Boosting methods, such as Gradient Boosting Machines (Friedman, 2001) and XGBoost (Chen and Guestrin,

2016), build trees sequentially, with each new tree attempting to correct the errors made by the previous ones, often achieving improved model accuracy (Masini et al., 2023).

Neural Networks (NNs) are another machine learning modeling framework for learning complex nonlinear relationships. Shallow NNs, typically with a single hidden layer of "neurons" using activation functions (like sigmoid or ReLU), act as universal function approximators (Hornik et al., 1989; Masini et al., 2023). Deep Learning (DL) extends this by using multiple hidden layers (deep NNs) to learn hierarchical representations of the data (Goodfellow et al., 2016; Lim and Zohren, 2021). Specific DL architectures tailored for sequential data include:

- **Convolutional Neural Networks (CNNs):** Adapted using 1D or causal convolutions, CNNs apply shared filters across time to extract local patterns. Dilated convolutions allow them to capture longer-range dependencies efficiently (Bai et al., 2018; Lim and Zohren, 2021). Techniques like WaveNet use exponentially increasing dilation rates (Oord et al., 2016).

- **Recurrent Neural Networks (RNNs):** Designed explicitly for sequences, RNNs maintain a hidden state (memory) that is updated at each time step. Variants like Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Units (GRU) (Chung et al., 2014) use gating mechanisms to control information flow, enabling them to learn long-term dependencies while mitigating issues like vanishing gradients (Lim and Zohren, 2021).

- **Attention Mechanisms and Transformers:** Originating from NLP, attention mechanisms allow a model to dynamically weigh the importance of different past inputs when making a prediction at a specific time step (Bahdanau et al., 2014; Vaswani et al., 2017). Transformers rely heavily on self-attention, processing the entire input sequence in parallel to capture complex dependencies, which has shown promise but also sparked debate regarding its effectiveness compared to simpler models for certain time series tasks (Lim et al., 2021; Zeng et al., 2022).

2.2. Forecasting Horizon and Multi-step Forecasting methods

The forecast horizon, H, significantly influences the choice of forecasting strategy, especially when predicting multiple steps ahead ($H > 1$). Generating the sequence of forecasts $\{\hat{y}_{k|t}\}_{k=1}^{H}$ can be approached in several ways, each with distinct characteristics and trade-offs (Ben Taieb et al., 2012; Lim and Zohren, 2021). The primary strategies discussed in the literature include:

- **Recursive (Iterative) Strategy:** This conceptually simple method applies a model trained only for one-step-ahead prediction iteratively. The forecast for step $t + 1$ is used as input to predict step $t + 2$, and so on. Its main advantage is requiring only a single model, making it computationally efficient to train. However, it is susceptible to error accumulation, as prediction errors from early steps can propagate and potentially degrade accuracy at longer lead times (Ben Taieb et al., 2012). The detailed recursive formulation is given in Eq. (3.3) in Section 3.1.1.2.

- **Direct Strategy:** This approach trains H separate models, each specialized to predict a specific lead time k. For example, one model predicts $t + 1$, another predicts $t + 2$, directly from inputs available at time t. By construction, it avoids the problem of error propagation faced by the recursive method. However, it requires training and storing H models, increasing computational burden. Furthermore, it assumes conditional independence between forecasts at different lead times, which may not hold true, potentially leading to less coherent forecast trajectories (Chevillon, 2007; Lim and Zohren, 2021). The direct prediction formula is shown in Eq. (3.4) in Section 3.1.1.2.

- **Multi-Input Multi-Output (MIMO) Strategy:** This method uses a single, often more complex, model (typically a vector-output model like certain neural networks) to predict all H future values simultaneously in one step. It balances computational efficiency (one model)

with the potential to capture dependencies between the forecast steps. However, it usually involves a more complex model specification and training procedure compared to the recursive approach (Taieb and Hyndman, 2012; Lim and Zohren, 2021). The MIMO formulation is presented in Eq. (3.5) in Section 3.1.1.2.

Next to these main strategies, hybrid approaches that combine elements, such as the DirRec strategy (combining Direct and Recursive) or using MIMO models recursively, have also been proposed (Ben Taieb et al., 2012). Comparative studies, such as the NN5 competition analysis (Taieb and Hyndman, 2012), have explored the relative performance of these strategies, finding that the best choice often depends on the specific dataset, forecast horizon, and the underlying model family used (Masini et al., 2023). Factors like the complexity of temporal dynamics, the length of the required horizon H, computational resources, and the need to model cross-horizon dependencies influence the optimal selection (De Gooijer and Hyndman, 2006). For precise mathematical definitions of the Recursive, Direct, and MIMO approaches, refer to Section 3.1.1.2.

2.3. Energy Forecasting Applications and Challenges

Building upon the general time series forecasting methods discussed in Section 2.1, this section delves into the specific domain of energy forecasting, particularly electricity load forecasting. Energy forecasting plays an important role in the planning and operation of power systems, supporting decisions ranging from real-time grid balancing to long-term infrastructure investments (Hong, 2014). The increasing penetration of variable renewable energy sources and the electrification of sectors like transport and heating add significant complexity and volatility to energy systems, making accurate load forecasts salient (Hong, 2014).

2.3.1. Temporal Scales

A comprehensive overview by Petropoulos et al., 2022 categorizes load forecasting tasks along two dimensions: forecast horizon and the treatment of uncertainty. The horizon typically distinguishes between:

- *Very Short-Term Forecasting (VSTLF):* Sub-hourly predictions, often used for real-time grid operations and frequency control.
- *Short-Term Forecasting (STLF):* Spanning from one hour up to one week ahead, relevant for day-ahead market participation, unit commitment, and dispatch scheduling.
- *Medium-Term Forecasting (MTLF):* Covering weeks to a year ahead, relevant for maintenance planning, fuel procurement, and resource adequacy assessment.
- *Long-Term Forecasting (LTLF):* Predicting demand several years into the future, essential for generation and transmission capacity expansion planning (Hong, 2014; Petropoulos et al., 2022).

Regarding uncertainty, forecasts can be deterministic (providing a single point prediction, $\hat{y}_{k|t}$) or probabilistic (providing prediction intervals or full predictive densities to quantify uncertainty) (Hong and Fan, 2016; Petropoulos et al., 2022).

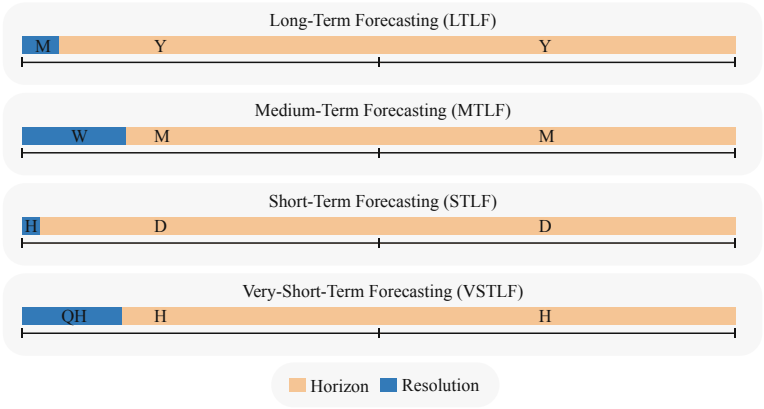


Figure 2.1.: Temporal Scales in Energy Forecasting

This thesis focuses on deterministic Very-Short-Term Forecasting. Further background on the specific choices and evaluation methods pertinent to energy demand forecasting can be found in Kazmi et al., 2023.

Energy time series, particularly electricity load, exhibit unique characteristics that present challenges for forecasting models. Load profiles are influenced by complex interactions between weather patterns (temperature, humidity, irradiance), calendar effects (time of day, day of week, holidays), socio-economic factors, and increasingly, the behavior of distributed energy resources (DERs) like rooftop PV, battery storage, and electric vehicles (Hong, 2014; Masini et al., 2023). This often results in non-stationarity, multiple seasonalities (daily, weekly, yearly), and nonlinear dependencies, rendering more traditional statistical methods inadequate, and motivating the use of more complex architectures like neural networks.

2.3.2. Spatial Scales

Given the breadth of electricity load forecasting, research often focuses on specific spatial aggregation levels. The following review examines contributions comparing forecasting algorithms at distinct scales taken from Houben et al., 2025a:

2.3.2.1. Building-Level Load Forecasting

At the most granular level, forecasting the load of individual commercial or residential buildings is essential for optimizing building energy management systems and participating in demand response programs. Yildiz et al., 2017 reviewed methods for day-ahead and peak load forecasting in commercial buildings, finding that ML models (NNs and tree-based) generally outperformed multivariate linear regression, although predicting peak load accurately remained challenging. Similarly, Lusis et al., 2017 compared several ML algorithms for residential load forecasting, noting comparable RMSE scores but differences in error distributions, with tree-based models showing less bias. The characteristics of the building load data itself also play a role; Hu et al., 2023 demonstrated that the dispersion level (coefficient of variation) of a building's load profile can indicate which model types might be most suitable.

2.3.2.2. Neighborhood and Low-Voltage Network Forecasting

Moving up in aggregation, forecasting load at the level of neighborhoods or low-voltage (LV) distribution feeders is important for distribution system operators (DSOs) managing grid constraints and voltage levels, especially with increasing DER penetration. Haben et al., 2019 evaluated algorithms on numerous LV feeder datasets, highlighting a relative lack of research focus on this specific scale compared to higher aggregation levels. Addressing this, Pinheiro et al., 2023 applied ML models (specifically tree-based) to forecast load for secondary substations across Portugal, significantly outperforming earlier benchmarks (Hong, 2010). Their work also emphasized the importance of considering criteria beyond mere statistical accuracy, such as model applicability, reproducibility, and interpretability, when selecting forecasting methods for operational use.

2.3.2.3. System-Level Load Forecasting

At higher aggregation levels, such as cities, regions, or entire transmission systems, load forecasting informs system-wide operations, market clearing, and generation scheduling. Research at this scale often involves benchmarking various advanced ML and DL methods. Elattar et al., 2020 developed and benchmarked a novel algorithm against different NN types and support vector regression variants using city-scale data. Wang et al., 2021 compared seven models, including tree-based (LightGBM) and deep learning approaches, on transmission system load data from the US (Ruggles et al., 2020), finding that Glubschi performed best and underscoring the impact of weather variables at this scale.

2.3.3. Net Load Forecasting

A particular challenge arises with the increasing installation of behind-the-meter distributed generation, primarily rooftop photovoltaic (PV) systems, and storage (*e.g.*, batteries). Forecasting the *net load*, the difference between the total consumption and local generation exchanged with the grid, is becoming pivotal for retailers and system operators. Net load profiles are highly volatile due to the intermittency of PV generation and the complex behavior introduced by battery storage and associated energy management systems (HEMS). Based on the review exhibited in Houben et al., 2025b, the following review included:

Net load forecasting can be performed with *direct* or *indirect* methods. Direct methods predict the net load time series without explicit decomposition, using a data-driven model that may take as input recent net load data, weather forecasts, and other features (*e.g.*, datetime information). The approach is similar to standard short-term load forecasting (Kazmi et al., 2023; Nti et al., 2020), but often includes additional covariates like solar irradiance, allowing the model to implicitly learn the latent PV profile (Kaytez, 2020; Alipour et al., 2020).

Indirect methods, conversely, explicitly decompose the net load profile. This typically involves using a physical model for behind-the-meter PV generation and a separate data-driven model (*e.g.*, gradient boosting regression tree) for the non-flexible (gross) load component (Wang et al., 2018). The outputs of these component models can then be combined either *additively* (simply subtracting predicted generation from predicted load) or in an *integrated* manner, where the predicted PV generation is used as an input feature to the load forecasting model (Kaur et al., 2016). Neural networks have also been integrated into indirect methods, for example, to handle varying PV meter data visibility (M. Sun et al., 2020) or using LSTMs for grid-scale net load forecasting (Rubasinghe et al., 2023). A challenge for both direct and indirect methods is capturing the impact of storage and HEMS control logic, which significantly alters net load shapes based on factors like electricity tariffs or user preferences.

2.3.4. Model Interpretability & Data Efficiency

While achieving high predictive accuracy is important in time series forecasting, the increasing complexity of state-of-the-art models, particularly Deep Learning (DL) approaches, presents significant challenges regarding their transparency and interpretability (Karniadakis et al., 2021; Carvalho et al., 2019; Xu et al., 2019). Many sophisticated models, such as deep neural networks (DNNs) or large ensembles (*e.g.* XGBoost), quickly become uninterpretable, also referred to as "black boxes ", where their internal logic and decision-making processes are not explainable (Karniadakis et al., 2021; Carvalho et al., 2019; Xu et al., 2019). This lack of transparency is a concern, especially in system-relevant domains like electricity grid operation.

The field of Explainable Artificial Intelligence addresses these challenges by developing methods that make the behavior and predictions of AI systems understandable to developers and stakeholders (Carvalho et al., 2019; Xu et al., 2019; Gilpin et al., 2018). Approaches are categorized as intrinsic or post-hoc (Carvalho et al., 2019; Dwivedi et al., 2023). On the one hand, intrinsic methods use models that are interpretable due to their simpler structure (*e.g.*, linear models, simple decision trees) or by imposing constraints like

sparsity during training (Carvalho et al., 2019). On the other hand, post-hoc methods are applied after a model is trained and attempt to explain the predictions of an existing model. These can be model-specific, using the internal structure of a particular algorithm class (*e.g.*, gradient-based methods for DNNs), treating the model as a black box and analyzing input-output relationships (*e.g.*, LIME, SHAP) (Carvalho et al., 2019; Dwivedi et al., 2023; Angelov et al., 2021). The goal is to provide explanations that are not only true to the model's behavior but also meaningful to its users (Carvalho et al., 2019; Gilpin et al., 2018).

A promising direction, particularly relevant for physical systems like energy forecasting, is *model-based deep learning* and *physics-informed learning* (Karniadakis et al., 2021; Bishop, 2013). This paradigm increases the transparency, accuracy, and data efficiency by explicitly incorporating domain knowledge or, in the latter case physical principles, into the learning process (Karniadakis et al., 2021). Rather than relying only on correlations in the data, these methods integrate known physical laws or system models as inductive biases (Karniadakis et al., 2021). This can be achieved through various means:

- **Physics-Informed Neural Networks (PINNs):** Integrating governing partial differential equations (PDEs) directly into the loss function as soft constraints during training (Karniadakis et al., 2021; Gokhale et al., 2022; Drgoňa et al., 2021).

- **Hybrid Architectures:** Combining data-driven components (like NNs) with physics-based models. For example, using a physical model for PV generation alongside an NN for load prediction, as explored in indirect net load forecasting (Shlezinger et al., 2023; Rubasinghe et al., 2023).

- **Architectural Priors:** Designing network architectures that inherently respect certain physical laws, such as conservation principles or symmetries (Karniadakis et al., 2021).

By constraining the learning process with physical knowledge, model-based approaches thus offer several advantages. Firstly, they can lead to more ac-

curate predictions, especially when data is scarce, because the model is constrained to physically plausible solutions (Karniadakis et al., 2021; Bishop, 2013). Secondly, they often enhance interpretability. For instance, hybrid models can provide intermediate outputs interpretable as physical quantities (*e.g.*, room ambient temperature from an RC-Model), offering insights into the model's internal workings (Gokhale et al., 2022). Thirdly, incorporating domain knowledge significantly improves *data efficiency*, reducing the amount of labeled data required for training compared to purely data-driven black-box models (Karniadakis et al., 2021). This is particularly valuable in energy applications where obtaining large, high-quality datasets can be difficult due to privacy considerations (Van Aubel and Poll, 2019).

2.4. Evaluation of Forecasting Performance

2.4.1. Conventional Error Metrics and Their Limitations

Forecasting models are typically evaluated using Euclidian metrics that draw statistics on the per timestep errors between predicted values ($\hat{y}_t$) and actual ground truth values (y_t) over an evaluation dataset. For example, a commonly used error metric in forecasting is the Root Mean Squared Error (RMSE), given below:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2} \tag{2.1}$$

A significant limitation of Euclidean distance-based metrics like the RMSE (Eq. (2.1)), is the *double penalty effect* (Keil and Craig, 2009). Specifically, a forecast that correctly predicts peak magnitude but is slightly mistimed is penalized both for the early/late prediction and the missed actual peak (see Figure 2.2). Moreover, when evaluating forecasts with RMSE, a constant forecast (see Figure 2.2b) may yield a lower error score than a forecast (see Figure 2.2a) that, despite its imperfections, conveys valuable information

about the timing and magnitude of peaks. This can lead to suboptimal model selection for a given purpose, motivating the development of novel error metrics, such as Dynamic Time Warping (Müller, 2007).

Note that a more detailed description of various evaluation metrics used in this thesis can be found in Section 3.1.1.5.

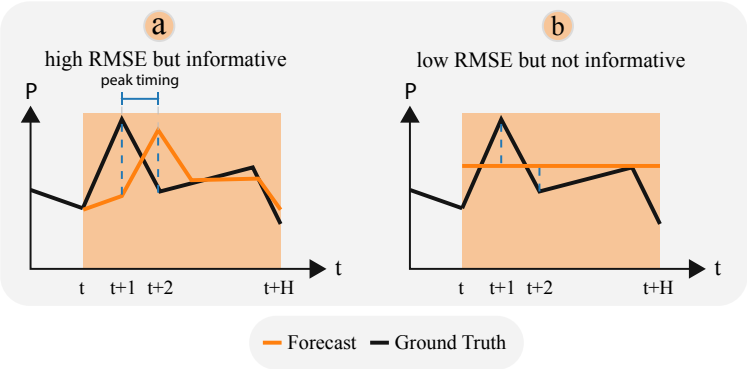


Figure 2.2.: Double Penalty Effect of Euclidian Error Metrics

2.4.2. Economic Evaluation of Energy Forecasts

It has been stated by Murphy, 1993 that *"forecasts possess no intrinsic value, they acquire value through their ability to influence decisions made by users of the forecasts"*. In other words, effective forecasts must be tailored to generate economic value in context. As the grid digitizes and load forecasting methods proliferate, studies on the real-world implications of forecast errors, *i.e.* their economic impact are becoming increasingly relevant. However, as noted Haben et al., 2021, "there are very few examples of the impact and role the forecast's accuracy has on the outputs of the application."

Partly motivated by this effect, several studies have adopted a application-driven evaluation. For instance, Ranaweera et al., 1997 assessed the economic costs of forecast errors via Monte Carlo simulations. More recently, Voss, 2020 evaluated forecasts within a Model Predictive Control framework and demonstrated that using a more peak-sensitive error metric improved peak

load reduction. Additionally, Putz et al., 2023 quantified the monetary impact of forecast errors for building load and domestic hot water forecasting, respectively. Gokhale et al., 2023 further evaluated a transfer learning approach for household load forecasting using a Temporal Fusion Transformer, considering both standard error metrics and operational costs.

2.5. Integration of Forecasts into Control Algorithms

The application of very-short-term energy forecasting in many operational contexts is to inform decision-making, particularly within automated control systems implemented to optimize the performance of energy assets, like DERs. While simple rule-based controllers often operate based on instantaneous feedback, optimization-based control strategies heavily rely on forecasts to anticipate future conditions and make proactive decisions (Stadler et al., 2015; Roslan et al., 2019). These optimization-based control problems are fundamentally sequential decision problems under uncertainty.

2.5.1. Sequential Decision Making under Uncertainty

Many control problems, especially in energy systems, involve making a sequence of decisions over time in the face of uncertainty to achieve some long-term objective (*e.g.*, minimizing operational cost). This process is known as sequential decision making under uncertainty. The standard mathematical framework for modeling such problems is the Markov Decision Process (MDP), typically formulated to minimize long-term costs.

An MDP is defined by the following components[1] (Powell, 2021; Busoniu et al., 2017; Sutton and Barto, 2020)

- **State Space $\mathcal{X}$:** A set of possible states $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$. The state vector $\mathbf{x}_t$ captures all relevant information about the system at time step

[1]Note that the notation differs slightly from the cited resouces, to match subsequent nomenclature in Section 3.1.2

t. It must satisfy the Markov property: the future evolution depends only on the current state $\mathbf{x}_t$ and action $\mathbf{u}_t$, not the entire past history. **Importantly**, the state space may include the forecasts of a time series coming from an arbitrary model (*e.g.,* the ones covered in Section 2.1).

- **Action Space $\mathcal{U}$:** A set of possible control actions $\mathbf{u} \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ that the controller can take in a given state.

- **State Transition Function f:** Describes the system dynamics. The next state $\mathbf{x}_{t+1}$ depends on the current state $\mathbf{x}_t$, the control action $\mathbf{u}_t$, and a random disturbance vector $\mathbf{w}_{t+1}$ representing exogenous uncertainties (like forecast errors or unmodeled dynamics): $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_{t+1})$. The function f implicitly defines the transition probabilities.

- **Stage Cost Function ℓ:** Assigns an immediate scalar cost $\ell_t = \ell(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_{t+1})$ to the transition occurring at step t. This cost can depend on the state, the action, and the realization of the uncertainty $\mathbf{w}_{t+1}$. We assume the cost is bounded.

- **Discount Factor $\gamma \in [0, 1)$:** A factor that discounts future costs relative to immediate ones.

The controller operates based on a policy $h : \mathcal{X} \to \mathcal{U}$, which is a mapping from states to actions, $\mathbf{u}_t = h(\mathbf{x}_t)$. The goal in an MDP is typically to find an optimal policy h^* that minimizes the expected cumulative discounted cost, known as the cost-to-go, from any starting state $\mathbf{x}_0$ (Powell, 2021; Sutton and Barto, 2020):

$$J^h(\mathbf{x}_0) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k \ell_k \middle| \mathbf{x}_0\right] = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k \ell(\mathbf{x}_k, h(\mathbf{x}_k), \mathbf{w}_{k+1}) \middle| \mathbf{x}_0\right] \tag{2.2}$$

where the expectation is taken over the sequence of states generated by following policy h and the stochastic transitions $\mathbf{x}_{k+1} = f(\mathbf{x}_k, h(\mathbf{x}_k), \mathbf{w}_{k+1})$.

A central concept in solving MDPs is the value function, representing expected future costs. The state-value function $V^h(\mathbf{x})$ is the expected cost-to-go starting from state $\mathbf{x}$ and following policy h. The action-value function (or

Q-function) $Q^h(\mathbf{x}, \mathbf{u})$ is the expected cost-to-go starting from state $\mathbf{x}$, taking action $\mathbf{u}$ first, and following policy h thereafter. These satisfy the Bellman equations (Busoniu et al., 2017; Sutton and Barto, 2020):

$$V^h(\mathbf{x}) = \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, h(\mathbf{x}), \mathbf{w}) + \gamma V^h(f(\mathbf{x}, h(\mathbf{x}), \mathbf{w}))\right] \tag{2.3}$$

$$\begin{aligned} Q^h(\mathbf{x}, \mathbf{u}) &= \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, \mathbf{u}, \mathbf{w}) + \gamma V^h(f(\mathbf{x}, \mathbf{u}, \mathbf{w}))\right] \\ &= \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, \mathbf{u}, \mathbf{w}) + \gamma Q^h(f(\mathbf{x}, \mathbf{u}, \mathbf{w}), h(f(\mathbf{x}, \mathbf{u}, \mathbf{w})))\right] \end{aligned} \tag{2.4}$$

where the expectation is over the random disturbance vector $\mathbf{w}$ for the next transition.

The optimal value functions, $V^*(\mathbf{x}) = \min_h V^h(\mathbf{x})$ and $Q^*(\mathbf{x}, \mathbf{u}) = \min_h Q^h(\mathbf{x}, \mathbf{u})$, represent the minimum possible expected cost-to-go. They satisfy the Bellman optimality equations:

$$V^*(\mathbf{x}) = \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, \mathbf{u}, \mathbf{w}) + \gamma V^*(f(\mathbf{x}, \mathbf{u}, \mathbf{w}))\right] \tag{2.5}$$

$$Q^*(\mathbf{x}, \mathbf{u}) = \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, \mathbf{u}, \mathbf{w}) + \gamma \min_{\mathbf{u}' \in \mathcal{U}} Q^*(f(\mathbf{x}, \mathbf{u}, \mathbf{w}), \mathbf{u}')\right] \tag{2.6}$$

An optimal policy h^* can be derived by acting greedily (minimizing) with respect to the optimal value functions:

$$h^*(\mathbf{x}) \in \arg\min_{\mathbf{u} \in \mathcal{U}} Q^*(\mathbf{x}, \mathbf{u}) \quad \text{or} \quad h^*(\mathbf{x}) \in \arg\min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\mathbf{w}}\left[\ell(\mathbf{x}, \mathbf{u}, \mathbf{w}) + \gamma V^*(f(\mathbf{x}, \mathbf{u}, \mathbf{w}))\right] \tag{2.7}$$

Solving an MDP means finding an optimal policy h^*. Dynamic Programming (DP) methods compute the optimal value function using the Bellman equations, typically requiring a model (f, ℓ, probability distribution of $\mathbf{w}$). Reinforcement Learning (RL) methods aim to find h^* or approximations of V^*/Q^* directly from interaction data, often without an explicit model. Control algorithms like Model Predictive Control, discussed next, provide another approach to approximating the solution to the sequential decision problem, using a model of the system and (state) forecasts to handle uncertainty.

2.5.2. Model Predictive Control

Model Predictive Control (MPC), or more specifically economic Model Predictive Control (eMPC), has emerged as a dominant paradigm for optimization-based control in energy systems (Wang et al., 2023; Stadler et al., 2015). MPC leverages a system model alongside forecasts of disturbances (*e.g.*, energy demand, renewable generation, market prices, contained in $\hat{\mathbf{y}}_{k|t}$) to solve an optimization problem over a finite prediction horizon H at each control step t. This optimization problem (Eq. (3.17) in Section 3.1.2) can be viewed as a practical method for approximating the solution to the Bellman optimality equation (e.g., Eq. (2.6)) over a finite horizon, using forecasts (representing the expected behavior of $\mathbf{w}$) instead of explicit expectations over future uncertainties. The first part of the optimal control sequence, $\mathbf{u}_0^*$, is applied, and the process is repeated at the next step with updated measurements and forecasts (receding horizon) (Morari and H. Lee, 1999). The quality of the forecasts (*e.g.*, load, PV generation, electricity prices) is therefore important to the performance of MPC controllers (Wang et al., 2023). Studies have demonstrated that improving forecast accuracy directly translates to better control performance, for example, reducing energy imbalances or operational costs in hybrid renewable energy systems (Bartolucci et al., 2019). MPC has been successfully applied to various energy systems, including residential energy hubs under flat tariffs (Negenborn et al., 2009), multi-energy systems with sector coupling under Time-of-Use tariffs (Gu et al., 2017), microgrids under demand charge tariffs (Gust et al., 2021), and systems interacting with variable market prices (Parisio et al., 2014). Advanced forecasting models, such as hybrid deep learning approaches, are increasingly being integrated into MPC frameworks to handle the complexities of modern building energy systems with renewables (Gao et al., 2023).

Despite its strengths, traditional MPC faces challenges, particularly in accurately modeling complex systems and handling uncertainty inherent in forecasts (Wang et al., 2023; Arroyo et al., 2022). While stochastic MPC methods exist to incorporate uncertainty, they often require detailed probabilistic descriptions of forecast errors ($\mathbf{w}$), which can be difficult to obtain (Arroyo et al., 2022), or can become computationally intractable. Furthermore, it is evident that many studies evaluating MPC performance either assume per-

fect forecasts or use simplified representations of forecast errors, lacking a thorough analysis of how realistic forecast inaccuracies propagate through the control decisions and impact outcomes like operational cost or constraint satisfaction (Kanwar et al., 2015; Moser et al., 2020). Another common limitation identified in the literature is the use of relatively short (*e.g.*, 4-12-hours) prediction horizons, which prevents the controller from anticipating variations further into the future, such as differences between weekdays and weekends. The focus on single tariff structures in many studies also limits the understanding of how forecast integration interacts with different economic signals.

2.5.3. Reinforcement Learning

Reinforcement Learning (RL) offers an alternative or complementary approach to MPC for solving the sequential decision-making problem defined by the MDP in Section 2.5.1, aiming to minimize long-term costs (Eq. (2.2)), and has gained traction in energy systems control (Powell, 2021). RL methods learn optimal policies (like h^* in Eq. (2.7)) or approximate optimal value functions (like Q^* or V^* representing minimum expected costs, Eqs. (2.6) and (2.5)) directly from interaction with the system (or a simulation), often without requiring an explicit model of the system dynamics f or cost function ℓ. Instead, they typically learn through trial-and-error, using observed transitions $(\mathbf{x}_t, \mathbf{u}_t, \ell_t, \mathbf{x}_{t+1})$ to update their estimates. Value-based RL methods, like Q-learning, iteratively estimate Q^* using updates derived from the Bellman optimality equation. Policy-based RL methods directly search for the optimal policy parameters. For large or continuous state/action spaces, RL heavily relies on function approximation (*e.g.*, linear models, neural networks) to represent value functions or policies compactly, implicitly capturing system dynamics and anticipating future states and costs (Kuznetsova et al., 2013; Wang et al., 2023). However, standard RL may not be able to ensure that the system is controlled to remain within its given operational constraints and often requires extensive training data and careful tuning (Arroyo et al., 2022; Wang et al., 2023). Comparisons between MPC and RL highlight their complementary strengths and weaknesses regarding model dependence, handling of uncertainty vs. constraints, adaptability, and computational de-

mands (Wang et al., 2023; Fu et al., 2023).

2.6. Summary of Gaps in the Literature & Contributions

Building on this review of the literature in energy forecasting and its related challenges, several gaps are evident, hindering the optimal application of these methods, particularly in the context deploying them in real systems for the control of distributed energy resources:

1. **Methodological Uncertainty Across Scales and Horizons:**
   Despite the proliferation of forecasting methods reviewed (Section 2.1), from classical statistics to diverse ML approaches including ensembles and various neural network architectures, there is no definitive guidance on which method performs best universally. Comparative studies (Hong et al., 2014; Januschowski et al., 2022; Yildiz et al., 2017; Lusis et al., 2017) often show context-dependent results, leaving practitioners uncertain about model selection, especially when dealing with the varying characteristics of energy data across different spatial aggregation levels (Section 2.3) and forecast horizons (Section 2.2). A comprehensive cross-scale benchmark is missing.

2. **Limitations of Conventional Evaluation Metrics:**
   The standard practice of evaluating forecasts using statistical error metrics like RMSE or MAE (Section 2.4) fails to capture the economic or operational relevance of the forecast quality. The double penalty effect illustrates how these metrics can misrepresent the utility of a forecast for tasks like peak management (Keil and Craig, 2009). As highlighted by the concept of economic value in forecasting (Murphy, 1993) and recent studies (Ranaweera et al., 1997; Haben et al., 2021; Putz et al., 2023), there is a clear gap between standard evaluation practices and assessing the actual impact of forecast errors on specific energy applications.

3. **Need for Interpretable and Data-Efficient Forecasting Models:** The increasing complexity of modern forecasting models, particularly DL techniques, often results in "black-box" systems lacking transparency (Section 2.3.4). This reduces trust and the understanding of forecast drivers (Carvalho et al., 2019; Dwivedi et al., 2023). Concurrently, the rise of DERs creates highly volatile net load profiles influenced by PV generation and HEMS control logic (Section 2.3.3), demanding forecasting methods that can capture these dynamics accurately. There is a distinct need for models that are not only accurate for net load but also interpretable and data-efficient, especially given that incorporating domain knowledge or physical priors shows promise for achieving these goals (Karniadakis et al., 2021; Shlezinger et al., 2023).

4. **Lack of Integrated Evaluation of Multi-Step Forecasts within Operational MPC Frameworks:** While various multi-step forecasting strategies exist (Section 2.2), and MPC is a recognized control paradigm (Section 2.5.2), there is a notable scarcity of studies that comprehensively integrate practical multi-step forecasting methods into an operational eMPC framework and then rigorously evaluate the economic and operational impact of forecasting uncertainty on real energy systems. Many evaluations of MPC performance either assume perfect foresight or use simplified forecast error models, without fully exploring how different multi-step forecasting characteristics affect the controller's performance and the achievable benefits under various real-world conditions such as diverse tariff structures and longer, multi-day operational horizons (Kanwar et al., 2015; Moser et al., 2020)

Addressing these challenges is highly relevant for developing forecasting techniques that support the data-driven planning, operation, and control of modern energy systems featuring high DER penetration.

In light of the gaps identified above, this thesis makes the following contributions, each addressing one or more of these challenges:

1. **Multi-Scale Load Forecasting Benchmark:** Directly addressing Gap 1, this thesis presents a comprehensive benchmarking study evalu-

ating six state-of-the-art machine learning forecasting algorithms across twelve distinct time series. These series span four different spatial aggregation levels (county, town, neighborhood, building) and multiple forecast horizons, providing empirical insights into relative model performance across diverse energy forecasting contexts (related to RQ1).

2. **Application-Driven Forecast Evaluation Metric (NLE):** To overcome the limitations of purely statistical metrics (Gap 2), this work introduces and utilizes a novel *Net Load Error (NLE)* metric. The NLE quantifies the economic impact of forecast errors specifically within a peak load management application (using BESS and demand charges), providing a tool to evaluate forecast "value" beyond simple accuracy (related to RQ1).

3. **Model-Based Interpretable & Data-Efficient Net Load Forecasting:** Tackling Gap 3, this work proposes a novel model-based deep learning architecture for net load forecasting in households with PV and BESS. By integrating a physical PV model and a differentiable HEMS optimization layer within the neural network, the model achieves improved accuracy while simultaneously enhancing interpretability (by predicting latent physical states like gross load and battery SOC) and demonstrating superior data efficiency compared to standard black-box approaches (related to RQ2).

4. **Framework for Integrating and Evaluating Multi-Step Forecasting in eMPC for Microgrids:** Directly addressing Gap 4, this thesis designs, implements, and evaluates a practical framework that couples a novel hybrid multi-step forecasting strategy (detailed in Section 3.2 for RQ3) with an eMPC controller for microgrid dispatch. This contribution includes: i) the development of a hybrid forecasting method tailored for eMPC, providing robust predictions over a 48-hour horizon by balancing recursive and direct forecasting strengths, ii) The integration of these forecasts into an eMPC designed for a real-world Renewable Energy Community, capable of optimizing for cost or CO_2 under various tariff structures, and iii) a systematic analysis of the "cost of uncertainty," quantifying the economic impact of realis-

tic forecast errors by comparing eMPC performance with real forecasts against a perfect foresight benchmark, thereby providing insights into the practical viability of such integrated systems.

3. Methods

This chapter details the methodologies developed and evaluated in this thesis and based on the three papers cited in Section 1.2. The focus lies on two interlinked tasks salient for managing distributed energy resources: short-term forecasting using Supervised Learning (SL) and sequential decision-making via Economic Model Predictive Control (eMPC). The research explores advanced SL techniques for prediction and leverages the eMPC framework not only for control but also for forecast evaluation and model enhancement.

Figure 3.1 illustrates the components of eMPC, where SL models act as the *predictor* component, providing necessary foresight regarding parameters in the techno-economic system model.

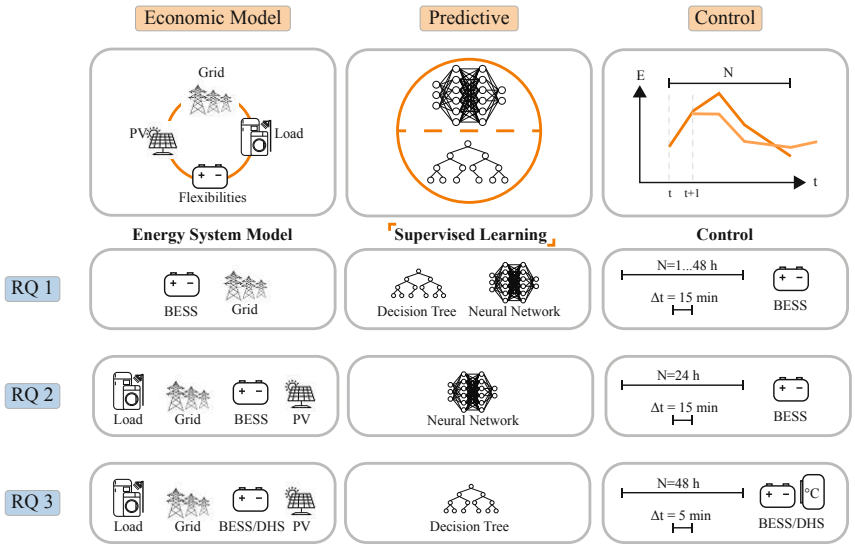


Figure 3.1.: Economic Model Predictive Control Framework Overview

This integration forms the basis for the methods applied across the research questions (presented in Section 1.2), although the specific role of the eMPC component varies significantly.

For RQ3, eMPC is applied conventionally, using SL forecasts for optimal multi-energy microgrid dispatch. In contrast, RQ1 uses the eMPC framework as a tool to *evaluate* the economic performance of various SL models, and as a result assessing their practical value. For RQ2, the concept is inverted: optimization logic derived from eMPC is incorporated *as a component within* the SL model itself to improve its capabilities by encoding system priors.

To keep a clear structure and consistent notation in the description of the methods, this chapter is structured as follows: In Section 3.1, a general description of Supervised Learning and Economic Model Predictive Control is provided, defining their respective problem statements, and laying out the nomenclature used throughout this thesis. This section provides the foundational building blocks, which are then instantiated and adapted in Section 3.2, separately delving into the methodology developed to answer each research question. For each research question, the respective section is structured to detail the specific:

- *Economic Model:* Specifying how the general energy system model from Section 3.1.2 is instantiated.
- *Predictive:* describing how the predictive model takes shape by instantiating the building blocks outlined in Section 3.1.1.
- *Control:* Providing details on the implementations or purpose of using eMPC, including the controlled or evaluated flexible assets and time series.

In Section 3.2.4 summarizes the common experimental setup, including software and hardware used.

3.1. General Methods & Preliminaries

3.1.1. Supervised Learning for Energy Forecasting

Supervised learning refers to a machine learning paradigm in which a model learns to map inputs to outputs by training on labeled example pairs (Bishop, 2006). For energy forecasting, supervised models predict future energy-related variables, such as electricity consumption or photovoltaic (PV) generation, based on historical measurements and external covariates (e.g., weather forecasts) (Hong, 2014; Wang et al., 2021). As mentioned in Section 2.5.1, accurate forecasts are indispensable for operating energy systems effectively, particularly when integrated into advanced control methods like eMPC (Stadler et al., 2015).

3.1.1.1. Notation and Problem Formulation

The supervised learning forecasting problem is defined with the following notation:

Time Indices

- T: Length of time series
- H: Control and Forecast horizon (equal in this thesis).
- L: Look-back window length (number of historical steps used).
- t: timestep index of generating a forecast or controlling an asset, where $t \in \{0, \dots, T\}$ (also referred to as *origin*).
- k: Lead time index within the forecast horizon, where $k \in \{1, \dots, H\}$.
- τ: Step index within the look-back window, where $\tau \in \{0, \dots, L\}$.

Dataset A training dataset $\mathcal{D}$ comprises T examples (Bishop, 1995):

$$\mathcal{D} = \left\{ \left(\mathbf{X}^{(t)}, \mathbf{y}^{(t)} \right) \right\}_{t=1}^{T}, \tag{3.1}$$

where each example indexed by t consists of:

- Input features $\mathbf{X}^{(t)}$: Features associated with time t, comprising historical features $\mathbf{x}_{t-\tau}^{\text{past}} \in \mathbb{R}^{n_{\text{past}}}$ $(\tau = 0, \ldots, L)$ and known future features $\mathbf{x}_{t+k}^{\text{future}} \in \mathbb{R}^{n_{\text{future}}}$ for future lead times $k = 1, \ldots, H$ relative to t (e.g., weather forecast for time $t + k$).

- Target sequence $\mathbf{y}^{(t)}$: The sequence of actual observed values for lead times $k = 1, \ldots, H$ relative to t, i.e., $\mathbf{y}^{(t)} = \{y_{t+k}\}_{k=1}^{H}$.

Forecasting Model Given parameters $\boldsymbol{\theta}$ and hyperparameters $\boldsymbol{\phi}$, the model $\mathcal{F}$ uses features $\mathbf{X}^{(t)}$ available at time t to predict the target sequence $\mathbf{y}^{(t)} = \{y_{t+k}\}_{k=1}^{H}$. We denote the forecast for lead time k made at time t as $\hat{y}_{k|t}$. This notation distinguishes the prediction from the ground truth value y_{t+k} and expresses the forecast origin time t.

The model output is the sequence:

$$\hat{\mathbf{y}}^{(t)} = \left\{ \hat{y}_{k|t} \right\}_{k=1}^{H} = \mathcal{F} \left(\left\{ \mathbf{x}_{t-\tau}^{\text{past}} \right\}_{\tau=0}^{L}, \left\{ \mathbf{x}_{k}^{\text{future}} \right\}_{k=1}^{H}; \boldsymbol{\theta}, \boldsymbol{\phi} \right) \tag{3.2}$$

3.1.1.2. Multi-step Ahead Forecasting Approaches

Generating the forecast sequence $\{\hat{y}_{k|t}\}_{k=1}^{H}$ for multiple steps ahead $(H > 1)$ requires specific strategies. As reviewed in Section 2.2, the main approaches differ in how the forecast for each step k is constructed (Ben Taieb et al., 2012; Taieb and Hyndman, 2012). With the notation introduced above, the three main methods are formalized as follows:

- **Recursive Method**: This strategy utilizes a single one-step-ahead model $\mathcal{F}_{1step}$ iteratively.

– The first forecast $\hat{y}_{1|t}$ is generated using available features $\mathbf{Z}_t$ at time t:

$$\hat{y}_{1|t} = \mathcal{F}_{1step}(\mathbf{Z}_t; \boldsymbol{\theta}_{1step})$$

– For subsequent steps ($k = 2, \ldots, H$), the model input $\hat{\mathbf{Z}}_{t+k-1}$ must incorporate previously generated forecasts $\{\hat{y}_{1|t}, \ldots, \hat{y}_{k-1|t}\}$ in place of unknown actual future values, potentially leading to error accumulation:

$$\hat{y}_{k|t} = \mathcal{F}_{1step}(\hat{\mathbf{Z}}_{t+k-1}; \boldsymbol{\theta}_{1step}) \tag{3.3}$$

Here, $\hat{\mathbf{Z}}_{t+k-1}$ contains known past features, known future covariates up to $\mathbf{x}_k^{\text{future}}$, and the sequence of prior forecasts $(\hat{y}_{1|t}, \ldots, \hat{y}_{k-1|t})$.

- **Direct Method**: This approach trains H separate models, $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(H)}$. Each model $\mathcal{F}^{(k)}$ is specifically trained to predict the target at lead time k, using only information available at the forecast origin t:

$$\hat{y}_{k|t} = \mathcal{F}^{(k)}\left(\left\{\mathbf{x}_{t-\tau}^{\text{past}}\right\}_{\tau=0}^{L},\ \mathbf{x}_k^{\text{future}};\ \boldsymbol{\theta}^{(k)}\right) \tag{3.4}$$

This formulation inherently avoids using intermediate forecasts as inputs, but neglects dependencies between timesteps (*e.g.*, consistent aggregate (daily) energy demand).

- **Multi-Input Multi-Output (MIMO)**: This strategy employs a single model $\mathcal{F}_{MIMO}$ designed to output the entire forecast vector $\{\hat{y}_{k|t}\}_{k=1}^{H}$ simultaneously. The model takes all relevant past features and all known future features across the horizon as input:

$$\left\{\hat{y}_{k|t}\right\}_{k=1}^{H} = \mathcal{F}_{MIMO}\left(\left\{\mathbf{x}_{t-\tau}^{\text{past}}\right\}_{\tau=0}^{L},\ \left\{\mathbf{x}_k^{\text{future}}\right\}_{k=1}^{H};\ \boldsymbol{\theta}_{MIMO}\right) \tag{3.5}$$

This allows the model to potentially capture dependencies across the forecast steps $k = 1, \ldots, H$ internally during training and inference.

Other strategies, such as combinations or variations of these (e.g., DirRec, DirMO), also exist (Ben Taieb et al., 2012). The choice of strategy has

an effect on model complexity, computational cost, and error characteristics across the forecast horizon.

3.1.1.3. Model Training

Model parameters $\boldsymbol{\theta}$ are learned by minimizing a loss function $\mathcal{L}$ over the training dataset $\mathcal{D}$ (Hastie et al., 2009):

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \frac{1}{T} \sum_{t=1}^{T} \mathcal{L}\left(\{y_{t+k}\}_{k=1}^{H}, \{\hat{y}_{k|t}(\boldsymbol{\theta})\}_{k=1}^{H}; \boldsymbol{\phi}\right). \tag{3.6}$$

Typical loss functions averaged over the dataset and horizon include Mean Squared Error (MSE) and Mean Absolute Error (MAE):

- **Mean Squared Error (MSE):**

$$\text{MSE} = \frac{1}{TH} \sum_{t=1}^{T} \sum_{k=1}^{H} \left(y_{t+k} - \hat{y}_{k|t}\right)^2 \tag{3.7}$$

- **Mean Absolute Error (MAE):**

$$\text{MAE} = \frac{1}{TH} \sum_{t=1}^{T} \sum_{k=1}^{H} \left|y_{t+k} - \hat{y}_{k|t}\right| \tag{3.8}$$

Optimization of θ is commonly accomplished through gradient-based methods, such as stochastic gradient descent or Adam (Kingma and Ba, 2017). For efficiency-related considerations these algorithms operate on batches (subsets of the training set), instead on the entire training data set.

3.1.1.4. Model Inference

Once the model has been trained, the optimized parameters $\boldsymbol{\theta}^*$ can be used to generate predictions on (unseen) input data $\mathbf{X}^{(t)}$:

$$\{\hat{y}_{k|t}\}_{k=1}^{H} = \mathcal{F}\left(\mathbf{X}^{(t)};\, \boldsymbol{\theta}^*,\, \boldsymbol{\phi}\right). \tag{3.9}$$

Inference typically involves the same preprocessing steps applied during training (Goodfellow et al., 2016).

3.1.1.5. Model Evaluation (Testing)

Once a model has been trained, and predictions for unseen data has been generated, its predictive performance can be evaluated. Figure 3.2 illustrates this standard approach, where historical data is divided, and the trained model (Phase 1) is applied to the independent *Testing Data* to generate outputs (predictions) for assessment (Phase 2).

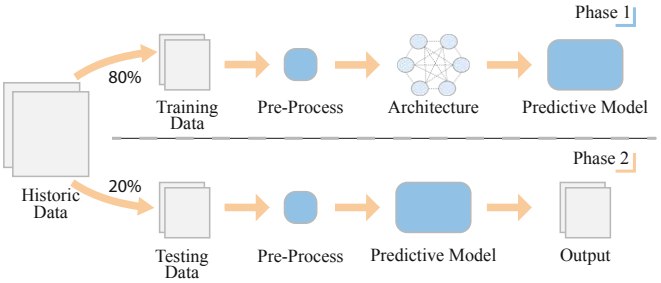


Figure 3.2.: Typical Train Test Split in Supervised Learning Model Development

This evaluation involves using metrics to quantify the discrepancy between the predicted values ($\hat{y}_{k|t}$) and actual ground truth values (y_{t+k}) over an

independent evaluation dataset (e.g., a test set not used during training) comprising N forecast origins. These metrics must not be the same as the loss function used during training.

Generally, an evaluation metric D computes an aggregate error score ϵ by comparing the sequences of predictions $\hat{\mathbf{y}}^{(t)} = \{\hat{y}_{k|t}\}_{k=1}^{H}$ with the corresponding sequences of actual values $\mathbf{y}^{(t)} = \{y_{t+k}\}_{k=1}^{H}$ across all N forecast origins in the evaluation set:

$$\epsilon = D\left(\{\mathbf{y}^{(t)}\}_{t=1}^{N}, \{\hat{\mathbf{y}}^{(t)}\}_{t=1}^{N}\right) \tag{3.10}$$

To arrive at the aggregate error score (e.g., overall RMSE, MAE) across an evaluation dataset, the process depicted in Figure 3.3 is followed: forecasts are generated originating from each timestep t in the evaluation set, the errors for each forecast sequence are computed, and then these errors are aggregated (e.g., averaged) according to the specific metric's formula.

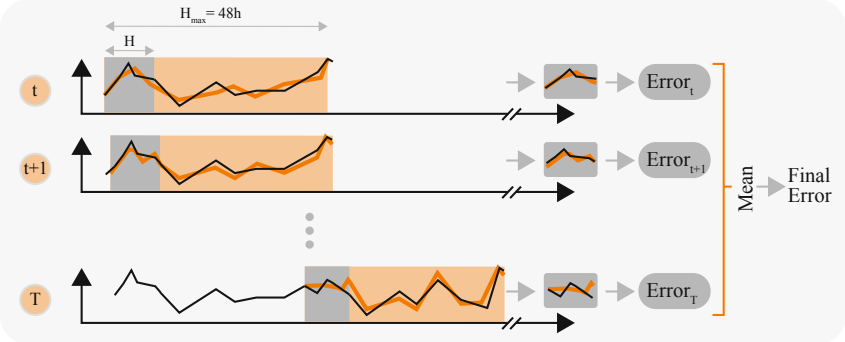


Figure 3.3.: Evaluation process: Forecasts are generated at each timestep t in the evaluation set for the horizon H. Errors are computed for each sequence (prediction vs. actual), and then aggregated across all sequences to obtain the final performance metric.

The specific metrics used throughout this thesis are detailed below (Equations (3.11)-(3.16)):

- **Mean Absolute Error (MAE):** Measures the average absolute dif-

ference. Lower values indicate better accuracy.

$$\text{MAE} = \frac{1}{NH} \sum_{t=1}^{N} \sum_{k=1}^{H} |y_{t+k} - \hat{y}_{k|t}| \tag{3.11}$$

- **Root Mean Squared Error (RMSE):** Calculates the square root of the average squared differences. It penalizes larger errors more heavily than MAE.

$$\text{RMSE} = \sqrt{\frac{1}{NH} \sum_{t=1}^{N} \sum_{k=1}^{H} (y_{t+k} - \hat{y}_{k|t})^2} \tag{3.12}$$

- **Mean Absolute Percentage Error (MAPE):** Expresses the average absolute error as a percentage of the actual values. It is scale-independent but undefined or problematic when actual values (y_{t+k}) are zero or close to zero.

$$\text{MAPE} = \frac{100\%}{N_{nz}} \sum_{t=1}^{N} \sum_{k \text{ s.t. } y_{t+k} \neq 0} \left| \frac{y_{t+k} - \hat{y}_{k|t}}{y_{t+k}} \right| \tag{3.13}$$

  where N_{nz} is the total count of non-zero y_{t+k} values in the evaluation set sums.

- **Normalized Root Mean Squared Error (nRMSE):** Normalizes the overall RMSE, typically by the range ($y_{max} - y_{min}$) or the mean ($\bar{y}$) of the ground truth data over a representative period (e.g., the training set), to allow the comparison across datasets with different scales:

$$nRMSE = \frac{RMSE}{y_{max} - y_{min}} \quad \text{or} \quad nRMSE = \frac{RMSE}{\bar{y}} \tag{3.14}$$

  where $RMSE$ is the overall RMSE calculated as per Eq. (3.12).

- **Coefficient of Determination (R^2):** Represents the proportion of the variance in the dependent variable that is predictable from the model, calculated over the evaluation set. Values closer to 1 indicate a

better fit.

$$R^2 = 1 - \frac{\sum_{t=1}^{N}\sum_{k=1}^{H}(y_{t+k} - \hat{y}_{k|t})^2}{\sum_{t=1}^{N}\sum_{k=1}^{H}(y_{t+k} - \bar{y})^2} \tag{3.15}$$

where $\bar{y}$ is the mean of all y_{t+k} ground truth values in the evaluation set.

- **Skill Score (SS):** Quantifies the relative improvement of the forecast model ($\mathcal{F}$) compared to a baseline forecast model (e.g., persistence, or a simple benchmark like Linear Regression). It is calculated based on an aggregate error metric D (e.g., overall RMSE or MAE) (Yang et al., 2020):

$$SS^{(D)} = 1 - \frac{D_{\mathcal{F}}}{D_{baseline}} \tag{3.16}$$

A value of larger than 0 indicates improvement over the baseline, 0 indicates no improvement, and negative values indicate the model performs worse than the baseline.

3.1.2. Economic Model Predictive Control (eMPC)

Economic Model Predictive Control (eMPC) optimizes a system by explicitly integrating economic objectives into a finite-horizon optimal control problem (Rawlings et al., 2012). Unlike classical Model Predictive Control (MPC), which primarily aims to track reference trajectories (Laurí et al., 2014), eMPC directly minimizes or maximizes economic criteria (e.g., cost, profit, efficiency), making it particularly applicable for tasks like energy management systems. This section introduces the general eMPC framework that serves as a foundational template for the specific applications detailed in Section 3.2.

3.1.2.1. Notation and Problem Formulation

At each control timestep t, the eMPC controller solves an optimization problem over a prediction horizon H. Mirroring the nomenclature introduced for SL in Section 3.1.1, the problem uses the following notation, where

$k \in \{0, \ldots, H-1\}$ denotes the discrete timestep (or lead time) within the optimization horizon relative to the current time t:

- **State vector:** $\mathbf{x}_k \in \mathbb{R}^{n_x}$. Represents the predicted state of the system at time $t+k$, derived iteratively starting from $\mathbf{x}_0 = \mathbf{x}_{current}$. The vector $\mathbf{x}_k$ comprises all relevant system states (e.g., energy stored, temperatures).

- **Control input vector:** $\mathbf{u}_k \in \mathbb{R}^{n_u}$. Represents the control actions to be applied during the interval $[t+k, t+k+1)$. The vector $\mathbf{u}_k$ includes all decision variables which are directly controlled (e.g., power flows, operating modes).

- **Forecast Input Vector** for step k: $\hat{\mathbf{y}}_{k|t} \in \mathbb{R}^{n_y}$. This vector consolidates the forecasts for n_y different external variables required by the controller at lead time k (corresponding to time $t+k$), based on information available at time t. These exogenous variables might include electrical load, renewable generation, energy prices, ambient temperature, etc. Each individual forecast component within this vector (e.g., the load forecast component $\hat{y}^{\text{load}}_{k|t}$) is (in this thesis) obtained from a supervised learning model $\mathcal{F}$ as described in Section 3.1.1.

- **State transition function:** $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_x}$. Describes the system dynamics, predicting the state at step $k+1$ based on step k.

- **Auxiliary Binary variables:** $\mathbf{z}_k \in \{0,1\}^{n_z}$

- **Stage cost function:** $\ell(\cdot)$ [units of cost, e.g., €]. Represents the economic cost incurred during interval k.

The eMPC optimization problem solved at time t is:

$$\min_{\substack{\mathbf{u}_{0:H-1} \\ \mathbf{z}_{0:H-1}}} \quad \sum_{k=0}^{H-1} \ell\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t}\right) + V_f(\mathbf{x}_H) \tag{3.17a}$$

$$\text{subject to} \quad \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t}), \quad k = 0, \ldots, H-1, \tag{3.17b}$$

$$\mathbf{x}_k \in \mathcal{X}, \quad k = 0, \ldots, H, \tag{3.17c}$$

$$\mathbf{u}_k \in \mathcal{U}, \quad k = 0, \ldots, H-1, \tag{3.17d}$$

$$g(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t}) \leq 0, \quad k = 0, \ldots, H-1, \tag{3.17e}$$

$$\mathbf{z}_k \in \{0,1\}^{n_z}, \quad k = 0, \ldots, H-1, \tag{3.17f}$$

$$\mathbf{x}_0 = \mathbf{x}_{current} \tag{3.17g}$$

Here, $\mathcal{X}$ and $\mathcal{U}$ represent feasible state and input sets defined by physical or operational limits. $g(\cdot)$ represents the constraints (e.g., power balance equations, comfort bounds). $V_f(\cdot)$ is an optional terminal cost function penalizing undesirable final states or approximating the long-term cost beyond the horizon H. $\mathbf{x}_{current}$ is the state measured or estimated at the current time t.

General Economic Cost Structure In this thesis, the stage cost $\ell_k = \ell(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t})$ includes components related to energy consumption or injection and demand charges:

$$\ell_k = \ell_k^{\text{energy}} + \ell_k^{\text{DC}}. \tag{3.18}$$

The energy cost ℓ_k^{energy} depends on the net power exchanged with the grid, P_k^{grid} (which is a function of $\mathbf{x}_k, \mathbf{u}_k, \hat{\mathbf{y}}_{k|t}$), and forecast electricity prices $\hat{\pi}_{k|t}^{\text{buy/sell}}$ (which are specific components within the vector $\hat{\mathbf{y}}_{k|t}$):

$$\ell_k^{\text{energy}} = \hat{\pi}_{k|t}^{\text{buy}} \cdot [P_k^{\text{grid}}]^+ \cdot \Delta t - \hat{\pi}_{k|t}^{\text{sell}} \cdot |[P_k^{\text{grid}}]^-| \cdot \Delta t, \tag{3.19}$$

where $[x]^+ = \max(x, 0)$, $[x]^- = \min(x, 0)$, and Δt is the timestep duration (*e.g.*, 15 minutes). The demand charge ℓ_k^{DC} relates to the peak power drawn from the grid, often formulated using an auxiliary peak variable p^h optimized over the horizon (or a relevant billing period subset):

$$\sum_{k \in \text{BillingPeriod}} \ell_k^{\text{DC}} \approx \pi^{\text{DC}} \cdot p^h \quad \text{where} \quad p^h \geq [P_k^{grid}]^+, \forall k \in \text{BillingPeriod}. \tag{3.20}$$

The exact formulation, including how p^h enters the objective (e.g., directly in the sum, or as part of V_f) depends on the specific application and tariff

structure.

General Constraint Formulation System operation is bound by various constraints:

- State dynamics $\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t})$ as in Eq. (3.17b).
- State limits $\mathbf{x}_k \in \mathcal{X}$ (e.g., $E^{min} \leq E_k \leq E^{max}$) as in Eq. (3.17c).
- Input limits $\mathbf{u}_k \in \mathcal{U}$ (e.g., $P^{ch,min} \leq P_k^{ch} \leq P^{ch,max}$) as in Eq. (3.17d).
- Constraints (operational & physical) $g(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t}) \leq 0$ (e.g., power balance equations) as in Eq. (3.17e).
- Logical constraints, often requiring binary variables $\mathbf{z}_k \in \{0,1\}^{n_z}$ as in Eq. (3.17f). A use of binaries is the prevention of simultaneous charging ($p_{a,k}^{ch} > 0$) and discharging ($p_{a,k}^{dis} > 0$) of a specific storage asset 'a':

$$p_{a,k}^{\mathrm{ch}} \leq P_a^{\mathrm{ch,max}} z_{a,k}^{\mathrm{ch}}, \tag{3.21a}$$
$$p_{a,k}^{\mathrm{dis}} \leq P_a^{\mathrm{dis,max}} z_{a,k}^{\mathrm{dis}}, \tag{3.21b}$$
$$z_{a,k}^{\mathrm{ch}} + z_{a,k}^{\mathrm{dis}} \leq 1, \tag{3.21c}$$
$$z_{a,k}^{\mathrm{ch}}, z_{a,k}^{\mathrm{dis}} \in \{0,1\}. \tag{3.21d}$$

  (where $p_{a,k}^{ch}, p_{a,k}^{dis}$ would be components of the control vector $\mathbf{u}_k$, and $z_{a,k}^{ch}, z_{a,k}^{dis}$ components of $\mathbf{z}_k$).

3.1.2.2. General Energy System Model

In this thesis the eMPC framework is applied to operate an energy system. As shown in Figure 3.4, the general energy system considered in this thesis comprises several interacting components:

- **Energy Demands:** These are the loads that the system must satisfy, primarily the electrical load, but can also include other demands like

domestic hot water (DHW) load y_k^{DHW}. These loads are uncontrollable but are uncertain, forming components of the forecast input vector $\hat{\mathbf{y}}_{k|t}$ (e.g., $\hat{y}_{k|t}^{L}, \hat{y}_{k|t}^{DHW}$).

- **On-site Generation:** This refers to variable renewable energy sources, such as photovoltaic (PV) generation. Similar to loads, on-site generation is forecasted and included in $\hat{\mathbf{y}}_{k|t}$ as $\hat{y}_{k|t}^{PV}$.

- **Flexible Assets:** These are dispatchable components within the system, such as Battery Energy Storage Systems (BESS), thermal storage (e.g., DHS), or controllable loads. Their operation is determined by the eMPC. Noteworthy components, as shown in Figure 3.4, include:

  - *State variables* ($\mathbf{x}_k$): Primarily the stored energy level in each asset (e.g., E_k^{BESS}, E_k^{DHS}). These states evolve based on charging/discharging actions and efficiencies (η^{ch}, η^{dis}).

  - *Control input variables* ($\mathbf{u}_k$): The power flows into or out of these assets (e.g., charging power $p_{a,k}^{ch}$, discharging power $p_{a,k}^{dis}$ for asset a). These are constrained by maximum power ratings ($P^{ch,max}, P^{dis,max}$) and energy capacities (E^{min}, E^{max}).

- **Grid Interaction:** The system can exchange energy with an external utility grid. The net power exchanged with the grid at timestep k, P_k^{grid}, is a variable determined by the energy balance of demands, on-site generation, and the dispatch of flexible assets. A positive P_k^{grid} typically denotes import (purchase) from the grid, while a negative value denotes export (sale).

- **Economic Parameters (Tariffs):** These define the costs and revenues associated with grid interaction and are essential for the economic objective function ℓ_k. As shown in Figure 3.4, these include electricity purchase prices (π^{buy}), sale prices (π^{sell}), and potentially demand charge rates (π^{DC}). These prices can be static, time-varying (e.g., TOU rates), or dynamic (e.g., real-time prices). If dynamic and not known perfectly in advance, their forecasts ($\hat{\pi}_{k|t}^{buy}, \hat{\pi}_{k|t}^{sell}$) become part of the

forecast input vector $\hat{\mathbf{y}}_{k|t}$.

The specific mathematical formulation of the state dynamics $f(\cdot)$, operational constraints $g(\cdot)$, and the instantiation of these components vary depending on the particular case study and research question, as detailed in Section 3.2.

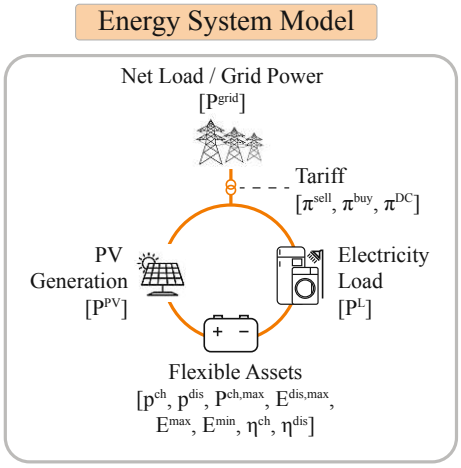


Figure 3.4.: Techno-Economic Energy System Model

3.1.2.3. Implementation via Receding Horizon

As is standard in MPC, the optimization problem (3.17) is solved repeatedly at each timestep t. Only the first element of the optimal control sequence, $\mathbf{u}_0^*$, is applied to the actual system (or simulation environment) for the interval $[t, t+\Delta t)$. The horizon then recedes: the system state evolves to $\mathbf{x}_{t+1}$ (which becomes $\mathbf{x}_{current}$ for the next iteration), new forecast input vectors $\hat{\mathbf{y}}^{(t+1)} = \{\hat{\mathbf{y}}_{k|t+1}\}_{k=0}^{H-1}$ are obtained, and the optimization is resolved at time $t+1$ (Rawlings et al., 2012). The simulation studies within RQ1, and RQ2 use an simulation-based approach, where the state update for the next optimization step ($\mathbf{x}_{current}$ at $t+1$) relies on the system model $f(\mathbf{x}_0^*, \mathbf{u}_0^*, \mathbf{z}_0^*, \hat{\mathbf{y}}_{0|t})$ used within the MPC, rather than direct measurements from a physical system.

3.1.2.4. Stability Considerations

While a rigorous analysis of closed-loop stability is a cornerstone of eMPC theory (Rawlings et al., 2012), this lies outside of the scope of this thesis. Still, it is important to acknowledge standard techniques used to promote stable closed-loop behavior. These typically involve the careful design of:

- Terminal cost functions $V_f(\mathbf{x}_H)$ that approximate the long-term economic performance or penalize deviations from desirable terminal states (Mayne et al., 2000).
- Terminal constraints that require the state $\mathbf{x}_H$ to lie within a predefined invariant or stabilizing set.
- Choosing a prediction horizon H that is sufficiently long relative to the system dynamics.

In economic eMPC, where the control objective is economic optimization rather than stabilization to a specific equilibrium, notions of *dissipativity* have been developed to assess average performance and ensure boundedness of the closed-loop trajectories—even in the absence of convergence to a fixed point (Angeli et al., 2012). In the methods employed in this thesis (Section 3.2), terminal costs or constraints are added for practical reasons (*e.g.*, to enforce a target final state to compare forecast accuracy fairly), rather than as part of a formal stability guarantee.

3.2. Methods Per Research Question

Building upon the general frameworks for Supervised Learning (Section 3.1.1) and Economic Model Predictive Control (Section 3.1.2), this section elaborates on the specific methodologies tailored to each research question. Table 3.1 provides a comparative overview of these approaches.

Table 3.1.: Overview of Methodologies per Research Question (explaning Figure 3.1 in more detail)

Feature	RQ1: Benchmarking	RQ2: Interpretable Forecasting	RQ3: Optimal Dispatch
Main Objective	Benchmark SL models; Economic evaluation via Net Load Error (NLE).	Interpretable Net Load Forecasting via Model-Based Deep Learning (DL).	Optimal Microgrid Dispatch (Cost/CO_2) using eMPC.
Core Framework	SL models evaluated using an eMPC-based NLE metric.	Model-Based DL: NN integrates physical PV model & embedded optimization.	Rolling Horizon eMPC using Mixed-Integer Linear Programming (MILP).
Energy System	Generic Load + BESS + Grid (for NLE evaluation).	Household: Load + PV + BESS + Grid.	REC: Load + PV + BESS + DHS + PtH + Grid.
Predictive Model	Trees (Direct) & Neural Networks (MIMO) benchmarked.	Custom NN (TiDE-based) + PVWatts physical model.	Hybrid XGBoost (Recursive short-term + One-shot long-term strategies).
Control / Optimization Policy	eMPC simulating Peak Shaving for NLE calculation.	Embedded HEMS Optimization (TOU Cost Min.) within forecast model.	eMPC solving MILP for Cost/CO_2 minimization (various tariffs).
Prediction Horizon (H)	1, 8, 24, 48 hours	24 hours	48 hours
Control Timestep (Δt)	1 hour	15 min	5 min

3.2.1. RQ1: Evaluating Forecasting Errors through the Net Load Error Framework

Research Question 1: *How do existing supervised learning-based forecasting methods perform in terms of statistical and economic error metrics for the task of peak load forecasting?*

To address RQ1, various state-of-the art model architectures were implement and benchmarked using conventional error metrics and the Net Load Error (NLE) framework (Houben et al., 2025a). The NLE quantifies the economic impact of load forecast errors within a peak load shaving context, specifically using an eMPC simulation as an evaluation tool for various time series forecasting algorithms.

3.2.1.1. Economic Model and Control Policy (NLE Framework)

The NLE framework simulates a simplified energy system (Figure 3.5) controlled via eMPC. This simulation isolates the cost impact specifically attributable to forecast errors ($\epsilon_{k|t} = y_{t+k} - \hat{y}_{k|t}$) under a daily peak demand charge tariff (π^{DC}) (Houben et al., 2025a).

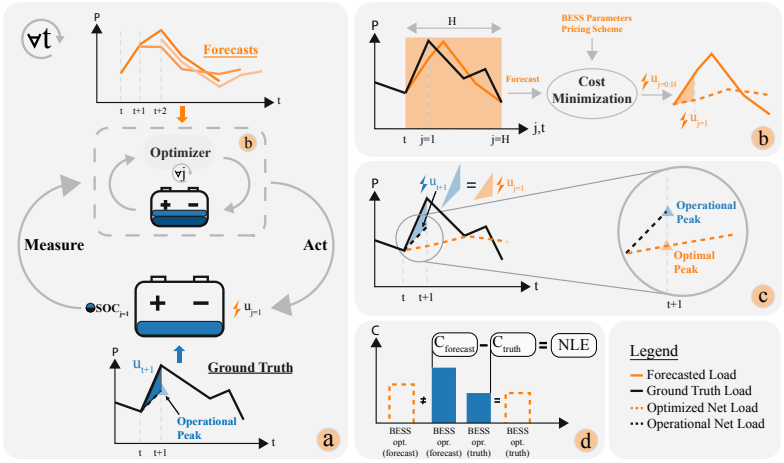


Figure 3.5.: Net Load Error Framework Schematic (Houben et al., 2025a)

Energy System Model (NLE Instantiation) This model instantiates the general eMPC framework (Section 3.1.2) with the following components:

- **Load**: The system considers an electrical load y_t [kW] (ground truth) and its forecast $\hat{y}_{k|t}$ [kW]. For the simulation, load values are scaled to $[0, 1]$. The forecast $\hat{y}_{k|t}$ serves as the single component of the forecast input vector $\hat{\mathbf{y}}_{k|t}$ for the eMPC.

- **BESS**: A single Battery Energy Storage System is modelled.

  - State vector $\mathbf{x}_k$: Represents the energy stored, E_k [kWh], i.e., $\mathbf{x}_k = [E_k]$.

  - State limits $\mathcal{X}$: Defined by the capacity E^{max}, so $0 \leq E_k \leq E^{max}$. $\mathcal{X} = [0, E^{max}]$.

  - Control input vector $\mathbf{u}_k$: Represents the charging/discharging power, P_k^{BESS} [kW], i.e., $\mathbf{u}_k = [P_k^{BESS}]$. Negative values denote discharging, positive values charging.

  - Input limits $\mathcal{U}$: Defined by the maximum power rating P^{max}, so

$-P^{max} \leq P_k^{BESS} \leq P^{max}$. $\mathcal{U} = [-P^{max}, P^{max}]$.

- Dynamics f: Instantiates Eq. (3.17b) with a lossless model: $E_{k+1} = E_k + P_k^{BESS} \cdot \Delta t$.

- Parameters: For comparability, E^{max} and P^{max} are set to 1.0 kWh and 1.0 kW respectively (Houben et al., 2025a).

- **Grid Interaction**: The net grid power within the optimization at step k is $P_k^{grid} = \hat{y}_{k|t} + P_k^{BESS}$. The **operational** grid power realized at time t based on the first implemented control action P_0^{BESS*} is $P_t^{grid,opr} = y_t + P_0^{BESS*}$.

- **Demand Charge Tariff**: A daily demand charge tariff with rate π^{DC} is applied ex-post based on the maximum operational grid power observed each day: $\max_{t \in day\, d}(P_t^{grid,opr})$. This differs from the general formulation Eq. (3.20) as the charge is calculated ex-post, not optimized directly within the horizon using p^h.

eMPC Optimization (Internal to NLE) The eMPC formulation Eq. (3.17) is instantiated to find the optimal control sequence $\mathbf{u}^*_{0:H-1} = \{P_k^{BESS*}\}_{k=0}^{H-1}$ and the minimum predicted peak power p^{h*}. Due to the ex-post calculation of the demand charge, the stage cost ℓ_k in Eq. (3.17a) is effectively zero. Instead, the objective minimizes a proxy for the peak power p^h over the horizon, combined with a terminal cost $V_f(E_H)$:

$$(\mathbf{u}^*_{0:H-1}, p^{h*}) = \underset{\mathbf{u}_{0:H-1}, p^h}{\arg\min} \quad \left[p^h + V_f(E_H)\right] \tag{3.22}$$
$$\text{where } V_f(E_H) = w_{E_H}(E_H - E_0)^2$$

subject to:

- BESS dynamics (instantiating Eq. (3.17b)): $E_{k+1} = E_k + P_k^{BESS} \cdot \Delta t$.

- State and Input Constraints (instantiating Eq. (3.17c), (3.17d)): $E_k \in [0, E^{max}]$, $P_k^{BESS} \in [-P^{max}, P^{max}]$.

- Initial State (Eq. (3.17g)): $E_0 = E_{current}$.

- Peak Power Path Constraint (instantiating Eq. (3.17e)): $p^h \geq \hat{y}_{k|t} + P_k^{BESS}$, for $k = 0, \ldots, H-1$. This links the decision variables $\mathbf{u}_k = [P_k^{BESS}]$ and p^h to the forecast input $\hat{y}_{k|t}$.

The terminal cost $V_f(E_H)$ penalizes the squared deviation of the final battery state E_H from its initial state E_0. The weight w_{E_H} is set to 10^2, a high value chosen to strongly discourage significant net changes in the battery's state of charge over the control horizon, thereby forcing the solution for all forecasting time series to be fairly comparable in terms of total costs (Houben et al., 2025a).

NLE Calculation Logic The NLE score compares the ex-post operational demand charge costs ($C^{opr} = \sum_d \max_{t \in day\, d}(P_t^{grid,opr}) \cdot \pi^{DC}$) from two simulations (Houben et al., 2025a):

1. **Actual Scenario**: Uses the forecast sequence $\{\hat{y}_{k|t}\}_{k=0}^{H-1}$ as input $\hat{\mathbf{y}}_{k|t} = [\hat{y}_{k|t}]$ to the eMPC at each step t.

2. **Ideal Scenario**: Uses the ground truth sequence $\{y_{t+k}\}_{k=0}^{H-1}$ as a perfect forecast input $\hat{\mathbf{y}}_{k|t} = [y_{t+k}]$ to the eMPC at each step t.

The simulation process (Fig 3.5) involves:

- *Optimization:* At each t, solve Eq. (3.22) using the relevant forecast ($\hat{y}_{k|t}$ or y_{t+k}) to find $\mathbf{u}^*_{0:H-1}$.

- *Implementation:* Apply only the first action $\mathbf{u}_0^* = [P_0^{BESS*}]$. Calculate operational net load using the actual load y_t: $P_t^{grid,opr} = y_t + P_0^{BESS*}$. If $\hat{y}_{0|t} \neq y_t$, $P_t^{grid,opr}$ will differ from the planned peak-minimized value.

- *State Update:* Update $E_{current}$ for the next step $t+1$ using $E_{t+1} = E_t + P_0^{BESS*}\Delta t$.

- *NLE Calculation:* After simulating the entire test period, calculate

total costs C^{opr}_{actual} and C^{opr}_{ideal} based on daily peaks of $P^{grid,opr}_t$. The NLE is the difference:

$$\text{NLE} = C^{opr}_{actual} - C^{opr}_{ideal} \tag{3.23}$$

NLE measures the extra cost incurred due to using the imperfect forecast $\hat{y}_{k|t}$ compared to perfect knowledge y_{t+k}.

3.2.1.2. Predictor: Machine Learning Model Benchmarking

This work applied the SL framework (Section 3.1.1), using methods detailed in (Houben et al., 2025a), to train and benchmark various algorithms for predicting the load y_{t+k}. Mathematical details of all algorithms and architectures are given in Appendix A.

- **Algorithms Benchmarked** (Table 3.2): Includes Random Forest (RF), XGBoost, LightGBM using the Direct strategy (Eq. (3.4)); and GRU (implemented as BlockRNN), N-BEATS, TFT using the MIMO strategy (Eq. (3.5)). A Linear Regression model (Direct strategy) served as a baseline. Specific algorithm details can be found in (Houben et al., 2025a) Appendix A and the original references (Table 3.2).

- **Features ($\mathbf{X}^{(t)}$):** Instantiated the general feature sets from Section 3.1.1. $\mathbf{x}^{\text{past}}$ included historical load values (lags up to $L = 72$ hours). $\mathbf{x}^{\text{future}}_k$ included time-based features (e.g., trigonometric day/week cycle encodings, holiday flags) and forecasts of air temperature (Houben et al., 2025a). An example trigonometric feature for the hour of day (relative to time $t + k$) is:

  $$x^{\text{hour,sin}}_k = \sin\left(\frac{2\pi \cdot \text{hour}(t+k)}{24}\right), \quad x^{\text{hour,cos}}_k = \cos\left(\frac{2\pi \cdot \text{hour}(t+k)}{24}\right) \tag{3.24}$$

  These $x^{\text{hour,sin}}_k$ and $x^{\text{hour,cos}}_k$ would be components of the vector $\mathbf{x}^{\text{future}}_k$.

- **Training & Tuning:** Models were trained minimizing the MSE loss

(Eq. (3.7)). Hyperparameters (ϕ) were optimized using Bayesian Optimization Hyperband (Biewald, 2020; Falkner et al., 2018).

- **Evaluation Metrics:** Performance was assessed using standard statistical metrics (RMSE, MAE, MAPE, R2 - Section 3.1.1.5), Skill Scores (Eq. (3.16)), error distribution analysis, and the NLE score described above (Houben et al., 2025a).

Table 3.2.: Overview of the SL algorithms benchmarked in RQ1 (Houben et al., 2025a) (Table 2)

Algorithm	**Type**	**Mechanism**	**Multi-Step**	**Reference**
Random Forest	Decision Tree Ensemble	Bagging	Direct	Breiman, 2001
XGBoost	Decision Tree Ensemble	Grad. Boosting & Depth First	Direct	Chen and Guestrin, 2016
LightGBM	Decision Tree Ensemble	Grad. Boosting & Breadth First	Direct	Ke et al., 2017
GRU	Deep Neural Network	Memory Gates	MIMO	Chung et al., 2014
N-BEATS	Deep Neural Network	Basis Expansion	MIMO	Oreshkin et al., 2019
TFT	Deep Neural Network	Attention	MIMO	Vaswani et al., 2017; Lim et al., 2021

3.2.2. RQ2: Integration of Household Energy Management into Net Load Forecasting Models

Research Question 2: *How can net load forecasting be improved for households with behind-the-meter PV systems, battery storage, and HEMS, while enhancing interpretability and data efficiency through novel architectures?*

In order to answer RQ2, this thesis explored improving household net load forecasting by integrating physical and operational knowledge into a predictive model. As illustrated in Figure 3.6, a model-based deep learning approach was developed (Houben et al., 2025b). This approach combines a physical PV model with a neural network that includes an embedded, differentiable optimization layer of the household's energy management system.

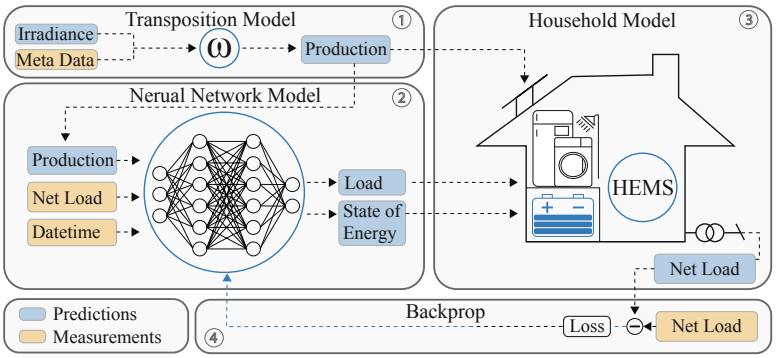


Figure 3.6.: Modeling framework overview for RQ2: Model-based deep learning using physical PV and HEMS optimization priors (Houben et al., 2025b).

As depicted in Figure 3.61), the transposition model generates a PV power forecast based on weather forecasts and PV metadata. In standard indirect modeling approaches, the output of this model would be subtracted from the net load to create training samples of residual (non-flexible) load, which can effectively be learned by a data-driven method. For the task at hand, however, this would not lead to a satisfactory decomposition into a non-

flexible load, as the resulting profile would still contain the battery power actions. In the approach developed in this work, the output of the transposition model is therefore utilized in two locations: i) as an input to the forecasting model (Figure 3.62)) and ii) as an input to the household model (Figure 3.63)). By utilizing the PV power prediction from the transposition model as an input to the neural network and explicitly employing the outputs of the neural network in the differentiable household layer, an interpretable and data-efficient multi-step ahead forecasting model of the net load is learned. It should be noted that although the neural network model and the household model are presented as two separate components in this work, they are trained jointly as a single model, with the household model functioning as the output layer (Houben et al., 2025b).

3.2.2.1. Energy System Model: Household Energy Management System (HEMS)

This component models the operational behavior of a household equipped with non-flexible load (y_t^{load}), photovoltaic generation (y_t^{PV}), and a battery energy storage system (BESS). The household operates under a time-of-use (TOU) purchase tariff (π_k^{buy}) and a flat feed-in tariff (π_k^{sell}) (Houben et al., 2025b). A more realistic version of this HEMS model (including battery efficiencies) was used to generate the ground truth net load data y_t^{net} for the study, while a simplified, more computationally efficient version (*i.e.*, assuming lossless BESS and no binary variables) is formulated as a differentiable optimization layer within the neural network forecasting model (Houben et al., 2025b).

This differentiable layer instantiates the general eMPC framework (Section 3.1.2) to simulate the HEMS decision-making over a horizon $H = 24$ hours.

- **State vector $\mathbf{x}_k$:** Represents the BESS energy stored, E_k [kWh], i.e., $\mathbf{x}_k = [E_k]$.
- **Control input vector $\mathbf{u}_k$:** Represents the BESS charge/discharge power, P_k^{BESS} [kW], i.e., $\mathbf{u}_k = [P_k^{BESS}]$.

- **Forecast Input Vector $\hat{\mathbf{y}}_{k|t}$:** This includes forecasts needed for the optimization at step k: the non-flexible load forecast $\hat{y}^{\text{load}}_{k|t}$, the PV production forecast $\hat{y}^{\text{PV}}_{k|t}$, the purchase price π^{buy}_{t+k}, and the feed-in tariff π^{sell}_{t+k}. Note that $\hat{y}^{\text{load}}_{k|t}$ and the initial state E_0 are provided by the main neural network component (see Section 3.2.2.2), while $\hat{y}^{\text{PV}}_{k|t}$ comes from the physical PV model, and prices are assumed known.

- **Objective Function:** Instantiates the general objective Eq. (3.17a) to minimize the predicted energy cost over the horizon, based on the energy cost definition Eq. (3.19). A terminal state constraint effectively acts as the terminal cost V_f. The specific problem is:

$$\min_{\mathbf{u}_{0:H-1}} \quad \sum_{k=0}^{H-1} \ell^{\text{energy}}_k \quad \text{s.t. terminal constraint on } E_H \tag{3.25}$$

  where the stage cost ℓ^{energy}_k is calculated using Eq. (3.19) with the predicted grid power $\hat{P}^{\text{grid}}_k$:

$$\hat{P}^{\text{grid}}_k = P^{BESS}_k + \hat{y}^{\text{load}}_{k|t} - \hat{y}^{\text{PV}}_{k|t} \tag{3.26}$$

  and using the known prices $\pi^{\text{buy}}_{t+k}, \pi^{\text{sell}}_{t+k}$ for $\hat{\pi}^{\text{buy}}_{k|t}, \hat{\pi}^{\text{sell}}_{k|t}$.

- **Constraints:** Instantiate the general constraints:

  - Dynamics (Eq. (3.17b)): $E_{k+1} = E_k + P^{BESS}_k \cdot \Delta t$ (lossless model used in the layer).

  - State Limits $\mathcal{X}$ (Eq. (3.17c)): $0 \leq E_k \leq E^{\text{cap}}$.

  - Input Limits $\mathcal{U}$ (Eq. (3.17d)): $-P^{\text{dis,max}} \leq P^{BESS}_k \leq P^{\text{ch,max}}$.

  - Terminal Constraint (related to V_f): $E_H = E^{\text{target}}_H$. The target E^{target}_H is set based on logic considering the time of day or previous HEMS operation (Houben et al., 2025b).

This embedded optimization layer allows the overall model to learn forecasting targets that are consistent with optimal HEMS operation based on the

predicted load, PV, and known tariffs.

3.2.2.2. Predictor: Deep Neural Network & Physical PV Model

The forecasting model $\mathcal{F}$ is a MIMO model, generating the final net load forecast $\hat{y}^{\text{net}}_{k|t}$ integrates three parts: a physical PV model, a deep neural network, and the differentiable HEMS optimization layer described above.

PV Forecasting Model: Irradiance-Power Transposition A physical model, specifically PVWatts (Dobos, 2014), is used to predict the PV generation $\hat{y}^{\text{PV}}_{k|t}$. This component corresponds to a specific function within the overall SL model $\mathcal{F}$. It uses weather forecasts (components of $\mathbf{x}^{\text{future}}_k$, e.g., $\widehat{\text{DNI}}_{k|t}, \widehat{\text{DHI}}_{k|t}, \widehat{\text{GHI}}_{k|t}$) and static PV system metadata ($P_{STC}, \theta_T, \theta_{A,arr}$, etc.) as inputs (Houben et al., 2025b). The steps involve calculating the plane-of-array irradiance $\hat{G}_{POA,k|t}$ using the Perez model (Perez et al., 1990) and then applying the solar power curve:

$$\hat{G}_{POA,k|t} = f_{Perez}(\widehat{\text{DNI}}_{k|t}, \widehat{\text{DHI}}_{k|t}, \widehat{\text{GHI}}_{k|t}, \text{sun angles}_k, \text{PV angles}) \tag{3.27}$$

$$\hat{y}^{\text{PV}}_{k|t} = \frac{P_{STC} \cdot \hat{G}_{POA,k|t}}{G_{\text{ref}}} \left(1 + \gamma(\hat{T}_{\text{cell},k|t} - T_{\text{ref}})\right) \eta_{inv} \tag{3.28}$$

where $\hat{T}_{\text{cell},k|t}$ is the predicted cell temperature at step k. The output $\hat{y}^{\text{PV}}_{k|t}$ is used both as input to the main DNN and the HEMS layer.

Deep Neural Network The core data-driven component is a neural network based on a modified Time-series Dense Encoder (TiDE) architecture (Das et al., 2024), as illustrated in Figure 3.7. TiDE is specifically designed for multi-step ahead forecasting and effectively processes covariates by distinguishing between past and future features (Houben et al., 2025b). This network, as part of the overall model $\mathcal{F}$, predicts the inputs required by the HEMS optimization layer: the non-flexible load sequence $\{\hat{y}^{\text{load}}_{k|t}\}_{k=1}^{H}$ (denoted $\hat{P}^{load}_{t:t+H}$ in Figure 3.7) and the initial BESS state $\hat{E}_t$ (denoted $\hat{e}_t$ in Figure 3.7, corresponding to $\mathbf{x}_{0,\text{HEMS}}$).

The model takes as input features derived from $\mathbf{X}^{(t)}$ (defined in Section 3.1.1):

- The historical features $\{\mathbf{x}_{t-\tau}^{\text{past}}\}_{\tau=0}^{L}$ (e.g., $p_{t-L:t}^{PV}, p_{t-L:t}^{net}$ in Figure 3.7) are first processed by n_r Residual Blocks (Eq. (3.34)) on a per-timestep basis. This yields the processed past feature sequence $\mathbf{X}_{t-L:t} \in \mathbb{R}^{L \times d_{feat}}$, as depicted in Figure 3.7.

- Similarly, the known future features $\{\mathbf{x}_{k}^{\text{future}}\}_{k=1}^{H}$ (e.g., $\hat{P}_{t:t+H}^{PV}, \delta_{t:t+H}^{day}, \delta_{t:t+H}^{qh}$ in Figure 3.7) are processed by n_r Residual Blocks per timestep, yielding the processed future feature sequence $\mathbf{X}_{t:t+H} \in \mathbb{R}^{H \times d_{feat}}$, also shown in Figure 3.7.

The learnable weights of these initial Residual Blocks are part of the model parameters $\boldsymbol{\theta}$. These blocks, further detailed in (He et al., 2016; Houben et al., 2025b), facilitate gradient flow and learn complex non-linear dependencies.

The concatenated sequence of these processed features, $\mathbf{X}_{t-L:t}$ and $\mathbf{X}_{t:t+H}$, forms the input to the subsequent Decoder module:

$$\mathbf{X}_{dec\_in} = \text{concat}(\mathbf{X}_{t-L:t}, \mathbf{X}_{t:t+H}) \quad \text{where } \mathbf{X}_{dec\_in} \in \mathbb{R}^{(L+H) \times d_{feat}} \tag{3.29}$$

The Decoder module incorporates a Self-Attention mechanism, an important component of the Transformer architecture (Vaswani et al., 2017), to weigh the importance of different parts of the input sequence $\mathbf{X}_{dec\_in}$ (Houben et al., 2025b). From $\mathbf{X}_{dec\_in}$, the Query ($\mathbf{Q}$), Key ($\mathbf{K}$), and Value ($\mathbf{V}$) matrices are computed as linear projections using learnable weight matrices $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V$ (which are components of $\boldsymbol{\theta}$):

$$\mathbf{Q} = \mathbf{X}_{dec\_in}\mathbf{W}^Q \tag{3.30}$$
$$\mathbf{K} = \mathbf{X}_{dec\_in}\mathbf{W}^K \tag{3.31}$$
$$\mathbf{V} = \mathbf{X}_{dec\_in}\mathbf{W}^V \tag{3.32}$$

where $\mathbf{W}^Q \in \mathbb{R}^{d_{feat} \times d_q}$, $\mathbf{W}^K \in \mathbb{R}^{d_{feat} \times d_k}$, and $\mathbf{W}^V \in \mathbb{R}^{d_{feat} \times d_v}$. Typically, $d_q = d_k$. The attention output matrix $\mathbf{G}_{att}$ (denoted $g^{(t)}$ in Figure 3.7) is

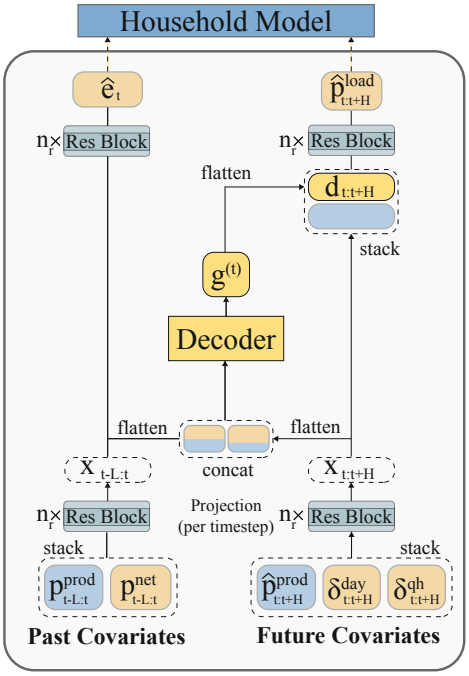


Figure 3.7.: Deep Neural Network Architecture based on a modified TiDE model (Houben et al., 2025b). The terms $\mathbf{X}_{t-L:t}$ and $\mathbf{X}_{t:t+H}$ represent processed past and future feature sequences, respectively, which form the input to the Decoder after concatenation. The Decoder output $g^{(t)}$ (below denoted $\mathbf{G}_{att}$) is then transformed into predictions $\hat{P}^{load}_{t:t+H}$ and $\hat{e}_t$.

then calculated using scaled dot-product attention:

$$\mathbf{G}_{att} = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{3.33}$$

where $\mathbf{G}_{att} \in \mathbb{R}^{(L+H)\times d_v}$. This output matrix $\mathbf{G}_{att}$ is then further processed by subsequent layers within the Decoder (often including more Residual Blocks and feed-forward networks, as generally depicted by the path to $\hat{P}^{load}_{t:t+H}$ and $\hat{e}_t$ in Figure 3.7) to generate the intermediary predictions of non-flexible load and the initial BESS state.

The formulation of a Residual Block is given by:

$$\mathbf{h}^{(l+1)} = \mathbf{h}^{(l)} + \text{ReLU}(\mathbf{W}^{(l)}\mathbf{h}^{(l)} + \mathbf{B}^{(l)}) \tag{3.34}$$

where $\mathbf{h}^{(l)}$ is the input to the l-th block, and $\mathbf{W}^{(l)}, \mathbf{B}^{(l)}$ are its learnable parameters (part of $\boldsymbol{\theta}$).

The entire model (PV model → DNN → HEMS layer) is trained end-to-end by minimizing a loss function (*i.e.*, MSE, Eq. (3.7)) calculated between the final predicted net load $\hat{y}^{\text{net}}_{k|t}$ (output by the HEMS layer) and the actual net load y^{net}_{t+k}.

3.2.3. RQ3: Optimal Operational Dispatch of a Multi-Energy Microgrid under Uncertainty

Research Question 3: *What is the economic impact of load and photovoltaic generation forecast errors in a multi-energy microgrid, subject to various electricity pricing systems?*

For RQ3, an eMPC framework was developed to optimally dispatch a multi-energy Renewable Energy Community (REC) testbed, specifically analyzing its performance under forecast uncertainty across various electricity pricing schemes (Houben et al., 2023).

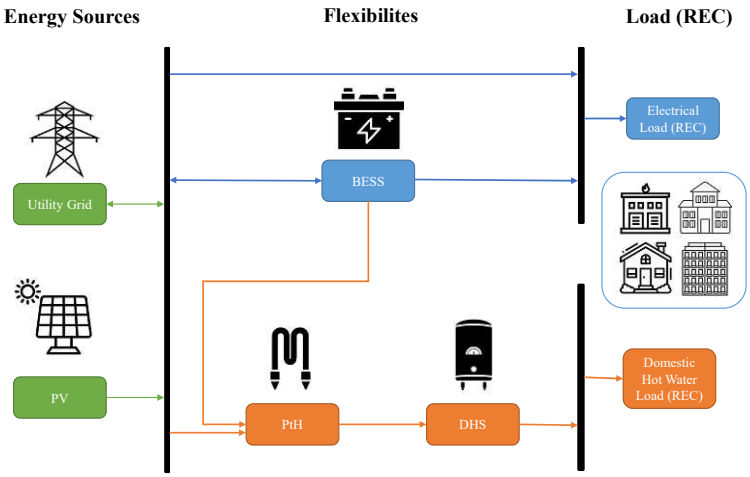


Figure 3.8.: Energy System Model for RQ3

3.2.3.1. Economic Model: Multi-Energy REC System and MILP Formulation

As shown in Figure 3.8, the energy system model represents the REC, including electrical loads (y_t^{load}), thermal loads (y_t^{DHW}), PV generation (y_t^{PV}), a Battery Energy Storage System (BESS), a Power-to-Heat unit (PtH), a Domestic Heat Storage (DHS), and grid connection (Houben et al., 2023). The system is formulated as a Mixed-Integer Linear Program (MILP) that instantiates the general eMPC framework (Section 3.1.2).

REC System State and Control To align with the general framework (Eq. (3.17)), we define the core vectors for the optimization at step k:

- **State vector $\mathbf{x}_k \in \mathbb{R}^2$:** Contains the energy stored in the BESS and DHS. $\mathbf{x}_k = [E_k^{BESS}, E_k^{DHS}]^T$. $(n_x = 2)$.
- **Control input vector $\mathbf{u}_k \in \mathbb{R}^4$:** Includes the primary power decisions for the storage assets. $\mathbf{u}_k = [P_k^{BESS,ch}, P_k^{BESS,dis}, P_k^{DHS,ch}, P_k^{DHS,dis}]^T$. $(n_u = 4)$. These represent the power flow into or out of the storage

units.

- **Binary decision variables $\mathbf{z}_k \in \{0,1\}^4$:** Includes variables to enforce unidirectional power flow for storage: $\mathbf{z}_k = [z_k^{BESS,ch}, z_k^{BESS,dis}, z_k^{DHS,ch}, z_k^{DHS,dis}]^T$. $(n_z = 4)$.

- **Forecast Input Vector $\hat{\mathbf{y}}_{k|t} \in \mathbb{R}^{n_y}$:** Contains forecasts for uncontrollable variables: $\hat{\mathbf{y}}_{k|t} = [\hat{y}_{k|t}^{\text{load}}, \hat{y}_{k|t}^{\text{PV}}, \hat{y}_{k|t}^{\text{DHW}}, \hat{c}_{k|t}^{CO2}]^T$. $(n_y = 4)$.

Other variables, such as grid power (P_k^{grid}), PtH input power ($P_k^{PtH,in}$), PV self-consumption ($P_k^{PV,onsite}$), etc., are treated as dependent variables determined by the energy balance constraints.

MILP Optimization Problem The MILP is solved at each decision step t over horizon H.

- **Objective:** Instantiates Eq. (3.17a). The stage cost $\ell_k = \ell(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \hat{\mathbf{y}}_{k|t})$ minimizes either Cost or CO_2.

  - Cost: $\ell_k = \ell_k^{\text{energy}} + \ell_k^{\text{DC}}$. ℓ_k^{energy} (Eq. (3.19)) uses prices $\hat{\pi}_{k|t}^{\text{buy/sell}}$ and the grid power P_k^{grid}. ℓ_k^{DC} handled via peak variable p^h constrained by dependent$[P_k^{grid}]^+$.

  - CO_2: $\ell_k = [P_k^{grid}]^+ \cdot \hat{c}_{k|t}^{CO2} \cdot \Delta t$.

- **Constraints:** Instantiate the general constraints (Eqs. (3.17b)-(3.17f)).

  - *Asset Dynamics (Eq.* (3.17b)*):* State transitions for E_{k+1}^{BESS} and E_{k+1}^{DHS} are defined by Eq. (3.2.3.1), using components of $\mathbf{x}_k$ and $\mathbf{u}_k$.

    $$E_{k+1}^a = E_k^a \cdot (1 - \Theta_{ES}^a \Delta t) + \left(\eta^{a,ch} P_k^{a,ch} - \frac{1}{\eta^{a,dis}} P_k^{a,dis}\right) \Delta t \quad \text{for } a \in \{\text{BESS, DHS}\}$$

  - *State and Input Limits (Eqs.* (3.17c)*,* (3.17d)*):* Define feasible

ranges $\mathcal{X}$ for $\mathbf{x}_k$ (e.g., $0 \leq E_k^a \leq Cap_a$) and $\mathcal{U}$ for $\mathbf{u}_k$ (e.g., $0 \leq P_k^{a,ch} \leq P^{a,ch,max}$, $0 \leq P_k^{a,dis} \leq P^{a,dis,max}$).

- *Constraints (Eq.* (3.17e)*):* The energy balance equations that link control inputs $\mathbf{u}_k$, forecasts $\hat{\mathbf{y}}_{k|t}$, and dependent power flows:
  - Electrical Balance: Determines net grid power $P_k^{grid} = P_k^{grid,buy} - P_k^{grid,sell}$. It ensures supply meets demand: $P_k^{grid} + \hat{y}_{k|t}^{PV} + P_k^{BESS,dis} = \hat{y}_{k|t}^{\text{load}} + P_k^{BESS,ch} + P_k^{PtH,in}$.
  - Thermal Balance: Determines the required electrical input to the PtH, $P_k^{PtH,in}$: $\eta_{PtH} P_k^{PtH,in} \Delta t + P_k^{DHS,dis} \Delta t = \hat{y}_{k|t}^{\text{DHW}} \Delta t + P_k^{DHS,ch} \Delta t$.
  - PtH Capacity: $0 \leq P_k^{PtH,in} \leq P^{PtH,max}$. This constrains the value determined by the thermal balance.
- *Binary Logic (Eq.* (3.17f)*):* Uses Eq. (3.21) applied to BESS and DHS, linking power variables in $\mathbf{u}_k$ to binary variables in $\mathbf{z}_k$.

The controller operates in a receding horizon, open-loop manner as described in Section 3.1.2.

3.2.3.2. Predictor: Hybrid Multi-Step Forecasting Method

Forecasts for electrical load ($\{\hat{y}_{k|t}^{\text{load}}\}_{k=1}^{H}$) and PV generation ($\{\hat{y}_{k|t}^{\text{PV}}\}_{k=1}^{H}$) are generated using a hybrid strategy combining two models based on the lead time k (Houben et al., 2023).

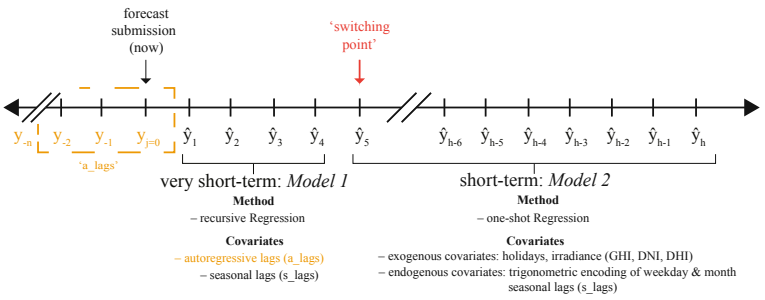


Figure 3.9.: Hybrid Forecasting Method: Combining recursive and MIMO outputs

- **Hybrid Strategy:** For lead times $k = 1, \ldots, s$, a short-term model (Model 1) using the Recursive strategy (Section 3.1.1.2, Eq. (3.3)) is employed. This model uses recent historical measurements (autoregressive features $\mathbf{x}^{\text{past}}$). For longer lead times $k = s+1, \ldots, H$, a separate model (Model 2) is used, which relies only on exogenous features ($\mathbf{x}_k^{\text{future}}$) such as time-based features and weather forecasts, avoiding the use of potentially error-prone recursive inputs. This study implemented Model 2 using a Direct-like strategy, training the model to predict each step $k > s$ based on exogenous inputs available at time t. The final forecast sequence is obtained by concatenating the outputs: $\{\hat{y}_{1|t}, ..., \hat{y}_{s|t}\}$ from Model 1 and $\{\hat{y}_{s+1|t}, ..., \hat{y}_{H|t}\}$ from Model 2.

- **Switching Point (s):** The optimal lead time s to switch from the recursive model to the exogenous-feature-based model is determined via cross-validation. It is typically chosen where the forecast error (e.g., nRMSE Eq. (3.14)) of the recursive model starts to exceed that of the non-recursive model (Houben et al., 2023).

- **Algorithm ($\mathcal{F}$):** XGBoost was selected as the underlying algorithm for both Model 1 and Model 2 based on cross-validation performance (Houben et al., 2023).

- **Features ($\mathbf{x}$):** Model 1 uses recent lags of the target variable ($\mathbf{x}^{\text{past}}$). Model 2 uses time features (e.g., hour, day of week) and weather forecasts (e.g., irradiance, temperature) as future features $\mathbf{x}_k^{\text{future}}$ (Houben et al., 2023).

3.2.4. Experimental Setup

The methodologies described were implemented primarily using **Python**. For SL model development (RQ1, RQ2, RQ3 predictor): used *Darts* (Herzen et al., 2022), *scikit-learn* (Pedregosa et al., 2011), *xgboost* (Chen and Guestrin, 2016), *PyTorch* (Paszke et al., 2019), and *Skforecast* (**Amatsk**). Experiment tracking via *Weights & Biases* (Biewald, 2020) (RQ1, RQ2). For Optimization and Simulation: RQ1 NLE used *Pyomo* with *CPLEX*. RQ2 used *cvxpylayers* (Agrawal et al., 2021). RQ3 used *GAMS* with *CPLEX* (**GAMS**). Simulations were used for (RQ1, RQ2). Hardware included Nvidia GPUs (RQ1, RQ2), cloud resources (RQ3 model selection), and servers (RQ3 eMPC) (Houben et al., 2025a; Houben et al., 2025b; Houben et al., 2023).

In summary, this section has detailed the specific methodological implementations undertaken for each of the three core research questions. Building upon the general principles of Supervised Learning for forecasting (Section 3.1.1) and Economic Model Predictive Control (Section 3.1.2), this chapter elaborated on the distinct energy system models, the tailored predictive algorithms and strategies, and the specific control or evaluation policies employed in RQ1, RQ2, and RQ3. This included the NLE framework for economic forecast evaluation, the model-based deep learning approach for interpretable net load forecasting, and the MILP-based eMPC for optimal multi-energy microgrid dispatch. The following section will synthesize these approaches, discussing their commonalities, differences, and overall contribution.

4. Case Studies and Results

Having detailed the underlying methodologies rooted in Supervised Learning (SL) and Economic Model Predictive Control (eMPC) in Chapter 3, this chapter now transitions to their practical application and evaluation. The specific methods developed and benchmarked are applied to distinct case studies designed to address the three core research questions (RQs) of this thesis. For each research question, the corresponding case study is first described, outlining the system context, data sources, and experimental design. Subsequently, the empirical results obtained from applying the respective methodologies are presented and analyzed. Figure 4.1 provides a visual overview, summarizing the case study context, indicating its level of spatial aggregation, and presenting the principal findings for each research question investigated in this work.

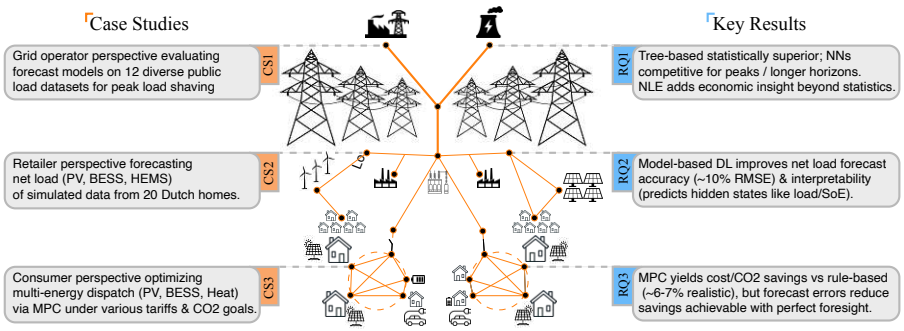


Figure 4.1.: Overview of case studies and results; linking the research questions to specific spatial scales.

4.1. RQ1: Benchmarking SL Models and Evaluating Economic Impact

This section details the case study setup and presents the results corresponding to Research Question 1, which focused on benchmarking supervised learning forecasting models across various spatial scales and evaluating their economic impact using the Net Load Error (NLE) framework developed in Section 3.2.1.1.

4.1.1. Case Study Description

The evaluation for RQ1 was performed on a diverse set of real-world electricity load time series to ensure the robustness and generalizability of the findings across different contexts (Houben et al., 2025a)(Sec 3).

This study drew on three open-source datasets, each including multiple load time series covering a broad range of geographies and consumer types. Specifically, the levels of spatial aggregation (referred to as *spatial scales*) between the datasets vary widely; with data collected at what will henceforth be referred to as county, town, neighborhood, and building levels. For instance, the county-level load time series, taken from the *EIA Cleaned Hourly Electricity Demand* dataset presented in (Ruggles et al., 2020), encompasses millions of end-users comprised of households, industries, transport, and public lighting, whereas the neighborhood-level dataset from the USA (Miller et al., 2020) consists primarily of commercial building loads. The diversity of datasets, summarized in Table 4.1, represents electricity demand across various scales and consumer types, forming a robust foundation for the subsequent evaluations.

From each dataset, three independent time series were selected, resulting in a total of 12 distinct load profiles for evaluation. A requirement for inclusion was at least two years of historical data to allow for one year of training data and separate summer/winter months for testing and validation, following the data splitting procedure depicted in Figure 4.2 (Houben et al.,

Table 4.1.: Overview of the datasets used for benchmarking forecasting models in RQ1 (taken from (Houben et al., 2025a)(Table 4).

Spatial Scale	Power Range	Res.	Country	Meter Label	Source
County	1–50 GW	60 min	USA	LDWP, BANC, NYIS	(Ruggles et al., 2020)
Town	5–50 MW	15 min	Portugal	MT_196, MT_279, MT_208	(Trindade, 2015)
Neighborhood	5–2000 kW	60 min	USA	Bull, Hog, Bobcat	(Miller et al., 2020)
Building	0–2 kW	60 min	USA	Be_Sandy, Be_Millie, Co_Joel	(Miller et al., 2020)

2025a)(Sec 3). Although sourced with varying native resolutions (Table 4.1), all data was resampled to a consistent hourly resolution ($\Delta t = 1$ hour) for this study. The preprocessing steps including outlier handling and missing value treatment, were applied uniformly. Additionally, hourly ambient temperature data, sourced via the Open-Meteo API (*Free Open-Source Weather API | Open-Meteo.com* 2023), was used as an exogenous predictor variable ($\mathbf{x}^{\text{future}}$) for the forecasting models, given its known influence on electricity consumption (Pardo et al., 2002; Behm et al., 2020).

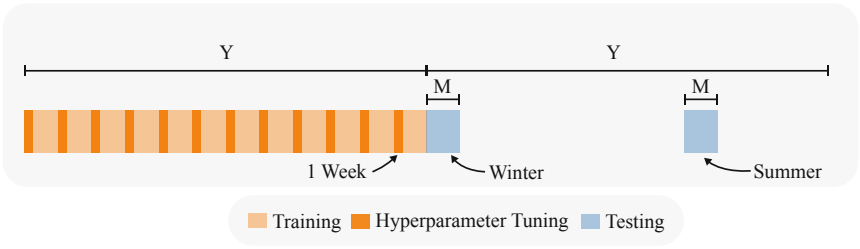


Figure 4.2.: Train and Test, Hyperparameter Tuning Data Splits (Houben et al., 2025a)

4.1.2. Results

The performance of the six benchmarked forecasting algorithms (detailed in Section 3.2.1.2) was evaluated on the 12 datasets across forecast horizons $H \in \{1, 8, 24, 48\}$ hours. This subsection summarizes the key findings regarding overall performance, the impact of seasonality, and the insights gained from the Net Load Error (NLE) analysis (Houben et al., 2025a)(Sec 4). The interpretation and discussion of these results are further elaborated in Chapter 5.

4.1.2.1. Overall Forecasting Performance

A qualitative comparison provides initial insights into model behavior. Figure 4.3 contrasts 24-hour ahead forecasts from tree-based models (left panel) and neural networks (right panel) against the ground truth for representative datasets from each spatial scale during days with extreme temperatures. As expected, load volatility increases significantly at lower levels of spatial aggregation (e.g., building vs. county). Both model types generally perform better on smoother, aggregated data. Qualitatively, tree-based models appear adept at capturing recurring patterns but sometimes under-predict extreme peaks, while neural networks show potential in reaching peak magnitudes but occasionally overestimate load during low-consumption periods and exhibit less smooth forecast trajectories (Houben et al., 2025a)(Sec 4.1).

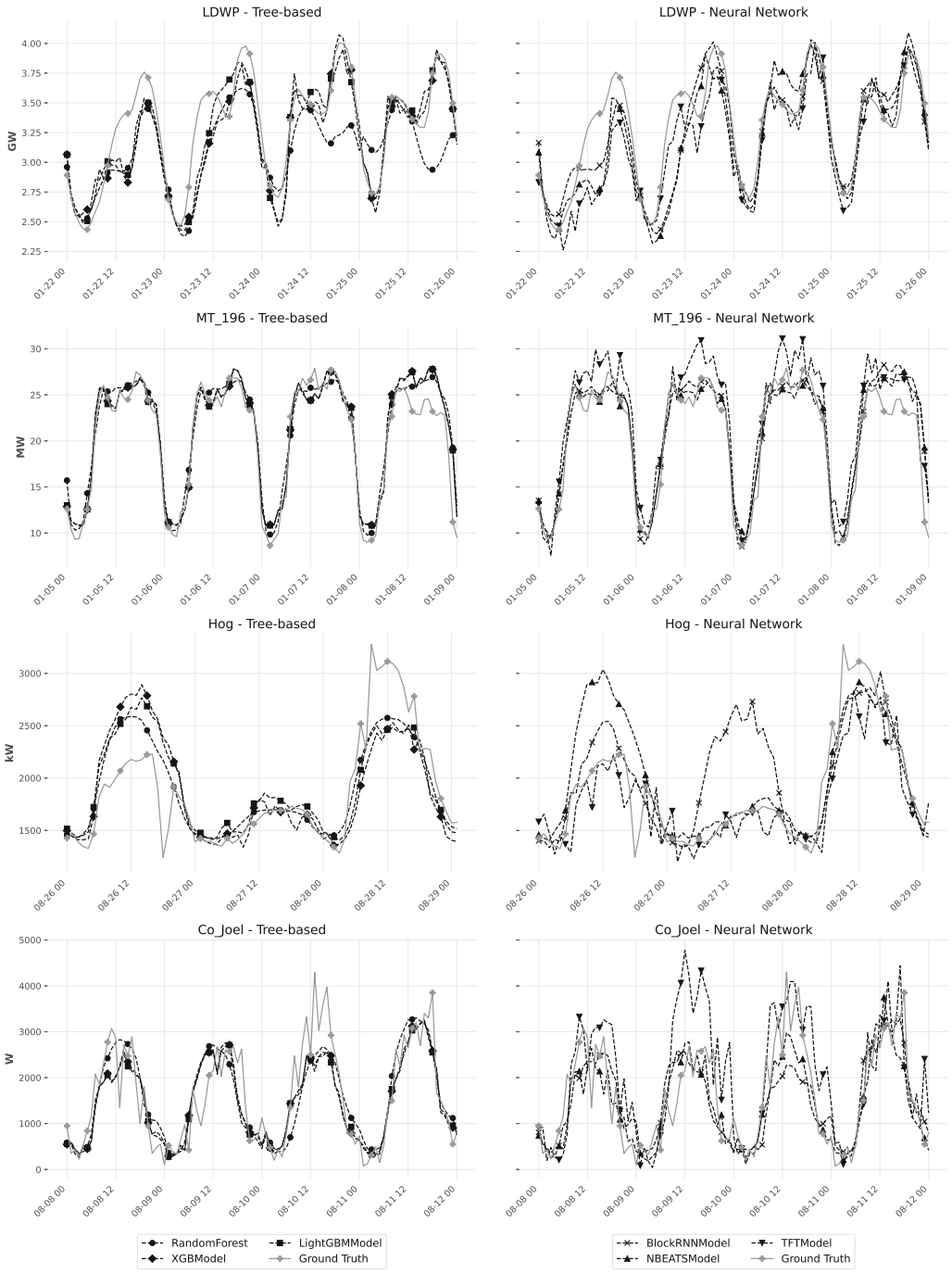


Figure 4.3.: Qualitative Comparison of Algorithm Types (Trees vs. NNs) across Spatial Scales for 24-Hour Ahead Forecasts (RQ1, taken from (Houben et al., 2025a)(Fig 5).

Quantitatively, model performance was compared by averaging error scores across datasets within each spatial scale and forecast horizon, grouping algorithms into Tree-based, Neural Network (NN), and the Linear Regression Benchmark categories (Figure 4.4). Generally, both tree-based and NN models outperform the simpler linear regression benchmark, especially for horizons $H \geq 8$ hours. The benchmark remains competitive only at very short horizons (1-4h) for the highly aggregated county-level data. Across most scales and metrics (MAE, MAPE, RMSE, R^2), tree-based algorithms demonstrated, on average, slightly better statistical performance than NNs. However, exceptions exist, notably at the county scale where NNs performed competitively on RMSE/MAE/R^2 and at the building scale for MAPE (though MAPE is less reliable here due to near-zero loads) (Houben et al., 2025a)(Sec 4.1.1).

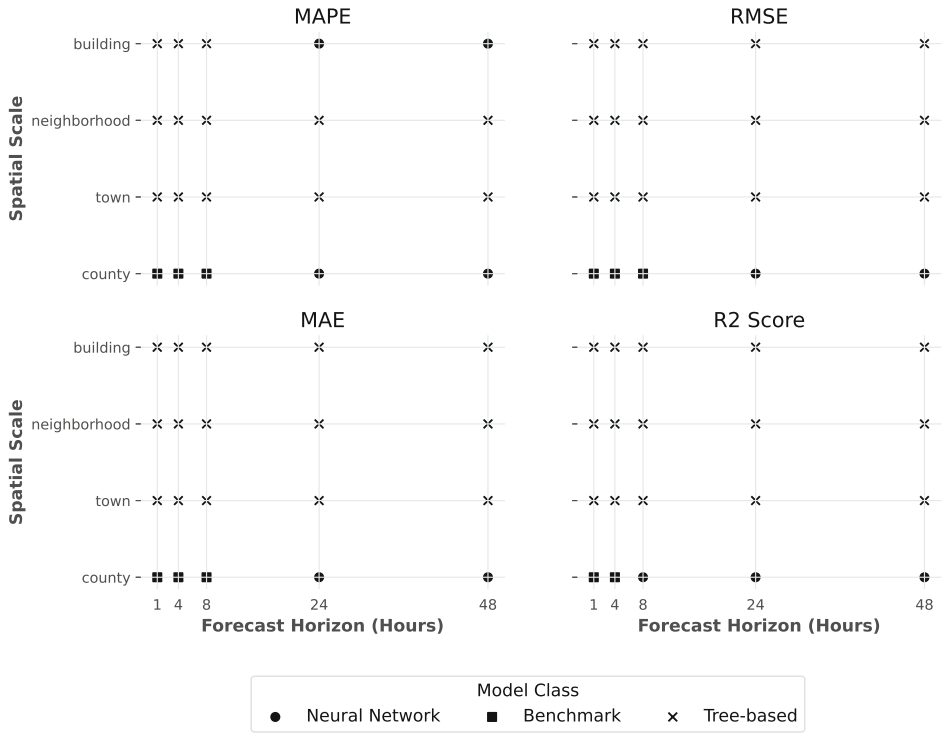


Figure 4.4.: Best Performing Model Type (Tree-based, NN, Benchmark) per Spatial Scale and Forecast Horizon based on Average Statistical Metrics (RQ1, taken from (Houben et al., 2025a)(Fig 6).

The relative performance is further illustrated using RMSE Skill Scores (Eq. (3.16)) against the linear regression benchmark (Figure 4.5, shown for county scale). While sophisticated models significantly outperform the benchmark at longer horizons ($H \geq 24h$), their skill scores can be negative at very short horizons (1-4h), particularly for NNs on aggregated data. Notably, NNs tend to match or exceed the performance of tree-based models around the 24-hour horizon mark for these county-level datasets (Houben et al., 2025a)(Sec 4.1.1).

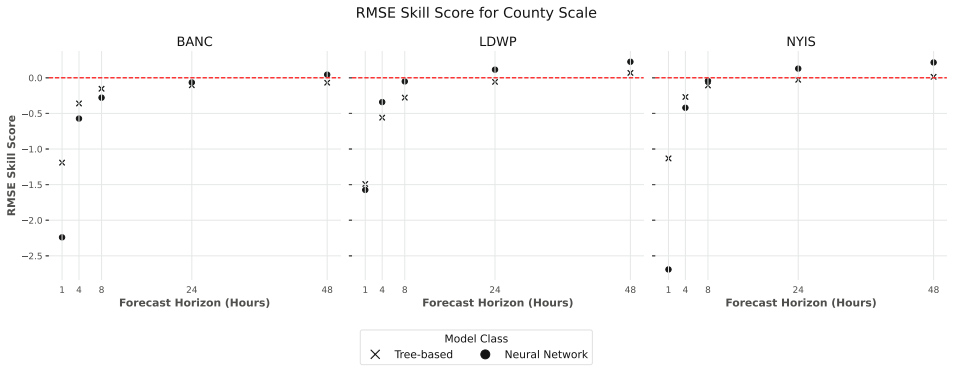


Figure 4.5.: RMSE Skill Score (Relative to Linear Regression Benchmark) for Tree-based vs. NN Model Classes on County Scale Datasets (RQ1, taken from (Houben et al., 2025a)(Fig 7).

4.1.2.2. Impact of Seasonality and Error Analysis

Performance variations between seasons (summer vs. winter test periods) were significant. Figure 4.6 shows RMSE skill scores for individual models on county-level data, separated by season. Most models exhibited lower skill scores (poorer performance relative to the benchmark) during summer compared to winter. This seasonal difference was consistent across most spatial scales, except for the building level (Houben et al., 2025a)(Sec 4.2). Examples of XGBoost forecasts for the LDWP dataset (Figures 4.7 and 4.8) illustrate this, showing accurate peak prediction in winter but underestimation in summer across various horizons.

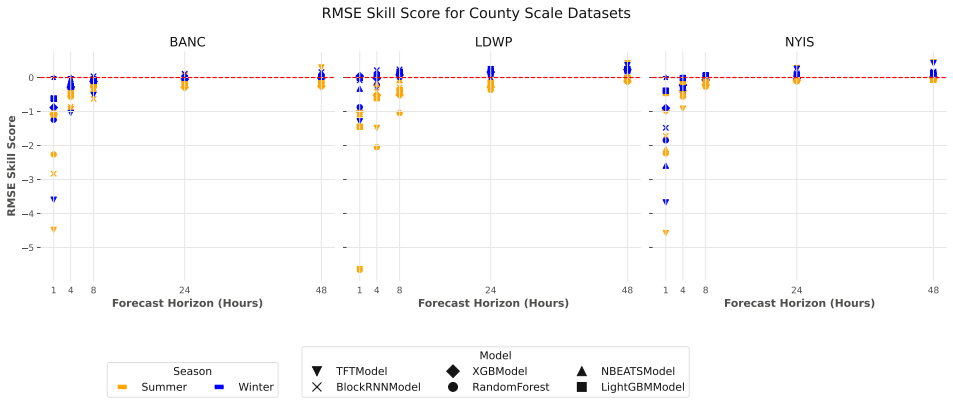


Figure 4.6.: RMSE Skill Scores for Individual Models on County Scale Datasets, Differentiated by Season (RQ1, taken from (Houben et al., 2025a)(Fig 8)).

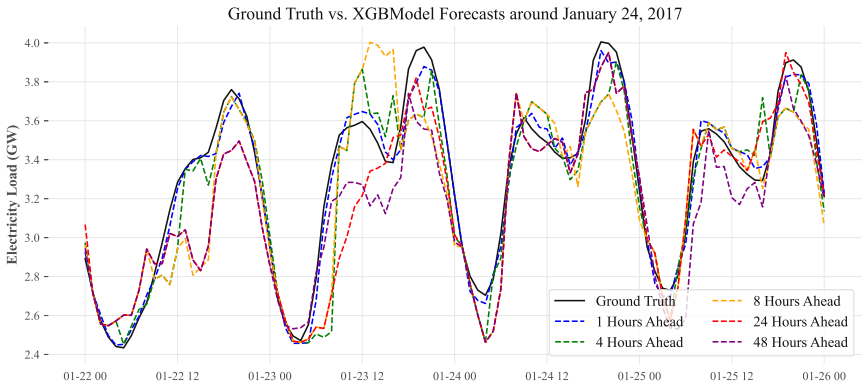


Figure 4.7.: Example XGBoost Forecasts (1h to 48h ahead) vs. Ground Truth on LDWP Dataset in Winter (RQ1, taken from (Houben et al., 2025a)(Fig 9)).

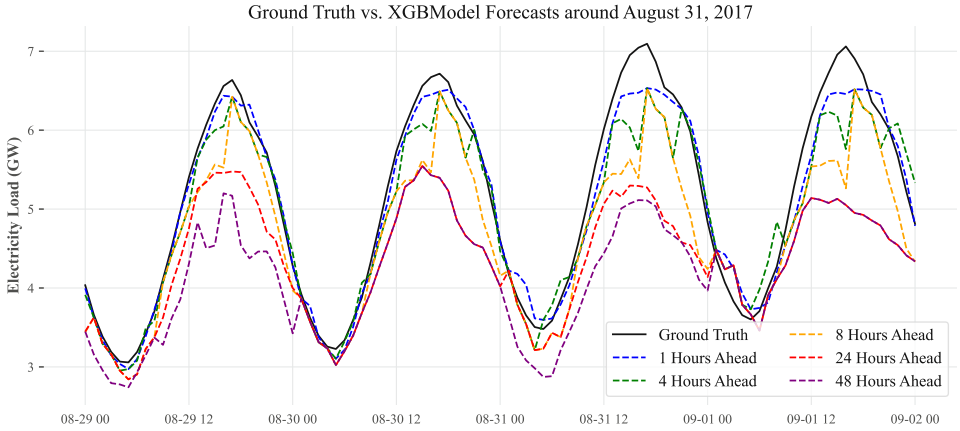


Figure 4.8.: Example XGBoost Forecasts (1h to 48h ahead) vs. Ground Truth on LDWP Dataset in Summer (RQ1, taken from (Houben et al., 2025a)(Fig 10)).

Analysis of error distributions ($\epsilon_t = y_t - \hat{y}_t$) confirmed these seasonal differences. Figure 4.9 shows error histograms for all models on the LDWP dataset (48h horizon). Summer errors tend to be right-skewed (positive tail, indicating under-prediction bias), while winter errors are generally less skewed. Table 4.2 summarizes skewness and kurtosis across scales. Summer forecasts consistently show higher positive skewness and higher kurtosis (heavier tails, more extreme errors) compared to winter for most scales and model types. Tree-based models generally exhibited less skew than NNs (Houben et al.,

2025a)(Sec 4.2).

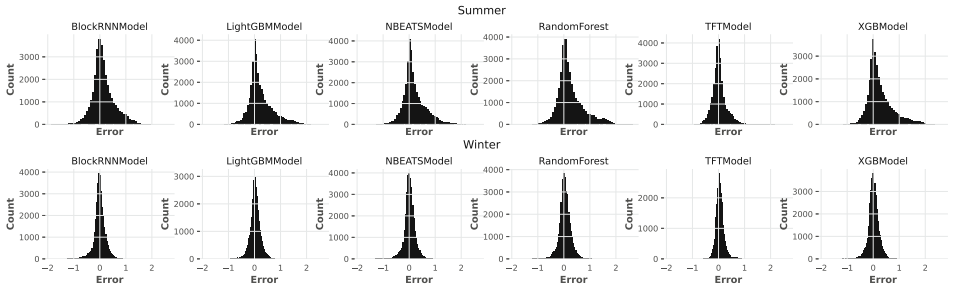


Figure 4.9.: Error Distributions ($\epsilon = y - \hat{y}$) for All Models on LDWP Dataset (H=48h), Comparing Summer vs. Winter Performance (RQ1, taken from (Houben et al., 2025a)(Fig 11)).

Table 4.2.: Comparison of Skewness and Kurtosis of Forecast Errors ($\epsilon = y - \hat{y}$) for Summer and Winter Test Periods, Averaged by Model Type and Spatial Scale (RQ1, taken from (Houben et al., 2025a)(Table 5).

Scale	Type	Skewness		Kurtosis	
		Summer	Winter	Summer	Winter
County	Neural Network	0.83	-0.06	9.36	7.07
	Tree-based	0.12	0.04	-9.67	9.85
Town	Neural Network	0.45	-0.43	3.87	2.66
	Tree-based	0.36	0.16	-7.31	8.70
Neighborhood	Neural Network	1.48	-0.07	16.74	5.55
	Tree-based	0.56	-0.38	-9.82	4.13
Building	Neural Network	-0.01	-0.09	1.35	4.25
	Tree-based	0.17	-0.22	1.71	3.22

Overall, the statistical analysis suggests tree-based models often provide lower average errors and more symmetric error distributions, while performance for all models degrades in summer compared to winter, likely due to different underlying load patterns.

4.1.2.3. Net Load Error (NLE) Results

Evaluating forecasts using the NLE score, which quantifies the economic impact of errors on peak load shaving performance (Section 3.2.1.1), provides a different perspective compared to purely statistical metrics (Houben et al., 2025a)(Sec 4.3). Figure 4.10 illustrates the process, showing how forecast errors (panel a) lead to different BESS dispatch (panel b) and ultimately result in higher operational peaks ($P^{grid,opr}$) compared to the ideal case using perfect forecasts (panel c). Under-predictions coinciding with actual peaks are particularly costly.

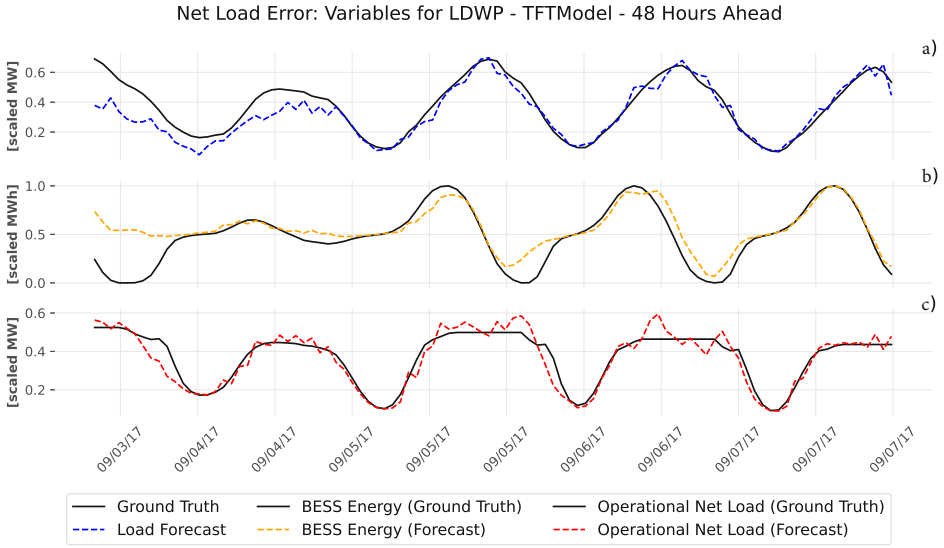


Figure 4.10.: Illustration of NLE Calculation Variables for TFT Model (H=48h) on LDWP Dataset: (a) Forecast vs. Ground Truth Load, (b) Resulting BESS Energy Trajectories, (c) Resulting Operational Net Grid Power (RQ1, taken from (Houben et al., 2025a)(Fig 12).

Comparing model classes based on the average NLE score (Figure 4.11) reveals the differences from the statistical evaluation:

- **NNs vs. Trees:** Neural Network models consistently achieved lower (better) NLE scores than both the benchmark and tree-based models on the county and often on the town scale datasets, particularly

for horizons $H \geq 8$ hours. This contrasts with the statistical results where trees often performed better. It suggests NNs, despite potentially higher average errors, may capture peak characteristics more effectively for demand charge reduction purposes.

- **Horizon Trend:** Unlike statistical errors which typically increase with horizon H, the NLE scores generally *decrease* (improve) as the forecast horizon increases. This indicates that having a longer foresight allows the eMPC controller within the NLE simulation to plan BESS operations more effectively for daily peak reduction, partially mitigating the negative impact of higher absolute forecast errors at longer lead times.

These findings highlight the importance of using application-driven metrics like NLE, as they can lead to different conclusions about model suitability compared to standard statistical measures alone, especially when the application focuses on specific events like peak load management (Houben et al., 2025a)(Sec 4.3).

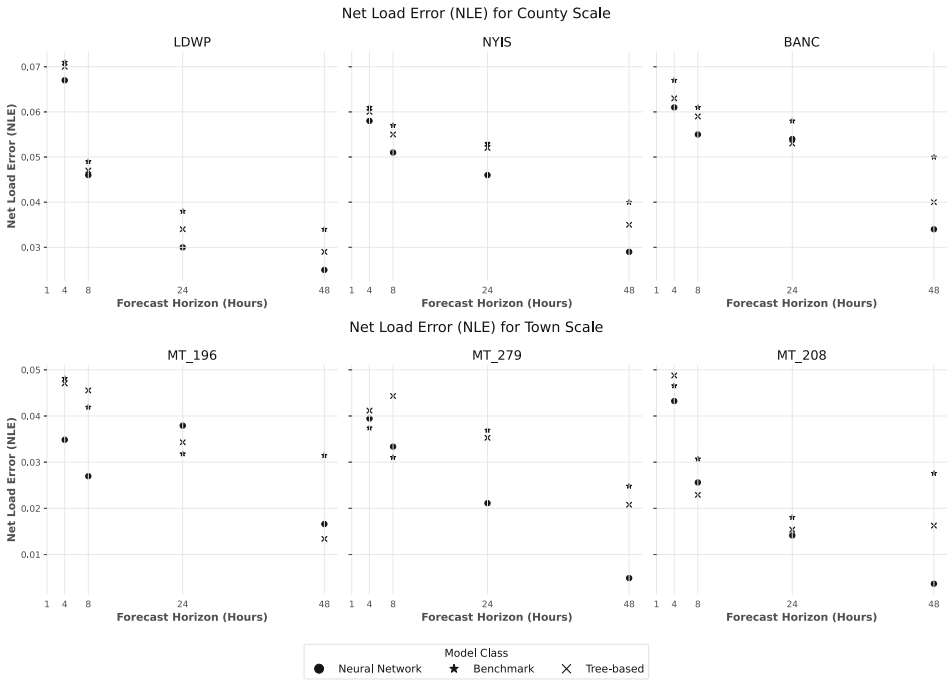


Figure 4.11.: Average Net Load Error (NLE) Scores for Model Classes (Tree-based, NN, Benchmark) on County and Town Level Datasets across Forecast Horizons (RQ1, taken from (Houben et al., 2025a)(Fig 13). Lower NLE is better.

4.2. RQ2: Interpretable Net Load Forecasting

This section addresses Research Question 2, focusing on improving household net load forecasting accuracy and interpretability by incorporating physical and operational knowledge into a deep learning model. It details the case study used for development and validation, followed by the numerical results demonstrating the effectiveness of the proposed model-based deep learning approach described in Section 3.2.2.

4.2.1. Case Study Description

The evaluation for RQ2 adopts the perspective of an energy retailer aiming to accurately forecast the 24-hour ahead net load ($H = 24\text{h}$, $\Delta t = 15\text{min}$) for its residential customers equipped with behind-the-meter energy systems (Houben et al., 2025b)(Sec IV-A].

4.2.1.1. System Context and Data Generation

The case study focuses on a cohort of 20 single-family households located in Utrecht, The Netherlands. Each household is assumed to be equipped with a rooftop Photovoltaic (PV) system, a Battery Energy Storage System (BESS), and a Home Energy Management System (HEMS). The households operate under a specified time-of-use (TOU) electricity purchase tariff (π_t^{buy}) and a flat feed-in tariff (π_t^{sell}) for exports (details specified in (Houben et al., 2025b)(Appendix A).

To provide a controlled environment for evaluating model interpretability, the ground truth net load time series (y_t^{net}) for these households were simulated rather than directly measured (Houben et al., 2025b)(Sec IV-B). This involved:

1. Using real-world, open-source data for the non-flexible appliance load (y_t^{load}) (Schlemminger et al., 2022) and PV generation (y_t^{prod}) (Visser et al., 2022) for the years 2014 & 2015. PV metadata (location, orientation, capacity) was also available from the source.

2. Simulating the optimal BESS operation for each household over the entire period using a HEMS optimization model. This model aimed to minimize energy costs under the given tariffs, similar to the one described in Section 3.2.2.1, but included BESS round-trip efficiencies

($\eta = 0.86$) for realism during simulation, using the state update:

$$E_{t+1} = E_t + \begin{cases} P_t^{BESS} \cdot \eta \cdot \Delta t & \text{if } P_t^{BESS} \geq 0 \quad \text{(charging)} \\ P_t^{BESS} \cdot \frac{1}{\eta} \cdot \Delta t & \text{if } P_t^{BESS} < 0 \quad \text{(discharging)} \end{cases} \tag{4.1}$$

3. Calculating the resulting net load $y_t^{net} = y_t^{load} - y_t^{prod} + P_t^{BESS}$.

This simulation process provided the ground truth y_t^{net} time series used for training and testing the forecasting models, while also yielding the underlying "hidden" states (non-flexible load y_t^{load} and optimal BESS state E_t) necessary for evaluating the interpretability of the proposed forecasting model (see Section 4.2.2.2). Figure 4.12 illustrates the components of the simulated data for a representative household.

4.2.1.2. Data Availability Scenarios

Recognizing that a retailer may not have perfect information about customer assets, the study evaluated the forecasting model under different data availability scenarios regarding the PV system specifications and the exact HEMS strategy employed by the household (Table 4.4) (Houben et al., 2025b)(Sec IV-C). The 'Unknown PV Specs' scenario involved inferring parameters like capacity, tilt, and azimuth via grid search. The 'Unknown Strategy' scenario involved training models assuming both a simple self-consumption HEMS and a TOU-aware HEMS, then selecting the better-performing one based on validation loss.

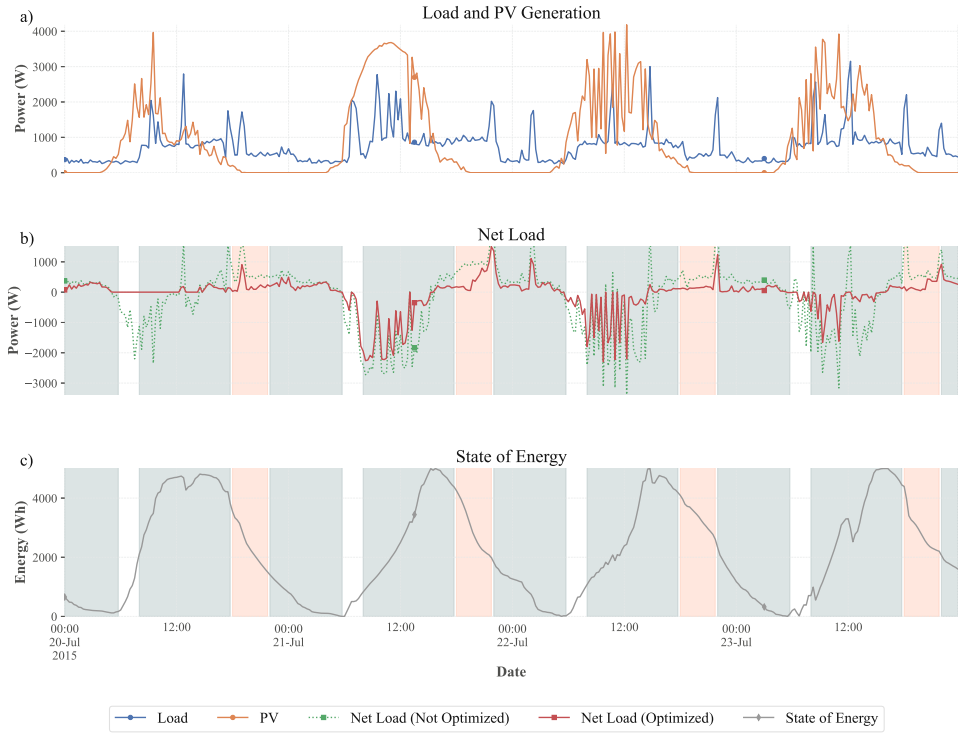


Figure 4.12.: Simulated data for a representative household (ID096/SFH10): (a) non-flexible load and PV generation, (b) resulting net load with and without optimal BESS operation, (c) optimal BESS state of energy. Background colors indicate TOU tariff periods (red: high price, grey: mid, white: low). (RQ2, adapted from (Houben et al., 2025b)(Fig 3).

Table 4.3.: 24-hour ahead net load forecasting performance comparison (MAE, RMSE, RMSE Skill Score vs. Naive) across models for summer and winter test periods (RQ2, adapted from (Houben et al., 2025b), Table II).

Season	Model	MAE (W)	RMSE (W)	SS-RMSE (%)
Summer	Naive	433	706	0.0
	DM1 (Direct TiDE)	343	506	28.3
	DM2 (Direct XGB)	422	597	15.4
	IM1 (Indirect XGB)	696	946	-33.9
	Proposed	**238**	**445**	**37.0**
Winter	Naive	480	778	0.0
	DM1 (Direct TiDE)	372	558	31.2
	DM2 (Direct XGB)	467	665	16.9
	IM1 (Indirect XGB)	760	1034	-37.2
	Proposed	**264**	**488**	**40.8**

Table 4.4.: Data Availability Scenarios considered for the Energy Retailer in RQ2 (adapted from (Houben et al., 2025b), Table I).

Scenario	BESS Specs	PV Specs	HEMS Strategy
Ideal: Full Specs & Known Strategy	✓	✓	Known
Known Specs, Unknown Strategy	✓	✓	Unknown / Inferred
Unknown PV Specs, Unknown Strategy	✓	Unknown / Inferred	Unknown / Inferred

4.2.1.3. Experimental Setup

The simulated net load data (2 years) was split chronologically into training (2014), validation (first half of 2015), and testing (second half of 2015) sets. Weather data for the PV model was sourced from the Open-Meteo API (*Free Open-Source Weather API | Open-Meteo.com* 2023). Model implementation used Python with the `Darts` library (Herzen et al., 2022), `cvxpylayers`, and `PyTorch`. Experiments were tracked using `Weights & Biases` (Biewald, 2020) and run on NVIDIA RTX 8000 GPUs (Houben et al., 2025b)(Sec IV-D).

4.2.2. Results

The performance of the proposed model-based deep learning forecaster (Section 3.2.2.2) was evaluated against several baseline methods across the different data availability scenarios, focusing on forecast accuracy, interpretability, and data efficiency (Houben et al., 2025b)(Sec V).

4.2.2.1. Forecasting Performance Evaluation

Comparison with Baseline Models The proposed model was compared against four baselines: Naive (persistence from previous day), DM1 (Direct TiDE model), DM2 (Direct XGBoost model), and IM1 (Indirect XGBoost predicting load and subtracting physical PV prediction). Table 4.3 summarizes the 24-hour ahead performance metrics (MAE, RMSE, RMSE Skill Score relative to Naive) averaged across all 20 households for the summer and winter test periods under the ideal data availability scenario.

The proposed model consistently achieved the lowest MAE and RMSE in both seasons, demonstrating superior accuracy. It yielded an RMSE improvement of approximately 37-41% over the naive baseline and a notable 9-10% improvement over the best performing standard deep learning baseline (DM1, which uses the same underlying TiDE architecture but without

the model-based components) (Houben et al., 2025b)(Sec V-A.1).

Impact of Data Availability Figure 4.13 shows the forecasting performance (RMSE and MAE) under the different data availability scenarios described in Section 4.2.1.2. The performance of the proposed model degrades slightly when the HEMS strategy is unknown but remains significantly better than the DM1 baseline (which uses no HEMS/PV/BESS priors). However, performance worsens considerably (RMSE increases by 40%) when PV specifications (particularly capacity) are also unknown, highlighting the importance of incorporating this physical information. The model's accuracy remained relatively robust even with imperfect knowledge, showcasing the benefit of the model-based structure (Houben et al., 2025b)(Sec V-A.2).

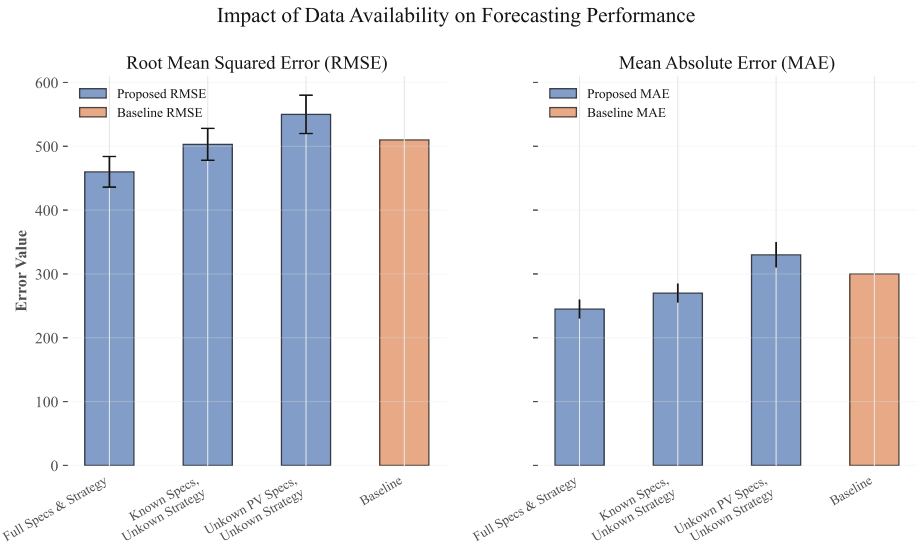


Figure 4.13.: Impact of data availability scenarios (Table 4.4) on forecasting performance (RMSE, MAE) compared to the DM1 baseline (RQ2, adapted from (Houben et al., 2025b)(Fig 4).

4.2.2.2. Model Interpretability and Data Efficiency

An advantage of the proposed model-based deep learning approach is its potential for enhanced interpretability and data efficiency compared to purely black-box methods.

Interpretability The model's architecture allows for the examination of intermediate, physically meaningful predictions (Houben et al., 2025b)(Sec V-B). As detailed in Section 3.2.2.2, the neural network component explicitly predicts the non-flexible load ($\hat{y}^{load}_{t+k|t}$) and the initial BESS state ($\hat{E}_t$) which serve as inputs to the final HEMS optimization layer. Figure 4.15 (left panel) compares these internal predictions against the ground truth values derived from the simulation dataset for a sample household. The model demonstrates a reasonable ability to learn the latent non-flexible load profile and track the BESS state of energy, providing valuable insights into the model's internal reasoning and the underlying system dynamics. Some discrepancies, particularly in the SoE prediction towards the end of the horizon, might relate to the formulation of the terminal state constraint and warrant further investigation.

Data Efficiency The study also investigated the model's performance when trained with limited data (Houben et al., 2025b)(Sec V-C). Figure 4.15 (right panel) plots the RMSE Skill Score (relative to the naive baseline) as a function of the percentage of training data utilized. The proposed model-based approach achieves positive skill scores (outperforming the baseline) even with very small data fractions (e.g., 10-20%). In contrast, the standard direct forecasting models (DM1 - TiDE, DM2 - XGBoost) require substantially more training data (e.g., 40-50% or more) to show similar improvements over the naive predictor. This superior data efficiency suggests that embedding physical and operational priors can reduce the reliance on large historical datasets, a significant advantage in practical applications where data may be scarce. It is noted, however, that effective training still requires the limited data samples to adequately represent seasonal variations present in the full dataset.

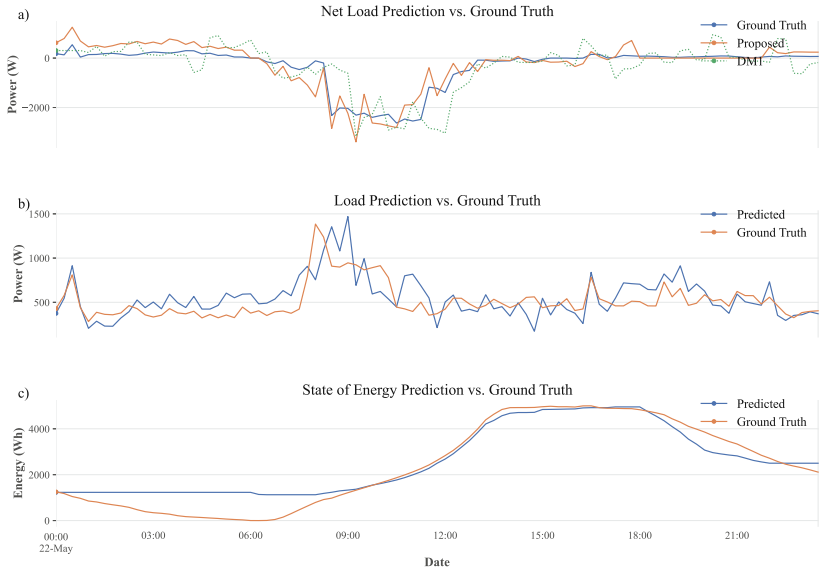


Figure 4.14.: Model Interpretability: Forecasts vs. Ground Truth for Net Load, Non-Flexible Load, and SoE (RQ2, adapted from (Houben et al., 2025b), Fig. 5).

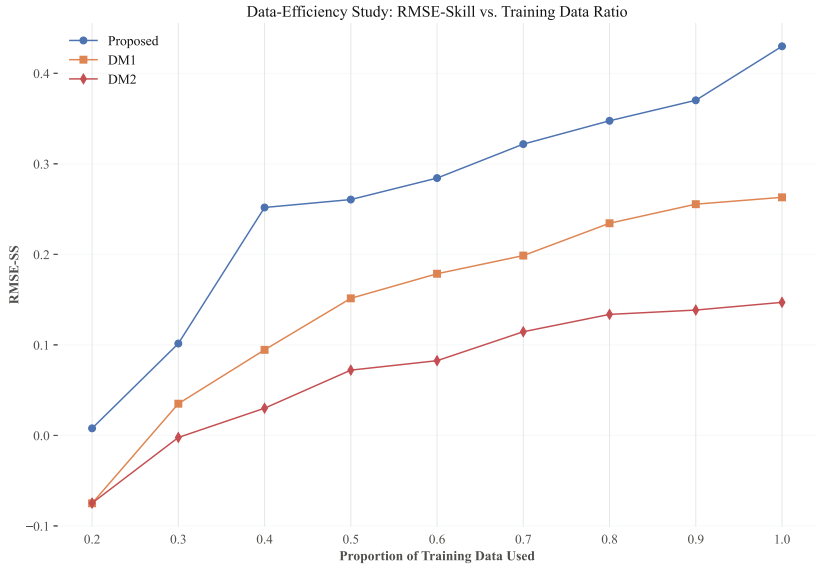


Figure 4.15.: Model Data Efficiency: RMSE Skill Score vs. Training Data Proportion (RQ2, adapted from (Houben et al., 2025b), Fig. 6).

4.3. RQ3: Optimal Dispatch of a Multi-Energy Microgrid under Uncertainty

This section addresses Research Question 3, which investigates the optimal dispatch of a multi-energy Renewable Energy Community (REC) using the eMPC framework developed in Section 3.2.3. It details the specific REC testbed used as a case study, the different operational scenarios evaluated, and presents the results regarding cost and CO_2 emission savings, particularly focusing on the impact of forecast uncertainty generated by the hybrid predictor (Section 3.2.3.2).

4.3.1. Case Study Description

The case study centers on a microgrid-enabled REC testbed located in a village in Carinthia, Austria, involving nine community participants and existing energy assets. This setup was previously presented in (Cosic et al., 2021) and provides a realistic context for evaluating the eMPC controller developed in this research (Houben et al., 2023), Sec 3].

4.3.1.1. Overview of the Energy Community

The REC testbed aims to optimally utilize locally generated renewable energy and manage flexible assets across both electrical and thermal sectors in near real-time (Houben et al., 2023), Sec 3.1]. The primary goal is to operate the available flexibilities according to the objective function defined in the eMPC (either cost or CO_2 minimization).

The components of the REC system are:

- **Generation:** An existing Photovoltaic (PV) system with a peak power of 17.68 kWp.

- **Loads:** Aggregated electrical demand and Domestic Hot Water (DHW) demand from the nine participants, treated as a single node.
- **Flexibilities:**
  - Battery Energy Storage System (BESS): 14 kWh capacity.
  - Power-to-Heat (PtH) system: 4 kW capacity.
  - Domestic Heat Storage (DHS): 27.5 kWh thermal capacity, coupled with the PtH.
- **Grid Connection:** Allows energy exchange with the main utility grid.

Based on forecasts for load (y^{load}) and PV generation (y^{PV}), and subject to tariff constraints, the eMPC controller determines the optimal dispatch for the BESS and the PtH/DHS system. The energy flows are illustrated in Section 3.2.3.1 in Figure 3.8. Detailed input parameters and time series data are provided in Appendix B.

4.3.1.2. Consideration of Tariff Scenarios

Five distinct electricity tariff scenarios were modelled to assess the eMPC controller's adaptability and performance under different economic conditions (Houben et al., 2023), Sec 3.2]. These scenarios directly influence the cost parameters ($\pi^{buy}_{t+k}, \pi^{sell}_{t+k}, \pi^{DC}$) in the cost minimization objective function (Section 3.2.3.1):

- **Flat Tariff (FT):** Constant purchase rate (29.84 €ct/kWh) and feed-in tariff (4 €ct/kWh), based on the local utility (KELAG) standard tariff (**KELAGtariff**).
- **Flat Tariff + Demand Charge (FT-DC):** FT scenario plus a monthly demand charge (16.78 €/kW) on the peak grid import.
- **Time-of-Use (TOU):** Variable purchase rates based on time of day

(On-peak: 35.8, Mid-peak: 29.84, Off-peak: 23.87 €ct/kWh), constant feed-in tariff.

- **Time-of-Use + Demand Charge (TOU-DC):** TOU scenario combined with the monthly demand charge.
- **Real-Time Pricing (RTP):** Flat purchase rate, but feed-in compensation based on hourly Austrian spot market prices (**SpotPrice**).

These scenarios are summarized in Table 4.5.

Table 4.5.: Considered tariff scenarios influencing the eMPC cost objective function, detailing utility purchase rates, sales rates, and demand charges (RQ3, adapted from (Houben et al., 2023), Table 2]).

Tariff Scenario	**Utility Purchase Rate (π^{buy}) [€ct/kWh]**	**Utility Sales Rate (π^{sell}) [€ct/kWh]**	**Demand Charge (π^{DC}) [€/kW/month]**
FT	29.84	4	0
FT-DC	29.84	4	16.78
TOU	35.8 (on), 29.84 (mid), 23.87 (off)	4	0
TOU-DC	35.8 (on), 29.84 (mid), 23.87 (off)	4	16.78
RTP	29.84	Hourly Market Prices	0

Table 4.6.: Optimization Cases (OCs) simulated, detailing the cost parameters used within the eMPC optimization and the corresponding objective function (RQ3, adapted from (Houben et al., 2023), Table 3]).

Case	**Cost Parameter Information Used in Optimization**	**Objective**
OC-FT	Flat Tariff ("FT")	Cost Min.
OC-FT-DC	Flat Tariff & Demand Charge ("FT-DC")	Cost Min.
OC-TOU	Time-Of-Use ("TOU")	Cost Min.
OC-TOU-DC	Time-Of-Use & Demand Charge ("TOU-DC")	Cost Min.
OC-RTP	Real-time Prices for Sales ("RTP")	Cost Min.
OC-CO_2	Marginal CO_2 Emission Factors	CO_2 Min.

4.3.1.3. Optimization Cases and Comparison Strategy

The performance of the eMPC controller (referred to as the Smart and Microgrid Controller, SMG, in Paper 3) was evaluated by simulating several *Optimization Cases (OCs)* over a one-month test period (May 2021) (Houben et al., 2023), Sec 3.3]. Each OC corresponds to running the eMPC with a specific objective function and awareness of one of the tariff structures (Table 4.6).

The results from these OCs (using realistic forecasts) were benchmarked against two references:

1. **Reference Case (RC):** Represents a standard rule-based "surplus charging" strategy, aiming for self-consumption maximization without considering tariff details or future conditions. It prioritizes using PV for load, then charging storage, then exporting surplus; grid import is used as a last resort. (Approximated in the study by a single optimization run with perfect foresight and a flat tariff).

2. **Perfect Foresight (PF):** Represents the theoretical optimum achievable by the eMPC if forecasts were perfect ($\hat{y} = y$). This was simulated by running the OCs using measured historical data instead of forecasts.

The comparison allows quantification of savings from the eMPC relative to basic control, and isolation of the performance loss due to forecast uncertainty (Houben et al., 2023), Sec 3.3, 3.5.1].

4.3.1.4. Operational Cost and CO_2 Calculation

It is important to note that the operational costs and CO_2 emissions are calculated ex-post based on the simulated grid interactions, considering the impact of forecast errors (Houben et al., 2023), Sec 3.4]. The actual utility exchange $P_t^{grid,opr}$ [kW] at each step t depends on the first step of the eMPC dispatch plan ($P_t^{grid,plan} = P_{k=0,t}^{grid,buy} - P_{k=0,t}^{grid,sell}$) and the net forecast error

e_t^{net} for that interval:

$$P_t^{grid,opr} = P_t^{grid,plan} + e_t^{net} \tag{4.2}$$

where the net forecast error combines load and PV errors (assuming PV curtailment is not modelled, and PV generation first serves local load):

$$e_t^{net} = e_t^{load} - e_t^{PV} \tag{4.3}$$
$$e_t^{load} = y_t^{load} - \hat{y}_{t|t-1}^{load} \tag{4.4}$$
$$e_t^{PV} = y_t^{PV} - \hat{y}_{t|t-1}^{PV} \tag{4.5}$$

This actual exchange $P_t^{grid,opr}$ is then used to calculate the realized costs based on the applicable tariff scenario rates $(\pi_t^{buy/sell}, \pi^{DC})$ according to the general structure in Section 3.1.2.1:

$$C_t^{rates} = \pi_t^{\text{buy}}[P_t^{grid,opr}]^+\Delta t - \pi_t^{\text{sell}}|[P_t^{grid,opr}]^-|\Delta t \tag{4.6}$$
$$C^{DC} = \pi^{DC} \cdot \max_{t\in\text{billing period}}\{[P_t^{grid,opr}]^+\} \tag{4.7}$$
$$C_{total} = \sum_{t\in\text{test period}} C_t^{rates} + C^{DC} \tag{4.8}$$

A similar calculation yields total CO_2 emissions using marginal emission factors instead of prices (see Appendix B).

4.3.1.5. Additional Analyses: Uncertainty Impact and Sensitivity

To further probe the system behavior, two additional analyses were conducted (Houben et al., 2023), Sec 3.5]:

1. **Foregone Savings due to Forecast Errors:** Calculated as the difference in operational savings (relative to RC) between the Perfect Foresight simulations and the simulations using real forecasts. This quantifies the economic penalty of imperfect predictions.

2. **Sensitivity to BESS Capacity:** The simulations (realistic forecast and perfect foresight) were repeated for larger BESS capacities (28, 42,

84 kWh, compared to the base 14 kWh) to understand the interplay between storage size, forecast accuracy, and achievable savings.

Figure 4.16 summarizes the comparison framework.

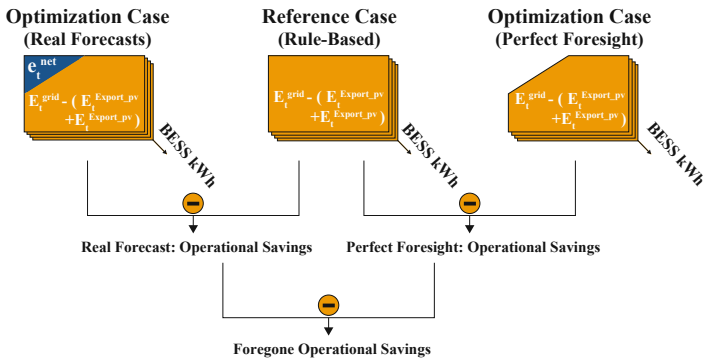


Figure 4.16.: Calculation Overview: Illustrates how operational savings (vs. RC) and foregone savings due to forecast errors are determined by comparing simulation results under different forecast assumptions (RQ3, adapted from (Houben et al., 2023), Fig 7]).

4.3.2. Results

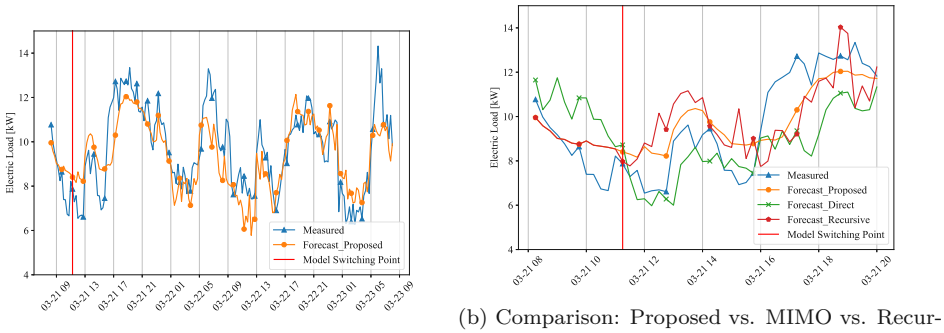
This section presents the results from applying the eMPC framework and the hybrid forecasting method to the REC case study. It covers the performance of the forecasting method itself, the optimal dispatch behavior under different scenarios, the resulting economic and environmental savings, and the impact of forecast uncertainty and BESS sizing (Houben et al., 2023), Sec 4].

4.3.2.1. Forecasting Method Performance

The performance of the hybrid forecasting method (Section 3.2.3.2) was evaluated during the model development phase using cross-validation on data

prior to the main test period (Houben et al., 2023), Sec 4.1].

Figure 4.17 visually compares the proposed method's 48-hour electrical load forecast against measurements and benchmarks. Panel (a) shows the full horizon, while panel (b) zooms in on the first 12 hours, comparing the proposed hybrid method against pure 'recursive' and 'MIMO' strategies using the same underlying XGBoost algorithm. The proposed method closely tracks measurements initially (leveraging Model 1's recursive approach with lags) and maintains better stability over longer horizons compared to the pure recursive method (benefitting from Model 2's one-shot approach with exogenous features). The MIMO method, while stable, struggles to capture the initial dynamics as effectively.



(a) Proposed hybrid forecast (48h)

(b) Comparison: Proposed vs. MIMO vs. Recursive (12h)

Figure 4.17.: XGBoost Electrical Load Forecast Comparison (21.03.2021, dev set): (a) Proposed hybrid method over 48 hours. (b) Close-up comparing Proposed, MIMO, and Recursive strategies over 12 hours (RQ3, adapted from (Houben et al., 2023), Fig 8]).

Table 4.7 quantitatively compares the cross-validated error scores (nRMSE, MAPE) and computation times. The proposed hybrid method achieves the best accuracy (lowest nRMSE and MAPE) with significantly lower training time than the MIMO method, although execution time is slightly higher than the pure recursive method due to the two-model structure.

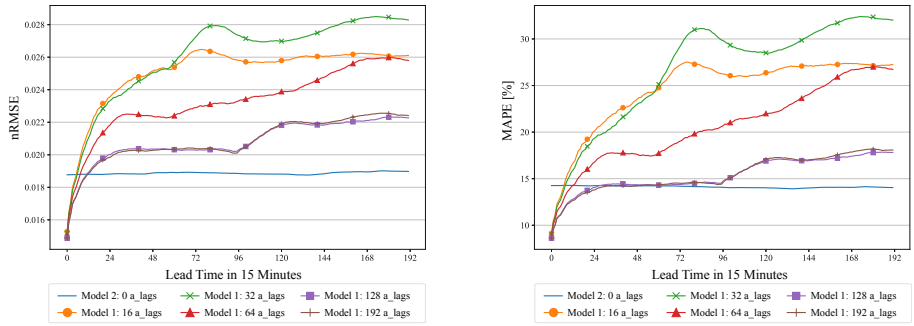
Table 4.7.: Comparison of cross-validated error scores and computational performance for different multi-step forecasting strategies using XGBoost for electrical load (H=48h) (RQ3, adapted from (Houben et al., 2023), Table B.1]).

Method	nRMSE	MAPE [%]	Training Time [s]	Execution Time [ms]
Recursive	2.30×10^{-2}	18.2	22.1	10.9
MIMO	1.95×10^{-2}	15.2	4531.0	40.8
Proposed Hybrid	$\mathbf{1.87 \times 10^{-2}}$	**13.8**	24.3	20.3

The determination of the optimal switching point s between Model 1 (recursive) and Model 2 (one-shot) is crucial. Figure 4.18 shows the error scores versus lead time for different configurations of Model 1 (varying number of autoregressive lags, 'lookback-window L'). The switching point is where the error of Model 2 (flat green line for load) intersects with the rising error of Model 1. Table 4.8 details the impact of 'lookback-window L' on performance and the resulting switching points. Using 128 lags for Model 1 was chosen as the optimal configuration, providing the best error score with acceptable computation time (Houben et al., 2023), Sec 4.1].

Table 4.8.: Comparison of Error Score, Computational Performance, and Resulting Switching Point for Different Numbers of Autoregressive Lags ('lookback-window L') in Model 1 for Electrical Load Forecasting (RQ3, adapted from (Houben et al., 2023), Table 4]). Bold indicates selected configuration.

L	Error Score		Computation Time		Switching Point (s)	
	nRMSE	MAPE [%]	Train [s]	Exec [ms]	nRMSE	MAPE
16	1.89×10^{-2}	14.1	0.1	10.9	6	7
32	1.88×10^{-2}	14.0	8.7	10.9	6	8
64	1.87×10^{-2}	13.9	12.6	14.1	8	12
128	$\mathbf{1.87 \times 10^{-2}}$	**13.8**	24.3	20.3	12	26
192	1.87×10^{-2}	13.8	36.4	21.9	13	27



(a) Electrical Load Forecasts – nRMSE (b) Electrical Load Forecasts – MAPE

Figure 4.18.: Lead-time vs. Error Score Plots: Intersection of Error Scores of Model 1 (varying lookback-window L) & Model 2 (flat green line) over the Forecast Horizon for Electrical Load, used to determine the switching point s (RQ3, adapted from (Houben et al., 2023), Fig 9]).

4.3.2.2. Optimal Operational Dispatch and Economic Analysis

The eMPC controller was simulated using the hybrid forecasts for the different tariff scenarios (OCs) and compared to the rule-based Reference Case (RC) and Perfect Foresight (PF) (Houben et al., 2023), Sec 4.2].

Figures 4.19 and 4.20 show example dispatch trajectories (storage SOC and grid exchange) for the OC-FT-DC and OC-CO2 cases, respectively, comparing the eMPC results (realistic forecast in yellow, perfect foresight in blue) with the RC (green). Under the demand charge scenario (Fig 4.19), the eMPC controller actively uses the BESS to shave peaks in grid import, attempting to keep imports below a threshold (visible as flat tops in the blue PF curve). Forecast errors (difference between yellow and blue grid exchange) disrupt this perfect shaving, leading to unintended peaks and higher costs. Under the CO_2 minimization scenario (Fig 4.20), the controller charges storage during low-emission periods (white/green background) and discharges during high-emission periods (grey background) to minimize the carbon footprint of grid imports.

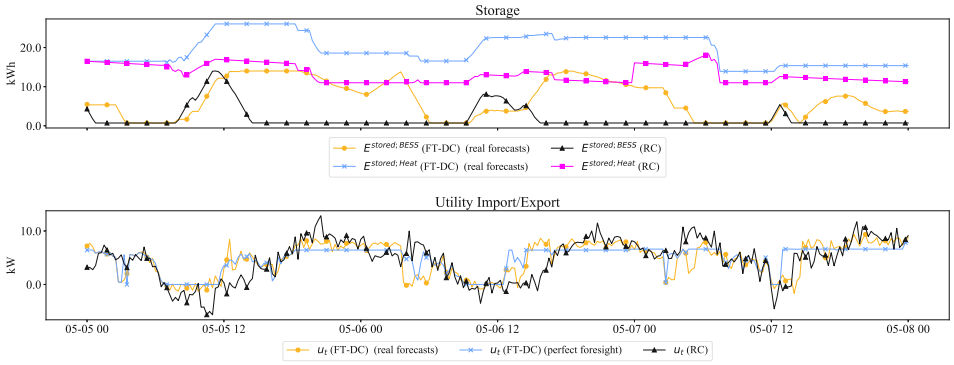


Figure 4.19.: Optimal Operational Dispatch for the Flat Tariff + Demand Charge Scenario (OC-FT-DC): Storage SOC (top) and Grid Exchange (bottom) comparing eMPC with realistic forecasts (yellow), perfect foresight (blue), and Reference Case (green) for May 5-8, 2021 (RQ3, adapted from (Houben et al., 2023), Fig 10]).

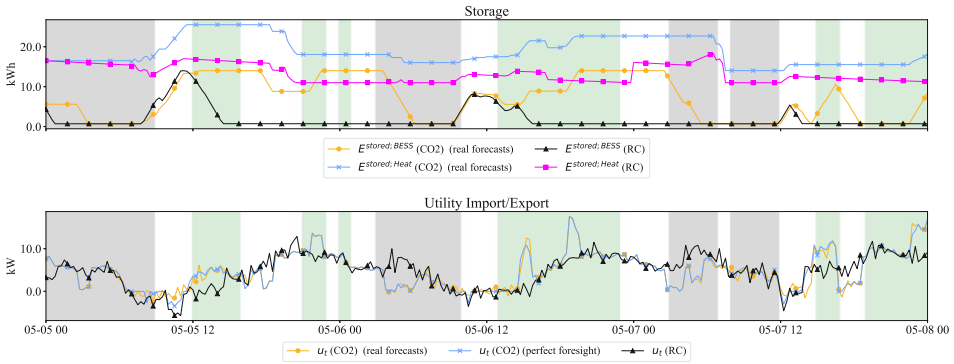


Figure 4.20.: Optimal Operational Dispatch for the CO_2 Minimization Scenario (OC-CO2): Storage SOC (top) and Grid Exchange (bottom) comparing eMPC with realistic forecasts (yellow), perfect foresight (blue), and Reference Case (green) for May 5-8, 2021. Background colors indicate marginal grid CO_2 intensity (RQ3, adapted from (Houben et al., 2023), Fig 11]).

Table 4.9 summarizes the total operational costs over the one-month test period for the eMPC controller (using realistic forecasts) compared to the RC across all tariff scenarios (for the 14 kWh BESS). The eMPC achieves the largest relative cost savings (6.3%) under the TOU scenario, where timing energy use is crucial. Savings are also positive under RTP (5.3

Table 4.9.: Operational Costs (C^{rates} from energy, C^{DC} from demand charge, C total) for Optimization Cases (OCs) with realistic forecasts vs. Reference Case (RC) under various tariff scenarios (Base Case: 14 kWh BESS) (RQ3, adapted from (Houben et al., 2023), Table 5]). u_t refers to total grid energy exchange [kWh].

	SMG Controller (eMPC with Forecasts)				Reference Case (RC)				Cost Savings (eMPC vs RC)	
Scenario	u_t [kWh]	C^{rates} [€]	C^{DC} [€]	C [€]	u_t [kWh]	C^{rates} [€]	C^{DC} [€]	C [€]	Abs. [€]	Rel. [%]
FT	7723.9	729.8	0	729.8	7762.7	733.5	0	733.5	3.7	0.5
FT-DC	9478.0	737.0	5069.1	5806.1	7762.7	733.5	5070.9	5804.4	-1.7	0.0
TOU	7417.1	687.3	0	687.3	7762.7	733.7	0	733.7	46.4	**6.3**
TOU-DC	7412.8	688.0	4923.4	5611.4	7762.7	733.7	5070.9	5804.6	193.2	3.3
RTP	7476.5	686.0	0	686.0	7762.7	724.0	0	724.0	38.8	5.3
CO_2[1]	8899.0	731.5	0	731.5	7762.7	733.5	0	733.5	2.0	0.3

Table 4.10.: Sensitivity of Relative Operational Savings (%) vs. Reference Case (RC) to BESS Capacity and Forecast Accuracy (Realistic Forecasts vs. Perfect Foresight) across different Optimization Cases (OCs) (RQ3, adapted from (Houben et al., 2023), Table 6]).

	Savings: Realistic Forecasts [%]				Savings: Perfect Foresight [%]			
Scenario	14 kWh	28 kWh	42 kWh	84 kWh	14 kWh	28 kWh	42 kWh	84 kWh
OC-FT	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
OC-FT-DC	0.0	7.1	3.3	10.6	24.7	38.1	36.6	35.7
OC-TOU	6.3	8.3	11.2	15.6	7.4	11.5	15.7	20.8
OC-TOU-DC	3.3	3.6	3.7	1.5	24.7	37.6	35.4	34.5
OC-RTP	5.2	5.8	6.4	7.7	4.7	6.0	7.3	8.9
OC-CO_2[2]	7.3	9.3	11.3	17.1	8.6	13.0	16.5	23.1

Table 4.11.: *CO_2 case minimizes emissions, costs shown for comparison.*; *Relative savings for OC-CO_2 refer to % reduction in kg-CO_2 emissions. All others are % reduction in €-costs.*

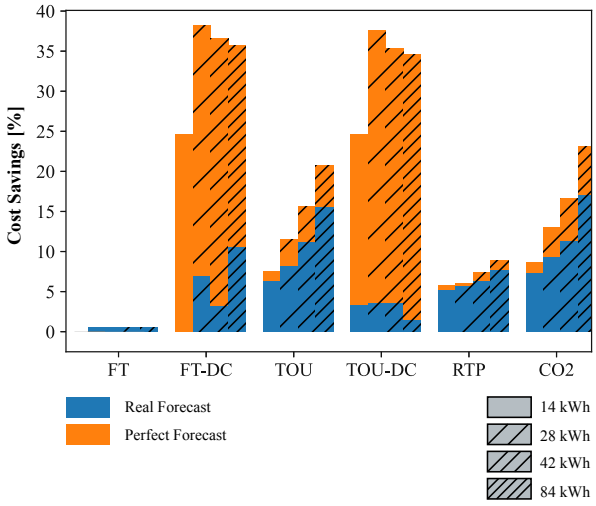
4.3.2.3. Impact of Uncertainty and BESS Capacity

The performance gap between realistic forecasts and perfect foresight highlights the significant economic impact of uncertainty (Houben et al., 2023), Sec 4.2.1]. Table 4.10 shows the relative operational savings achieved by the eMPC (compared to RC) under both realistic forecasts and perfect foresight, also exploring sensitivity to BESS capacity (14 kWh to 84 kWh).

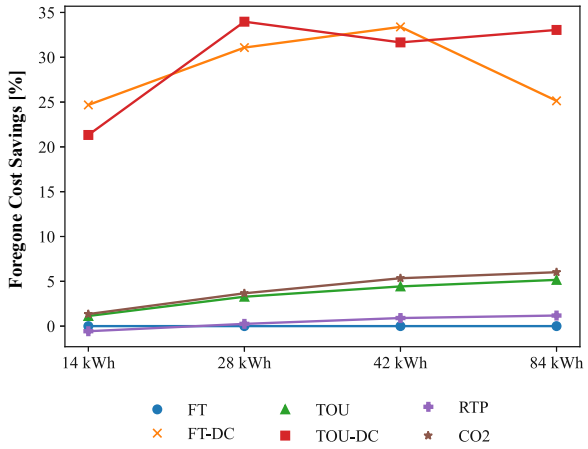
Figure 4.21 visually contrasts the savings achievable with realistic forecasts (blue bars) versus perfect foresight (orange bars) across BESS sizes. Key observations include:

- **Potential vs. Reality:** With perfect forecasts, significant savings (e.g., 25% for FT-DC/TOU-DC with 14kWh BESS, increasing further with size) are possible. However, realistic forecast errors drastically reduce these savings (Panel a).
- **Foregone Savings:** The difference between perfect foresight savings and realistic savings represents the cost of uncertainty ("Foregone Savings"). Panel (b) shows this gap generally increases with larger BESS capacity, particularly for demand charge scenarios (FT-DC, TOU-DC). This implies that while larger batteries offer more flexibility potential, realizing that potential becomes increasingly dependent on high forecast accuracy.
- **Sub-linear Scaling:** Savings increase sub-linearly with BESS capacity, even under perfect foresight, consistent with optimal planning findings (Cosic et al., 2021).

These results underscore that forecast accuracy is an important determinant of the economic viability of MPC microgrids, especially if they have flexibility and are exposed to complex tariffs, like demand charges (Houben et al., 2023), Sec 4.2.1].



(a) Operational savings (vs. RC)



(b) Foregone savings due to forecast errors

Figure 4.21.: Sensitivity Analysis of Operational Savings relative to Reference Case (RC): (a) Compares percentage savings achieved with realistic vs. perfect forecasts across BESS sizes. (b) Shows the foregone savings (difference between perfect and realistic) due to forecast errors. Relative savings for OC-CO_2 refer to % kg-CO_2 reduction. Others are % €-cost reduction. Adapted from (Houben et al., 2023), Fig 12).

5. Synthesis

This thesis covers the development and economic evaluation of supervised learning methods for short-term forecasting and the control of distributed energy resources (DERs). Through three interconnected research questions (RQs), explored in dedicated publications (Houben et al., 2025a; Houben et al., 2025b; Houben et al., 2023), this work navigates the terrain of forecasting electricity load across various scales, develops interpretable models for prosumer households with storage, and analyzes the real-world performance of advanced microgrid control under forecast uncertainty. Chapter 3 establishes the methodological groundwork, detailing the common frameworks of Supervised Learning (SL) and Economic Model Predictive Control (eMPC), while Chapter 4 presents the specific case studies and empirical findings for each research question.

This chapter synthesizes these individual parts by examining the evolution of the aforementioned forecasting techniques, the beautiful interplay between prediction accuracy and control performance, and the relevance of these findings for different stakeholders operating within the energy transition. Further, by integrating the insights obtained from benchmarking diverse algorithms (RQ1), developing model-based deep learning approaches (RQ2), and evaluating optimal dispatch under uncertainty (RQ3), this synthesis provides a comprehensive view on the challenges and opportunities in leveraging data-driven forecasting methods for the control of energy systems. Finally, this chapter places the specific contributions of the thesis into a larger context, drawing conclusions that span across the individual studies and offering consolidated recommendations.

5.1. Advancements in Forecasting: From Benchmarking to Application-Specific Models

A central theme connecting all three research questions is the role of supervised learning-based forecasting methods for energy applications. This thesis explores this from multiple angles: establishing a baseline performance landscape and the limits of traditional evaluation (RQ1), pioneering interpretable and data-efficient architectures for complex prosumer systems (RQ2), and implementing practical hybrid strategies for operational control systems (RQ3), together underscoring a progression towards more application-centric forecasting.

Establishing Baselines and Unveiling the Limits of Statistical Metrics (RQ1) RQ1 provided a starting point by systematically comparing standard tree-based ensembles (RF, XGBoost, LightGBM) and neural networks (GRU, N-BEATS, TFT) across diverse spatial scales (Houben et al., 2025a) (Section 4.1). The findings, based on traditional statistical metrics like RMSE and MAE, indicated that tree-based models often demonstrate superior average statistical accuracy and robustness, potentially due to inherent regularization and simpler structures compared to the benchmarked deep NNs, which occasionally showed signs of overfitting (see Table 4.2). This study also underscored the challenge of seasonality, particularly during summer months with non-normal load distributions, suggesting the limitations of standard global preprocessing and pointing towards the need for season-specific modeling approaches (see Section 4.1).

However, the pivotal contribution of RQ1 was its clear demonstration of the inadequacy of relying solely on statistical metrics. The introduction of the Net Load Error (NLE) metric, tailored for a peak-shaving application under demand charges, painted a different picture, in that Neural networks, despite sometimes higher average statistical errors, consistently outperformed tree-based models in NLE terms. This was the case, especially for longer horizons ($H \geq 8$ hours) on more aggregated datasets. Furthermore, this divergence suggests NNs may better capture the magnitude and timing of peak

events salient for specific economic objectives like demand charge reduction during extreme weather. This insight directly translates into the thesis's central argument: forecasting models must be evaluated through the lens of their intended application. The NLE serves as a compelling proof-of-concept for such application-driven evaluation, moving beyond the "double penalty" problem of metrics like RMSE (see Figure 2.2 and emphasizing that practitioners must select or develop error metrics congruent with their specific operational goals. This necessity for application-aware evaluation naturally led to exploring models designed with specific applications in mind.

Enhancing Interpretability and Data Efficiency with Model-Based Deep Learning (RQ2) Building on RQ1's call for application-centric approaches, the contribution to answer RQ2 went beyond benchmarking existing 'black-box' models to developing an application-tailored method for net load forecasting. The model architecture prioritizing interpretability and data efficiency for prosumer systems with PV, BESS, and HEMS logic (see Section 4.2). The core challenge identified was that standard net load forecasting often overlooks the BESS/HEMS internal operational strategy, treating its impact as mere noise.

The proposed solution in RQ2 is a model-based deep learning approach integrating physical priors (a PVWatts model) and operational logic (a simplified, differentiable HEMS optimization layer) into a modified TiDE neural network 3.2.2. This hybrid architecture yielded significant benefits: improved accuracy (around 10% RMSE reduction over baselines), enhanced interpretability via prediction of hidden operational states (like non-flexible load and BESS State of Energy), superior data efficiency, and reasonable robustness to missing system information (though sensitive to certain parameters like PV capacity). RQ2 thus showcased a pathway from evaluating models via their application (RQ1) to designing models *for* their application, demonstrating how customized architectures incorporating domain knowledge can bring about more accurate, trustworthy, and data-efficient forecasts for complex systems like the examined households. This work also addresses the need for explainable AI in energy systems (Section 2.3.4), with differentiable optimization layers (Agrawal et al., 2021) emerging as an effective technique.

Integrating and Evaluating Forecasting within an Operational eMPC Framework (RQ3) While RQ2 focused on embedding application logic within the forecast model for aimed to be used by energy retailers, RQ3 focused on integrating forecasts within an eMPC system for a multi-energy Renewable Energy Community. The primary aim was not merely to achieve the lowest statistical forecast error in isolation, but to provide reliable multi-step predictions effective for real-time eMPC. Furthermore, the goal was to evaluate the economic and operational impact of inherent forecast uncertainty in such an integrated system (see 4.3). This work responds to a gap where many studies evaluate MPC assuming perfect foresight or using overly simplified error models (Kanwar et al., 2015; Moser et al., 2020).

The novel hybrid forecasting method developed in RQ3 balanced accuracy over a 48-hour horizon with computational feasibility for the eMPC (Table 4.7). However, the more profound contribution was the subsequent analysis of this integrated forecasting-control system. The results starkly quantified the "cost of uncertainty": while the eMPC significantly outperformed a rule-based approach even with imperfect forecasts (*e.g.*, up to 6.3% cost savings under TOU tariffs), these savings were substantially eroded compared to a perfect foresight scenario (*e.g.*, potential 24.7% TOU-DC savings diminishing to 3.3% with realistic forecasts) (see Table 4.10). This gap, representing foregone savings due to forecast errors, increased with larger BESS capacities. This highlights another insight: while more flexibility offers greater theoretical benefits, its effective utilization is increasingly hostage to high forecast accuracy (see Figure 4.21b). This empirical assessment provides relevant evidence of the practical limitations faced by advanced control systems due to real-world forecast inaccuracies.

Evolving Perspectives on Energy Forecasting Across the RQs, this thesis shows a clear progression in its approach to energy forecasting. It begins by setting the scene and stating the fundamental need for application-aware evaluation (RQ1), then transitions to embedding domain-specific knowledge to create more proficient and transparent models for complex systems (RQ2). And finally it culminates in integrating a computationally efficient data-driven forecasting model into an eMPC to control a real system, in which

the economic consequences of forecast imperfections were rigorously measured (RQ3). The collective work underscores that selecting or developing a forecasting method is not a one-size-fits-all solution; it demands consideration of the application's specific goals, system complexity, data availability, and computational limits. The major takeaway is the transition from evaluating forecasts in isolation based on statistical metrics towards a paradigm of co-designing and co-evaluating forecasting methods within the context of the decision-making or control task, which they are intended to support.

5.2. Integrating Forecasting and Control: Performance, Uncertainty, and Value

The preceding discussion on forecasting advancements naturally leads to the central tenet of this thesis: energy forecasting is rarely an end in itself but derives its primary value from enabling more effective decision-making and control. The three research questions explore different sides of this tight relationship between the SL predictor and the control logic, providing insights into how forecast quality shapes operational performance.

The Indispensable Link: Forecasts as Direct Inputs to Optimization All three research questions explicitly show the role of forecasting outputs as direct inputs to optimization frameworks, albeit through varied integration mechanisms. In RQ1, forecasts from diverse SL models were essential inputs for the eMPC simulation within the NLE framework, which served as an application-based evaluation tool (see Section 3.2.1.1). RQ3 applied hybrid SL forecasts directly feeding a rolling horizon MILP-based eMPC controller for the REC (see Section 3.2.3.1). RQ2 demonstrated the tightest integration, embedding the HEMS optimization logic as a differentiable layer within the forecasting model itself (see Section 3.2.2.1). These integrations highlight the diverse ways that forecasting can inform automated decision-making in modern energy systems.

Quantifying the Economic Impact of Forecast Uncertainty: The "Cost of Uncertainty" An important finding of this thesis, predominantly emerging from RQ3, is the substantial disparity between the theoretical potential of optimized control (assuming perfect foresight) and the actual performance achievable under realistic forecast uncertainty. While the eMPC controller in RQ3 delivered significant improvements over rule-based strategies even with imperfect forecasts (*e.g.*, up to 6.3% cost and 7.3% CO_2 savings), these gains were considerably lower than the 25% cost savings (for demand charge scenarios with a 14 kWh BESS, rising to over 35% with larger batteries) achievable with perfect knowledge (see Table 4.9, and Table 4.10).

This *cost of uncertainty*, quantified by the foregone savings (Figure 4.21b), illustrates that forecast errors are a noteworthy impediment to realizing the full added-value of flexible DERs under advanced control. This impact is highly sensitive to the tariff structure (demand charges being particularly vulnerable) and the degree of system flexibility (larger BESS magnifying both potential gains and the detrimental effects of uncertainty). This finding underscores a trade-off: investments in greater flexibility (*e.g.*, larger storage) must be paralleled by advancements in forecasting and control systems capable of effectively harnessing that flexibility.

Towards a Holistic View of Forecasting and Control Integration The collective findings from this thesis paint a picture of a complex, symbiotic relationship between forecasting and control. Advanced control strategies like eMPC hold immense promise for optimizing DER management, but the realization of this potential is inextricably tied to the quality and, importantly, the *relevant characteristics* of the input forecasts. Standard statistical metrics prove insufficient for assessing a forecast's suitability for a given control task; application-driven metrics (like the NLE) or direct evaluation within the intended control framework (as in RQ3's eMPC analysis) are indispensable (see Section 4.1, and 4.3).

Forecast errors inevitably degrade control performance, with the severity of this degradation being highly contingent on the specific economic objective (*e.g.*, sensitivity to demand charges) and the available system flexibility.

While advanced forecasting techniques, whether model-based (RQ2) or tailored hybrid strategies (RQ3), can enhance prediction quality, a degree of residual uncertainty is an inescapable reality. This points towards the need for future research, as discussed in Section 5.4, to focus on developing control strategies inherently robust to forecast errors (*e.g.*, stochastic or robust MPC) and, in parallel, forecasting methods that are explicitly optimized for specific economic outcomes or control objectives rather than solely for statistical accuracy. Ultimately, the integration of forecasting and control cannot be approached as two unrelated problems; their co-design and co-evaluation are vital for unlocking the full potential of DERs.

5.3. Implications for Stakeholders

The research across the three RQs offers valuable insights not only for the academic community but also for practitioners.

Grid Operators For grid operators, the findings underscore the need to move beyond conventional statistical metrics when evaluating forecasting tools for specific operational tasks like peak shaving. As shown in RQ1, metrics like the NLE can reveal superior performance from models that might seem less accurate by traditional standards (see Section 4.5). Operators should invest in developing or adopting such *application-specific evaluation frameworks*. Furthermore, careful attention to *seasonal variations* in forecast accuracy and potential model biases (RQ1) is pivotal for reliable operations. Especially when planning to leverage flexibility from DERs, operators should account for the significant impact of *forecast uncertainty* on achievable benefits (RQ3). This might involve developing or adopting control strategies that are more robust to prediction errors.

Energy Retailers Energy retailers dealing with prosumers—households with PV, BESS, and HEMS—require sophisticated forecasting models that understand the underlying operational logic, as highlighted by the motivation for

RQ2. *Model-based deep learning* (RQ2) presents a promising avenue, offering not only improved accuracy but also enhanced interpretability, which can aid in understanding and segmenting customer behavior and predicting their net demand more effectively (see Section). However, the performance of such models depends significantly on access to *customer asset data* (*e.g.*, PV capacity), emphasizing the need for data-sharing mechanisms or robust parameter inference techniques. Furthermore, as retailers manage portfolios under increasingly dynamic tariffs, the challenge presented by forecast uncertainty (RQ1, RQ3) become paramount for risk management and service innovation.

End-Users, RECs, and Aggregators For end-users, including those participating in Renewable Energy Communities or aggregation schemes, optimized control offers cost and CO_2 savings (RQ3). However as mentioned above, these benefits are highly sensitive to forecast accuracy, leading to a gap between theoretical potential and real-world outcomes. This highlights a trade-off: *investing in greater flexibility* (*e.g.*, larger BESS) only yields maximal returns if coupled with high-quality forecasting and control systems capable of effectively utilizing that flexibility (RQ3)(see Figure(4.21). For many end-users, participating in Energy Communities that provide professional forecast and control management might be essential for maximizing the value derived from their DERs.

Broader Energy Transition Context This research reinforces the pivotal role of advanced forecasting and control as pivotal enablers for the large-scale integration of DERs. It calls for a paradigm shift in evaluation of forecasts, moving beyond statistical accuracy towards application-aware and economically meaningful assessment of forecasting tools. Successfully managing the inherent uncertainty of future energy systems calls for sophisticated tools that effectively integrate data-driven approaches with domain knowledge. Ultimately, fostering a tighter synergy between forecasting methodologies and control applications is crucial for unlocking the full economic and environmental value of DER flexibility, thereby empowering new actors like Virtual Power Plants or Energy Communities to contribute effectively to a

more decentralized, resilient, and sustainable energy system.

5.4. Limitations and Future Research Directions

While this thesis provides valuable insights into the evaluation, development, and application of supervised learning for forecasting and controlling DERs, it also features some major limitations, which are now highlighted. Given these limitations, new avenues for future research arise.

5.4.1. Limitations of the Presented Research

- **Scope of Case Studies and Data:** The geographic scope of the studies (USA, Netherlands, Austria) is specific, and the number of sites/households in RQ2 and RQ3, while providing valuable depth, restricts the statistical power for drawing universally generalizable conclusions. Further validation across more diverse datasets and geographic regions is warranted to confirm the broader applicability of the findings (Houben et al., 2025a; Houben et al., 2025b; Houben et al., 2023).

- **Forecast Input Assumptions:** The use of measured weather data as a proxy for perfect weather forecasts (RQ1, RQ3), and the assumption of perfect foresight for electricity prices and CO_2 intensity factors (RQ3), represent simplifications of real-world operational conditions. This likely leads to an overestimation of the achievable performance of forecasting and control systems when faced with true multi-faceted operational uncertainty (Houben et al., 2025a) (Section 5.3), (Houben et al., 2023) (Section 1.2, 4.2.1).

- **Model and Parameter Assumptions:** The accuracy of model-based approaches (like in RQ2) inherently relies on the fidelity of the underlying physical models (*e.g.*, PV model) and the simplified representations of complex components like HEMS/BESS. Parameter identification, especially for nuanced BESS characteristics (*e.g.*, degradation, precise

efficiencies under varying conditions), remains a significant challenge and was largely assumed to be known or handled simplistically in RQ2 (Houben et al., 2025b) (Section VI).

- **NLE Metric Specificity:** The NLE metric, while a valuable proof-of-concept for application-driven evaluation, is specifically tailored to a peak-shaving application with demand charges. Its direct applicability to other grid services (*e.g.*, frequency response, imbalance reduction) is limited due to its design (asymmetry, decoupling from broader network/market dynamics) (Houben et al., 2025a) (Section 5.2.1).

- **Computational and Tuning Constraints:** The performance of some machine learning algorithms, particularly deep neural networks in RQ1, might have been constrained by fixed hyperparameter tuning budgets. More extensive tuning could potentially unlock further performance gains (Houben et al., 2025a) (Section 5.1, 5.3). Similarly, model simplifications, such as the lossless BESS in the differentiable HEMS layer in RQ2, were sometimes necessary for computational tractability or to ensure differentiability (Houben et al., 2025b).

- **Simulation Environment fidelity:** The primary use of open-loop simulations for eMPC evaluation (RQ1 for NLE) neglects real-world complexities such as latencies, actuator imprecision, and unmodelled system dynamics, which could further degrade performance in live deployments.

5.4.2. Future Research Directions

Building upon the limitations and findings of this thesis, a few avenues for future research are worth noting:

- **Advancing Application-Driven Evaluation and Economically-Motivated Forecasting:** Develop and validate a broader suite of sophisticated application-driven forecasting metrics beyond NLE. These should be tailored to grid operations (*e.g.*, imbalance cost minimization,

congestion management) and should ideally incorporate more detailed market and network contextual information (Houben et al., 2025a) (Section 5.2.1, Section 6). A parallel and relevant direction is the development of forecasting models whose loss functions are directly derived from the economic objectives of the target application, thus inherently learning to prioritize forecast characteristics that maximize operational value.

- **Improving Forecast Robustness:** Enhance model resilience to challenges like seasonality, non-stationarity, and diverse system configurations. Techniques such as transfer learning and, also, comprehensive probabilistic forecasting is a promising avenue. This is because quantifying forecast uncertainty reliably is a prerequisite for advanced stochastic control methods and for providing operators with risk-aware decision frameworks (Houben et al., 2025a) (Section 6).

- **Enhancing Model-Based and Interpretable Forecasting for Complex Systems:** Extend the model-based deep learning approach demonstrated in RQ2 to a wider array of complex energy systems, including those with electric vehicles, heat pumps, and multi-vector energy interactions. A significant sub-challenge here is advancing online parameter identification techniques (*e.g.*, for BESS degradation, thermal building parameters) to ensure that embedded physical models remain accurate over time and across different installations. Validating these richer models on larger, more diverse datasets could be be essential (Houben et al., 2025b) (Section VI).

- **Deepening the Integration of Forecasting and Control under Uncertainty:** Bridge the gap highlighted by the "cost of uncertainty" by developing advanced uncertainty-aware control strategies. This includes practical implementations of stochastic MPC, robust MPC, and reinforcement learning approaches that can explicitly account for probabilistic forecasts or forecast error bounds. Furthermore, exploring hierarchical or distributed control architectures could offer ways to ameliorate the impact of forecast errors (Houben et al., 2023) (Section 5).

- **Expanding System Scope and Real-World Validation:** Broaden the geographic and systemic scope of case studies to enhance generalizability. Future work should explicitly forecast and integrate uncertainties from other inputs like dynamic electricity prices, CO_2 intensity factors, and weather forecasts (rather than using perfect foresight or measured data). Analyzing system performance within specific emerging market frameworks (*e.g.*, peer-to-peer trading, local flexibility markets, Virtual Power Plants) and, most importantly, validating findings through real-world pilots and closed-loop experiments will be important for demonstrating real impact (Houben et al., 2023) (Section 5).

Addressing these research directions will be central for developing the next generation of energy forecasting and control systems. Such systems are indispensable for managing the increasingly complex power grids of the future.

6. Conclusions and Outlook

This thesis investigated the economic evaluation and application of supervised learning methods for short-term forecasting and the control of distributed energy resources (DERs). By addressing the three interrelated research questions focused on benchmarking forecasting methods (RQ1), developing interpretable models for prosumer net load (RQ2), and evaluating advanced microgrid control under uncertainty (RQ3), this work provides a multi-faceted perspective on leveraging data-driven techniques in the energy transition.

The research consistently highlights the necessity of moving beyond developing and evaluating forecasts in isolation based purely on statistical metrics. RQ1 demonstrated, through the novel Net Load Error (NLE) metric, that standard accuracy measures fail to capture the economic value of forecasts for specific applications like peak shaving, where neural networks could outperform statistically superior tree-based models. RQ2 underscored the benefits of incorporating domain knowledge, showing that a model-based deep learning approach integrating physical PV and operational HEMS priors improves net load forecast accuracy, interpretability, and data efficiency for complex prosumer systems. Finally, RQ3 quantified the substantial impact of forecast uncertainty on the performance of economic Model Predictive Control (eMPC) for a multi-energy microgrid. While eMPC significantly outperforms basic control strategies, the gap between achievable savings under realistic forecasts and the theoretical potential under perfect foresight is large, particularly when managing significant flexibility under complex tariffs like demand charges. This "cost of uncertainty" emphasizes that forecast quality is a limiting factor in realizing the full benefits of DERs and advanced control.

The main contributions of this thesis include the comprehensive multi-scale

benchmarking study, the introduction of the NLE metric for application-driven forecast evaluation, the development of the interpretable model-based deep learning architecture for net load, and the integrated analysis quantifying the interplay between forecasting accuracy and eMPC performance in a real-world REC context. Limitations of this work include the geographic scope of the case studies, simplifying assumptions regarding forecast inputs (weather, prices), underlying system model parameters, the specificity of the NLE metric, and the use of open-loop simulations.

Future research should focus on developing more sophisticated application-driven evaluation metrics for diverse grid services, enhancing forecast robustness to seasonality and non-stationarity (*e.g.*, via probabilistic methods or online adaptation), extending model-based approaches to include more DER types (like EVs) and better parameter identification, and developing uncertainty-aware control strategies (*e.g.*, stochastic/robust MPC) and forecasting methods optimized directly for economic outcomes rather than statistical accuracy. Expanding the system scope to include market interactions and broader geographic contexts is also vital.

Ultimately, this research underscores that forecasting and control are inextricably linked. Their co-design and evaluation within the specific application context are paramount for effectively integrating DERs, managing flexibility, and enabling a successful transition towards a more decentralized and sustainable energy future.

7. References

Books

Bishop, Christopher M (1995). *Neural networks for pattern recognition*. Oxford university press. ISBN: 0-19-853864-2 (cit. on p. 39).

Box, George E. P., Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung (May 2015). *Time Series Analysis: Forecasting and Control*. en. Google-Books-ID: rNt5CgAAQBAJ. John Wiley & Sons. ISBN: 978-1-118-67492-5 (cit. on p. 15).

Busoniu, Lucian, Robert Babuska, Bart De Schutter, and Damien Ernst (July 2017). *Reinforcement Learning and Dynamic Programming Using Function Approximators*. en. 0th ed. CRC Press. ISBN: 978-1-4398-2109-1. DOI: 10.1201/9781439821091. URL: https://www.taylorfrancis.com/books/9781439821091 (visited on 05/02/2025) (cit. on pp. 27, 29).

European Commission. Joint Research Centre. (2020). *Energy communities: an overview of energy and social innovation*. en. LU: Publications Office. URL: https://data.europa.eu/doi/10.2760/180576 (visited on 03/02/2025) (cit. on p. 7).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep learning*. MIT press. ISBN: 0-262-33737-1 (cit. on pp. 16, 42).

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). "Overview of Supervised Learning". en. In: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Ed. by Trevor Hastie, Robert Tibshirani, and Jerome Friedman. New York, NY: Springer, pp. 9–41. ISBN: 978-0-387-84858-7. DOI: 10.1007/978-0-387-84858-7_2. URL: https://doi.org/10.1007/978-0-387-84858-7_2 (visited on 11/11/2024) (cit. on p. 41).

Hong, Tao (2010). *Short term electric load forecasting*. North Carolina State University. ISBN: 1-124-47772-1 (cit. on p. 21).

Pérez-Arriaga, Ignacio and Christopher Knittle (2016). *Utility of the future: An mit energy iniative response to an industry in transition*. MIT Energy Initiative (cit. on p. 5).

Powell, Warren B. (2021). "From Reinforcement Learning to Optimal Control: A Unified Framework for Sequential Decisions". en. In: *Handbook of Reinforcement Learning and Control*. Ed. by Kyriakos G. Vamvoudakis, Yan Wan, Frank L. Lewis, and Derya Cansever. Cham: Springer International Publishing, pp. 29–74. ISBN: 978-3-030-60990-0. DOI: 10.1007/978-3-030-60990-0_3. URL: https://doi.org/10.1007/978-3-030-60990-0_3 (visited on 04/30/2025) (cit. on pp. 27, 28, 31).

Seneviratne, S. I. et al. (2021). "Weather and climate extreme events in a changing climate". en. In: ed. by V. P. Masson-Delmotte, A. Zhai, S. L. Pirani, and C. Connors. Cambridge, UK: Cambridge University Press, pp. 1513–1766. ISBN: 978-1-00-915789-6. URL: https://centaur.reading.ac.uk/101846/ (visited on 05/05/2025) (cit. on p. 5).

Sutton, Richard S. and Andrew Barto (2020). *Reinforcement learning: an introduction*. en. Second edition. Adaptive computation and machine learning. Cambridge, Massachusetts London, England: The MIT Press. ISBN: 978-0-262-03924-6 (cit. on pp. 27–29).

Taieb, Souhaib Ben and Rob J Hyndman (2012). *Recursive and direct multi-step forecasting: the best of both worlds*. Vol. 19. Department of Econometrics and Business Statistics, Monash Univ. (cit. on pp. 18, 39).

Journal Articles

Agrawal, Akshay, Shane Barratt, and Stephen Boyd (Aug. 2021). "Learning Convex Optimization Models". In: *IEEE/CAA Journal of Automatica Sinica* 8.8. Conference Name: IEEE/CAA Journal of Automatica Sinica, pp. 1355–1364. ISSN: 2329-9274. DOI: 10.1109/JAS.2021.1004075. URL: https://ieeexplore.ieee.org/abstract/document/9459585 (visited on 07/09/2024) (cit. on pp. 71, 109).

Alipour, Mohammadali, Jamshid Aghaei, Mohammadali Norouzi, Taher Niknam, Sattar Hashemi, and Matti Lehtonen (Aug. 2020). "A novel

electrical net-load forecasting model based on deep neural networks and wavelet transform integration". en. In: *Energy* 205, p. 118106. ISSN: 0360-5442. DOI: 10.1016/j.energy.2020.118106. URL: https://www.sciencedirect.com/science/article/pii/S0360544220312135 (visited on 01/09/2023) (cit. on p. 22).

Angeli, David, Rishi Amrit, and James B. Rawlings (July 2012). "On Average Performance and Stability of Economic Model Predictive Control". In: *IEEE Transactions on Automatic Control* 57.7, pp. 1615–1626. ISSN: 1558-2523. DOI: 10.1109/TAC.2011.2179349. URL: https://ieeexplore.ieee.org/document/6099558 (visited on 04/10/2025) (cit. on p. 51).

Angelov, Plamen P., Eduardo A. Soares, Richard Jiang, Nicholas I. Arnold, and Peter M. Atkinson (2021). "Explainable artificial intelligence: an analytical review". en. In: *WIREs Data Mining and Knowledge Discovery* 11.5. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/widm.1424, e1424. ISSN: 1942-4795. DOI: 10.1002/widm.1424. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/widm.1424 (visited on 05/01/2025) (cit. on p. 24).

Arroyo, Javier, Carlo Manna, Fred Spiessens, and Lieve Helsen (Mar. 2022). "Reinforced model predictive control (RL-MPC) for building energy management". In: *Applied Energy* 309, p. 118346. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2021.118346. URL: https://www.sciencedirect.com/science/article/pii/S0306261921015932 (visited on 05/01/2025) (cit. on pp. 30, 31).

Asensio, Miguel, Gregorio Muñoz-Delgado, and Javier Contreras (2017). "Bi-level approach to distribution network and renewable energy expansion planning considering demand response". In: *IEEE Transactions on Power Systems* 32.6. Publisher: IEEE, pp. 4298–4309. ISSN: 0885-8950 (cit. on p. 6).

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014). "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (cit. on p. 16).

Bartolucci, Lorenzo, Stefano Cordiner, Vincenzo Mulone, and Marina Santarelli (Apr. 2019). "Hybrid renewable energy systems: Influence of short term forecasting on model predictive control performance". In: *Energy* 172, pp. 997–1004. ISSN: 0360-5442. DOI: 10.1016/j.energy.2019.01.104.

URL: https://www.sciencedirect.com/science/article/pii/S0360544219301045 (visited on 05/01/2025) (cit. on p. 30).

Behm, Christian, Lars Nolting, and Aaron Praktiknjo (2020). "How to model European electricity load profiles using artificial neural networks". In: *Applied Energy* 277. Publisher: Elsevier, p. 115564. ISSN: 0306-2619 (cit. on p. 74).

Ben Taieb, Souhaib, Gianluca Bontempi, Amir F. Atiya, and Antti Sorjamaa (June 2012). "A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition". In: *Expert Systems with Applications* 39.8, pp. 7067–7083. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2012.01.039. URL: https://www.sciencedirect.com/science/article/pii/S0957417412000528 (cit. on pp. 17, 18, 39, 40).

Bishop, Christopher M (2006). "Pattern recognition and machine learning". In: *Springer google schola* 2, pp. 645–678 (cit. on p. 38).

Bishop, Christopher M. (Feb. 2013). "Model-based machine learning". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 371.1984. Publisher: Royal Society, p. 20120222. DOI: 10.1098/rsta.2012.0222. URL: https://royalsocietypublishing.org/doi/full/10.1098/rsta.2012.0222 (visited on 05/01/2025) (cit. on pp. 24, 25).

Blokhuis, Erik, Bart Brouwers, Eric van der Putten, and Wim Schaefer (Oct. 2011). "Peak loads and network investments in sustainable energy transitions". In: *Energy Policy*. Sustainability of biofuels 39.10, pp. 6220–6233. ISSN: 0301-4215. DOI: 10.1016/j.enpol.2011.07.021. URL: https://www.sciencedirect.com/science/article/pii/S0301421511005453 (visited on 01/26/2024) (cit. on p. 6).

Breiman, Leo (1996). "Bagging predictors". In: *Machine learning* 24. Publisher: Springer, pp. 123–140. ISSN: 0885-6125 (cit. on pp. 15, 145).

Breiman, Leo (2001). "Random forests". In: *Machine learning* 45. Publisher: Springer, pp. 5–32. ISSN: 0885-6125 (cit. on pp. 15, 59, 144).

Carvalho, Diogo V., Eduardo M. Pereira, and Jaime S. Cardoso (Aug. 2019). "Machine Learning Interpretability: A Survey on Methods and Metrics". en. In: *Electronics* 8.8. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute, p. 832. ISSN: 2079-9292. DOI: 10.3390/

electronics8080832. URL: https://www.mdpi.com/2079-9292/8/8/832 (visited on 05/01/2025) (cit. on pp. 23, 24, 33).

Chevillon, Guillaume (2007). "Direct Multi-Step Estimation and Forecasting". en. In: *Journal of Economic Surveys* 21.4. _eprint: https://onlinelibrary.wiley.com/doi/pdf/6419.2007.00518.x, pp. 746–785. ISSN: 1467-6419. DOI: 10.1111/j.1467-6419.2007.00518.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6419.2007.00518.x (visited on 04/29/2025) (cit. on p. 17).

Cosic, Armin, Michael Stadler, Muhammad Mansoor, and Michael Zellinger (Dec. 2021). "Mixed-integer linear programming based optimization strategies for renewable energy communities". In: *Energy* 237, p. 121559. ISSN: 0360-5442. DOI: 10.1016/j.energy.2021.121559. URL: https://www.sciencedirect.com/science/article/pii/S0360544221018077 (visited on 04/17/2025) (cit. on pp. 93, 105).

De Gooijer, Jan G. and Rob J. Hyndman (Jan. 2006). "25 years of time series forecasting". In: *International Journal of Forecasting*. Twenty five years of forecasting 22.3, pp. 443–473. ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2006.01.001. URL: https://www.sciencedirect.com/science/article/pii/S0169207006000021 (visited on 04/29/2025) (cit. on pp. 14, 15, 18).

Drgoňa, Ján, Aaron R. Tuor, Vikas Chandan, and Draguna L. Vrabie (July 2021). "Physics-constrained deep learning of multi-zone building thermal dynamics". In: *Energy and Buildings* 243, p. 110992. ISSN: 0378-7788. DOI: 10.1016/j.enbuild.2021.110992. URL: https://www.sciencedirect.com/science/article/pii/S0378778821002760 (visited on 07/19/2024) (cit. on p. 24).

Dwivedi, Rudresh, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, and Rajiv Ranjan (Jan. 2023). "Explainable AI (XAI): Core Ideas, Techniques, and Solutions". In: *ACM Comput. Surv.* 55.9, 194:1–194:33. ISSN: 0360-0300. DOI: 10.1145/3561048. URL: https://dl.acm.org/doi/10.1145/3561048 (visited on 05/01/2025) (cit. on pp. 23, 24, 33).

Elattar, Ehab E., Nehmdoh A. Sabiha, Mohammad Alsharef, Mohamed K. Metwaly, Amr M. Abd-Elhady, and Ibrahim B. M. Taha (Oct. 2020). "Short term electric load forecasting using hybrid algorithm for smart cities". en. In: *Applied Intelligence* 50.10, pp. 3379–3399. ISSN: 1573-7497.

DOI: 10.1007/s10489-020-01728-x. URL: https://doi.org/10.1007/s10489-020-01728-x (visited on 12/26/2023) (cit. on p. 22).

Elman, Jeffrey L (1990). "Finding structure in time". In: *Cognitive science* 14.2. Publisher: Wiley Online Library, pp. 179–211. ISSN: 0364-0213 (cit. on p. 148).

Friedman, Jerome H. (Oct. 2001). "Greedy function approximation: A gradient boosting machine." In: *The Annals of Statistics* 29.5. Publisher: Institute of Mathematical Statistics, pp. 1189–1232. ISSN: 0090-5364, 2168-8966. DOI: 10.1214/aos/1013203451. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.full (visited on 11/11/2024) (cit. on p. 15).

Fu, Yangyang, Shichao Xu, Qi Zhu, Zheng O'Neill, and Veronica Adetola (June 2023). "How good are learning-based control v.s. model-based control for load shifting? Investigations on a single zone building energy system". In: *Energy* 273, p. 127073. ISSN: 0360-5442. DOI: 10.1016/j.energy.2023.127073. URL: https://www.sciencedirect.com/science/article/pii/S036054422300467X (visited on 05/01/2025) (cit. on p. 32).

Gao, Yuan, Yuki Matsunami, Shohei Miyata, and Yasunori Akashi (Feb. 2023). "Model predictive control of a building renewable energy system based on a long short-term hybrid model". In: *Sustainable Cities and Society* 89, p. 104317. ISSN: 2210-6707. DOI: 10.1016/j.scs.2022.104317. URL: https://www.sciencedirect.com/science/article/pii/S2210670722006217 (visited on 05/02/2025) (cit. on p. 30).

Gardner Jr., Everette S. (1985). "Exponential smoothing: The state of the art". en. In: *Journal of Forecasting* 4.1. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002 pp. 1–28. ISSN: 1099-131X. DOI: 10.1002/for.3980040103. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/for.3980040103 (visited on 04/29/2025) (cit. on p. 14).

Gjorgievski, Vladimir Z, Snezana Cundeva, and George E Georghiou (2021). "Social arrangements, technical designs and impacts of energy communities: A review". In: *Renewable Energy* 169. Publisher: Elsevier, pp. 1138–1156. ISSN: 0960-1481 (cit. on p. 7).

Gokhale, Gargya, Bert Claessens, and Chris Develder (May 2022). "Physics informed neural networks for control oriented thermal modeling of build-

ings". In: *Applied Energy* 314, p. 118852. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2022.118852. URL: https://www.sciencedirect.com/science/article/pii/S0306261922002884 (visited on 07/19/2024) (cit. on pp. 24, 25).

Gu, Wei, Zhihe Wang, Zhi Wu, Zhao Luo, Yiyuan Tang, and Jun Wang (Sept. 2017). "An Online Optimal Dispatch Schedule for CCHP Microgrids Based on Model Predictive Control". In: *IEEE Transactions on Smart Grid* 8.5, pp. 2332–2342. ISSN: 1949-3061. DOI: 10.1109/TSG.2016.2523504. URL: https://ieeexplore.ieee.org/abstract/document/7406729 (visited on 05/01/2025) (cit. on p. 30).

Gust, Gunther, Tobias Brandt, Salman Mashayekh, Miguel Heleno, Nicholas DeForest, Michael Stadler, and Dirk Neumann (July 2021). "Strategies for microgrid operation under real-world conditions". In: *European Journal of Operational Research* 292.1, pp. 339–352. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2020.10.041. URL: https://www.sciencedirect.com/science/article/pii/S0377221720309188 (visited on 05/01/2025) (cit. on p. 30).

Haben, Stephen, Siddharth Arora, Georgios Giasemidis, Marcus Voss, and Danica Vukadinović Greetham (Dec. 2021). "Review of low voltage load forecasting: Methods, applications, and recommendations". In: *Applied Energy* 304, p. 117798. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2021.117798. URL: https://www.sciencedirect.com/science/article/pii/S0306261921011326 (cit. on pp. 26, 32).

Haben, Stephen, Georgios Giasemidis, Florian Ziel, and Siddharth Arora (Oct. 2019). "Short term load forecasting and the effect of temperature at the low voltage level". In: *International Journal of Forecasting* 35.4, pp. 1469–1484. ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2018.10.007. URL: https://www.sciencedirect.com/science/article/pii/S0169207018301870 (visited on 11/29/2023) (cit. on p. 21).

Herzen, Julien, Francesco Lässig, Samuele Giuliano Piazzetta, Thomas Neuer, Léo Tafti, Guillaume Raille, Tomas Van Pottelbergh, Marek Pasieka, Andrzej Skrodzki, and Nicolas Huguenin (2022). "Darts: User-friendly modern machine learning for time series". In: *The Journal of Machine Learning Research* 23.1. Publisher: JMLRORG, pp. 5442–5447. ISSN: 1532-4435 (cit. on pp. 71, 89).

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8. Publisher: MIT press, pp. 1735–1780. ISSN: 0899-7667 (cit. on pp. 16, 148, 154).

Hong, Tao (2014). "Energy forecasting: Past, present, and future". In: *Foresight: The International Journal of Applied Forecasting* 32. Publisher: International Institute of Forecasters, pp. 43–48 (cit. on pp. 18–20, 38).

Hong, Tao and Shu Fan (July 2016). "Probabilistic electric load forecasting: A tutorial review". en. In: *International Journal of Forecasting* 32.3, pp. 914–938. ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2015.11.011. URL: https://www.sciencedirect.com/science/article/pii/S0169207015001508 (visited on 01/19/2023) (cit. on p. 19).

Hong, Tao, Pierre Pinson, and Shu Fan (2014). "Global energy forecasting competition 2012". In: *International Journal of Forecasting* 30.2. Publisher: Elsevier, pp. 357–363. ISSN: 0169-2070 (cit. on p. 32).

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White (1989). "Multilayer feedforward networks are universal approximators". In: *Neural networks* 2.5. Publisher: Elsevier, pp. 359–366. ISSN: 0893-6080 (cit. on pp. 16, 153).

Hou, Qingchun, Ning Zhang, Ershun Du, Miao Miao, Fei Peng, and Chongqing Kang (2019). "Probabilistic duck curve in high PV penetration power system: Concept, modeling, and empirical analysis in China". In: *Applied Energy* 242. Publisher: Elsevier, pp. 205–215. ISSN: 0306-2619 (cit. on p. 6).

Houben, Nikolaus, Armin Cosic, Michael Stadler, Muhammad Mansoor, Michael Zellinger, Hans Auer, Amela Ajanovic, and Reinhard Haas (2023). "Optimal dispatch of a multi-energy system microgrid under uncertainty: A renewable energy community in Austria". In: *Applied Energy* 337. Publisher: Elsevier, p. 120913. ISSN: 0306-2619 (cit. on pp. 11, 66, 67, 69–71, 93, 94, 96–107, 115, 117, 118).

Houben, Nikolaus, Han Li, Tianzhen Hong, Hans Auer, Amela Ajanovic, and Reinhard Haas (2025a). "Short-term Electricity Load Forecasting: Application-Driven Evaluation of Machine Learning Models Across Spatial and Temporal Scales." In: *Engineering Applications of Artificial Intelligence* (under review). Publisher: Elsevier (cit. on pp. 10, 20, 53–59, 71, 73–84, 107, 108, 115–117).

Houben, Nikolaus, Lennard R. Visser, Wilfried G. J. H. M. van Sark, Hans Auer, Amela Ajanovic, and Reinhard Haas (2025b). "Interpretable net load forecasting with model-based deep learning for behind-the-meter generation and storage". In: *IEEE Transactions on Smart Grid* (under review) (cit. on pp. 11, 22, 60–65, 71, 85–92, 107, 115–117).

Hu, Maomao, Bruce Stephen, Jethro Browell, Stephen Haben, and David CH Wallom (2023). "Impacts of building load dispersion level on its load forecasting accuracy: Data or algorithms? Importance of reliability and interpretability in machine learning". In: *Energy and Buildings* 285. Publisher: Elsevier, p. 112896. ISSN: 0378-7788 (cit. on p. 21).

Hyndman, Rob J. and Yeasmin Khandakar (July 2008). "Automatic Time Series Forecasting: The forecast Package for R". en. In: *Journal of Statistical Software* 27, pp. 1–22. ISSN: 1548-7660. DOI: 10.18637/jss.v027.i03. URL: https://doi.org/10.18637/jss.v027.i03 (visited on 04/29/2025) (cit. on p. 15).

Januschowski, Tim, Yuyang Wang, Kari Torkkola, Timo Erkkilä, Hilaf Hasson, and Jan Gasthaus (2022). "Forecasting with trees". In: *International Journal of Forecasting* 38.4. Publisher: Elsevier, pp. 1473–1481. ISSN: 0169-2070 (cit. on p. 32).

Kanwar, Aastha, Diego I. Hidalgo Rodríguez, Jan Von Appen, and Martin Braun (Jan. 2015). "A Comparative Study of Optimization- and Rule-Based Control for Microgrid Operation". en. In: ed. by Andreas Kubis, Christian Rehtanz, Anton Shapovalov, Dominik Hilbrich, and Ewa Plota. Publisher: Technische Universität Dortmund. DOI: 10.17877/DE290R-7262. URL: http://eldorado.tu-dortmund.de/handle/2003/33969 (visited on 05/01/2025) (cit. on pp. 31, 33, 110).

Karniadakis, George Em, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang (June 2021). "Physics-informed machine learning". en. In: *Nature Reviews Physics* 3.6. Publisher: Nature Publishing Group, pp. 422–440. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00314-5. URL: https://www.nature.com/articles/s42254-021-00314-5 (visited on 07/12/2024) (cit. on pp. 23–25, 33).

Kaur, Amanpreet, Lukas Nonnenmacher, and Carlos F. M. Coimbra (Nov. 2016). "Net load forecasting for high renewable energy penetration grids". en. In: *Energy* 114, pp. 1073–1084. ISSN: 0360-5442. DOI: 10.1016/j.energy.2016.08.067. URL: https://www.sciencedirect.com/

science/article/pii/S036054421631180X (visited on 01/19/2023) (cit. on p. 23).

Kaytez, Fazil (Apr. 2020). "A hybrid approach based on autoregressive integrated moving average and least-square support vector machine for long-term forecasting of net electricity consumption". en. In: *Energy* 197, p. 117200. ISSN: 0360-5442. DOI: 10.1016/j.energy.2020.117200. URL: https://www.sciencedirect.com/science/article/pii/S0360544220303078 (visited on 01/19/2023) (cit. on p. 22).

Kazmi, Hussain, Chun Fu, and Clayton Miller (July 2023). "Ten questions concerning data-driven modelling and forecasting of operational energy demand at building and urban scale". en. In: *Building and Environment* 239, p. 110407. ISSN: 0360-1323. DOI: 10.1016/j.buildenv.2023.110407. URL: https://www.sciencedirect.com/science/article/pii/S0360132323004341 (visited on 06/26/2023) (cit. on pp. 20, 22).

Ke, Guolin, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu (2017). "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in neural information processing systems* 30 (cit. on pp. 59, 145, 148).

Keil, Christian and George C Craig (2009). "A displacement and amplitude score employing an optical flow technique". In: *Weather and Forecasting* 24.5. Publisher: American Meteorological Society, pp. 1297–1308. ISSN: 1520-0434 (cit. on pp. 25, 32).

Kerscher, Selina and Pablo Arboleya (2022). "The key role of aggregators in the energy transition under the latest European regulatory framework". In: *International Journal of Electrical Power & Energy Systems* 134, p. 107361. ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2021.107361. URL: https://www.sciencedirect.com/science/article/pii/S0142061521006001 (cit. on p. 6).

Kuznetsova, Elizaveta, Yan-Fu Li, Carlos Ruiz, Enrico Zio, Graham Ault, and Keith Bell (Sept. 2013). "Reinforcement learning for microgrid energy management". In: *Energy* 59, pp. 133–146. ISSN: 0360-5442. DOI: 10.1016/j.energy.2013.05.060. URL: https://www.sciencedirect.com/science/article/pii/S0360544213004817 (visited on 05/01/2025) (cit. on p. 31).

Laurí, D., B. Lennox, and J. Camacho (Jan. 2014). "Model predictive control for batch processes: Ensuring validity of predictions". In: *Journal*

of Process Control 24.1, pp. 239–249. ISSN: 0959-1524. DOI: 10.1016/j.jprocont.2013.11.005. URL: https://www.sciencedirect.com/science/article/pii/S0959152413002308 (visited on 03/16/2025) (cit. on p. 45).

Lim, Bryan, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister (2021). "Temporal fusion transformers for interpretable multi-horizon time series forecasting". In: *International Journal of Forecasting* 37.4. Publisher: Elsevier, pp. 1748–1764. ISSN: 0169-2070 (cit. on pp. 16, 59, 153, 154).

Lim, Bryan and Stefan Zohren (Feb. 2021). "Time-series forecasting with deep learning: a survey". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 379.2194. Publisher: Royal Society, p. 20200209. DOI: 10.1098/rsta.2020.0209. URL: https://royalsocietypublishing.org/doi/full/10.1098/rsta.2020.0209 (visited on 04/29/2025) (cit. on pp. 15–18).

Lindberg, K. B., P. Seljom, H. Madsen, D. Fischer, and M. Korpås (June 2019). "Long-term electricity load forecasting: Current and future trends". In: *Utilities Policy* 58, pp. 102–119. ISSN: 0957-1787. DOI: 10.1016/j.jup.2019.04.001. URL: https://www.sciencedirect.com/science/article/pii/S0957178719300116 (visited on 05/05/2025) (cit. on p. 5).

Lusis, Peter, Kaveh Rajab Khalilpour, Lachlan Andrew, and Ariel Liebman (Nov. 2017). "Short-term residential load forecasting: Impact of calendar effects and forecast granularity". In: *Applied Energy* 205, pp. 654–669. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2017.07.114. URL: https://www.sciencedirect.com/science/article/pii/S0306261917309881 (cit. on pp. 21, 32).

M. Sun, T. Zhang, Y. Wang, G. Strbac, and C. Kang (Jan. 2020). "Using Bayesian Deep Learning to Capture Uncertainty for Residential Net Load Forecasting". In: *IEEE Transactions on Power Systems* 35.1, pp. 188–201. ISSN: 1558-0679. DOI: 10.1109/TPWRS.2019.2924294 (cit. on p. 23).

Masini, Ricardo P, Marcelo C Medeiros, and Eduardo F Mendes (2023). "Machine learning advances for time series forecasting". In: *Journal of economic surveys* 37.1. Publisher: Wiley Online Library, pp. 76–111. ISSN: 0950-0804 (cit. on pp. 15, 16, 18, 20).

Mayne, D. Q., J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert (June 2000). "Constrained model predictive control: Stability and optimality".

In: *Automatica* 36.6, pp. 789–814. ISSN: 0005-1098. DOI: 10.1016/S0005-1098(99)00214-9. URL: https://www.sciencedirect.com/science/article/pii/S0005109899002149 (visited on 11/05/2024) (cit. on p. 51).

Miller, Clayton, Anjukan Kathirgamanathan, Bianca Picchetti, Pandarasamy Arjunan, June Young Park, Zoltan Nagy, Paul Raftery, Brodie W. Hobson, Zixiao Shi, and Forrest Meggers (Oct. 2020). "The Building Data Genome Project 2, energy meter data from the ASHRAE Great Energy Predictor III competition". en. In: *Scientific Data* 7.1. Number: 1 Publisher: Nature Publishing Group, p. 368. ISSN: 2052-4463. DOI: 10.1038/s41597-020-00712-x. URL: https://www.nature.com/articles/s41597-020-00712-x (visited on 10/02/2023) (cit. on pp. 73, 74).

Morari, Manfred and Jay H. Lee (May 1999). "Model predictive control: past, present and future". In: *Computers & Chemical Engineering* 23.4, pp. 667–682. ISSN: 0098-1354. DOI: 10.1016/S0098-1354(98)00301-9. URL: https://www.sciencedirect.com/science/article/pii/S0098135498003019 (visited on 05/01/2025) (cit. on p. 30).

Moser, A., D. Muschick, M. Gölles, P. Nageler, H. Schranzhofer, T. Mach, C. Ribas Tugores, I. Leusbrock, S. Stark, F. Lackner, and A. Hofer (Mar. 2020). "A MILP-based modular energy management system for urban multi-energy systems: Performance and sensitivity analysis". In: *Applied Energy* 261, p. 114342. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2019.114342. URL: https://www.sciencedirect.com/science/article/pii/S030626191932029X (visited on 05/01/2025) (cit. on pp. 31, 33, 110).

Müller, Meinard (2007). "Dynamic time warping". In: *Information retrieval for music and motion*. Publisher: Springer, pp. 69–84. ISSN: 3540740473 (cit. on p. 26).

Murphy, Allan H (1993). "What is a good forecast? An essay on the nature of goodness in weather forecasting". In: *Weather and forecasting* 8.2. Publisher: American Meteorological Society, pp. 281–293. ISSN: 1520-0434 (cit. on pp. 26, 32).

Nosratabadi, Seyyed Mostafa, Rahmat-Allah Hooshmand, and Eskandar Gholipour (2017). "A comprehensive review on microgrid and virtual power plant concepts employed for distributed energy resources schedul-

ing in power systems". In: *Renewable and Sustainable Energy Reviews* 67. Publisher: Elsevier, pp. 341–363. ISSN: 1364-0321 (cit. on p. 7).

Nti, Isaac Kofi, Moses Teimeh, Owusu Nyarko-Boateng, and Adebayo Felix Adekoya (2020). "Electricity load forecasting: A systematic review". In: *Journal of Electrical Systems and Information Technology* 7.1. Publisher: Springer, pp. 1–19. ISSN: 2314-7172 (cit. on p. 22).

Nuytten, Thomas, Bert Claessens, Kristof Paredis, Johan Van Bael, and Daan Six (Apr. 2013). "Flexibility of a combined heat and power system with thermal energy storage for district heating". In: *Applied Energy* 104, pp. 583–591. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2012.11.029. URL: https://www.sciencedirect.com/science/article/pii/S0306261912008227 (visited on 03/02/2025) (cit. on p. 6).

Olivares, Kin G., Cristian Challu, Grzegorz Marcjasz, Rafał Weron, and Artur Dubrawski (2023). "Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx". In: *International Journal of Forecasting* 39.2, pp. 884–900. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2022.03.001. URL: https://www.sciencedirect.com/science/article/pii/S0169207022000413 (cit. on p. 152).

Oreshkin, Boris N, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio (2019). "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting". In: *arXiv preprint arXiv:1905.10437* (cit. on pp. 59, 151).

Oreshkin, Boris N., Grzegorz Dudek, Paweł Pełka, and Ekaterina Turkina (2021). "N-BEATS neural network for mid-term electricity load forecasting". In: *Applied Energy* 293, p. 116918. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2021.116918. URL: https://www.sciencedirect.com/science/article/pii/S0306261921003986 (cit. on p. 151).

Pardo, Angel, Vicente Meneu, and Enric Valor (2002). "Temperature and seasonality influences on Spanish electricity load". In: *Energy Economics* 24.1. Publisher: Elsevier, pp. 55–70. ISSN: 0140-9883 (cit. on p. 74).

Parisio, Alessandra, Evangelos Rikos, and Luigi Glielmo (Sept. 2014). "A Model Predictive Control Approach to Microgrid Operation Optimization". In: *IEEE Transactions on Control Systems Technology* 22.5, pp. 1813–1827. ISSN: 1558-0865. DOI: 10.1109/TCST.2013.2295737.

URL: https://ieeexplore.ieee.org/abstract/document/6705582 (visited on 05/01/2025) (cit. on p. 30).

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, and Luca Antiga (2019). "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems* 32 (cit. on p. 71).

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg (2011). "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12. Publisher: JMLR. org, pp. 2825–2830. ISSN: 1532-4435 (cit. on pp. 71, 144).

Peplinski, McKenna and Kelly T Sanders (Dec. 2023). "Residential electricity demand on CAISO Flex Alert days: a case study of voluntary emergency demand response programs". en. In: *Environmental Research: Energy* 1.1. Publisher: IOP Publishing, p. 015002. ISSN: 2753-3751. DOI: 10.1088/2753-3751/ad0fda. URL: https://dx.doi.org/10.1088/2753-3751/ad0fda (visited on 03/02/2025) (cit. on p. 6).

Perez, Richard, Pierre Ineichen, Robert Seals, Joseph Michalsky, and Ronald Stewart (1990). "Modeling daylight availability and irradiance components from direct and global irradiance". In: *Solar energy* 44.5. Publisher: Elsevier, pp. 271–289. ISSN: 0038-092X (cit. on p. 63).

Petropoulos, Fotios, Daniele Apiletti, Vassilios Assimakopoulos, Mohamed Zied Babai, Devon K Barrow, Souhaib Ben Taieb, Christoph Bergmeir, Ricardo J Bessa, Jakub Bijak, and John E Boylan (2022). "Forecasting: theory and practice". In: *International Journal of Forecasting* 38.3. Publisher: Elsevier, pp. 705–871. ISSN: 0169-2070 (cit. on p. 19).

Pinheiro, Marco G., Sara C. Madeira, and Alexandre P. Francisco (Feb. 2023). "Short-term electricity load forecasting—A systematic approach from system level to secondary substations". In: *Applied Energy* 332, p. 120493. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2022.120493. URL: https://www.sciencedirect.com/science/article/pii/S0306261922017500 (visited on 11/29/2023) (cit. on p. 21).

Putz, Dominik, Michael Gumhalter, and Hans Auer (Nov. 2021). "A novel approach to multi-horizon wind power forecasting based on deep neural architecture". In: *Renewable Energy* 178, pp. 494–505. ISSN: 0960-

1481. DOI: 10.1016/j.renene.2021.06.099. URL: https://www.sciencedirect.com/science/article/pii/S0960148121009654 (visited on 09/18/2023) (cit. on p. 151).

Putz, Dominik, Michael Gumhalter, and Hans Auer (Mar. 2023). "The true value of a forecast: Assessing the impact of accuracy on local energy communities". In: *Sustainable Energy, Grids and Networks* 33, p. 100983. ISSN: 2352-4677. DOI: 10.1016/j.segan.2022.100983. URL: https://www.sciencedirect.com/science/article/pii/S2352467722002284 (visited on 11/29/2023) (cit. on pp. 27, 32).

Ranaweera, Damitha K, George G Karady, and Richard G Farmer (1997). "Economic impact analysis of load forecasting". In: *IEEE Transactions on Power Systems* 12.3. Publisher: IEEE, pp. 1388–1392. ISSN: 0885-8950 (cit. on pp. 26, 32).

Roslan, M. F., M. A. Hannan, Pin Jern Ker, and M. N. Uddin (Apr. 2019). "Microgrid control methods toward achieving sustainable energy management". In: *Applied Energy* 240, pp. 583–607. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2019.02.070. URL: https://www.sciencedirect.com/science/article/pii/S0306261919303678 (visited on 05/01/2025) (cit. on p. 27).

Rubasinghe, Osaka, Tingze Zhang, Xinan Zhang, San Shing Choi, Tat Kei Chau, Yau Chow, Tyrone Fernando, and Herbert Ho-Ching Iu (2023). "Highly accurate peak and valley prediction short-term net load forecasting approach based on decomposition for power systems with high PV penetration". In: *Applied Energy* 333, p. 120641. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2023.120641. URL: https://www.sciencedirect.com/science/article/pii/S0306261923000053 (cit. on pp. 23, 24).

Ruggles, Tyler H., David J. Farnham, Dan Tong, and Ken Caldeira (May 2020). "Developing reliable hourly electricity demand data through screening and imputation". en. In: *Scientific Data* 7.1. Number: 1 Publisher: Nature Publishing Group, p. 155. ISSN: 2052-4463. DOI: 10.1038/s41597-020-0483-x. URL: https://www.nature.com/articles/s41597-020-0483-x (visited on 04/25/2023) (cit. on pp. 22, 73, 74).

Schlemminger, Marlon, Tobias Ohrdes, Elisabeth Schneider, and Michael Knoop (2022). "Dataset on electrical single-family house and heat pump

load profiles in Germany". In: *Scientific Data* 9.1. Publisher: Nature Publishing Group, pp. 1–11. ISSN: 2052-4463 (cit. on pp. 85, 163).

Shlezinger, Nir, Jay Whang, Yonina C. Eldar, and Alexandros G. Dimakis (May 2023). "Model-Based Deep Learning". In: *Proceedings of the IEEE* 111.5. Conference Name: Proceedings of the IEEE, pp. 465–499. ISSN: 1558-2256. DOI: 10.1109/JPROC.2023.3247480. (Visited on 07/12/2024) (cit. on pp. 24, 33).

Stadler, Michael, Salman Mashayekh, Sankar Narayan, Nicholas DeForest, and Tobias Brandt (2015). "Supervisory Controller for PV and Storage Microgrids". In: (cit. on pp. 27, 30, 38).

Tibshirani, Robert (Jan. 1996). "Regression Shrinkage and Selection Via the Lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288. ISSN: 0035-9246. DOI: 10.1111/j.2517-6161.1996.tb02080.x. URL: https://doi.org/10.1111/j.2517-6161.1996.tb02080.x (visited on 04/29/2025) (cit. on p. 15).

Van Aubel, Pol and Erik Poll (July 2019). "Smart metering in the Netherlands: What, how, and why". en. In: *International Journal of Electrical Power & Energy Systems* 109, pp. 719–725. ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2019.01.001. URL: https://www.sciencedirect.com/science/article/pii/S0142061518335142 (visited on 01/11/2023) (cit. on p. 25).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). "Attention is all you need". In: *Advances in neural information processing systems* 30 (cit. on pp. 16, 59, 64, 153, 154).

Visser, Lennard R., Boudewijn Elsinga, Tarek A. AlSkaif, and Wilfried G. J. H. M. van Sark (July 2022). "Open-source quality control routine and multi-year power generation data of 175 PV systems". In: *Journal of Renewable and Sustainable Energy* 14.4. Publisher: American Institute of Physics, p. 043501. DOI: 10.1063/5.0100939. URL: https://aip.scitation.org/doi/abs/10.1063/5.0100939 (visited on 01/06/2023) (cit. on pp. 85, 163).

Wang, Dan, Wanfu Zheng, Zhe Wang, Yaran Wang, Xiufeng Pang, and Wei Wang (June 2023). "Comparison of reinforcement learning and model predictive control for building energy system optimization". In: *Applied Thermal Engineering* 228, p. 120430. ISSN: 1359-4311. DOI:

10.1016/j.applthermaleng.2023.120430. URL: https://www.sciencedirect.com/science/article/pii/S1359431123004593 (visited on 04/30/2025) (cit. on pp. 30–32).

Wang, Yi, Ning Zhang, Qixin Chen, Daniel S. Kirschen, Pan Li, and Qing Xia (May 2018). "Data-Driven Probabilistic Net Load Forecasting With High Penetration of Behind-the-Meter PV". In: *IEEE Transactions on Power Systems* 33.3. Conference Name: IEEE Transactions on Power Systems, pp. 3255–3264. ISSN: 1558-0679. DOI: 10.1109/TPWRS.2017.2762599 (cit. on p. 23).

Wang, Zhe, Tianzhen Hong, Han Li, and Mary Ann Piette (May 2021). "Predicting city-scale daily electricity consumption using data-driven models". en. In: *Advances in Applied Energy* 2, p. 100025. ISSN: 2666-7924. DOI: 10.1016/j.adapen.2021.100025. URL: https://www.sciencedirect.com/science/article/pii/S2666792421000184 (visited on 04/25/2023) (cit. on pp. 22, 38).

Winters, Peter R. (Apr. 1960). "Forecasting Sales by Exponentially Weighted Moving Averages". In: *Management Science* 6.3. Publisher: INFORMS, pp. 324–342. ISSN: 0025-1909. DOI: 10.1287/mnsc.6.3.324. URL: https://pubsonline.informs.org/doi/abs/10.1287/mnsc.6.3.324 (visited on 04/29/2025) (cit. on p. 15).

Wu, Qing, Zhe Jiang, Kewei Hong, Liu Huazhong, Laurence Yang, and Jihong Ding (Feb. 2021). "Tensor-Based Recurrent Neural Network and Multi-Modal Prediction With Its Applications in Traffic Network Management". In: *IEEE Transactions on Network and Service Management* PP, pp. 1–1. DOI: 10.1109/TNSM.2021.3056912 (cit. on p. 149).

Yang, Dazhi et al. (Nov. 2020). "Verification of deterministic solar forecasts". In: *Special Issue on Grid Integration* 210, pp. 20–37. ISSN: 0038-092X. DOI: 10.1016/j.solener.2020.04.019. URL: https://www.sciencedirect.com/science/article/pii/S0038092X20303947 (cit. on p. 45).

Yang, Ying, Yang Zhang, Pietro Elia Campana, and Jinyue Yan (2017). "Peak-shaving and profit-sharing model by Aggregators in residential buildings with PV– a case study in Eskilstuna, Sweden". In: *Energy Procedia* 142, pp. 3182–3193. ISSN: 1876-6102. DOI: https://doi.org/10.1016/j.egypro.2017.12.488. URL: https://www.sciencedirect.com/science/article/pii/S1876610217362434 (cit. on p. 6).

Yildiz, Baran, Jose I Bilbao, and Alistair B Sproul (2017). "A review and analysis of regression and machine learning models on commercial building electricity load forecasting". In: *Renewable and Sustainable Energy Reviews* 73. Publisher: Elsevier, pp. 1104–1122. ISSN: 1364-0321 (cit. on pp. 21, 32).

Zeng, Ailing, Muxi Chen, Lei Zhang, and Qiang Xu (2022). "Are transformers effective for time series forecasting?" In: *arXiv preprint arXiv:2205.13504* (cit. on p. 16).

Zou, Hui and Trevor Hastie (Apr. 2005). "Regularization and Variable Selection Via the Elastic Net". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 67.2, pp. 301–320. ISSN: 1369-7412. DOI: 10.1111/j.1467-9868.2005.00503.x. URL: https://doi.org/10.1111/j.1467-9868.2005.00503.x (visited on 04/29/2025) (cit. on p. 15).

Conference Papers

Chen, Tianqi and Carlos Guestrin (Aug. 2016). "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. arXiv:1603.02754 [cs], pp. 785–794. DOI: 10.1145/2939672.2939785. URL: http://arxiv.org/abs/1603.02754 (visited on 04/23/2023) (cit. on pp. 15, 59, 71, 145).

Falkner, Stefan, Aaron Klein, and Frank Hutter (2018). "BOHB: Robust and efficient hyperparameter optimization at scale". In: PMLR, pp. 1437–1446. ISBN: 2640-3498 (cit. on p. 58).

Gilpin, Leilani H., David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal (Oct. 2018). "Explaining Explanations: An Overview of Interpretability of Machine Learning". In: *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 80–89. DOI: 10.1109/DSAA.2018.00018. URL: https://ieeexplore.ieee.org/abstract/document/8631448 (visited on 05/01/2025) (cit. on pp. 23, 24).

Gokhale, Gargya, Jonas Van Gompel, Bert Claessens, and Chris Develder (2023). "Transfer Learning in Transformer-Based Demand Forecasting For Home Energy Management System". In: pp. 458–462 (cit. on p. 27).

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: pp. 770–778 (cit. on pp. 64, 153).

Negenborn, R.R., M. Houwing, B. De Schutter, and J. Hellendoorn (Mar. 2009). "Model predictive control for residential energy resources using a mixed-logical dynamic model". In: *2009 International Conference on Networking, Sensing and Control*, pp. 702–707. DOI: 10.1109/ICNSC.2009.4919363. URL: https://ieeexplore.ieee.org/abstract/document/4919363 (visited on 05/01/2025) (cit. on p. 30).

Rawlings, James B, David Angeli, and Cuyler N Bates (2012). "Fundamentals of economic model predictive control". In: IEEE, pp. 3851–3861. ISBN: 1-4673-2066-8 (cit. on pp. 45, 50, 51).

Saboori, H., M. Mohammadi, and R. Taghe (Mar. 2011). "Virtual Power Plant (VPP), Definition, Concept, Components and Types". In: *2011 Asia-Pacific Power and Energy Engineering Conference*. ISSN: 2157-4847, pp. 1–4. DOI: 10.1109/APPEEC.2011.5749026. URL: https://ieeexplore.ieee.org/abstract/document/5749026?casa_token=8OQdtbyuWCUAAAAA:fKGspAd6CuFd-6kKxEjdL8HdX54PAhxJi17nlWX4EhvyDAMAkaLOqQ58-uljkkkAs8yafq0 (visited on 03/02/2025) (cit. on p. 6).

Voss, Marcus (June 2020). "Permutation-Based Residential Short-term Load Forecasting in the Context of Energy Management Optimization Objectives". In: *Proceedings of the Eleventh ACM International Conference on Future Energy Systems*. e-Energy '20. New York, NY, USA: Association for Computing Machinery, pp. 231–236. ISBN: 978-1-4503-8009-6. DOI: 10.1145/3396851.3397731. URL: https://dl.acm.org/doi/10.1145/3396851.3397731 (visited on 11/29/2023) (cit. on p. 26).

Xu, Feiyu, Hans Uszkoreit, Yangzhou Du, Wei Fan, Dongyan Zhao, and Jun Zhu (2019). "Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges". en. In: *Natural Language Processing and Chinese Computing*. Ed. by Jie Tang, Min-Yen Kan, Dongyan Zhao, Sujian Li, and Hongying Zan. Cham: Springer International Publishing, pp. 563–574. ISBN: 978-3-030-32236-6. DOI: 10.1007/978-3-030-32236-6_51 (cit. on p. 23).

Other sources

Free Open-Source Weather API | Open-Meteo.com (2023). URL: https://open-meteo.com/ (visited on 12/29/2023) (cit. on pp. 74, 89).

Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun (Apr. 2018). *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. arXiv:1803.01271 [cs]. DOI: 10.48550/arXiv.1803.01271. URL: http://arxiv.org/abs/1803.01271 (visited on 04/29/2025) (cit. on p. 16).

Biewald, Lukas (2020). *Experiment Tracking with Weights and Biases*. URL: https://wandb.ai/site/research,%20http://wandb.ai/site/research (visited on 05/21/2023) (cit. on pp. 58, 71, 89).

Chung, Junyoung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio (Dec. 2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. arXiv:1412.3555 [cs]. DOI: 10.48550/arXiv.1412.3555. URL: http://arxiv.org/abs/1412.3555 (visited on 04/29/2025) (cit. on pp. 16, 59, 148, 153).

Das, Abhimanyu, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu (Apr. 2024). *Long-term Forecasting with TiDE: Time-series Dense Encoder*. arXiv:2304.08424 [cs, stat]. DOI: 10.48550/arXiv.2304.08424. URL: http://arxiv.org/abs/2304.08424 (visited on 07/15/2024) (cit. on p. 63).

Denholm, Paul, Matthew O'Connell, Gregory Brinkman, and Jennie Jorgenson (2015). *Overgeneration from solar energy in california. a field guide to the duck chart*. Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States) (cit. on p. 6).

Dobos, A. (Sept. 2014). *PVWatts Version 5 Manual*. en. Tech. rep. NREL/TP-6A20-62641, 1158421, NREL/TP–6A20–62641, 1158421. DOI: 10.2172/1158421. URL: http://www.osti.gov/servlets/purl/1158421/ (visited on 01/22/2023) (cit. on p. 63).

EIA (2025). *As solar capacity grows, duck curves are getting deeper in California - U.S. Energy Information Administration (EIA)*. URL: https://www.eia.gov/todayinenergy/detail.php?id=56880 (visited on 05/05/2025) (cit. on p. 7).

Energy, US Department of (2025). *Demand Response*. en. URL: https://www.energy.gov/oe/demand-response (visited on 03/02/2025) (cit. on p. 6).

Kingma, Diederik P. and Jimmy Ba (Jan. 2017). *Adam: A Method for Stochastic Optimization*. arXiv:1412.6980. DOI: 10.48550/arXiv.1412.6980. URL: http://arxiv.org/abs/1412.6980 (visited on 11/11/2024) (cit. on p. 41).

Ofgem (2025). *Ofgem decision P415 'Facilitating Access to Wholesale Markets for Flexibility Dispatched by VLPs* (cit. on p. 7).

Oord, Aaron van den, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu (Sept. 2016). *WaveNet: A Generative Model for Raw Audio*. en. arXiv:1609.03499 [cs]. DOI: 10.48550/arXiv.1609.03499. URL: http://arxiv.org/abs/1609.03499 (visited on 04/29/2025) (cit. on p. 16).

The ESO's Demand Flexibility Service | ESO (2024). URL: https://www.nationalgrideso.com/industry-information/balancing-services/demand-flexibility-service/esos-demand-flexibility-service (visited on 02/04/2024) (cit. on p. 6).

Trindade, Artur (2015). *ElectricityLoadDiagrams20112014*. DOI: 10.24432/C58C86. URL: https://archive-beta.ics.uci.edu/dataset/321 (visited on 04/25/2023) (cit. on p. 74).

Zhou, Ella, David Hurlbut, and Kaifeng Xu (Sept. 2021). *A Primer on FERC Order No. 2222: Insights for International Power Systems*. en. Tech. rep. NREL-TP-5C00-80166, 1823766, MainId:42369, NREL–TP–5C00–80166, 1823766, MainId:42369. DOI: 10.2172/1823766. URL: https://www.osti.gov/servlets/purl/1823766/ (visited on 04/09/2025) (cit. on p. 7).

Appendices

Appendix A.

Mathematical Details of Machine Learning Algorithms

A.1. Linear Regression

Linear Regression is a classical statistical method for modeling linear relationships between variables, and it has been in use for well over a century. While it is neither a decision tree-based method nor a neural network, this work includes its prediction as a benchmark. The rudimentary mathematical expression is given by Eq. (A.1):

$$\hat{y}_{pred} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_M x_M \tag{A.1}$$

where,

- $\hat{y}_{pred}$: Predicted target value (scalar, e.g., one step $\hat{y}_{1|t}$)
- x_m: Input feature m (a component of $\mathbf{X}^{(t)}$ for a given forecast origin t)
- β_m: Model coefficients (learnable parameters, components of $\boldsymbol{\theta}$)
- M: Number of features

The training process consists of finding the optimal combination of parameters $\boldsymbol{\beta}$ (part of $\boldsymbol{\theta}$) to linearly transform the feature vectors into the target vector. This can be achieved by solving a system of M equations, or as performed in sklearn's implementation, through gradient descent Pedregosa et al., 2011.

A.2. Random Forest

Random Forest Breiman, 2001 is a decision tree ensemble algorithm, which for the context of forecasting means that predictions are obtained by the averaging of M_{trees} functions (Classification and Regression Trees) as shown in Eq. (A.2). For a given input $\mathbf{X}^{(i)}$ from the dataset:

$$\hat{\mathbf{y}}^{(i)} = \frac{1}{M_{trees}} \sum_{m_{tree}=1}^{M_{trees}} p_{m_{tree}}(\mathbf{X}^{(i)}) \; ; \; p_{m_{tree}} \in P \tag{A.2}$$

where,

- $p_{m_{tree}}$: A classification and regression tree (CART)
- P: The space of regression trees
- $\mathbf{X}^{(i)}$: Input feature vector for sample i
- $\hat{\mathbf{y}}^{(i)}$: Predicted output sequence (or value) for sample i
- M_{trees}: Number of trees in the forest

For each regression tree $p_{m_{tree}}$ the covariate space is recursively partitioned based on discrete or Boolean values into subsets to minimize a cost criterion. Figure A.1 exemplifies this, where the top-most node - the *parent node* - splits the data into two more groups, the *child nodes*. This process is repeated until the splits are found that optimally separate the data based on the cost crite-

rion and other hyperparameters, such as maximum tree depth. The order of the features and the numerical value by which the data are split are systematically chosen to minimize the gap between the forecasts and the observed values. If a split for a particular branch does not reduce the cost criterion (by a significant margin), the process is stopped. In this way, terminal nodes, or *leaf nodes* are created, which can later be used for forecasting.

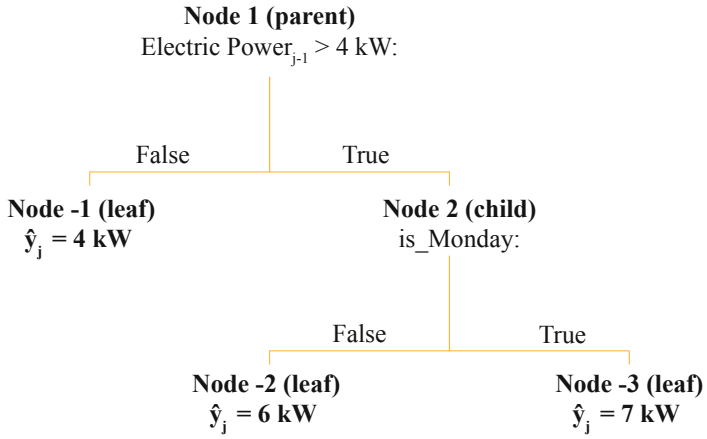


Figure A.1.: Regression Tree $p_{m_{tree}}$ for time series forecasting: a simple example of how a regression tree would split features for electrical load forecasting.

The random forest algorithm is trained through *bagging* - short for *bootstrap aggregation* - which involves creating subsets of the original dataset by randomly selecting samples with replacement. Each subset is used to train an individual tree Breiman, 1996. The objective of this method is to introduce diversity among the trees and improve the model's ability to generalize.

A.3. XGBoost & LightGBM

Like Random Forest, Extreme Gradient Boost (XGBoost) algorithm Chen and Guestrin, 2016 and the Light Gradient Boosting Machine (LightGBM) Ke et al., 2017 are tree-based algorithms. However unlike Random Forest,

which builds trees independently, these two algorithms build trees sequentially, where each new tree is built to correct the errors of the preceding ensemble. Furthermore, XGBoost and LightGBM leverage a technique called *gradient tree-boosting*. In the XGBoost algorithm, the cost criterion used to learn the tree set P is a *regularized* cost function given by Eq. (A.3).

$$\mathcal{L}_{obj}(\boldsymbol{\theta}) = \sum_{i=1}^{N_{train}} l(y^{(i)}, \hat{y}^{(i)}) + \sum_{m_{tree}=1}^{M_{trees}} \Omega(p_{m_{tree}}) \tag{A.3}$$

Here l is a differentiable loss function (e.g., mean squared error, part of $\mathcal{L}$ in Eq. (3.6)) that measures the distance between the forecast $\hat{y}^{(i)}$ (prediction for sample i from the current ensemble of M_{trees} trees) and the target $y^{(i)}$. Furthermore, the regularization term $\Omega(p)$ is given by Eq. (A.4), and penalizes the complexity of the model. This is achieved by considering the number of leaf nodes N_{leaf} and the dimensionality of their respective weights $\mathbf{w}$.

$$\Omega(p) = \gamma N_{leaf} + \frac{1}{2}\lambda ||\mathbf{w}||^2 \tag{A.4}$$

where,

- γ: Pruning hyperparameter
- λ: Smoothing hyperparameter (L2 regularization term)
- N_{leaf}: Number of leaf nodes in a tree p
- $\mathbf{w}$: Vector of leaf weights (scores) in tree p

Gradient Tree Boosting: In XGBoost, Eq. (A.3) is optimized in an iterative manner. Formally, let $\hat{y}^{(i,z)}$ be the forecast of the i-th training example at the z-th iteration (i.e., with z trees). Implied by *boosting*, the algorithm greedily adds a tree $p_z(\mathbf{X}^{(i)})$ to minimize the objective given by Eq. (A.5).

$$\mathcal{L}_{obj}^{(z)} = \sum_{i=1}^{N_{train}} l(y^{(i)}, \hat{y}^{(i,z-1)} + p_z(\mathbf{X}^{(i)})) + \Omega(p_z) \tag{A.5}$$

In practice, Eq. (A.5) is approximated by using a second-order Taylor expansion of the loss function:

$$\mathcal{L}_{obj}^{(z)} \simeq \sum_{i=1}^{N_{train}} [l(y^{(i)}, \hat{y}^{(i,z-1)}) + g_i p_z(\mathbf{X}^{(i)}) + \frac{1}{2} h_i p_z^2(\mathbf{X}^{(i)})] + \Omega(p_z) \tag{A.6}$$

where,

- g_i: First-order gradient: $\partial_{\hat{y}^{(i,z-1)}} l(y^{(i)}, \hat{y}^{(i,z-1)})$
- h_i: Second-order gradient: $\partial^2_{\hat{y}^{(i,z-1)}} l(y^{(i)}, \hat{y}^{(i,z-1)})$

Omitting the constant terms $l(y^{(i)}, \hat{y}^{(i,z-1)})$ and defining $I_e = \{i | q(\mathbf{X}^{(i)}) = e\}$ as the set of all data samples assigned to leaf node e of tree p_z (where $q(\mathbf{X}^{(i)})$ maps an instance to a leaf index), and $p_z(\mathbf{X}^{(i)}) = w_{q(\mathbf{X}^{(i)})}$, Eq. (A.6) is rewritten as (focusing on terms related to p_z):

$$\tilde{\mathcal{L}}^{(z)}(p_z) \simeq \sum_{e=1}^{N_{leaf}} \left[\left(\sum_{i \in I_e} g_i \right) w_e + \frac{1}{2} \left(\sum_{i \in I_e} h_i + \lambda \right) w_e^2 \right] + \gamma N_{leaf} \tag{A.7}$$

To find the optimal weight w_e^* of a leaf e, for a fixed tree structure p_z:

$$w_e^* = -\frac{\sum_{i \in I_e} g_i}{\sum_{i \in I_e} h_i + \lambda} \tag{A.8}$$

Substituting w_e^* in Eq. (A.7), the quality score of a fixed tree structure p_z can be calculated according to Eq. (A.9).

$$\tilde{\mathcal{L}}^{(z)}(p_z) = -\frac{1}{2} \sum_{e=1}^{N_{leaf}} \frac{\left(\sum_{i \in I_e} g_i\right)^2}{\sum_{i \in I_e} h_i + \lambda} + \gamma N_{leaf} \tag{A.9}$$

Incremental Algorithm: As it is intractable to enumerate all possible trees, a greedy algorithm that starts from a single parent node and iteratively adds branches is used instead. Let I_L and I_R be the training sample sets of left and right nodes after a split. With $I = I_L \cup I_R$, the incremental loss reduction, or the gain G, from this split can be calculated by Eq. (A.10)

$$G = \frac{1}{2}\left[\frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{\left(\sum_{i\in I} g_i\right)^2}{\sum_{i\in I} h_i + \lambda}\right] - \gamma \tag{A.10}$$

The term γ represents the penalty for adding a new leaf (or rather, the difference in tree complexity before and after the split, which results in a penalty for adding a split if the gain from the sum of squared scores does not offset it).

While LightGBM follows similar principles as XGBoost, the major difference between the two algorithms is the tree growth strategy. Specifically, LightGBM chooses the leaf node that has the maximum reduction in the loss function (or impurity) at each step (leaf-wise growth). The algorithm selects the best split based on the feature that contributes the most to reducing the loss, leading to a more aggressive search for informative features. This approach can lead to deeper trees with fewer leaf nodes. For further reading regarding the mathematical difference compared to XGBoost, the original manuscript by Ke et al. Ke et al., 2017 is recommended.

A.4. Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) is a Deep Neural Network, specifically a recurrent neural network (RNN). These were developed to model sequential, time-dependent data. Since RNNs' inception in the late 1980s (see Elman, 1990), the Long Short Term Memory (LSTM) model by Hochreiter and Schmidhuber Hochreiter and Schmidhuber, 1997 has been a popular recurrent architecture. With the introduction of gates, LSTM addressed the issue of vanishing gradients and thus improved the gradient-flow during backpropagation. The Gated Recurrent Unit (GRU) Chung et al., 2014 is a simpler variant of the

LSTM, omitting the memory gate.

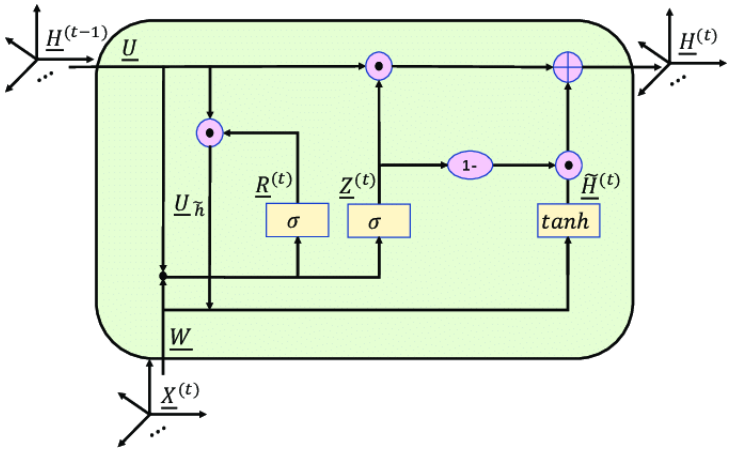


Figure A.2.: Architecture of a Gated Recurrent Unit taken from Wu et al., 2021. In this diagram, X_t corresponds to an input $\mathbf{x}_s$ at sequence step s, h_{t-1} to $\mathbf{S}_{s-1}$, and h_t to $\mathbf{S}_s$.

As shown in Figure A.2, for an input sequence (e.g., derived from $\{\mathbf{x}_{t-\tau}^{\text{past}}\}_{\tau=0}^{L}$), each GRU cell at sequence step s takes as input the hidden state $\mathbf{S}_{s-1}$ from the previous cell and the current input $\mathbf{x}_s$. It computes the hidden state $\mathbf{S}_s$ based on a combination of gating mechanisms: the reset gate $\mathbf{R}_s$ and the update gate $\mathbf{Z}_s$. These gates determine how the information flows through the cell and which information is stored or forgotten.

$$\mathbf{R}_s = \sigma(\mathbf{W}_r[\mathbf{S}_{s-1}, \mathbf{x}_s] + \mathbf{b}_r) \tag{A.11}$$

$$\mathbf{Z}_s = \sigma(\mathbf{W}_z[\mathbf{S}_{s-1}, \mathbf{x}_s] + \mathbf{b}_z) \tag{A.12}$$

$$\tilde{\mathbf{S}}_s = \tanh(\mathbf{W}_h[\mathbf{R}_s \odot \mathbf{S}_{s-1}, \mathbf{x}_s] + \mathbf{b}_h) \tag{A.13}$$

$$\mathbf{S}_s = (1 - \mathbf{Z}_s) \odot \mathbf{S}_{s-1} + \mathbf{Z}_s \odot \tilde{\mathbf{S}}_s \tag{A.14}$$

where,

- $\mathbf{R}_s$: Reset gate vector at sequence step s
- $\mathbf{Z}_s$: Update gate vector at sequence step s

- $\mathbf{S}_s$: Hidden state vector at sequence step s
- $\tilde{\mathbf{S}}_s$: Candidate hidden state vector at sequence step s
- $\mathbf{x}_s$: Input vector at sequence step s (e.g., $\mathbf{x}_{t-(L-s)}$)
- $\mathbf{W}_r, \mathbf{W}_z, \mathbf{W}_h$: Weight matrices (part of $\boldsymbol{\theta}$)
- $\mathbf{b}_r, \mathbf{b}_z, \mathbf{b}_h$: Bias vectors (part of $\boldsymbol{\theta}$)
- σ: Sigmoid activation function
- tanh: Hyperbolic tangent activation function
- $\odot$: Element-wise multiplication
- $[\cdot, \cdot]$: Concatenation operator

This architecture allows GRUs to learn long-term dependencies while being more efficient than LSTMs due to the reduced number of gates. The gating mechanisms, specifically the reset and update gates, allow the GRU to capture various time-scale patterns in the data, making them suitable for tasks like time-series prediction.

Note that in most Recurrent Neural Network (RNN) implementations for time series forecasting, the model is designed to predict one step into the future and then relies on its own predictions to forecast further. This method is known as the *recursive* multi-step ahead forecasting (see Section 3.1.1.2, or original Section 2.2). However, in the work presented here, a different approach for the GRU is applied, termed the "BlockRNN", which represents a *MIMO* formulation of the forecasting task (see Section 3.1.1.2). The BlockRNN does not recursively unroll into the future up to the desired forecast horizon H. Instead, at forecast origin t, the final hidden layer $\mathbf{S}_t$ (after processing the input sequence $\mathbf{X}^{(t)}$) is directly mapped to the forecast sequence $\hat{\mathbf{y}}^{(t)}$ using a final linear layer, as shown in Eq. (A.15).

$$\hat{\mathbf{y}}^{(t)} = \mathbf{W}_f \mathbf{S}_t + \mathbf{b}_f \tag{A.15}$$

where,

- $\hat{\mathbf{y}}^{(t)}$: Forecast sequence $\{\hat{y}_{k|t}\}_{k=1}^{H}$ of H elements, made at origin t
- $\mathbf{S}_t$: Final hidden state from GRU processing inputs related to origin t
- $\mathbf{W}_f$: Final weight matrix (e.g., of size (output_features$\times H$, hidden_size))
- $\mathbf{b}_f$: Final bias vector (e.g., of size output_features $\times$ H)

(Dimensions of $\mathbf{W}_f, \mathbf{b}_f$ depend on whether output is univariate or multivariate per step).

A.5. N-BEATS

N-BEATS is a deep neural network architecture that was specifically developed for time series forecasting, presented in the context of beating common benchmarks on the M3, M4 and TOURISM datasets Oreshkin et al., 2019. It has also been effectively used in energy forecasting (e.g., Putz et al., 2021; Oreshkin et al., 2021). Although it is not a recurrent architecture, it achieves remarkable depth through stacking multiple blocks of fully-connected neural networks, each predicting the future outputs (forecast) and their contribution to the decomposition of the input (backcast). As shown in Figure A.3, each block consists of a sequence of fully connected layers, each using the historic residual of the previous block as input. As the authors write in Oreshkin et al., 2021: "the architecture runs a residual recursion over the entire input window and sums block outputs to make its final forecast".

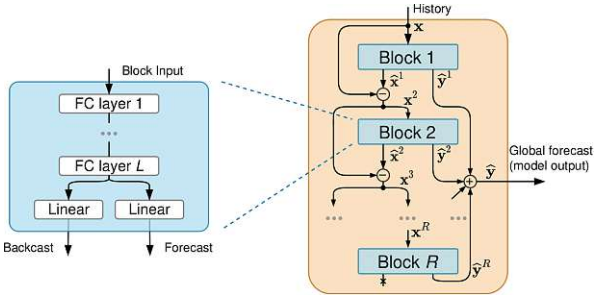


Figure A.3.: Architecture of N-BEATS with exogenous feature support taken from Olivares et al., 2023.

Formally, assume there are R residual blocks with N_L hidden layers each, and l and r denote the superscripts of a particular layer in a block. The input to the first block, $x^{(0)}$, is derived from the model's input features $\mathbf{X}^{(t)}$ (typically the look-back portion $\{\mathbf{x}_{t-\tau}^{\text{past}}\}_{\tau=0}^{L}$). The fully connected layer with weights $\mathbf{W}^{r,l}$ and bias term $\mathbf{b}^{r,l}$ can be expressed by Eq. (A.16).

$$FC^{r,l}(h^{r,l-1}) = \text{ReLU}(\mathbf{W}^{r,l}h^{r,l-1} + \mathbf{b}^{r,l}) \tag{A.16}$$

Then the recursive operation of N-BEATS is given by Eq. (A.17).

$$\begin{aligned} x^{(r)} &= \text{ReLU}[x^{(r-1)} - \hat{x}^{(r-1)}_{\text{backcast}}], \quad \text{(input to block r)} \\ h^{r,1} &= FC^{r,1}(x^{(r)}), \\ &\vdots \\ h^{r,N_L} &= FC^{r,N_L}(h^{r,N_L-1}), \\ \hat{x}^{(r)}_{\text{backcast}} &= \mathbf{B}^{r}_{\text{backcast}}h^{r,N_L}, \\ \hat{y}^{(r)}_{\text{forecast}} &= \mathbf{B}^{r}_{\text{forecast}}h^{r,N_L} \end{aligned} \tag{A.17, A.18}$$

The final model forecast is $\hat{\mathbf{y}}^{(t)} = \sum_{r=1}^{R} \hat{y}^{(r)}_{\text{forecast}}$. Note that even though the algorithm performs a recursive step between blocks, it strictly belongs to the MIMO type as discussed in Section 3.1.1.2, as each block's forecast part directly contributes to the final multi-step output without feeding predictions back as inputs in a time-recursive manner.

A.6. Temporal Fusion Transformer (TFT)

Temporal Fusion Transformer (TFT) by Lim et al. Lim et al., 2021 is a variant of the original transformer neural network (as presented in Vaswani et al., 2017). It combines recurrent neural network layers (e.g., GRU Chung et al., 2014 or LSTM) and self-attention mechanisms.

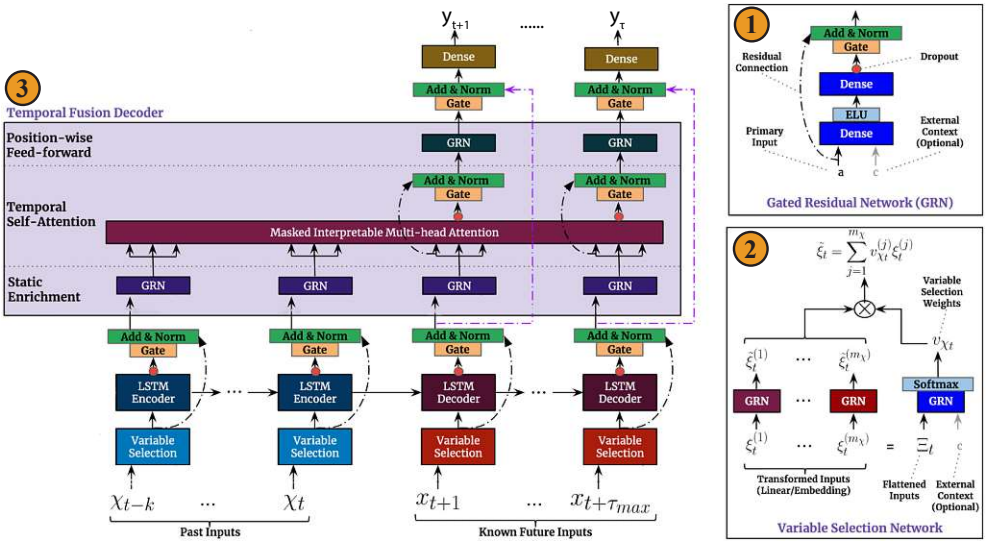


Figure A.4.: Architecture of the Deterministic Temporal Fusion Transformer, adapted from Lim et al., 2021.

The architecture, shown in Figure A.4, consists of the following differentiable building blocks:

1. *Gated Residual Network (GRN)*: Multi-layer perceptron (see Hornik et al., 1989) with two layers and a residual connection (see He et al., 2016). The residual connection allows the network to skip over any unused components of the architecture. As shown in Figure A.4, the GRN component is used throughout the architecture, *i.e.* in the Variable Selection Network and before and after the self-attention layer.

2. *Variable selection networks (VSN)*: This component combines multiple instances of the GRN. Particularly, for each input variable ξ_v (which

can be a static variable, a past time-varying input from $\{\mathbf{x}_{t-\tau}^{\text{past}}\}_{\tau=0}^{L}$, or a future known input from $\{\mathbf{x}_{k}^{\text{future}}\}_{k=1}^{H}$), its representation is flattened across its specific time dimension (if applicable) to $\boldsymbol{\Xi}_v$. This is then used to produce variable-specific weights w_{ξ_v}, shown in Eq. (A.19).

$$w_{\xi_v} = \text{softmax}(\text{GRN}_{w_\xi}(\boldsymbol{\Xi}_v)) \tag{A.19}$$

As the decoder rolls out over time, each feature vector $\xi_v^{(s)}$ (variable v at its own sequence step s) is input into its own GRN and weighted with w_{ξ_v}. Note that past inputs $\{\mathbf{x}_{t-\tau}^{\text{past}}\}$ and future known inputs $\{\mathbf{x}_{k}^{\text{future}}\}$ have their own VSNs, with shared weights across all time steps t of forecast origin. As static features are not used in this work, the context vector c can be ignored in Figure A.4.

3. *Temporal Fusion Decoder*: This component combines LSTM encoder layers (see Hochreiter and Schmidhuber, 1997) with a modified self-attention layer (see Vaswani et al., 2017). The idea of the authors in Lim et al., 2021 is that the self-attention layer, which is effective in modeling long-term dependencies in a sequence, can benefit from the localized enhancement of the LSTM. The authors use the scaled dot-product attention, as given by Eq. (A.20).

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_{attn}}}\right)\mathbf{V} \tag{A.20}$$

where,

- $\mathbf{Q}, \mathbf{K}, \mathbf{V}$: Linear projections of the transformed feature vectors $\overline{\boldsymbol{\xi}}$ called queries, keys and values.
- d_{attn}: Width dimension of the key tensor, which is a model choice.

As proposed in the original transformer paper, the attention mechanism is framed in a multi-head setup (see Vaswani et al., 2017), however, here with the additive aggregation of attention heads to improve interpretability, see Eq. (A.21). (Note: The original TFT aggregates

heads before a final linear transformation).

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \left(\frac{1}{M_H} \sum_{h=1}^{M_H} \text{Attention}(\mathbf{Q}\mathbf{W}_Q^h, \mathbf{K}\mathbf{W}_K^h, \mathbf{V}\mathbf{W}_V^h) \right) \mathbf{W}_O \tag{A.21}$$

where,

- $\mathbf{W}_Q^h, \mathbf{W}_K^h, \mathbf{W}_V^h$: Weight matrices to project queries, keys, and values for head h.
- $\mathbf{W}_O$: Output weight matrix for combining heads.
- M_H: Number of attention heads.

Note that while the original implementation of the TFT is designed for probabilistic forecasting, this work uses the deterministic version.

Appendix B.

Optimization Input Data, System Parameters, and Net Load Error Details

B.1. Optimization Input Data from Case Study (RQ3)

The responsible Distribution System Operator (DSO) in this region is the KNG-Kärnten Netz GmbH, who provided - together with meo Energy GmbH - the electrical load profile measurement data of the entire community. In this process, the collected historical load data starting from June 2020 until beginning of May 2021 are used for training the forecasting model, which enables the multi-step forecasting given in subsection *2.2.1* of RQ3 using the queried real-time data collected during the controller testing period. The test period started at the beginning of May 2021 and lasted one month. The 15 minute based historical and real-time electrical load data gathered during the open-loop-test period are shown in Figure B.1.

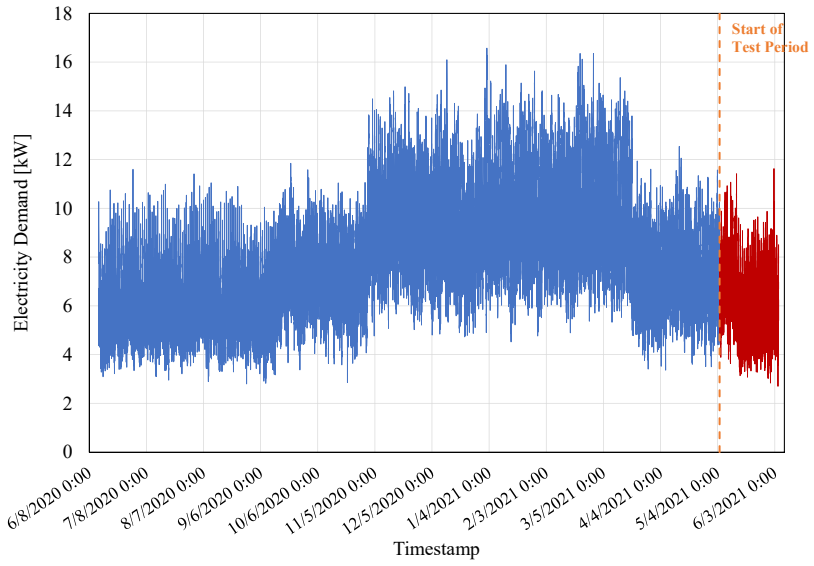


Figure B.1.: The electrical demand profile in a 15 minute timestep of the entire REC: the dashed vertical line indicates the end of the historical data set used for the forecast model training and the begin of the considered test period.

In addition, meo Energy GmbH provided the PV production measurement data for the existing PV plant at the site. The historical measurement data used for training the forecasting model is from the end of February 2021 to the beginning of May 2021. The data from the PV system is recorded in a 2-minute interval and resampled to a 15 minute interval by calculating the mean value for each measurement over 15 minute time intervals. The 15 minute based historical and real-time PV production data gathered during the open-loop-test period are shown in Figure B.2.

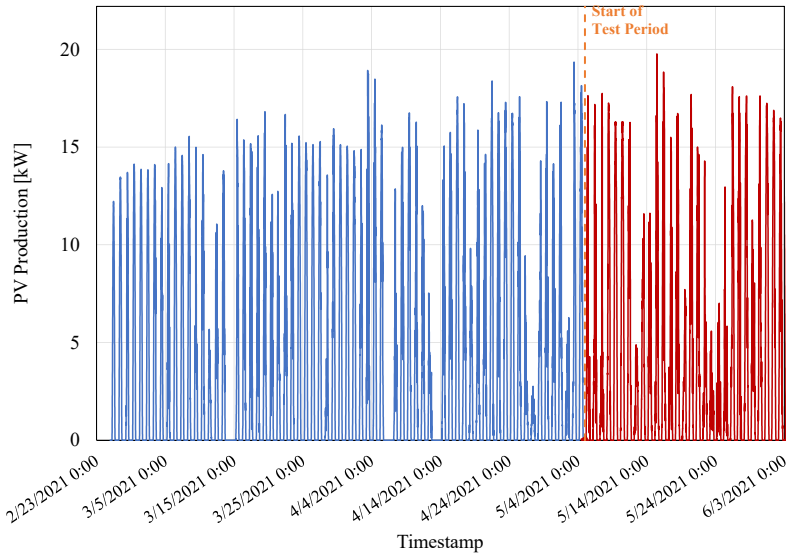


Figure B.2.: The solar PV production data in a 15 minute timestep of the considered PV system at the site: the dashed vertical line indicates the end of the historical data set used for the forecast model training and the begin of the considered test period.

Besides the PV production measurement data, the global horizontal irradiance, direct normal irradiance, and diffuse horizontal irradiance are also used as an exogenous covariate for the PV forecasting model (*Model 2* in RQ3), as described in subsection *2.2.1* of RQ3. For this purpose, both historical and real-time global horizontal irradiance data provided by **solcast** at the REC test site on a half-hourly basis are used. The 30-min based historical and real-time global horizontal irradiance data are shown in Figure B.3.

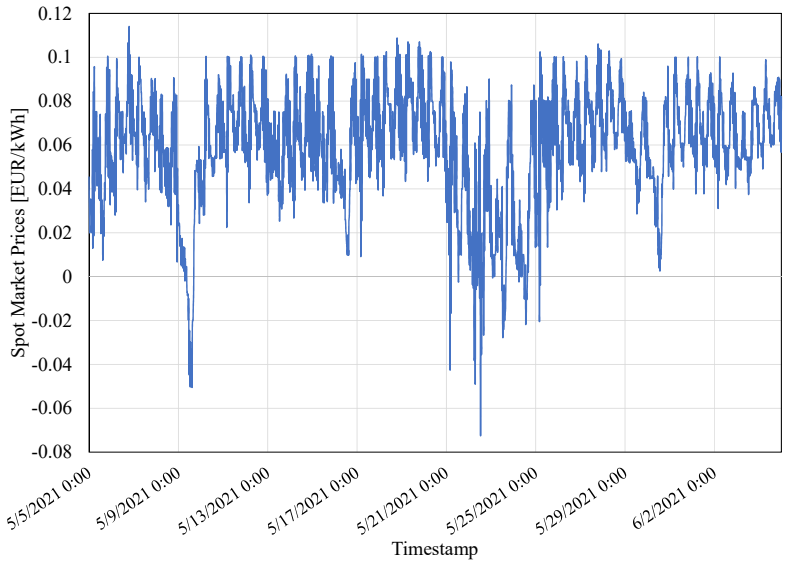


Figure B.3.: The GHI data in a 30-minute timestep at the site. The dashed vertical line indicates the end of the historical data set used for the forecast model training and the begin of the considered test period.

In order to assess the operational CO_2 emissions of the energy community correctly, the CO_2 content of the purchased electricity from the utility also needs to be considered. The marginal CO_2 emissions rate of the purchased electricity is considered in $kgCO_2eq/kWh$ in this study and is given by **CO2data** on hourly basis. The marginal CO_2 emissions rate indicates the carbon footprint of the electricity being consumed in a given location zone based on the Austrian electricity mix and import mix of electricity sources. The hourly-based real-time marginal CO_2 emissions of the utility electricity for the considered open-loop-test period in Austria are shown in Figure B.4. The hourly EXAA spot market prices for the same period are shown in Figure B.5.

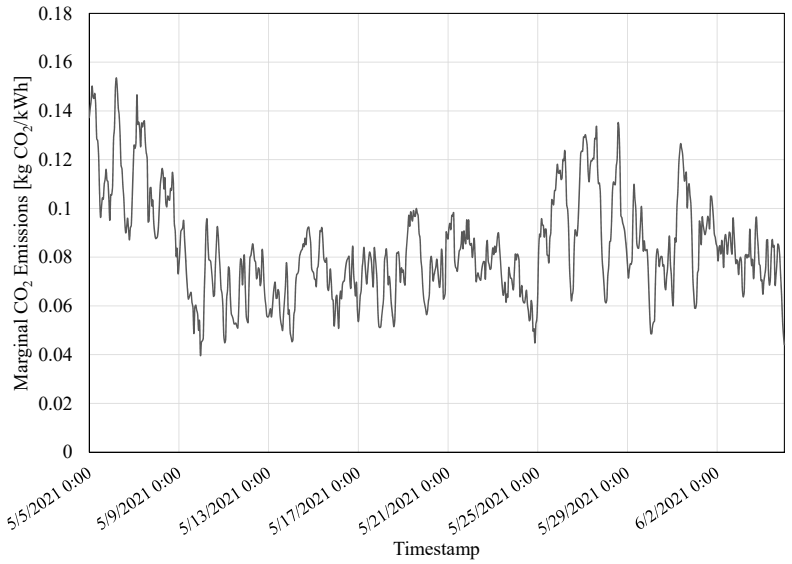


Figure B.4.: The hourly marginal CO_2 emissions of the utility electricity for the considered open-loop-test period in Austria **CO2data**.

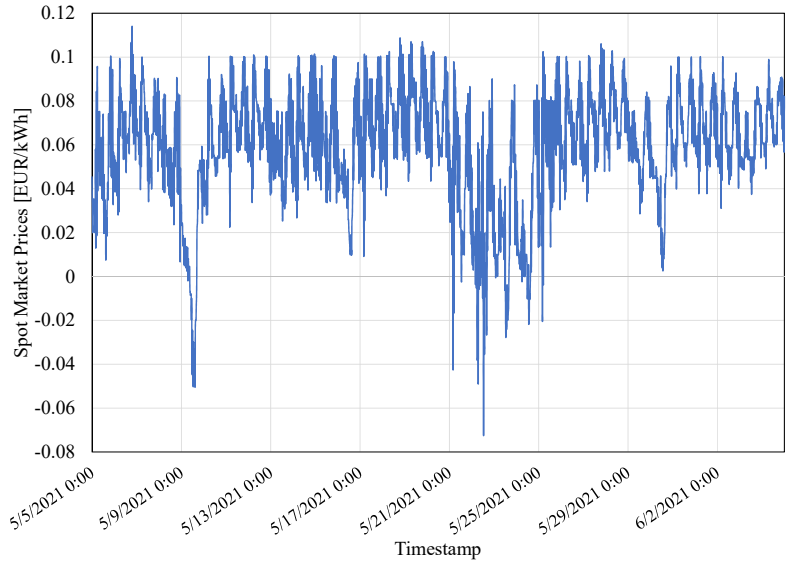


Figure B.5.: The hourly EXAA spot market prices for the considered open-loop-test period in Austria **SpotPrice**.

B.1.1. Domestic-Hot-Water Profile Generation (SAX Method)

As described in RQ3, domestic hot water (DHW) load is not forecast, but randomly generated based on historic data. In short, at runtime the DHW profile is generated by selecting a pre-processed profile of a random day from the time period before the 'test period'. The elaboration of this method and the 15 minute DHW demand profile of a household in the REC. Note that in this work, the DHW load $y_{k_h}^{DHW}$ (demand at optimization step k_h) is not forecast, but rather generated by an analysis with Symbolic Aggregate approXimation (SAX) **Lin2003**, as implemented by the python library *pyts* **Faouzi2020**. In this method, a normalized time series is discretized by dividing each entry into an equivalent range, usually denoted by a letter (from a-z). Formally, bins $\mathcal{B} = \{\beta_1, ..., \beta_a\}$ are constructed from a $N(0, 1)$ Gaussian probability density function. Each bin has an equal area (probability), given by $1/a$. Then each bin is labeled with a letter, starting with $a = \beta_1, b = \beta_2, ...$. Next, the normalized observations of a time series are mapped to each bin, and assigned the respective letter. This effectively transforms the time series into a string of letters. To reconstruct the DHW profile at run time, a string of letters of a random day is drawn from samples obtained from historic data, and each letter is replaced by its respective bin mean. Note that in the original version by **Lin2003**, binning is preceded by a dimensionality reduction of the time series via piece-wise aggregate approximation (PAA). For this research, however, this step was skipped, as the data was resampled to the desired interval prior entering SAX. Figure B.6 shows an example DHW demand profile.

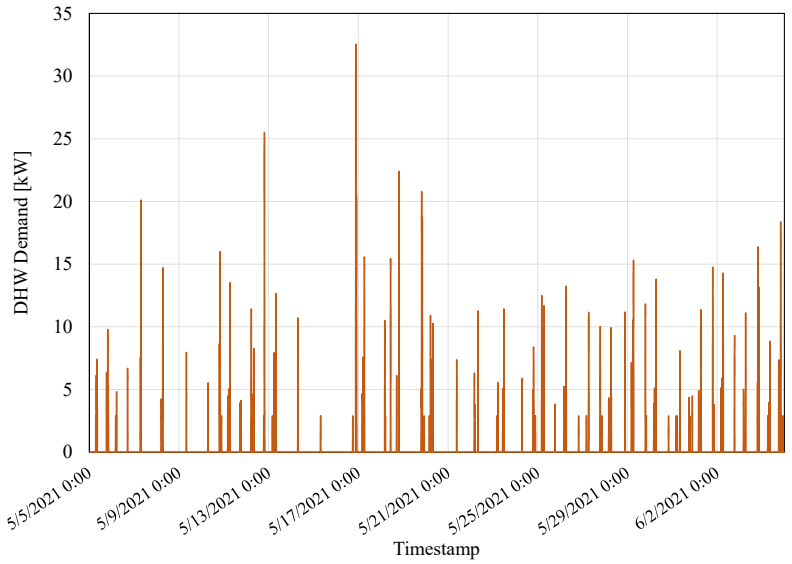


Figure B.6.: The DHW demand profile in a 15 minute timestep of a household in the REC for the considered open-loop-test period at the site.

B.1.2. Techno-Economical Parameters for REC Case Study (RQ3)

The technical parameters for the battery energy storage system (BESS) and the domestic heat storage (DHS) include i.a. the charging and discharging efficiencies (η^{ch}, η^{dis}), the maximum charge and discharge rates (related to $\mathcal{U}$), the minimum SOC (part of $\mathcal{X}$) and the heat loss rate of the storages. These parameters are given in Table B.1 (**Tesla2020**; **SelfDischarge2018**) and in Table B.2.

Parameter	Value
Charging Efficiency of BESS [%]	95
Discharging Efficiency of BESS [%]	95
Maximum Charge Rate [% per hour]	34
Maximum Discharge Rate [% per hour]	34
Minimum SOC of BESS [%]	5
Self-Discharge Rate of BESS [% per hour]	0.2

Table B.1.: The technical parameters of the Li-ion based BESS (**Tesla2020**; **SelfDischarge2018**).

The technical parameters for the power-to-heat system and the domestic heat storage include the efficiency of the PtH element, the charging and discharging efficiencies, the maximum charge and discharge rates, the minimum SOC and the heat loss rate of a typical domestic heat storage (hot water boiler). These parameters are given in Table B.2 and are based on assumptions derived from analysis of measured power-to-heat and domestic heat storage system data.

Parameter	Value
Efficiency of PtH element [%]	100
Charging Efficiency of DHS [%]	100
Discharging Efficiency of DHS [%]	95
Maximum Charge Rate [% per hour]	14.5
Maximum Discharge Rate [% per hour]	100
Minimum SOC of DHS [%]	60
Heat Loss Rate of DHS [% per hour]	1

Table B.2.: The technical parameters of the domestic heat storage.

B.2. Hardware and Tariff Information of Households (RQ2)

For the study in RQ2, 20 households were drawn from the PV dataset of Visser et al., 2022 matched with corresponding load data from Schlemminger et al., 2022. The specific (PV ID, SFH ID) pairs are: (ID003, SFH3), (ID089,

SFH4), (ID002, SFH5), (ID095, SFH9), (ID096, SFH10), (ID113, SFH12), (ID109, SFH16), (ID139, SFH18), (ID052, SFH19), (ID028, SFH21), (ID042, SFH22), (ID070, SFH23), (ID101, SFH27), (ID021, SFH28), (ID103, SFH29), (ID061, SFH30), (ID012, SFH31), (ID142, SFH32), (ID108, SFH36), (ID165, SFH38). The tilt angles range from 14° to 50°, and the average DC power capacity is approximately 2.4 kWp. All households have a uniform battery configuration, consisting of a 5 kWh energy capacity (E^{max}), 2.5 kW (dis-)charging limit ($P^{ch,max}, P^{dis,max}$), and 0.86 roundtrip efficiency. They are subject to a time-of-use tariff (including taxes and grid charges, defining $\pi^{buy}_{k|t}$) of 0.06 €/kWh from 00:00–06:00, 0.12 €/kWh from 06:00–08:00, 0.06 €/kWh from 08:00–18:00, and 0.20 €/kWh from 18:00–24:00. A flat feed-in tariff ($\pi^{sell}_{k|t}$) of 0.08 €/kWh applies for exported electricity.

B.2.1. NLE Data Assumptions

The NLE metric depends as much on the the forecast errors as on the assumptions regarding the stylized energy system. The goal in devising the system setup, was to provide a transparent and simple energy system that is transferable to real grid operations. To make results comparable across datasets, the pricing scheme and BESS specifications are parameterized by the scaled load time series (see Eq. (B.1)).

$$y_s^{scaled} = \frac{y_s - \min(\mathbf{y}_{hist})}{\max(\mathbf{y}_{hist}) - \min(\mathbf{y}_{hist})} \tag{B.1}$$

where y_s is a value from the time series and $\mathbf{y}_{hist}$ is a representative historical segment of the target variable. Based on this scaled load time series the price signals and storage are parameterized as a daily demand charge price $\pi^{DC} = 1$ (relative unit), a BESS C-Rate and Capacity of 1 (relative units). The linear optimization problem was solved with the CPLEX solver, leading to a total calculation time of the net load error of 2 minutes for a month of forecasts in hourly temporal resolution.

B.3. Hyperparameters

B.3.1. Neural Network Dropout Values (RQ1)

The following table shows dropout values for the best performing neural network models from RQ1.

Table B.3.: Dropout values for the best performing neural network models (RQ1)

Scale	Location	GRU	N-BEATS	TFT
County	Los_Angeles	0.0	0.0	0.2
	New_York	0.2	0.1	0.2
	Sacramento	0.2	0.0	0.0
Town	town_0	0.2	0.1	0.2
	town_1	0.0	0.0	0.0
	town_2	0.1	0.0	0.2
Neighborhood	neighborhood_0	0.1	0.2	0.2
	neighborhood_1	0.1	0.2	0.1
	neighborhood_2	0.0	0.0	0.1
Building	building_1	0.0	0.2	0.1
	building_2	0.2	0.1	0.2
	building_3	0.1	0.1	0.2

B.3.2. Darts and XGBoost Hyperparameters (RQ2)

Following the Darts argument naming convention, all implemented models in RQ2 were trained with: ‘input$_{c}hunk_{l}ength$‘ $= 192(correspondstolook - backwindow\mathrm{L} \approx 192)$, ‘output$_{c}hunk_{l}ength$‘ $= 96(correspondstoforecasthorizon\mathrm{H}{=}96)$, ‘$hidden_{s}ize$‘ $= 32$, ‘$dropout$‘ $= 0.1$, ‘$batch_{s}ize$‘ $= 32$, ‘$n_{e}pochs$‘ $= 100$, ‘$random_{s}tate$‘ $= None, andanAdamoptimizer$ $1e{-}3).Additionallyduetotheimplementationoftheattentiondecoder, thefinalproposedmodelinRQ2$ ‘$n_{h}eads$‘ $= 4$, ‘$d_{m}odel$‘ $= 64$, ‘$dim_{f}eedforward$‘ $= 512$.

Furthermore the baseline models based on the XGBoost model in RQ2, was trained with ‘lags‘=192, ‘output$_{c}hunk_{l}ength$‘ $= 96$, ‘$n_{e}stimators$‘ $= 100$, ‘$max_{d}epth$‘ $= 6$, ‘$booster$‘ $=' gbtree', and$‘$objective$‘ $=' reg : squarederror'$.

B.4. Weights & Biases Repository Supplementary (RQ1)

The complete hyperparameters tuning runs, training runs and results of the study in RQ1 are available in the various *Weights & Biases* repositories. In these repository, different acronyms mapping to the dataset names were opted for as follows: 'LDWP' to 'Los_Angeles', 'BANC' to 'Sacramento', 'NYIS' to 'New_York', 'MT_196' to 'town_0', 'MT_279' to 'town_1', 'MT_208' to 'town_2', 'Bull' to 'neighborhood_0', 'Hog' to 'neighborhood_1', 'Bobcat' to 'neighborhood_2', 'Be_Sandy' to 'building_0', 'Be_Millie' to 'building_1', and 'Co_Joel' to 'building_2'.

Appendix C.

Supplementary Results

C.1. Supplementary Forecasting Results (RQ3)

Table C.1 compares the cross-validated error scores (nRMSE, MAPE as defined in Section 3.1.1.5) over the forecast horizon and computation time of different algorithms with the best hyperparameter combination for the electrical load data of the test-bed from RQ3. For comparison, all covariates were kept identical in all runs. The error score is the timestep-weighted average of *Model 1* and *Model 2* from RQ3. That is, the number of timesteps in which *Model 1* was executed (up to the switching point) represents the weighting of its performance metrics, and the same is true for *Model 2*. Combinations where *Model 1* uses a different algorithm than *Model 2* (also known as *regime switching*) have not been tested, but may be explored in future work.

	Error Score		Computational Time [s]	
	nRMSE	MAPE [%]	Training Time	Execution Time
Linear Regression	2.01×10^{-2}	16.2	0.1	0.4
Random Forest Regression	1.93×10^{-2}	14.4	209.4	2.4
XGBoost Regression	1.87×10^{-2}	13.9	14.2	0.3
Support Vector Regression	2.47×10^{-2}	18.1	35.6	0.4

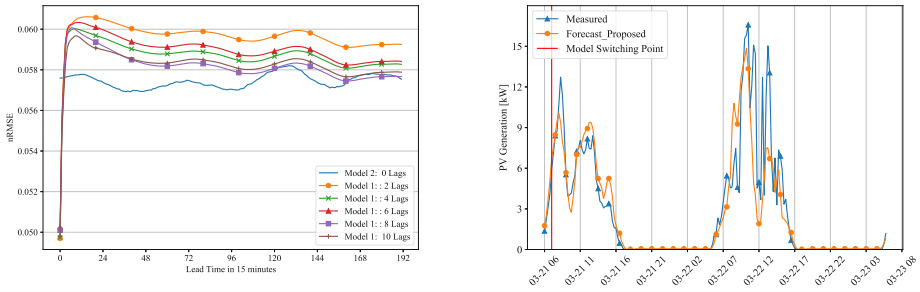
Table C.1.: Comparison of Algorithms for Electrical Load based on Error Scores and Computational Performance (RQ3).

Based on these results from RQ3, the XGBoost Regression algorithm was selected for electrical load forecasts. In further implementations that might

require even less computational time, the forecasting method presented in subsection 2.2.1 of RQ3 allows seamless use of other algorithms with faster computations, such as is *Linear Regression*. The switching point calculation for the forecasts of PV generation were performed according to Table C.2.

	Error Score	Computational Time		Switching Point
L	nRMSE	Training Time [s]	Execution Time [ms]	nRMSE
2	0.0575	3.2	0.1	2
4	0.0575	3.5	0.1	3
6	0.0574	3.8	0.3	2
8	0.0574	3.8	0.3	2
10	0.0574	4.0	0.3	2

Table C.2.: Comparison of Error Score and Computational Performance of *Model 1* Configurations for Photovoltaic Generation (RQ3). Here L refers to the number of lags (look-back window length) for Model 1.



(a) nRMSE Trajectory of the photovoltaic 'dev' data set

(b) Solar PV Power Generation Forecast on 21.03.2021

Figure C.1.: PV Forecasts (RQ3): nRMSE-Trajectory of PV Forecasts for one day (rolling) and 48-hours ahead forecasts of photovoltaic power generation on a "dev" data set (before the case-study).

C.2. Detailed Optimization and Economic Analysis Results (RQ3)

The remaining 4 optimization cases from RQ3, which were not shown in the results section of that paper, are exhibited here:

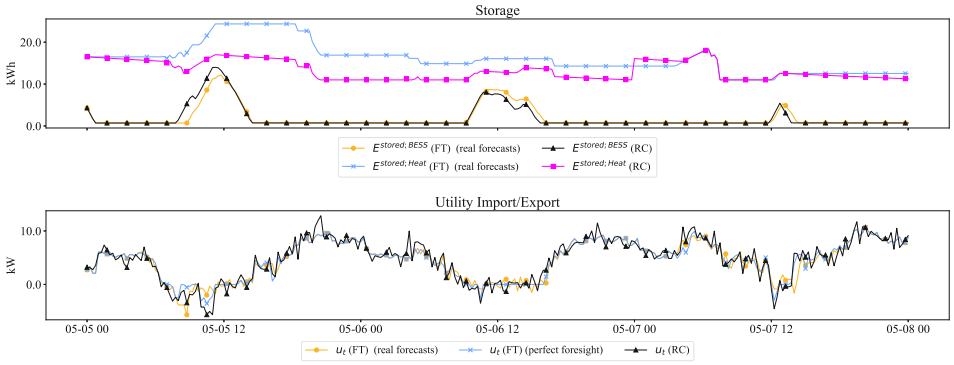


Figure C.2.: Optimal operational dispatch for the OC-FT case vs. reference case (RC) from 05.05.2021 until 08.05.2021 (RQ3).

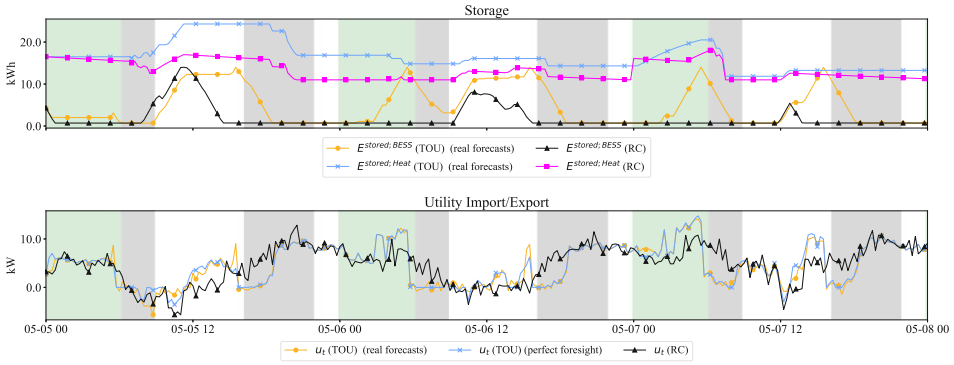


Figure C.3.: Optimal operational dispatch for the OC-TOU case vs. reference case (RC) from 05.05.2021 until 08.05.2021 (RQ3). The background colors correspond to: gray: "on" (35.8 €ct/kWh) , white: "mid" (29.84 €ct/kWh, green: "off" (23.87 €ct/kWh).

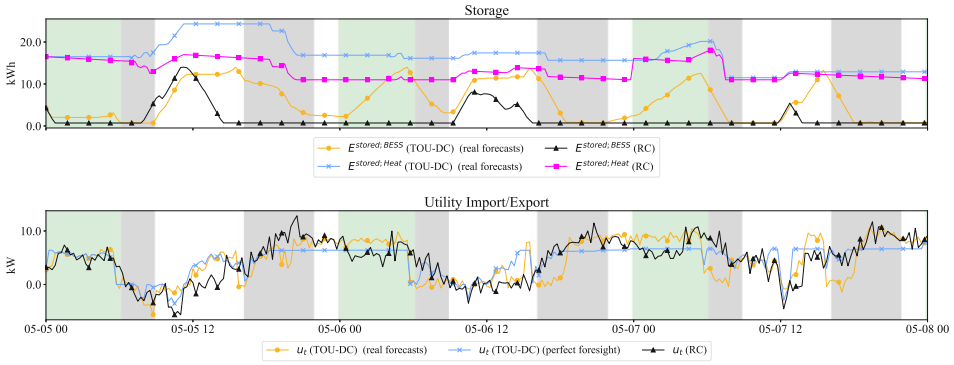


Figure C.4.: Optimal operational dispatch for the OC-TOU-DC case vs. reference case (RC) from 05.05.2021 until 08.05.2021 (RQ3). The background colors correspond to: gray: "on" (35.8 €ct/kWh) , white: "mid" (29.84 €ct/kWh, green: "off" (23.87 €ct/kWh).

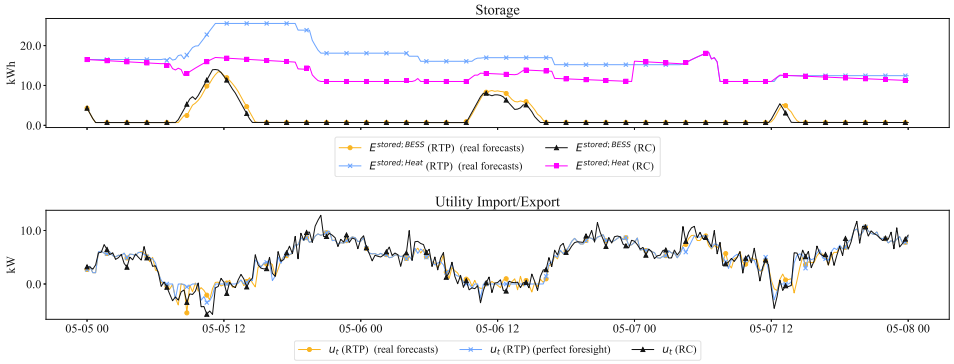


Figure C.5.: Optimal operational dispatch for the OC-RTP case vs. reference case (RC) from 05.05.2021 until 08.05.2021 (RQ3).

Declarations

C.3. Declaration of Authorship and Originality

I hereby declare that this thesis was composed independently by myself, and that no source other than those referenced has been used. All quotations or ideas from other works have been clearly cited. Where co-authors have contributed to the presented work, their contributions are explicitly acknowledged within the corresponding chapters or publications. Any citation errors or omissions found within this thesis are entirely unintentional and will be rectified promptly upon notification.

C.4. Use of Artificial Intelligence

Aritificial Intelligence (AI) assistance was employed strictly for language refinement (*e.g.*, re-wording or embellishment), proofreading, and formatting (*i.e.,* table and figure arrangement); all content, results, research insights, original contributions, and conceptual frameworks were developed exclusively by the author and co-authors listed in the respective paper citation.