UNIVERSITÀ DEGLI STUDI DI MESSINA

DIPARTIMENTO DI INGEGNERIA

DOTTORATO IN INGEGNERIA
INDUSTRIALE E DELL'INFORMAZIONE
CICLO XXXVIII
SSD: ING-INF 04

Interpretable Modeling Frameworks for Dynamical Systems: From First Principles to Explainable AI in Smart and Industrial Applications

---

Author:
Francesca SAPUPPO

Supervisors:
Prof. Maria Gabriella XIBILIA
Prof. Luca PATANÈ

Coordinator:
Prof. Edoardo Proverbio

AA 2024-2025

UNIVERSITÀ DEGLI STUDI DI MESSINA
DIPARTIMENTO DI INGEGNERIA

Abstract

Dottorato in Ingegneria Industriale e dell'Informazione
Ciclo XXXVIII
AA 2024-2025

**Interpretable Modeling Frameworks
for Dynamical Systems:
From First Principles to Explainable AI
in Smart and Industrial Applications**

by Francesca SAPUPPO

This dissertation develops and applies a unified methodological framework for modeling dynamical systems that places interpretability, transparency, and explainability at the center of model selection and design. The framework replaces a purely source-of-knowledge view with a dual-axis taxonomy that distinguishes four classes, white-, knowledge-driven grey-, data-driven grey-, and black-box, and operationalizes movement between them via calibration, physics-guided approximation of hard submodels, and surrogate modeling. It couples taxonomy-aware pipelines with practical indicators of model complexity, transparency, and explanatory adequacy, and it is validated across three application domains.

In smart sensing, the thesis introduces a first-principle, dual-carrier partial differential equations model implemented via finite element method for bacterial-cellulose ionic transducers, preserving mechanistic clarity while enabling a hybrid 2D–1D simulation. The model is experimentally validated and used to explain the roles of curvature, advection, and interfacial phenomena. Transparent data-driven models are then benchmarked: Auto-Regressive with eXogenous Inputs and Finite Input Response (FIR) excel in one-step prediction, whereas Nonlinear FIR provides superior long-horizon simulation.

In industrial soft-sensing, classical linear regressors and their nonlinear counterparts are compared with Koopman-operator-based state-space representations on processes such as Sulfur Recovery Units and distillation. Results show that knowledge-driven grey-box formulations can reconcile accuracy with structural insight, while data-driven-grey models retain auditability at higher flexibility; symbolic regression supports verifiable feature–response laws.

In energy systems, the framework is applied to vehicle-to-grid (V2G) forecasting. Koopman-based state-space models and machine-learning pipelines are complemented by SHapley Additive exPlanation (SHAP) to expose feature attributions and decision logic, delivering accurate predictions of aggregated available capacity together with actionable explanations.

In summary, the thesis proposes and applies a unified methodological framework that re-anchors model selection and design around interpretability, transparency, and explainability, and shows, through cross-domain studies, how hybrid paradigms can satisfy modern scientific, industrial, and financial requirements for trustworthy prediction and decision-making.

Acknowledgements

I warmly thank Prof. Maria Gabriella Xibilia and Prof. Luca Patanè for their constant personal and scientific support. Thanks to them I returned to my first passion, research, which I had set aside to work in industry and teaching but never forgot. I also thank Prof. Riccardo Caponetto, whose steady encouragement, even before this PhD, first prompted me to come back to research.

I am grateful to them for letting me being part of the Automation research group where respect for people comes first. This environment allowed me to restart this journey with a lighter heart and peace of mind, trusting in clear and honest communication that enables efficient, effective work and, beyond that, true friendship.

I also thank the Department of Engineering at the University of Messina for welcoming me. The calm, collaborative, and open environment made this journey more pleasant and effective. I am grateful to the PhD Board, and in particular to the Coordinator, Prof. Edoardo Proverbio, for fostering a climate of collaboration and simplicity throughout the program.

I thank Prof. Salvatore Graziani, from the Measurements Group of the Department of Electrical, Electronic and Computer Engineering of the University of Catania, for many helpful and stimulating discussions on smart sensors.

This journey has been enriched by the insights, critical feedback, and collaborative spirit of many colleagues, to whom I am truly grateful.

Finally, I am deeply grateful to my family for their constant support.

Fundings. The smart-sensor activities were supported by the European Union-NextGenerationEU, Mission 4, Component 1, "Green SENSing systems based on Bacterial Cellulose (SENS-BC)," CUP J53D23003460006. The Vehicle-to-Grid (V2G) studies were supported by the MASE - Consiglio Nazionale delle Ricerche within the project RICERCA DI SISTEMA 22-24 -21.2 Progetto Integrato Tecnologie di accumulo elettrochimico e termico. CUP Master: B53C22008540001, UNIME-DI-RdS22_24: J43C23000670001.

Contents

List of Figures

List of Tables

List of Abbreviations

AAC	Aggregated Available Capacity
ADAM	Adaptive Moment Estimation (optimizer)
ADH	Adherence (metric)
ANFIS	Adaptive Neuro-Fuzzy Inference System
ARX	AutoRegressive with eXogenous inputs
BC	Bacterial Cellulose
BMS	Battery Management System
CNN	Convolutional Neural Network
DC	Debutanizer Column
DL	Deep Learning
DMD	Dynamic Mode Decomposition
DMDc	Dynamic Mode Decomposition with Control
DT	Decision Tree
EAP	Electro Active Polymer
ES	Energy Supply
EV	Electric Vehicle
FCD	Floating Car Data
FCR	Frequency Containment Reserve
FIR	Finite Impulse Response
GBDT	Gradient-Boosted Decision Tree
GP	Genetic Programming
GP-NLS	Genetic Programming with Nonlinear Least Squares
GPS	Global Positioning System
HAVOK	Hankel Alternative View of Koopman
HDMD	Hankel Dynamic Mode Decomposition
HDMDc	Hankel Dynamic Mode Decomposition with Control
HEV	Hybrid Electric Vehicle
HODMD	High-Order Dynamic Mode Decomposition
ICE	Internal Combustion Engine
IoT	Internet of Things
KPI	Key Performance Indicator
LIME	Local Interpretable Model-agnostic Explanations
LPG	Liquefied Petroleum Gas
LR	Linear Regression
LSTM	Long Short-Term Memory
MAML	Model-Agnostic Meta-Learning
MAPE	Mean Absolute Percentage Error
MDL	Minimum Description Length
MEA	Monoethanolamine
ML	Machine Learning
MPC	Model Predictive Control
MSE	Mean Squared Error

MSA	Multi-Step-Ahead
MSA-HDMDc	Multi-Step-Ahead Hankel DMD with Control
NAR	Nonlinear AutoRegressive
NARX	Nonlinear AutoRegressive with eXogenous inputs
NFIR	Nonlinear Finite Impulse Response
NLR	Nonlinear Regression
NN	Neural Network
OD	Origin-Destination
ODE	Ordinary Differential Equation
PCA	Principal Component Analysis
PCR	Principal Component Regression
PDE	Partial Differential Equation
PD	Peak Demand
PHEV	Plug-in Hybrid Electric Vehicle
PINN	Physics-Informed Neural Network
PLSR	Partial Least-Squares Regression
POD	Proper Orthogonal Decomposition
PySR	Python Symbolic Regression (library)
RF	Random Forest
ReLU	Rectified Linear Unit
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
SARIMA	Seasonal AutoRegressive Integrated Moving Average
SEC	Schedulable Energy Capacity
SG	Smart Grid(s)
SHAP	Shapley Additive Explanations
SINDy	Sparse Identification of Nonlinear Dynamics
SoC	State of Charge
SR	Symbolic Regression
SRU	Sulfur Recovery Unit
STLSQ	Sequential Thresholded Least Squares
SVD	Singular Value Decomposition
SWS	Sour Water Stripping
V2G	Vehicle-to-Grid
VED	Vehicle Energy Dataset
XAI	Explainable Artificial Intelligence
XGBoost	Extreme Gradient Boosting
R^2	Coefficient of Determination

To my family

Chapter 1

Introduction

The increasing complexity of modern engineered systems, spanning advanced materials, industrial processes, and multi-energy networks, demands modeling frameworks that jointly optimize predictive accuracy, *interpretability*, and *transparency*. Traditional *white-box* models, grounded in first-principles, offer mechanistic insights, but can become computationally burdensome and, in practice, effectively opaque when scaled to high-dimensional or strongly nonlinear regimes. Conversely, purely *data-driven black-box* approaches deliver flexibility and strong predictive power, yet often lack *explainability* and struggle to generalize beyond their training distributions. In safety-critical, clinical, industrial, and financial contexts, where predictions inform actions with material risk, decision-makers increasingly require models whose internal logic and failure modes can be scrutinized, justified, and audited.

This thesis, titled *Interpretable Modeling Frameworks for Dynamical Systems*, advances a paradigm shift from the traditional classification of models by *source of knowledge*, i.e., physics versus data, to a richer taxonomy that explicitly incorporates *interpretability*, *transparency*, and *explainability* as first-class design objectives. We identify a gap in current practice and propose a *cartographic* view, a methodological map within which hybrid models can be positioned and navigated as they blend domain knowledge and empirical evidence. This map is not merely conceptual, it is used throughout the thesis to contextualize model design choices, to make trade-offs explicit, for example capability versus complexity, inference speed versus fidelity, and to guide transitions between classical and modern modeling regimes.

The thesis operationalizes this perspective across three application arenas, smart green materials, industrial soft-sensing, and energy systems with an emphasis on smart mobility and vehicle-to-grid (V2G), to show that interpretability can be engineered at multiple layers of the pipeline, from structure, for example, physics-informed forms and symbolic expressions, to learning, for example, constrained identification and Koopman-operator-based state-space formulations, to explanation interfaces, for example local and global attributions. Rather than treating applications as ends in themselves, each domain serves as a testbed for exploring the spectrum from first-principles Partial Differential Equations (PDEs) through symbolic regression (SR), linear and nonlinear system identification, AutoRegressive with eXogenous inputs (ARX) and Finite Impulse Response (FIR), Nonlinear ARX (NARX) and Nonlinear FIR (NFIR), Hankel Dynamic Mode Decomposition with control (HDMDc) and its multi-step-ahead variant (MSA-HDMDc), to machine-learning (ML) pipelines coupled with eXplainable Artificial Intelligence (XAI). In doing so, the work demonstrates how models can move on the map, toward greater transparency without forfeiting accuracy, by judiciously hybridizing physics and data, calibrating where needed, constraining where useful, and explaining where required.

In summary, the thesis proposes and applies a unified methodological framework that re-anchors model selection and design around interpretability, transparency, and

explainability, and shows, through cross-domain studies, how hybrid paradigms can satisfy modern scientific, industrial, and financial requirements for trustworthy prediction and decision-making.

The rest of the thesis will guide you through this methodological–application journey across the following chapters:

- **Chapter 2 - Modeling Framework: Knowledge, Data, Interpretability and Explainability** introduces the modeling framework, presenting the taxonomy of white-, knowledge-driven and data-driven grey-, and black-box approaches, and positioning them in terms of physical knowledge, data-driven parameter estimation, and interpretability requirements. This chapter provides the theoretical foundation and a novel taxonomic perspective for the case studies developed in the following chapters.

- **Chapter 3 - Smart Sensors based on Green Materials: From White- to Data-Driven Models** presents the development of bacterial cellulose-based smart sensors. It investigates a first-principle model implemented via a multiphysics finite-element (FEM) approach, a simplified FEM models, and data-driven black-box formulations, discussing trade-offs in terms of accuracy, complexity, and sustainability.

- **Chapter 4 - Soft Sensors for Industrial Applications: Linear Regression, Dynamic Mode Decomposition and Symbolic Regression** focuses on industrial process monitoring. It applies state-space models, in particular multi-step-ahead Hankel dynamic mode decomposition with control (MSA-HDMDc) based on the Koopman operator and SR, to the Sulfur Recovery Unit (SRU) and to the Debutanizer Column (DC) case, combining data-driven discovery of equations with interpretability.

- **Chapter 5 - Vehicle-to-Grid Availability Prediction: From Data-Driven Grey State-Space to Explainable AI** addresses forecasting of aggregated available capacity (AAC) in V2G systems. It compares state-space data-driven representation (HDMDc) and linear regression (ARX/FIR) with nonlinear data-driven methods (LSTM, NARX), and incorporates XAI techniques such as SHapley Additive exPlaintion (SHAP) to enhance interpretability in decision support for energy providers.

- **Chapter 6** concludes the thesis by synthesizing cross-domain insights on interpretable modeling, *placing all implemented models across the three application fields within the proposed two-axis taxonomy*, and outlining future research directions in hybrid modeling for dynamical systems.

This organization emphasizes a coherent narrative that progresses from material-level smart sensing, to industrial process monitoring, and finally to large-scale energy systems. Each chapter exemplifies the interplay between modeling accuracy, interpretability, and domain relevance, thus consolidating the thesis contribution to the field of interpretable dynamical system modeling.

Chapter 2

Modeling Framework: Knowledge, Data, Interpretability and Explainability

2.1 Introduction

The modeling of engineering and industrial processes is traditionally built upon two primary approaches: white-box modeling and black-box modeling. White-box modeling constructs models based on established physical relations and deterministic equations, emphasizing model creation using prior knowledge. In contrast, black-box modeling employs parametric models calibrated with real-world data obtained from the process, using experimental data as the primary information source. Instead of relying solely on a single source of knowledge, grey-box identification combines both prior knowledge and experimental data, merging the strengths of the two approaches (Bohlin, 2006).

This widely accepted definition (Ljung, 1999) serves as a foundation for classifying models. However, with the continuous growth of machine learning (ML) techniques, particularly deep neural networks, the traditional distinction has shifted from the source of knowledge toward the aspects of explainability and transparency (Pintelas, Livieris, and Pintelas, 2020). The internal mechanics and reasoning of a model decision-making process have become more critical for distinguishing between white- and black-box models than the original knowledge source (Shakerin and Gupta, 2020). This transformation reflects a change in perspective, moving from model development to its transparent and practical application in industrial contexts. While one definition is based on the way a model is constructed, the other focuses on the requirements that the model must satisfy to be usable (Wiemer et al., 2023).

An emerging trend shows a transition from black-box models toward white-box models in sectors where decisions are critical, such as healthcare, finance, and defense (Aiosa, Palesi, and Sapuppo, 2023; Gao et al., 2024; Wood, 2024). This movement highlights the importance of developing transparent white-box models and integrating white- and black-box methods to ensure that results can be effectively communicated and justified to decision-makers (Rudin, 2019). Nevertheless, this tendency focuses exclusively on the explainability of the model rather than its information source (Loyola-González, 2019).

An opposite development is also noticeable. With the ongoing increase in computational power, the complexity of physical simulations has also risen (Mittal and Tolk, 2019), leading to white-box models that, although based on prior knowledge, lack practical explainability due to their high complexity.

This paradigm shift significantly impacts the selection, development, and deployment of models in engineering and industrial applications. As explainability and transparency gain importance, models traditionally classified as white-box or black-box are being re-evaluated based on their capacity to deliver interpretable and actionable insights. White-box models, although grounded in physical principles, may lose usability when their complexity obscures their inner workings. Conversely, black-box models, such as deep neural networks, often achieve high predictive accuracy but struggle to meet the increasing demand for interpretability in decision-critical applications.

These developments highlight the growing importance of hybrid approaches, such as grey-box models, which aim to balance the transparency of white-box techniques with the adaptability and data-driven strengths of black-box approaches. However, as the criteria distinguishing white- and black-box models evolve, the classification of various grey-box models becomes increasingly ambiguous.

This growing ambiguity calls for a more refined framework capable of systematically balancing interpretability and complexity. As hybrid strategies blur the lines between traditional modeling categories, a structured classification framework helps guide model selection and application. Such a framework supports trade-off evaluation, strengthens decision-making, and fosters trust in ML systems. Moreover, it promotes innovation by identifying opportunities to integrate complementary features, ultimately improving communication and alignment during model development.

To orient the reader, the remainder of the chapter is organized as follows. Section 2.2.1 formalizes model development as a two-phase process (model-class selection and data fitting) and instantiates it for white-, black-, and grey-box approaches. Section 2.3 introduces cross-cutting design requirements (complexity, transparency, interpretability, explainability, domain knowledge, computational effort, and realism). Section 2.4 proposes the dual-axis taxonomy and distinguishes *knowledge-driven grey* from *data-driven grey*, including pathways for moving from white→grey and black→grey. Section 2.5 operationalizes placement on the taxonomy via quantitative and qualitative metrics detailing the qualitative rubric). The chapter closes with conclusions that link this framework to the application chapters.

2.2 Modeling Approaches by Knowledge Source and Model Class

This section sets the groundwork for the taxonomy by first stating a general formulation of model development and then contrasting how white-, black-, and grey-box approaches instantiate that formulation.

2.2.1 General Formulation of Model Development

Regardless of the specific "color" category assigned to a "box", Ljung, 1999 distinguishes two main stages in the creation of a model: the *model class selection phase*, where the model class is defined, and the *fitting phase*, where the internal parameters are adjusted using data.

Both model specification and parameter fitting are algorithmic procedures, which can be expressed mathematically as:

$$M(x_N, t, \theta) \rightarrow z(t|\theta), \tag{2.1}$$

$$\min_{\theta} L[y_N, z_N(\theta)], \tag{2.2}$$

where M denotes the model class, which contains a set of adjustable parameters θ, and L is a loss function minimized by estimating θ to improve the correspondence between the model output z and the measured data y_N, for a given input x over time t. This formalism provides a unified basis from which the different model types can be differentiated.

2.2.2 First-Principle (White-Box) Modeling

Following Ljung's framework (Ljung, 1999), white-box models follow the same two phases: modeling and fitting. Their key distinction lies in the modeling phase, where M is formulated based on prior theoretical knowledge, such as first-principles equations or domain-specific rules. The structure of the model often makes the parameters θ directly interpretable in terms of physical properties or system dynamics.

Although the fitting phase is necessary, it is typically straightforward: θ is optimized to minimize the loss function L, aligning the predicted output $z(t|\theta)$ with experimental observations y_N. This optimization ensures data consistency without undermining the inherent transparency of the model. This property is the reason white-box models are often considered transparent and interpretable. However, the reliance on explicit mathematical representation can make them computationally demanding (Ralph et al., 2021). A classical example of a white-box model is one described by differential equations.

2.2.3 Data-Driven (Black-Box) Modeling

In a similar way, black-box models adhere to the modeling–fitting structure, but their main emphasis is placed on the fitting phase. In this category, the parameters θ are estimated almost entirely from empirical data, with the objective of producing the closest agreement between model outputs and measured responses.

The modeling phase in black-box approaches is minimal, often limited to selecting a general-purpose model class M that has no explicit connection to the underlying physical or logical nature of the system. Common examples include neural networks (Dayhoff and DeLeo, 2001), support vector machines (Dinov, 2018), and linear regression models (Guidotti et al., 2018), which are versatile enough to approximate complex nonlinear mappings.

In these models, the parameter fitting process is central, adjusting θ to minimize L so that $z(t, \theta)$ reproduces y_N. As the number of adjustable parameters grows, interpretability and transparency usually decrease, even though predictive power may increase.

2.2.4 Hybrid (Grey-Box) Modeling

As described in the introduction, grey-box models serve as a middle ground between white-box and black-box approaches. In line with Ljung's general modeling process, grey-box models define M to combine known physical principles with parameters θ estimated from real-world measurements.

The purpose of grey-box modeling is to mitigate the limitations of each pure approach: it compensates for the low transparency of black-box models (Loyola-González, 2019) and reduces the computational burden often associated with white-box models (Li et al., 2021b; Li et al., 2021c).

Depending on the application, grey-box models can be implemented either as sequential or parallel combinations of white-box and black-box structures (Yang et al., 2017; Sohlberg and Jacobsen, 2008). However, this hybrid configuration requires additional effort in both model formulation (modeling) and parameter calibration (fitting).

2.3 Novel Requirements for Model Design

Beyond the traditional modeling approach, contemporary deployments impose *application-level* requirements that are not fully captured by model class alone. We therefore formalize these *novel applicative requirements* as explicit design objectives that cut across white-, grey-, and black-box approaches, and detail them below.

2.3.1 Complexity

Complexity is a relative property of a modeling approach characterized by attributes such as the number of tunable parameters, the depth of representational abstractions, the degree of nonlinearity, and the extent of variable coupling that can yield emergent behaviors resistant to straightforward human interpretation. In general, greater complexity confers greater capacity: more complex models are typically better able to capture intricate patterns, represent high-dimensional dependencies, and address problems that simpler models struggle to solve.

2.3.2 Transparency

Transparency in modeling refers to the extent to which model creation, parameter extraction, and output generation can be understood and explained. It includes three sub-aspects: model transparency, design transparency and algorithmic transparency (Roscher et al., 2020). While some methods, like kernel-based models (Hofmann, Schölkopf, and Smola, 2008), are often transparent in structure, design choices may lack clarity. Neural networks, despite clear input-output structures, involve heuristic design and hyperparameter tuning, reducing transparency.

2.3.3 Interpretability

In the context of black- and white-box models interpretability refers to the ability to present the internal properties or decisions of a model in understandable terms to humans (Roscher et al., 2020). It involves mapping abstract model concepts, such as predictions, into forms comprehensible to users. For black-box models, interpretability often relies on post hoc methods, such as surrogate models (Ribeiro, Singh, and Guestrin, 2016), feature importance analysis (König et al., 2021), or visual tools like saliency maps (Hohman et al., 2019). White-box models, due to their inherent transparency, facilitate interpretation by design. Achieving interpretability often requires data involvement and may depend on heuristic approaches when algorithmic explanations are complex or infeasible.

2.3.4 Explainability

In modeling, explainability refers to the ability to provide clear and understandable reasons or justifications for a model's predictions or decisions. It builds on interpretability by contextualizing model behavior with domain knowledge. While

interpretability focuses on understanding model components, explainability emphasizes clarifying the reasoning behind decisions, often combining interpretation tools, transparency, and domain-specific insights to provide meaningful explanations (Roscher et al., 2020).

2.3.5 Incorporation of Domain Knowledge

Incorporating domain or theoretical knowledge into modeling enhances explainability, improves performance, and helps address small data scenarios. It encompasses expertise or information specific to a field, ranging from mathematical equations and rules in the sciences to engineering workflows, world knowledge, or expert intuition. Integration involves three key aspects: the type of knowledge, its representation and transformation, and its application in the ML pipeline (Rueden et al., 2021). This can occur during data preparation, hypothesis design, training, or evaluation. Leveraging domain knowledge aligns models with real-world applications, making them more interpretable and effective.

2.3.6 Computational Effort

Computational effort or computational complexity impacts both white-box and black-box models differently. The life-cycle of a model can be roughly separated into two phases: the development phase, where a model is developed, and the application phase, where a finalized model is used. Black-box models generally require significant resources in the development phase during data fitting and optimization, whereas white-box models have a tendency to rather require more in the application phase (Boos et al., 2023). Nonetheless, although the processing power of computers grows continuously, computational effort and complexity remain critical considerations (Shahcheraghian, Madani, and Ilinca, 2024).

2.3.7 Model Realism and Fidelity

The level of realism refers to how accurately the model reflects the underlying system. White-box models, while grounded in physical laws and theoretical principles, often rely on simplifications, which limit their level of realism. However, as the complexity of a white-box model increases and the number of model parameters grows, its ability to capture real-world behaviors improves. This improvement, nevertheless, comes with the drawback of a higher computational complexity (Fujimoto et al., 2017). In contrast, black-box models achieve realism by leveraging empirical data, allowing them to model complex systems effectively. However, this data-driven approach may introduce overfitting or fail to incorporate underlying causal relationships, reducing interpretability.

2.3.8 Transferability (External Validity & Portability)

We define *transferability* as the ability of a model trained on a *source* setting (site, user population, season, device, or operating regime) to sustain performance and calibrated uncertainty in a *target* setting subject to distribution shift, i.e., covariate, label, or prior shift (Quiñonero-Candela et al., 2009). Transferability is distinct from in-distribution generalization: it concerns performance under domain change and the effort required to adapt (fine-tune) the model (Pan and Yang, 2010; Weiss, Khoshgoftaar, and Wang, 2016). Approaches that encode stable mechanisms—via

physics-informed inductive biases or causal invariances—tend to transfer more reliably (Karniadakis et al., 2021; Pearl, 2009); conversely, purely data-driven models may require domain adaptation or domain generalization techniques to mitigate dataset bias (Torralba and Efros, 2011; Wang et al., 2022). Transferability interacts with our other requirements: lower complexity and modularity curb overfitting; transparency and interpretability facilitate shift diagnosis and safe adaptation; incorporation of domain knowledge promotes invariance; and computational effort trades off between re-training costs and adaptation gains. In practice, we assess transferability via target-domain error and calibration, *transfer ratio* (target vs. source performance), and checks for negative transfer, using the OOD metrics in §2.5 (e.g., E_{OOD}).

2.4 A Dual-Axis Taxonomy for Models

With the rising complexity of models in engineering needed to correctly map reality, a clear tendency is emerging towards hybrid modeling such as grey-box models. The number of possibilities to numerically model reality is also growing with further research, offering multiple permutations to combine different modeling approaches. Due to the limits of white- and black-box models, their definition shifted from their modeling approaches to their application requirements. Both white-box and black-box models have inherent deficits that limit their application in certain scenarios, leading to a growing preference for hybrid grey-box approaches. Generally speaking, the tendency towards grey-box models stems from the goal to eliminate at least one modeling weakness by incorporating one or more of the presented requirements on modeling (see Section 2.3). In that regard a recommendation for action does not exist. At present, there is no standardized procedure that guides a researcher in strategically extending a given base model to address specific weaknesses. Nonetheless, in the traditional sense there are essentially two paths to model improvement: moving from white-box to grey-box, and moving from black-box to grey-box. Here a novel terminology is presented; it unites the modern point of view to modeling with the traditional one. This terminology includes a subdivision of two different types of grey-box models. The goal of this proposed view is to further break down these three basic terms by their relation to each other.

The terminology proposed here aims to depict white- and black-box models on a dual-axis scale, as shown in Figure 2.1. The horizontal axis represents the complexity of a model as defined in Section 2.3. The vertical axis represents the two phases of modeling (see Section 2.2.1) on a continuous scale. The upper values represent the modeling phase, while the lower values the fitting phase. It reflects the interpretability, to which extent the internal workings of the model and its decision-making processes can be understood by humans since it is based on domain knowledge. If the proposed model is read as a geographical map, then the following accounts: the more north a model is located the more it is defined by the modeling phase, while the more south it is, the more it is defined by the fitting phase. Likewise, on the complexity axis, the more west a model is located the easier it is to be fully understood by a human, while the further east it moves, the more intricate it becomes.

This mapping results in four quadrants. The first quadrant represents a true white-box model: it is a knowledge-based modeling approach, transparent, explainable, interpretable thus easy to understand for a human. On the contrary, in the fourth quadrant, the true black-box model is positioned: it is a modeling approach based on empirical data fitting, highly complex and opaque but also highly powerful in their ability to map non-linearity.

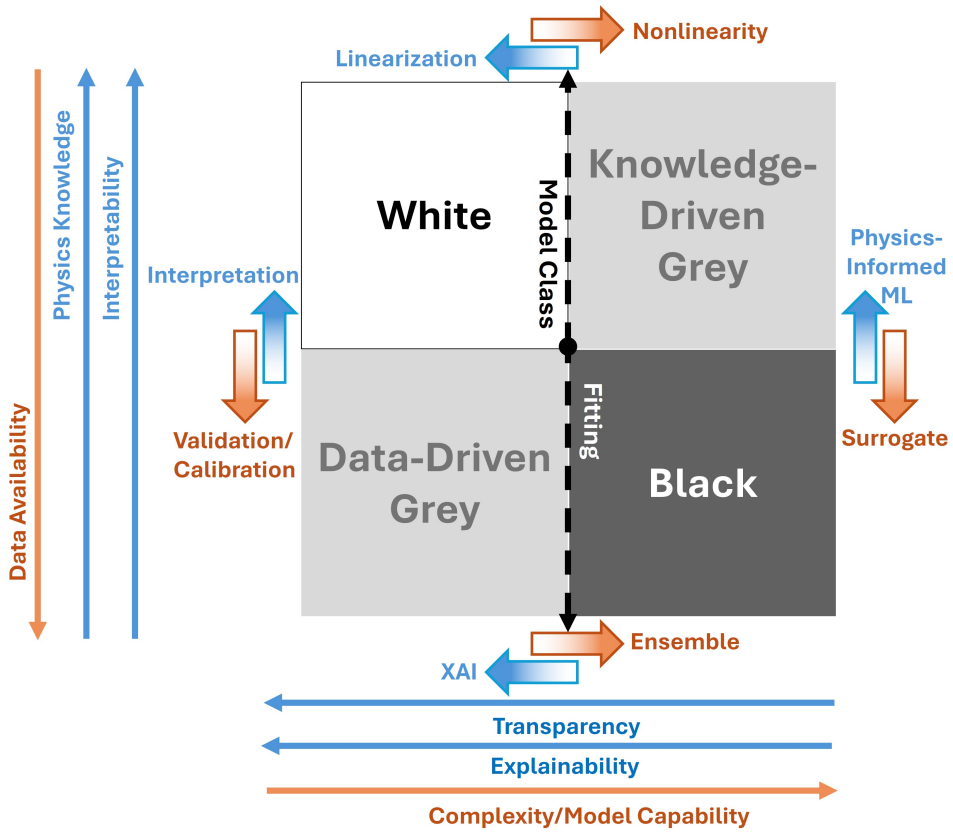


FIGURE 2.1: Placement of the implemented models within the dual-axis taxonomy.

The second and third quadrants are both different types of grey-box models. The second quadrant, which, compared to the first quadrant, increases in complexity but remains consistent in interpretability: it is called knowledge-driven grey. By increasing the complexity and thus model capability, this kind of grey-box model loses transparency and explainability. However, in its core, it remains a model, which is developed by the modeling phase.

In contrast, the third quadrant depicts a model, which, identical to a true black-box model, is based on the fitting phase of model design but equivalent in transparency and explainability with white-box models. We propose the name data-driven grey. Although developed with mainly empirical data, a data-driven grey model maintains sufficient transparency to ensure that its decision-making process is comprehensible to a human observer.

2.4.1 From White to Grey

Transitioning from white-box to grey-box involves integrating data-driven components. A first pathway is *calibration*, i.e., using data to estimate uncertain parameters in an otherwise physics-specified model, which shifts the approach toward *data-driven-grey* because the structure remains mechanistic while parameters are data-identified (Mostafavi et al., 2018). By contrast, different pathways cross the *knowledge-driven grey* quadrant. When modeling complexity, due mainly to nonlinearities, makes it impractical to encode intricate constitutive or closure relations, embedding localized black-box approximators within a physics scaffold preserves the governing balances and state semantics while learning only the intractable mappings (Shahcheraghian,

Madani, and Ilinca, 2024). Further, for computational efficiency, replacing selected submodels, or, in some cases, the full solver, with surrogates trained on high-fidelity simulations (or tightly constrained by physical laws) accelerates inference yet keeps the mechanistic architecture, interfaces, and constraints explicit, thus remaining in knowledge-driven grey (Stöcker et al., 2023; Böttcher et al., 2021a; Böttcher et al., 2021b; Alizadeh, Allen, and Mistree, 2020; Hashemi, Beheshti, and Mohammadi, 2025).

2.4.2 From Black to Grey

Transitioning from black-box to grey-box involves integrating interpretability, explainability and transparency by embedding reasoning into the model structure. One widely known approach aiming for improved transparency is eXplainable Artificial Intelligence (XAI) (Cheng et al., 2025; Rane and Paramesha, 2024). It provides insights into the model decision-making, bridging the gap between the model opaque internal workings and end-user interpretability. Another approach to enhance interpretability is by incorporating domain knowledge, which is aimed for in Physics-Informed ML (PIML) (Xu et al., 2023). PIML integrates physical laws, constraints, and governing equations directly into the black-box model. By embedding physical laws and domain insights, PIML can reduce the dependence on large datasets, thereby moving the model upward within the proposed taxonomy, improving generalization, and minimizing false discoveries, making it particularly suitable for engineering applications where data may be sparse or costly to obtain (Mackay and Nowell, 2023). In this direction linearization might further improve the operator understanding of the underlying processes.

2.5 Metrics for the Dual-Axis Taxonomy

Modern deployments require models to be placed on the proposed taxonomy with evidence, not opinion. We therefore pair *quantitative* metrics—numerical indicators that can be computed and compared across models—with *qualitative* rubrics—expert judgements capturing governance, documentation, and human factors. Quantitative scores enable reproducible mapping (e.g., complexity, capacity, generalization, physics alignment), while qualitative rubrics capture aspects that resist formalization (e.g., safety processes, human-simulatability). Used together, they provide a balanced basis for selection and auditing.

2.5.1 Quantitative Metrics

i) Model Complexity / Capacity

Parameter count and sparsity.

$$P = \|\boldsymbol{\theta}\|_0, \qquad S_0 = 1 - \frac{\|\boldsymbol{\theta}\|_0}{P_{\max}}, \qquad S_1 = \frac{\|\boldsymbol{\theta}\|_1}{\|\boldsymbol{\theta}\|_{1,\max}}. \tag{2.3}$$

Here, $\boldsymbol{\theta} \in \mathbb{R}^{P_{\max}}$ is the parameter vector, P is the number of nonzero parameters (ℓ_0 pseudo-norm), S_0 and S_1 are sparsity indicators normalized by admissible maxima $P_{\max}$ and $\|\boldsymbol{\theta}\|_{1,\max}$, respectively (Tibshirani, 1996).

Rademacher complexity (capacity indicator).

$$\mathfrak{R}_n(\mathcal{F}) = \mathbb{E}_{\boldsymbol{\sigma},\mathbf{x}}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i f(x_i)\right]. \tag{2.4}$$

$\mathcal{F}$ is a function class, $\{x_i\}_{i=1}^n$ a sample of size n, and $\{\sigma_i\}_{i=1}^n$ i.i.d. Rademacher variables with $\mathbb{P}(\sigma_i = \pm 1) = \frac{1}{2}$; the expectation is over data $\mathbf{x}$ and signs $\boldsymbol{\sigma}$ (Bartlett and Mendelson, 2003; Vapnik, 1995).

Parsimony–fit criteria (AIC/BIC).

$$\mathrm{AIC} = 2k - 2\ln\hat{L}, \qquad \mathrm{BIC} = k\ln n - 2\ln\hat{L}. \tag{2.5}$$

k is the number of free parameters, $\hat{L}$ the maximized likelihood, and n the number of observations (Akaike, 1974; Schwarz, 1978).

Minimum Description Length (MDL).

$$\mathrm{MDL} = L(\text{model}) + L(\text{data} \mid \text{model}). \tag{2.6}$$

$L(\cdot)$ denotes code length in bits for a chosen coding scheme (Rissanen, 1978).

Computational effort / footprint.

$$E = P_{\text{avg}} \cdot t, \qquad \text{report (FLOPs, } t, M, E). \tag{2.7}$$

P_{avg} is average power draw, t wall-clock time, M peak memory, and E energy; FLOPs quantify arithmetic cost (Strubell, Ganesh, and McCallum, 2019; Schwartz et al., 2020).

ii) Transparency / Interpretability / Explainability

Like many other aspects in the field of model interpretability, measuring the quality of an explanation is still an open question, as the definition of interpretability itself lacks mathematical rigour (Marcinkevičs and Vogt, 2023). Several possibilities have been considered in the literature, such as including humans in the loop to measure the quality of the explanation. However, this approach is inefficient as it would limit the scalability of the approach itself, and most importantly, prone to bias due to the mere presence of humans in the evaluation process. Some metrics have been proposed to evaluate the quality of the explanation with mathematical rigour by focusing on the robustness of the explanation itself (Aldeia and De França, 2022). The concept of robustness as a metric revolves around the idea that the explanation provided by a local explainer should not change with small perturbations of the explained observation. An additional metric called adherence is here proposed, which aims to precisely define the similarity between a ground-truth equation and the equation provided by a symbolic regressor or to compare to candidate models.

Global surrogate fidelity.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(f(x_i) - g(x_i)\right)^2}{\sum_{i=1}^{n}\left(f(x_i) - \bar{f}\right)^2}. \tag{2.8}$$

f is the reference (complex) model, g a transparent surrogate (e.g., linear or tree), and $\bar{f} = \frac{1}{n}\sum_i f(x_i)$; higher R^2 indicates better global explainability via surrogates (Ribeiro, Singh, and Guestrin, 2016).

Completeness (Integrated Gradients).

$$\sum_{j=1}^{d} \Phi_j(x) = f(x) - f(x_0), \tag{2.9}$$

with baseline x_0; equality defines the IG *completeness* axiom (Sundararajan, Taly, and Yan, 2017).

Expression complexity (closed-form models).

$$C_{\text{expr}} = \#\text{nodes (or terms)}, \qquad D_{\text{tree}} = \text{tree depth}. \tag{2.10}$$

C_{expr} and D_{tree} summarize symbolic or tree model size; lower values indicate simpler, more readable structures (Udrescu and Tegmark, 2020).

Structural compliance (monotonicity).

$$M = \frac{1}{|\mathcal{P}|} \sum_{(x,x')\in\mathcal{P}} \mathbb{1}\{x \preceq x' \Rightarrow f(x) \leq f(x')\}, \tag{2.11}$$

where $\mathcal{P}$ is a set of ordered input pairs under partial order $\preceq$; $M \in [0,1]$ measures monotonicity adherence.

Dimensional consistency (SR).

$$D = \frac{\#\text{dimensionally valid terms}}{\#\text{total terms}}, \tag{2.12}$$

with $D \in [0,1]$; larger D indicates better unit consistency in symbolic expressions (Udrescu and Tegmark, 2020).

Stability (S) Stability can be defined as the degree to which the local explanation for a given point changes compared to its neighbours. A high value of stability indicates that a small change in the explained feature causes a significant change in the explainer, which means that the explanation is not reliable because it is not locally stable.

$$S(\hat{f}, \Psi, \boldsymbol{x}) = E_{x'\sim N_x}\left(||\Psi(\hat{f}, \mathbf{x}) - \Psi(\hat{f}, \mathbf{x}')||_2^2\right) \tag{2.13}$$

Eq. 2.13 evaluates the mean distance between the original explanation Ψ and the one calculated using the members of a sample neighbourhood $N_{\boldsymbol{x}}$. The neighbourhood consists of samples obtained by perturbing the original sample and can therefore be generated in different ways. One way to achieve this is to use a normal distribution centered in $\mathbf{x}$ to generate the neighbourhood. However, a multivariate normal distribution may be more representative of the training data, as the input space may be unevenly distributed, especially in high-dimensional cases.

Infidelity (INFD) The measure of explanation infidelity focuses on the idea that given the subset of most relevant feature explanations, the explainer should ascribe high values even after a significant perturbation is applied.

$$INFD(\hat{f}, \Psi, \boldsymbol{x}) = E_{\boldsymbol{p} \in \mu_{\boldsymbol{x}}} \left((\boldsymbol{p}^T \Psi(\hat{f}, \boldsymbol{x}) - (\hat{f}(\boldsymbol{x}) - \hat{f}(\boldsymbol{x} - \boldsymbol{p})))^2 \right) \tag{2.14}$$

Analogous to what was previously stated for stability, the significant perturbation $\mathbf{p}$ should be picked from a suitable set $\mu_{\boldsymbol{x}}$ which contains the perturbations generated from the instance $\boldsymbol{x}$ considering the aforementioned multivariate normal distribution and a suitable value of η to ensure that the perturbation is significant enough.

Jaccard stability (J_k) The Jaccard index is a popular metric for expressing the similarity between two sets. Formally, it is defined as the cardinality of the intersection over the cardinality of the union. This metric can be extended in terms of explainability by considering the k most important measured features from the explanation Ψ, ordered by their relevance, and registering their change within the neighbourhood. This metric does not provide any information about the gain or loss of importance of individual characteristics. Therefore, the index should be calculated for multiple values of k for the index to be truly meaningful.

$$J_k(\hat{f}, \Psi, \boldsymbol{x}) = E_{\boldsymbol{p} \in \mu_{\boldsymbol{p}}} \left(\frac{|\tilde{\Psi}(\boldsymbol{x})_k \cap \tilde{\Psi}(\boldsymbol{x'})_k|}{|\tilde{\Psi}(\boldsymbol{x})_k \cup \tilde{\Psi}(\boldsymbol{x'})_k|} \right) \tag{2.15}$$

Adherence (Adh) The similarity metrics in the literature focus on the similarity of strings. This hardly applies to the field of mathematics, as functions and numbers have a deep meaning that needs to be treated properly and not just as a composition of characters. Also, equations can be expressed in many equivalent forms, and numbers can be rounded or approximated, so all these special cases need to be considered in the formulation. Therefore, a new similarity index named adherence has been introduced as follows:

$$Adh(E, \hat{E}) = \left(\frac{(\boldsymbol{\Phi}_E \cap \boldsymbol{\Phi}_{\hat{E}}) \cdot \boldsymbol{w}_{E,\hat{E}}}{|\boldsymbol{\Phi}_E \cup \boldsymbol{\Phi}_{\hat{E}}|} \right) \tag{2.16}$$

where $\boldsymbol{\Phi}_E$ and $\boldsymbol{\Phi}_{\hat{E}}$ are the set of nodes forming the ground truth equation and the regressed equation, respectively, while $\boldsymbol{w}_{E,\hat{E}}$ is a vector of weights for each common node in both equations, representing the difference between the original nodes coefficients w and the regressed ones w' for each common node:

$$\boldsymbol{w}_{E,\hat{E}} = \exp\left(-\frac{(w - w')^2}{2l^2} \right) \tag{2.17}$$

Adopting this index, the similarity between equations can be calculated as a weighted sum of common terms, where the role of weight is crucial to define the distance between similar nodes. This definition achieves both the formalization of the distance between terms approximated by the search process, but also defines an active penalty for models with high size losses l, which in this case will be the determination coefficient R^2 score, to bound the variance of the Gaussian distribution between 0 and 1. Using a tree representation for the equation, the score can be calculated recursively on the tree branches, evaluating the similarity of a similar subtree in the regressed equation. Furthermore, the applicability of the metric is limited to

situations where the underlying equation is either already known or where there is some prior knowledge of the system that facilitates the formulation of a ground truth equation, such as in gray-box approaches.

iii) Physics Knowledge / Constraint Alignment

Physics-loss ratio (PINNs / constrained training).

$$\rho_{\text{phys}} = \frac{\mathcal{L}_{\text{phys}}}{\mathcal{L}_{\text{phys}} + \mathcal{L}_{\text{data}}}, \qquad \mathcal{L}_{\text{phys}} = \frac{1}{n}\sum_{i=1}^{n} \left\| \mathcal{N}[u_{\boldsymbol{\theta}}]\,(x_i) \right\|_2^2. \tag{2.18}$$

$\rho_{\text{phys}} \in [0,1]$ quantifies the share of physics enforcement in training; $\mathcal{N}[\cdot]$ is the PDE residual operator, $u_{\boldsymbol{\theta}}$ the model solution, and $\{x_i\}$ collocation points (Raissi, Perdikaris, and Karniadakis, 2019).

Constraint violation norm.

$$V_{\text{con}} = \left\| \mathbf{c}(x, \boldsymbol{\theta}) \right\|_2, \tag{2.19}$$

where $\mathbf{c}$ stacks equality/inequality constraint residuals; lower V_{con} indicates better feasibility.

Global sensitivity.

$$S_i = \frac{\text{Var}_{X_i}\big(\mathbb{E}[Y \mid X_i]\big)}{\text{Var}(Y)}, \qquad S_i^{\text{Tot}} = 1 - \frac{\text{Var}_{X_{\sim i}}\big(\mathbb{E}[Y \mid X_{\sim i}]\big)}{\text{Var}(Y)}. \tag{2.20}$$

$Y = f(X)$ is the scalar output, $X = (X_1, \ldots, X_p)$ the inputs, S_i the first-order and S_i^{Tot} the total-effect sensitivity indices (Saltelli et al., 2008). Morris screening reports mean absolute elementary effects μ^* and dispersion σ (Morris, 1991).

Profile likelihood (identifiability).

$$\text{PL}(\theta_j) = \max_{\boldsymbol{\theta}_{-j}} \ell(\boldsymbol{\theta}_{-j}, \theta_j), \tag{2.21}$$

with log-likelihood $\ell(\cdot)$, scalar θ_j, and nuisance parameters $\boldsymbol{\theta}_{-j}$; curvature of PL informs practical identifiability (Raue et al., 2009).

iv) Generalization / Calibration / Robustness

Data efficiency.

$$E(n) = a\, n^{-\alpha} + b, \tag{2.22}$$

where $E(n)$ is generalization error at sample size n, $\alpha > 0$ the data-efficiency slope, and $a, b \geq 0$ constants.

Generalization gap and shift.

$$\Delta = E_{\text{test}} - E_{\text{train}}, \qquad \Delta_{\text{OOD}} = E_{\text{OOD}} - E_{\text{ID}}, \tag{2.23}$$

with E_{train}, E_{test} in-distribution errors and E_{ID}, E_{OOD} errors under in-/out-of-distribution shifts.

Probabilistic calibration.

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|, \qquad \mathrm{BS} = \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left(p_{ik} - y_{ik}\right)^2. \tag{2.24}$$

ECE is the expected calibration error over bins $\{B_m\}$, with $\mathrm{acc}(\cdot)$ empirical accuracy and $\mathrm{conf}(\cdot)$ mean confidence; BS is the Brier score for K-class probabilistic predictions p_{ik} and one-hot labels y_{ik} (Niculescu-Mizil and Caruana, 2005; Brier, 1950).

Distribution-free coverage.

$$\hat{c} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{y_i \in \Gamma_\alpha(x_i)\}, \tag{2.25}$$

where $\Gamma_\alpha(x)$ is a $(1-\alpha)$ predictive set and $\hat{c}$ the empirical coverage; valid conformal predictors target $\hat{c} \approx 1 - \alpha$ (Angelopoulos and Bates, 2023).

v) Performance indicators

To evaluate the accuracy of the models, the following performance indicators are used:

Root Mean Square Error (RMSE) The RMSE measures the standard deviation of the residuals (prediction error). It is defined as:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{2.26}$$

where y_i stands for the actual values, $\hat{y}_i$ for the predicted values and n for the number of observations. Lower RMSE values indicate better model performance.

Mean Absolute Error (MAE) The MAE calculates the average absolute difference between the actual and predicted values. It is given by:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{2.27}$$

MAE is an intuitive measure of prediction accuracy, with lower values indicating better model performance.

Coefficient of Determination (R^2) The R^2 value indicates how well the model explains the variance in the data. It is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{2.28}$$

where $\bar{y}$ is the mean value of the actual output. A value close to 1 indicates a good fit of the model, while a value close to 0 indicates poor predictive performance.

Notes. All quantitative scores should be complemented by qualitative rubrics (e.g., human-simulatability, documentation quality, governance readiness) to form a balanced evaluation.

2.5.2 Qualitative Metrics

- **Domain-knowledge strength** (None/Weak/Moderate/Strong): extent of physics/theory encoded.
- **Parameter semantics** (Low–High): share of parameters with direct physical meaning.
- **Documentation & code transparency**: clarity of assumptions, data lineage, and implementation.
- **Human-simulatability**: ability for an expert to trace predictions on small inputs within a fixed time budget.
- **Replicability**: ease of exact reproduction with artifacts and seeds.
- **Safety / constraint compliance**: presence of hard constraints, guards, or certified bounds.
- **Governance readiness**: audit trails, versioning, change logs for regulated contexts.

2.6 Conclusions

This chapter introduced a dual-axis taxonomy and a set of metrics that jointly frame model selection around interpretability, transparency, and explanatory adequacy. We clarified the grey-box split into knowledge-driven and data-driven variants and outlined concrete levers for moving across the map. The next chapters apply this scaffold—placing each model within the taxonomy and using the proposed metrics to justify design choices.

Chapter 3

Smart Sensors based on Green Materials: From White- to Data-Driven Models

3.1 Introduction

This chapter investigates bacterial–cellulose (BC) ionic transducers from a modeling perspective anchored to the thesis taxonomy (see Section 2.4). We first formulate a *first-principles* partial-differential-equation (PDE) model for mechanoelectrical transduction (White-Box). We then implement it via the *finite element method* (FEM), which retains the physics-first structure yet requires calibrated parameters, boundary conditions, meshing choices, and time-integration settings. Finally, we identify *data-driven* input–output models from experiments. In linear and nonlinear regressions as Finite Impulse Response (FIR), AutoRegressive with eXogenous Input(ARX), Nonlinear FIR (NFIR) and Nonlinear ARX (NARX), parameters are fit from data while the mapping remains transparent. For completeness, we include a deep recurrent baseline as Long Short-Term Memory(LSTM), which delivers high capability at the expense of intrinsic transparency. Throughout, we emphasize interpretability, transparency, and explainability as primary design criteria in addition to predictive accuracy and computational efficiency.

The remainder of the chapter is organized as follows. Section 3.2 introduces smart materials and emphasizes how application context shapes requirements, motivating rigorous modeling to ensure reliable, tunable, and deployable performance. It then outlines the research gap, namely, the absence of unified and interpretable modeling approaches tailored to these application environments, setting the stage for the framework developed in this thesis. Section 3.4 details materials, device geometry, and the measurement setup. Section 3.5 presents the continuum framework for coupled mechanics–ion transport–electrostatics, and in particular Section 3.5.4 its FEM realization. Section 3.5.5 compares simulations with experiments and analyzes position-dependent response and ionic currents. Section 3.6 develops and benchmarks data-driven models (FIR/ARX,NFIR/NARX), LSTM for 1-step and free-run prediction. Section 3.7 closes with a synthesis, including a partial taxonomy figure that situates each model used in this chapter.

3.2 Smart Materials Background

Electroactive polymers (EAPs) have attracted growing interest in industrial applications due to their potential to enable flexible, lightweight, and tunable electromechanical devices (Wang et al., 2016). Their ability to convert electrical energy into

mechanical motion (electromechanical transduction) and mechanical deformation into electrical signals (mechanoelectrical transduction) makes them suitable for applications ranging from biomedical devices and soft robotics to aerospace and consumer electronics (Park et al., 2022; He et al., 2019a; Jiang et al., 2023; El-Atab et al., 2020).

In this context, recent advances in soft robotics underscore the increasing role of intelligent polymer-based actuators in multifunctional applications such as drug delivery, wearable systems, and biomimetic motion, as reviewed in (Sarker, Ul Islam, and Islam, 2024; Cui et al., 2020). These materials provide unique opportunities for developing compliant and programmable robotic systems, with ionic polymer–metal composites (IPMCs) standing out as promising electrically driven actuators due to their rapid response and mechanical robustness (Jiang et al., 2023).

Among EAPs, ionic electroactive polymers (IEAPs), such as IPMCs and carbon-polymer composites, have been extensively studied due to their large deformation under low voltage and bio-inspired actuation capabilities. While actuation electromechanical transduction in IEAPs has been extensively characterized (Nemat-Nasser and Li, 2000; Pugal, 2012; Newbury, 2002; Newbury and Leo, 2002; Jiang et al., 2023; Sarker, Ul Islam, and Islam, 2024) in terms of mass transport principles and actuation mechanism, the inverse mechanoelectrical transduction process in IEAP-based sensors remains comparatively underexplored, despite promising applications in biomedical sensing (He et al., 2019b; Patanè et al., 2024a; Pugal et al., 2013), vibration energy harvesting and self-generating sensing (Yang, Yang, and Wang, 2023b). This research gap is particularly relevant in the context of flexible sensing transducers development, where dynamic ion redistribution under mechanical deformation offers a rich mechanism for signal generation but lacks comprehensive multiphysics modeling frameworks. In parallel with the functional advances of EAPs, the electronics industry is undergoing a transformation toward environmentally sustainable materials and processes. This shift has led to the emergence of green electronics, in which biodegradable and bio-derived materials play a central role.

BC, produced by strains such as Acetobacter xylinum and Gluconacetobacter hansenii, is a leading candidate due to its renewable origin, high purity and porous structure suitable for functionalization with ionic liquids (ILs). Although chemically identical to plant-derived cellulose, BC is produced via microbial fermentation using green feeding media (e.g. banana peel (Sijabat et al., 2020)) in standard laboratory environments. Unlike plant-based cellulose, its production avoids deforestation, yields a highly pure material, and results in a fully biodegradable substrate suitable for flexible, printable, and biocompatible electronic devices (Esa, Tasirin, and Abd.Rahman, 2014). Its mechanical resilience and environmental compatibility enable use in soft robotics, wearable electronics, smart packaging and biomedical systems (Huang et al., 2022).

When BC is functionalized with ILs and coated with conductive polymers, it becomes electroactive and is suitable for use in both actuators and sensors (Yuen et al., 2020; Pasquale et al., 2019b; Trigona et al., 2021). These BC-IL-based composites can be therefore classified as IEAPs, offering a sustainable alternative to synthetic polymers by combining functional performance with compatibility for roll-to-roll manufacturing processes (Trigona et al., 2021; Wang, Tavakoli, and Tang, 2019; Pasquale et al., 2020; Pasquale et al., 2019a). Importantly, IL derived from biomolecules, so-called bioionic liquids (BILs), have been shown to improve biocompatibility, cytocompatibility, and environmental safety, thereby expanding the use of BC-based IEAPs in biomedical applications such as implantable sensors and smart drug-delivery devices (Kanjilal et al., 2023). These bioinspired systems are at the forefront of smart tissue

engineering, molecular diagnostics, and regenerative medical electronics. In particular, the dual functionality of BC-based transducers, as both sensors and actuators, has opened new possibilities for soft robotic systems that require both adaptability and environmental integration. Recent reviews highlight the role of electroactive biomaterials like BC-IL composites in responsive biointerfaces and distributed robotic actuation (El-Atab et al., 2020). These materials support deformation-based sensing and controlled mechanical motion within the same lightweight and flexible platform.

3.3 Research Gap in BC-based Sensors Modeling

Despite this potential, the design and optimization of BC-based IEAP sensors face significant challenges. On the one hand, EAP modeling frameworks, including black-box and gray-box and machine learning approaches (Yang, Yang, and Wang, 2023b; Caponetto et al., 2013a; Caponetto et al., 2014c; Vahabi et al., 2011; Punning et al., 2009; Jiang et al., 2023), rely on empirical data and lack physical insight and generalizability across different materials and operating conditions. On the other hand, first-principles models based on computational methods have proven effective for the simulation of multiphysical systems for different compound materials. These models are able to address a variety of physical domains, including carrier transport (Caponetto et al., 2014a; He et al., 2019b), mechanical deformation (Auenhammer et al., 2024), thermal properties (Li and Zhang, 2024), magneto-mechanical interaction at the microscale (Moreno-Mateos et al., 2022). However, their high computational cost often makes them impractical for fast design iterations, an essential requirement for industrial prototyping and real-time system optimization, thus highlighting the inherent trade-off between simulation accuracy and computational efficiency. Addressing this trade-off is central to enabling high-throughput simulation pipelines for material screening and rapid sensor optimization.

More specifically, in the IEAP modeling environment, white-box models based on first-principles have been successfully developed for actuation purposes (Pugal, 2012; Caponetto et al., 2014b; Caponetto et al., 2014a; Caponetto et al., 2013b; Jiang et al., 2023). However, their application to mechanoelectrical sensor technology remains very limited in the literature and is typically restricted to simplified single-ion transport models within solvent-based systems, primarily water-based electrolytes (He et al., 2019b; Pugal et al., 2013; Pugal et al., 2010; Jiang et al., 2023). In the case of BC-based sensors, existing modeling efforts are still in a preliminary stage and primarily rely on gray-box approaches (Trigona et al., 2021; Caponetto et al., 2023b).

Against this backdrop, we examine multiphysics FEM alongside data-driven (grey/black-box) identification, each offering distinct advantages for building interpretable and efficient models for BC-based IEAP sensors. To overcome current limitations while keeping interpretability central, we introduce a first-principles modeling framework explicitly tailored to BC-based ionic transducers that captures dual-carrier transport (diffusion, advection, electromigration) in porous substrates, yet remains computationally tractable via a hybrid 2D–1D decomposition. This lays the foundation for high-throughput simulation pipelines tailored to soft intelligent systems that integrate sensing, actuation, and adaptive responsiveness within a single sustainable platform, leveraging the demonstrated potential of BC composites in soft robotics and wearable bioelectronics.

To implement this, we integrate a two-dimensional FEM for mechanical deformation with a series of one-dimensional FEM formulations for ionic transport and electric potential generation. This dimensional decomposition is designed to preserve

the fundamental physics of electromechanical coupling while reducing the computational effort. This facilitates fast parametric studies and supports the design and optimization of sensors at an early stage. The framework supports predictive material design, enabling the identification of structure–function relationships and critical design parameters, with direct validation through experimental measurements. Particular attention will be paid to ensuring that the model is compatible with data from real experimental setups to enable meaningful calibration and validation.

Ultimately, this methodology aims to provide a robust and scalable simulation framework for the design of next-generation green electronic devices based on BC-IL composites. To this end, the main contributions and novelties of this work include the introduction of a first-principles-based model specifically tailored for BC-IL sensors, establishing a bridge to data-driven identification within one unified workflow.

We next delineate the data-driven research gap. Data-driven grey-box models attempt to simplify the representation by incorporating empirical data while maintaining a certain degree of interpretability (Caponetto et al., 2023a). Black-box models, which can utilize machine learning techniques, are a viable alternative that allows for data-driven approximations of the sensor's response (Yang, Yang, and Wang, 2023a). There are only preliminary models for BC-based sensors in the literature (Trigona et al., 2021; Caponetto et al., 2023b). The similarities of BC with other IEAPs could be used to develop models for BC-based transducers. Different classes of models for IEAPs are proposed in the literature: Black-box models with neural networks (Yang, Yang, and Wang, 2023b), gray box models (Vahabi et al., 2011; Punning et al., 2009) including fractional order systems (Caponetto et al., 2013a) and white box models based on computational fluid dynamics (CFD) and FEM in multiphysics domains (Caponetto et al., 2014b; He et al., 2019c) are some examples. White Box models rely on a detailed understanding of the system's governing physics, providing insights into both spatial and temporal dynamics. While it ensures high interpretability and accuracy, its computational complexity limits its efficiency for fast simulations, making it less practical for real-time applications. Grey-box model accuracy depends on the quality of prior knowledge and available data, limiting generalizability in highly nonlinear or poorly understood systems.

Within the unified framework advanced in this chapter, experimental measurements on BC-based sensors are therefore used to derive and compare dynamic grey- and black-box models—evaluated in both one-step-ahead prediction and ∞-step simulation—to complement the FEM insights, calibrate critical closures, and deliver deployable predictors for control and design. Collectively, the hybrid 2D–1D FEM core and the data-driven layer form a single methodology that addresses the accuracy–cost and insight–generalization trade-offs across modern scientific and industrial applications.

3.4 Experimental Section

This section details the device geometry, fabrication, and the mechanoelectrical test bench used to acquire synchronized displacement–voltage datasets for model calibration and validation.

3.4.1 Sensor Geometry and Manufacturing

The BC-based sensor is a composite material produced using the following materials: purified BC films (CBP-GS0010) purchased from BioFaber (Italy), approximately

A4 size and average thickness t_{BC}; 1-Ethyl-3-Methylimidazolium tetrafluoroborate (EMIM-BF_4) purchased from Sigma Aldrich and used as IL, CLEVIOS™ PH 1000; an aqueous dispersion of poly(3,4-ethylenedioxythiophene)/polystyrene sulfonate (PEDOT:PSS), which was purchased from Heraeus and used for electrode fabrication. The fabrication process began with BC, which was processed in squared samples measuring 5 cm $\times$ 5 cm. Each sample was immersed in EMIM-BF_4 for 24 h at room temperature in a laboratory desiccator using calcium chloride as the desiccant. After impregnation, excess IL was removed with filter paper. At this stage, the BC-IL composite contained approximately 8% by weight of EMIM-BF_4. PEDOT:PSS electrodes were then deposited on both sides of the composite using a 25μm film spreader. Finally, the transducers were trimmed to the desired geometry. The geometry of the device is presented in Figure 3.2(a) and its parameters are reported in Table 3.1, where L is the total length of the sample, W is the width of the entire sample and L_c is the length of the clamped copper electrode. In particular, T_{BC} and T_{PSS} are the thickness of the BC and PEDOT:PSS electrodes, respectively. They were determined from scanning electron microscopy (SEM) micrographs obtained using a SEM EVO (Zeiss, Cambridge, UK) instrument equipped with an energy dispersive X-ray microanalysis (EDX) facility. The analyses were performed by setting a high electron beam voltage (EHT) of 20 kV and using a LaB6 (Lanthanum Hexaboride) emitter as the electron source. To perform the SEM analysis, the samples were coated with a thin gold film applied by a sputtering process using an Agar Sputter Coater AGB7340 spray coating machine (Assing, Italy).

3.4.2 Measurement Setup

The experimental configuration is illustrated in Figure 3.1: (a) computer-aided design (CAD) rendering of the mechanical system, (b) schematic of the complete experimental setup, and (c) photograph of the assembled device showing the cantilever-mounted sensor in its working configuration. A vertical threaded rod with two adjustable nuts was used to fine-tune the vertical positioning of the cantilever beam. A dedicated bracket attached to the threaded rod served as a support for the transducer. A custom rectangular housing was designed to accommodate a pair of rigid electrodes and the BC-based sensor. These electrodes are responsible for detecting the electrical signal generated by the composite material during deformation. To facilitate the connection between the electrodes and the external recording system, two conduit channels were integrated into the housing design. To ensure mechanical symmetry and proper housing, two identical brackets were fabricated and mounted opposite each other, enclosing the BC-based transducer and the pair of electrodes. The entire support structure was mounted on a base plate, which provided a stable foundation for the experimental setup. The mechanical excitation of the transducer was achieved with the help of an electromagnet located in a circular housing mounted on the base plate (see Figure 3.1(a)). The electromagnet used was the HS-P65*30 model from Heschen with a diameter of 65 mm and a height of 30 mm and a maximum DC voltage of 24 V. To drive the electromagnet, a power amplifier circuit was implemented with an OPA547 operational amplifier configured with two resistors of 4.7 kΩ and 6.8 kΩ. The actuation signal was synthesized in MATLAB®, then exported and loaded into a waveform generator (Keysight 33220A). The output signal from the waveform generator was fed into the power amplifier, which drives the electromagnet. This produces a motion of the tip of the BC sensor. The mechanical actuation was accomplished using two cylindrical magnets (mass: 0.32 g; radius: 2 mm; height: 3 mm) mounted symmetrically on both sides of the transducer near its free end at a distance $L_M = 3.7cm$

from the base clamp, commonly referred to as the beam tip. The displacement of the beam tip was monitored via a Baumer 12U6460-S35A laser displacement sensor with a resolution of approximately 2 μm and adjustable sensitivity. The laser was focused at a distance of $L_L = 3.2cm$ from the base clamp. The mechanical deformation of the composite led to a charge accumulation on the PEDOT:PSS electrodes. This accumulation was measured as the open circuit (OC) voltage, hereafter referred as sensor voltage. For the OC measurements, the electrodes were connected directly to an Agilent Infiniium MSO9064A digital oscilloscope. This simultaneously acquired the laser signal, allowing synchronous analysis of the mechanical input and electrical output.

3.5 First-Principle Dual Carrier Multiphysics Model

The BC-based transducer can convert mechanical deformation into electrical signal and vice versa and can act as both a sensor and an actuator. In this study, we investigate a BC-based sensor infused with EMIM-BF_4 as IL and coated with PEDOT:PSS as conductive polymers. This setup is illustrated in Figure 3.2(a). When configured as a cantilever and used as a sensor, the mobile IL ions, $EMIM^+$ (cation) and BF_4^- (anion), redistribute within the BC matrix in response to an external force applied, inducing mechanical deformation (see Figure 3.2(b)). This ion redistribution leads to the generation of an electrical potential in the BC, which can be measured at the electrodes as open-circuit (OC) voltage, hereinafter referred to the sensor output.

Multiphysics models for IEAPs have been explored in the literature (Pugal et al., 2010; Caponetto et al., 2014b; He et al., 2019b). These models typically couple three physical domains: a mechanical model to compute structural deformation, a chemical model to describe ionic transport, and an electrical model to calculate the internal voltage potential. The system behavior is generally governed by PDEs defined over the spatial domain $\boldsymbol{r} = (x, y, z)$ and time t. However, in the existing approaches, the chemical domain is limited to a single mobile ionic species, often assuming either cationic or anionic transport while treating the counter-ion as immobile or uniformly distributed. This simplification neglects the dynamics of both ionic carriers, which can be significant in systems based on ILs where both cations and anions are mobile and participate in the charge redistribution process (Figure 3.2(b)). Figure 3.2(c) presents the proposed multiphysics simulation framework, which couples mechanical deformation, ionic transport, and voltage generation through multiphysics integrating mechanical, chemical, and electrical models. Each of these sub-models is described in detail in the following section. The assumptions of isotropy, homogeneity, temporal invariance and linearity are made for all materials.

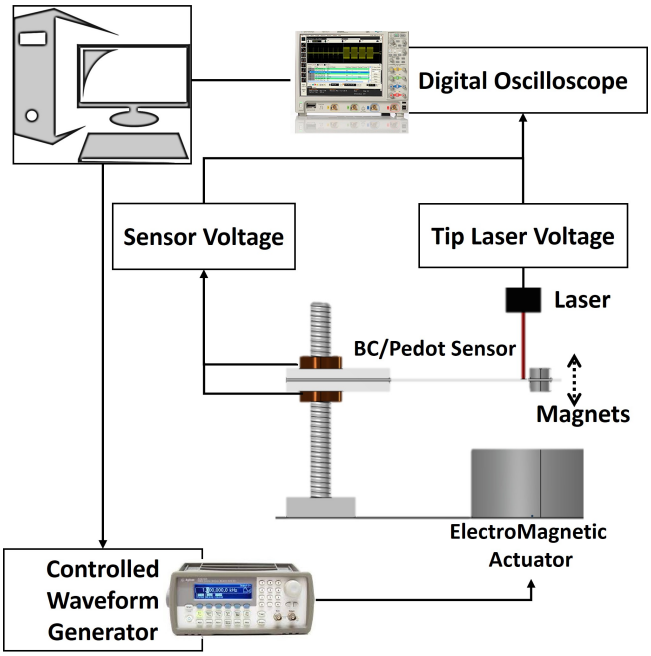
3.5.1 Mechanical Model

Newton's second law relates the displacement ($\boldsymbol{d}$) with the volumetric force ($\boldsymbol{F}$), as reported in (3.1).

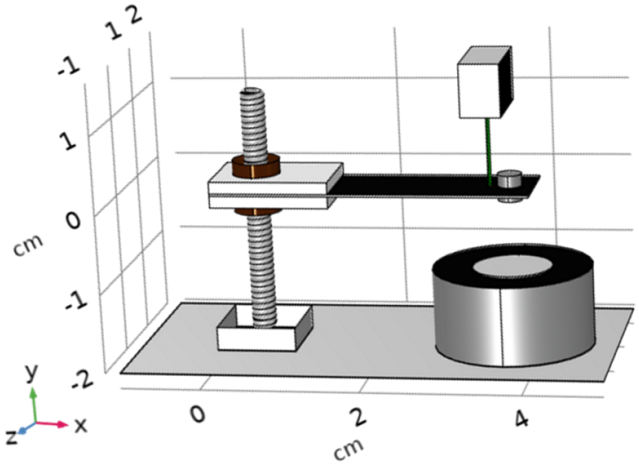
$$\rho \frac{\partial^2 \boldsymbol{d}(\boldsymbol{r}, t)}{\partial t^2} - \nabla \cdot c \nabla \boldsymbol{d}(\boldsymbol{r}, t) = \boldsymbol{F}(\boldsymbol{r}, t) \tag{3.1}$$

where ρ is material density and c is the Navier constant. The polymer pressure (p) is related to the σ_{ij} stress tensor components as in (3.2):

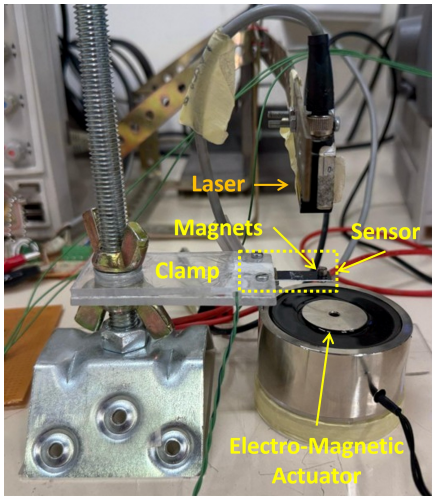
$$p(\boldsymbol{r}, t) = \frac{\sigma_{11} + \sigma_{22} + \sigma_{33}}{3} \tag{3.2}$$



(a)



(b)



(c)

FIGURE 3.1: Mechanical setup: (a) global schematic of the acquisition system, (b) 3D schematic, (c) laboratory setup.

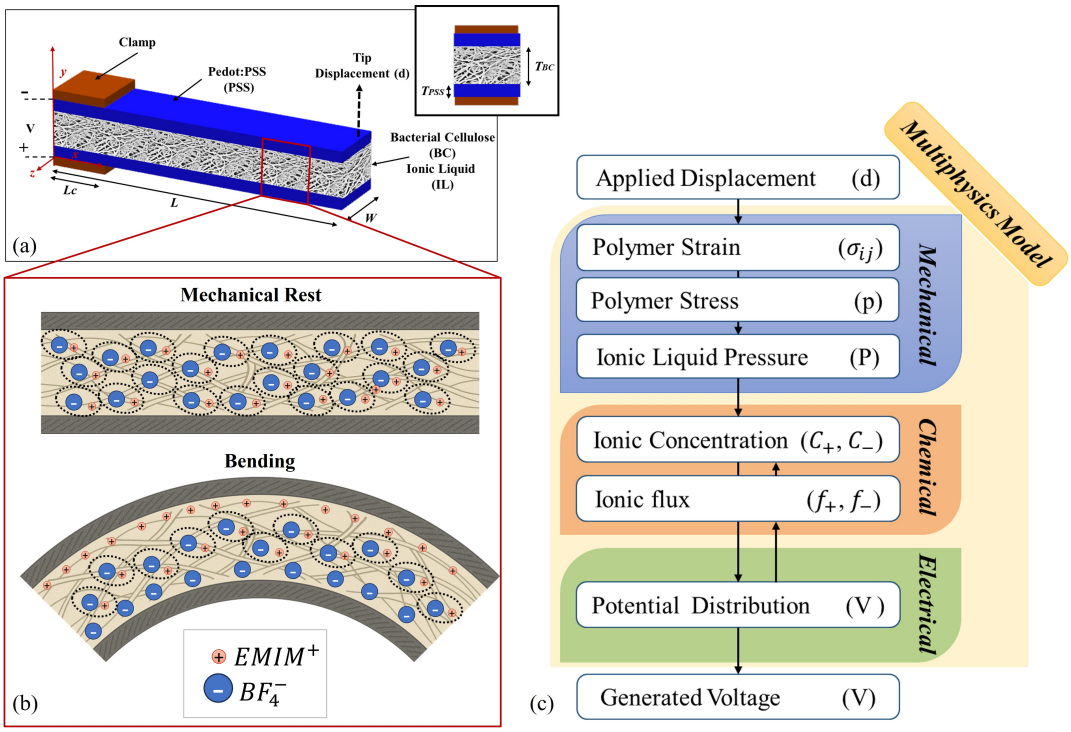


FIGURE 3.2: Multiphysics modeling framework for BC-based ionic sensors. (a) Geometry of the BC-based sensor configured as a cantilever, with PEDOT:PSS electrodes applied to both sides of the IL-infused BC membrane and voltage measured in a open circuit configuration. Geometric parameters are reported in Table 3.1 and the complete setup in the Methods section. (b) Schematic illustration of the dual-ion sensing mechanism showing the ionic distribution at rest and its redistribution under bending deformation. It highlights the role of both cations and anions in charge separation. (c) Overview of the proposed multiphysics simulation framework, coupling mechanical deformation with ionic transport and voltage generation, used to compute the sensor electrical response to tip displacement.

Then, using the momentum conservation law in (3.3) it is possible to relate the the hydrostatic pressure p in polymer with the fluid pressure (P):

$$\nabla P(\boldsymbol{r},t) = -\nabla p(\boldsymbol{r},t) \tag{3.3}$$

3.5.2 Chemical Model

A key novelty of this work lies in the chemical model, where both cations and anions are considered as mobile carriers, unlike conventional models that assume a single mobile species. Accordingly, distinct transport equations are formulated for each ionic species to capture their individual contributions to the total ionic current. The flux of cations $EMIM^+$ and of the anion BF_4^- are related to the time variation of their respective concentrations C_+ and C_-, the applied potential (V), and to the fluid pressure (P). It is calculated by the Nernst-Planck law as follows:

$$\begin{aligned}\frac{\partial C_i(\boldsymbol{r},t)}{\partial t} = \nabla \cdot \Big(& D_i \nabla C_i(\boldsymbol{r},t) + z_i \mu_i F C_i(\boldsymbol{r},t) \nabla V(\boldsymbol{r},t) \\ & + \mu_i C_i(\boldsymbol{r},t) D_{v_i} \nabla P(\boldsymbol{r},t) \Big)\end{aligned} \tag{3.4}$$

where D_i is the diffusion coefficient, z_i is the charge number, μ_i is the mobility, D_{v_i} is the molar volume of species i, and F denotes the Faraday constant.

The total ionic current density can be expressed as in (3.5)

$$\boldsymbol{J_{tot}}(\boldsymbol{r},t) = \sum_{i=1}^{N} z_i F \boldsymbol{f_i}(\boldsymbol{r},t) \tag{3.5}$$

where the general form of the Nernst-Planck equation for the total flux f_i of an ion species i in (3.6) consists of three distinct components: diffusion, electromigration, and advection:

$$\begin{aligned}\boldsymbol{f_i}(\boldsymbol{r},t) = & -D_i \nabla C_i(\boldsymbol{r},t) - z_i \mu_i F C_i(\boldsymbol{r},t) \nabla V(\boldsymbol{r},t) \\ & - \mu_i C_i(\boldsymbol{r},t) D_{v,i} \nabla P\end{aligned} \tag{3.6}$$

$$\boldsymbol{J_{i_{Diff}}}(\boldsymbol{r},t) = z_i F \boldsymbol{J_{i_{Diff}}}(\boldsymbol{r},t) = z_i F D_i \nabla C_i(\boldsymbol{r},t) \tag{3.7}$$

This term represents the ion current density due to concentration gradients, driving the species from regions of high concentration to low concentration.

$$\boldsymbol{J_{i_{EM}}}(\boldsymbol{r},t) = z_i F \boldsymbol{f_{i_{EM}}}(\boldsymbol{r},t) = z_i^2 \mu_i F^2 C_i(\boldsymbol{r},t) \nabla V(\boldsymbol{r},t)) \tag{3.8}$$

This term represents the ion current density due to the electric potential gradient ∇V, and is proportional to μ_i, the ion's mobility, and to the factor $z_i F$ representing the electric charge of the species in terms of Faraday's constant.

$$\boldsymbol{J_{i_{Ad}}}(\boldsymbol{r},t) = z_i F \boldsymbol{f_{i_{Ad}}}(\boldsymbol{r},t) = z_i F \mu_i C_i D_{v_i} \nabla P(\boldsymbol{r},t) \tag{3.9}$$

This term represents the transport of mass by the bulk motion of a fluid. The pressure gradient ∇P drives the species contained in the IL in response to an external pressure field, which in this case is the polymer pressure p due to deformation as in (3.3).

3.5.3 Electrical Model

The potential distribution (V) inside the BC is calculated in the electrical model solution by using the Poisson equation as reported in the following:

$$-\epsilon_0\epsilon_r\nabla^2 V(\boldsymbol{r},t) = \rho_0 + \sum_{i=1}^{N} z_i e C_i(\boldsymbol{r},t) \tag{3.10}$$

where ϵ_0 is the absolute dielectric constant, ϵ_r is the relative dielectric constant, ρ_0 is the permanent charge density. The ion concentrations C_i are the coupling factors to the chemical model introduced in (3.4). Within the PEDOT:PSS domain, the electric potential was assumed to be governed by Ohm law and the current continuity equation, which was implemented separately.

3.5.4 FEM Model Implementation

The model is implemented by FEM simulations in Comsol Multiphysics®, which solve PDEs over both spatial and temporal domains and allow coupling between the physical domains.

i) Mechanical Model

The mechanical deformation of the BC-based sensor was simulated using a two-dimensional FEM model in the xy-plane representing the cantilever configuration of the device (Figure 3.3). The clamped region (Ω_C^{2D}) was fixed with a zero displacement, while a prescribed sinusoidal displacement in the vertical direction was applied to the free tip. The sinusoidal excitation mimics the mechanical input used in the experimental characterization and allows a controlled analysis of the deformation-induced electrochemical phenomena. The model geometry was discretized with a physics controlled quadrilateral mesh, with a minimum element size of 3 μm and a maximum element size of 800 μm. This provides sufficient resolution near regions with strong curvature and high strain gradient, such as the clamping zone, while ensuring the efficiency of the calculation in less critical regions. The FEM analysis was performed using a linear-elastic formulation where the displacement field was solved as in Section 3.5.1, from which field variables such as strain, stress and derived polymer pressure p were calculated as in (3.2). The polymer pressure value of the mechanical model was extracted along selected cut lines (CL) in y-direction at 0.1 mm, 0.5 cm, 1 cm, 1.5 cm and 2 cm from the fixed bracket edge (L_C). It serves as a coupling variable for the mechanoelectric model as in (3.3).

ii) Simplified 1D Chemical and Electrical Model

The ionic transport and the distribution of the electric potential within the BC layer were modeled using a one-dimensional FEM formulation along the y-axis cutlines, orthogonal to the plane of mechanical deformation. The model considers two mobile ionic species, a cation and an anion, via the Nernst-Planck equation as in (3.4). The local net charge density, defined as the difference between the cation and anion concentrations weighted by their respective charges, served as the source term for the Poisson equation (3.10), which was solved to obtain the electric potential distribution and calculate the voltage at the electrode terminals for each cutline along the y-axis (Figure 3.3). For simplicity, each cutline model is treated independently, neglecting lateral coupling effects and the influence of the PEDOT:PSS electrode along the x-axis

within the 1D domain. At the outer ends of the BC domain (Ω_{BC}^{1D}), zero-flux Neumann boundary conditions were applied to represent ion confinement in the material. Dirichlet boundary conditions were introduced at the interfaces between the BC (Ω_{BC}^{1D}) and the PEDOT:PSS electrodes (Ω_{PSS}^{1D}) to ensure continuity of electric potential at the interfaces. A Dirichlet boundary condition that sets the electric potential to zero ($V = 0V$) was applied to the outer boundary of the upper electrode to define the electrical ground reference.

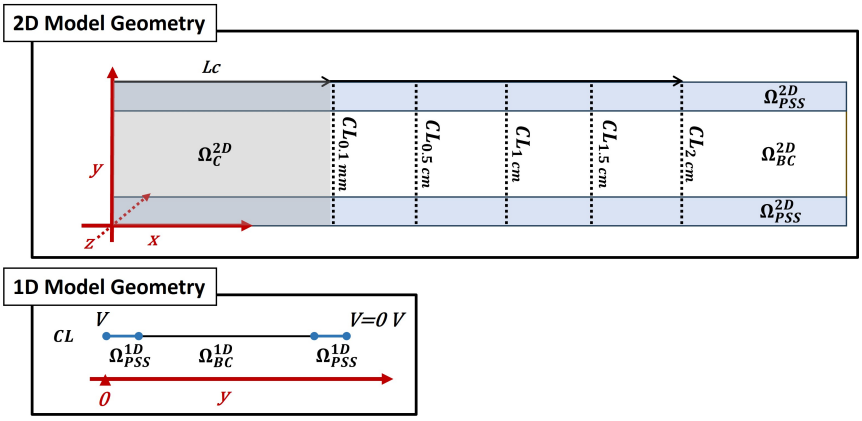


FIGURE 3.3: FEM Model Geometry: 2D geometry for the mechanical domain model, and 1D geometry for the mechano-electrical transduction model at the different y-axis cutlines

iii) Simulation Parameters

The model parameters are determined by experiments or from the literature. When necessary, scaling factors were determined through calibration by comparing the macroscopic model outputs with experimentally measurable field quantities at the selected excitation frequency. Since the model was calibrated under a specific frequency condition, its application to different excitation frequencies may require recalibration to ensure modeling accuracy. Table 3.2 contains the parameters associated with the mechanical model. Table 3.3 focuses on the electrochemical model and the ionic species, and Table 3.4 summarizes the parameters related to the electrical response of the sensor. The Young modulus of the BC-IL-PEDOT sample (E) was determined by dynamic mechanical analysis (DMA) using a 2000 TA DMA produced by Triton Technology Ltd (London, UK). The membrane modulus was determined by applying a sinusoidal force at a frequency of 10 Hz at a working temperature of approximately 25°C to a rectangular sample in single-cantilever mode. Calibration was performed before each measurement using the automatic procedure in the control software of the device. The initial ion concentration C_0 was calculated based on the amount of ionic liquid present in the sample and the number of moles was determined by dividing this value by the molecular weight of the ionic liquid. The resulting value was converted to moles per cubic meters taking into account the sample volume.

TABLE 3.1: Geometrical parameters.

Parameter description	Value	Unit
Total length, L	4	cm
Total width, W	1	cm
Clamp region length, L_c	1.5	cm
Thickness of BC, T_{BC}	320	μm
Thickness of PEDOT, T_{PSS}	25	μm

TABLE 3.2: Mechanical Parameters

Parameter description	Value	Unit
Young modulus of BC-IL-PEDOT compound, E	1.64	GPa
Poisson ratio of BC, ν_{BC} (Lopez-Sanchez et al., 2014)	0.01	-
Poisson ratio of PEDOT, ν_{PSS} (Agbaoye et al., 2017)	0.435	-
Mass density of BC, ρ_{BC}	0.904	g/cm^3
Mass density of PEDOT, ρ_{PSS} (Crispin et al., 2006)	1.28	g/cm^3

TABLE 3.3: Chemical parameters.

Parameter description	Value	Unit
Initial ion concentration, C_0 (EMIM-BF_4)	740.88	mol/m^3
Cation diffusion constant (Hayamizu et al., 2004), D_{cat}	4×10^{-10}	m^2/s
Anion diffusion constant (Hayamizu et al., 2004), D_{an} (BF_4)	4.4×10^{-10}	m^2/s
Cation molar volume (G. Montalban et al., 2015), $D_{v_{cat}}$	1.15×10^{-4}	m^3/mol
Anion molar volume (G. Montalban et al., 2015), $D_{v_{an}}$ (BF_4)	3.85×10^{-5}	m^3/mol
Cation charge number, Z_{cat}	1	-
Anion charge number, Z_{an}	-1	-
Universal gas constant, R	8.31	J/(K·mol)
Temperature, T	298.15	K
Cation mobility, μ_{cat}	$D_{cat}/(R \cdot T)$	s·mol/kg
Anion mobility, μ_{an}	$D_{an}/(R \cdot T)$	s·mol/kg

TABLE 3.4: Electrical parameters.

Parameter description	Value	Unit
Electrical conductivity of PEDOT, σ_{PSS} (Ouyang et al., 2004)	10	S/m
Effective absolute dielectric permittivity of BC-IL-PEDOT compound, $\epsilon = \epsilon_r \epsilon_0$	Calibrated (Figure 3.6)	F/m

3.5.5 Experimental and FEM-Based Analysis

In this section, a combined experimental and numerical investigation of the BC-IL sensor under dynamic loading is presented. First, experimental measurements are discussed and then compared with simulations performed with a multiphysics FEM framework. The analysis is divided into subsections dealing with experimental

data acquisition and model results, including 2D mechanical response, 1D mechano-electrical model calibration and ionic current density evaluation on selected y-axis cutlines.

i) Experimental Acquisition

The mechanoelectrical properties of the BC-based composites were evaluated by mounting the samples in a cantilever configuration. In this setup, the free end of the cantilevered transducer was subjected to controlled mechanical excitation. Figure 3.4 shows the acquisition of an imposed sinusoidal tip motion at a selected frequency of 8 Hz with a peak amplitude of $d_{\mathrm{MAX}} = 0.27$ mm, measured using a laser detector. The voltage response of the sensor shows a periodic signal with a peak amplitude of $V_{\mathrm{p}} = 0.14$ mV, which determines an experimental ratio of 0.52 V/m (corresponding to an attenuation of $A_{exp} = -6dB$) and a phase delay of $\Phi_{exp} = -30.7°$.

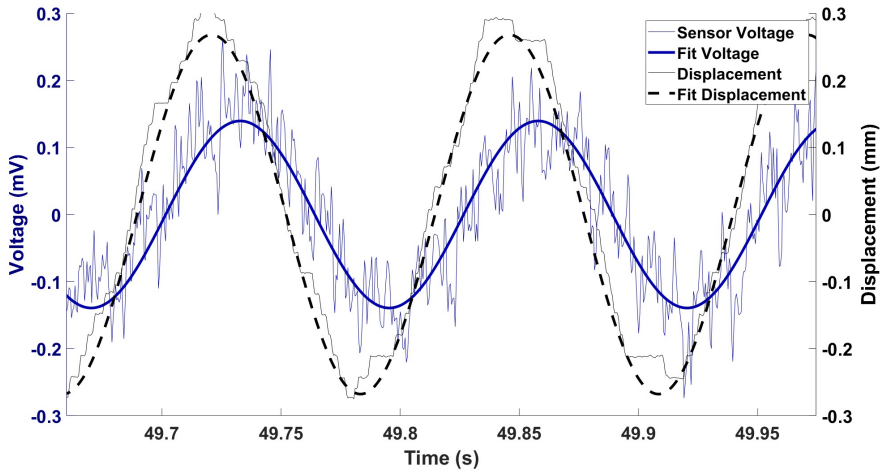


FIGURE 3.4: Experimental response of the BC sensor infused with EMIM-BF_4 to sinusoidal input displacements with amplitude $d_{MAX} = 0.27$ mm and frequency $f_d = 8$ Hz. The recorded signals were fitted using single-frequency sinusoids to filter noise and accurately extract amplitude and phase shift parameters.

ii) Mechanical Model

A 2D FEM simulation was performed in the xy-plane by applying a sinusoidal displacement at the cantilever tip, with a frequency of 8 Hz and a peak amplitude of $d_{\mathrm{MAX}} = 0.27$ mm. This displacement matches the experimental measurement shown in Figure 3.4 and obtained using the setup detailed in the Methods section.
The solution of the 2D mechanical model in terms of polymer pressure (p) is shown in Figure 3.5. The pressure distribution shows higher values in regions with maximum curvature and near the electrode boundaries. The pressure values were extracted along five selected y-axis cutlines ($CL_{0.1mm}$, $CL_{0.5cm}$, CL_{1cm}, $CL_{1.5cm}$, CL_{2cm}) considered at different distances from the fixed clamp and detailed in Figure 3.3 of the Methods section ii).

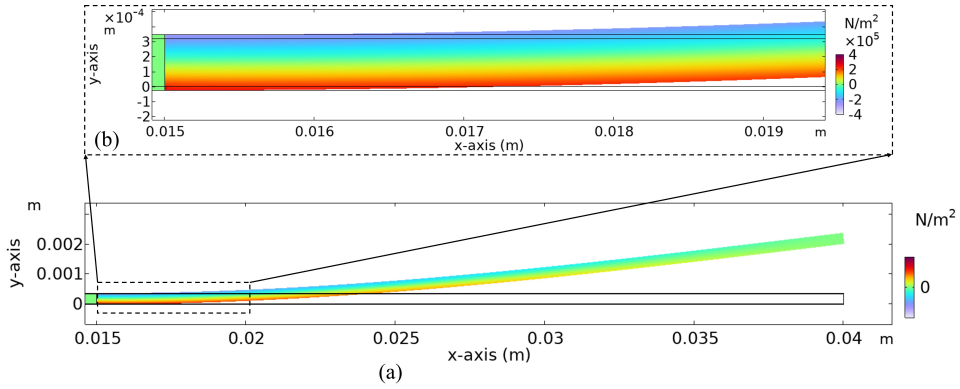


FIGURE 3.5: Pressure field distribution in the polymer domain (a) computed from the 2D mechanical model at the time of maximum applied displacement, including in (b) a zoomed-in view of the area exhibiting the highest pressure values and spatial gradient.

iii) Model Calibration

Effective absolute dielectric permittivity ϵ of the compound material plays a critical role in determining the amplitude and phase characteristics of the response of the BC-IL sensor. This dependence results from the influence of ϵ on the capacitive behavior of the system, which determines the dynamics of charge redistribution during mechanical deformation.

A 1D FEM model was employed along the y-axis cutlines, perpendicular to the mechanical deformation plane, to simulate ion migration and the associated electric potential distribution across the BC layer (details in Section ii)). The calibration of the permittivity of the BC sensor ϵ was performed by a parametric analysis of the model response for the cutline $CL_{0.1\,\mathrm{mm}}$ configuration by minimizing cost function determined as the sum of the normalized root mean square errors (RMSEs) of the simulation results with respect to the experimentally observed amplitude ratio A_{exp} and phase shift Φ_{exp} for a sinusoidal input at 8 Hz. The search space for permittivity was defined based on literature values reported for similar systems, including sensors made of graphene and chitosan with the ionic liquid EMIM-BF_4, which have a permittivity of 2.8×10^{-2} F/m (He et al., 2019b), and IPMCs based on water-hydrated Nafion, whose permittivity is between 5×10^{-2} and 6×10^{-1} F/m (Pugal, 2012). Figure 3.6 illustrates the trend of the total RMSE, calculated as the sum of the normalized RMSEs for attenuation and phase shift, and highlights the optimal permittivity value $\epsilon = 2.24 \times 10^{-2}$ F/m, which minimizes the cost function. The value determined is within 10% of the chitosan-based reference and thus confirms its agreement with comparable material systems. This optimum setting corresponds to discrepancies with respect to experimental values of 3.4×10^{-2} V/m for the amplitude ratio A and 1.6° for the phase Φ.

iv) Position-Dependent Sensor Response and Curvature Effects

Figure 3.7(a) illustrates the simulated response of the sensor voltage evaluated along the five selected y-axis cutline models. The results show a clear dependence of the electrical response of the composite in accordance with the position along the beam, which correlates with the spatial distribution of the mechanical stress and pressure gradient in Figure 3.5. In particular, the cutline closest to the fixed clamp ($CL_{0.1mm}$) shows the highest amplitude response, which agrees with the proposed transduction

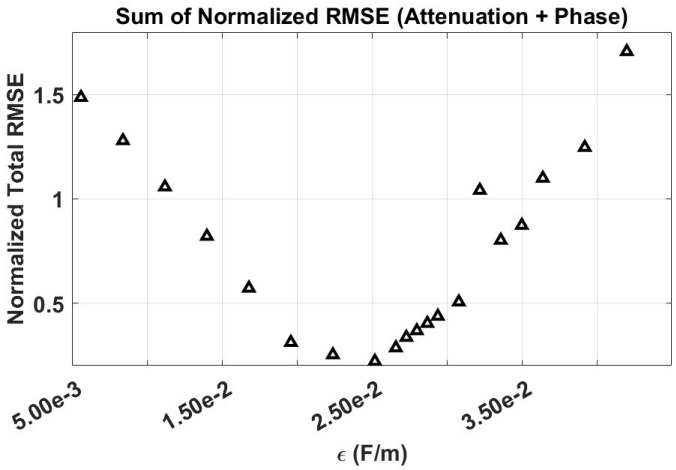


FIGURE 3.6: Calibration curve for the BC-sensor permittivity ϵ. The plot shows the sum of the normalized RMSEs for the attenuation and phase shift between the sensor voltage and the tip displacement, computed with respect to the experimental reference values A_{exp} and phase shift Φ_{exp} for a sinusoidal input at 8 Hz.

mechanism. This region corresponds to the section of maximum curvature of the cantilever. This increased curvature leads to a higher pressure gradient in the y direction, which directly affects the advection term in the Nernst-Planck equation and triggers ionic movement and charge separation. At larger distances from the clamp, both the pressure and its spatial gradient decrease due to the lower curvature, resulting in a smaller electrical response. These results emphasize the importance of local deformation and stress distribution in shaping the overall sensor response. In particular, large deformations near the clamp region can induce sideways microcracks in soft material electrodes, as reported in previous studies (Zhang et al., 2021; Moreno-Mateos and Steinmann, 2024), leading to increased local resistivity. As a result, the highly deformed zones near the clamp can be electrically decoupled from the rest of the sensor, maintaining a localized electrical response. This observation supports the assumption of neglecting the influence of the PEDOT electrodes in the transverse (x) direction.

v) Ionic Current Densities

The behavior of ionic transport was analyzed by examining the temporal evolution of the total ionic current density as in (3.5) and its components, diffusion, electromigration, and advection, at specific time instants of both the input displacement cycle and the resulting sensor voltage response. In Figure 3.7(b)-(c) broken y-axis is used to highlight interfacial effects, as the current densities remain nearly constant in the inner region of the BC. The first analysis reveals the dynamic interplay between mechanically induced pressure gradients and ionic transport. It highlights how advection dominates during displacement extrema, while electromigration becomes prominent when the displacement velocity approaches zero. Figure 3.7(b), corresponding to the maximum displacement, shows that advection is the dominant component due to the fluid motion induced by the structural deformation. In contrast, the diffusion and electromigration currents counteract the advection as they are driven by residual electric fields from the previous moments and local concentration

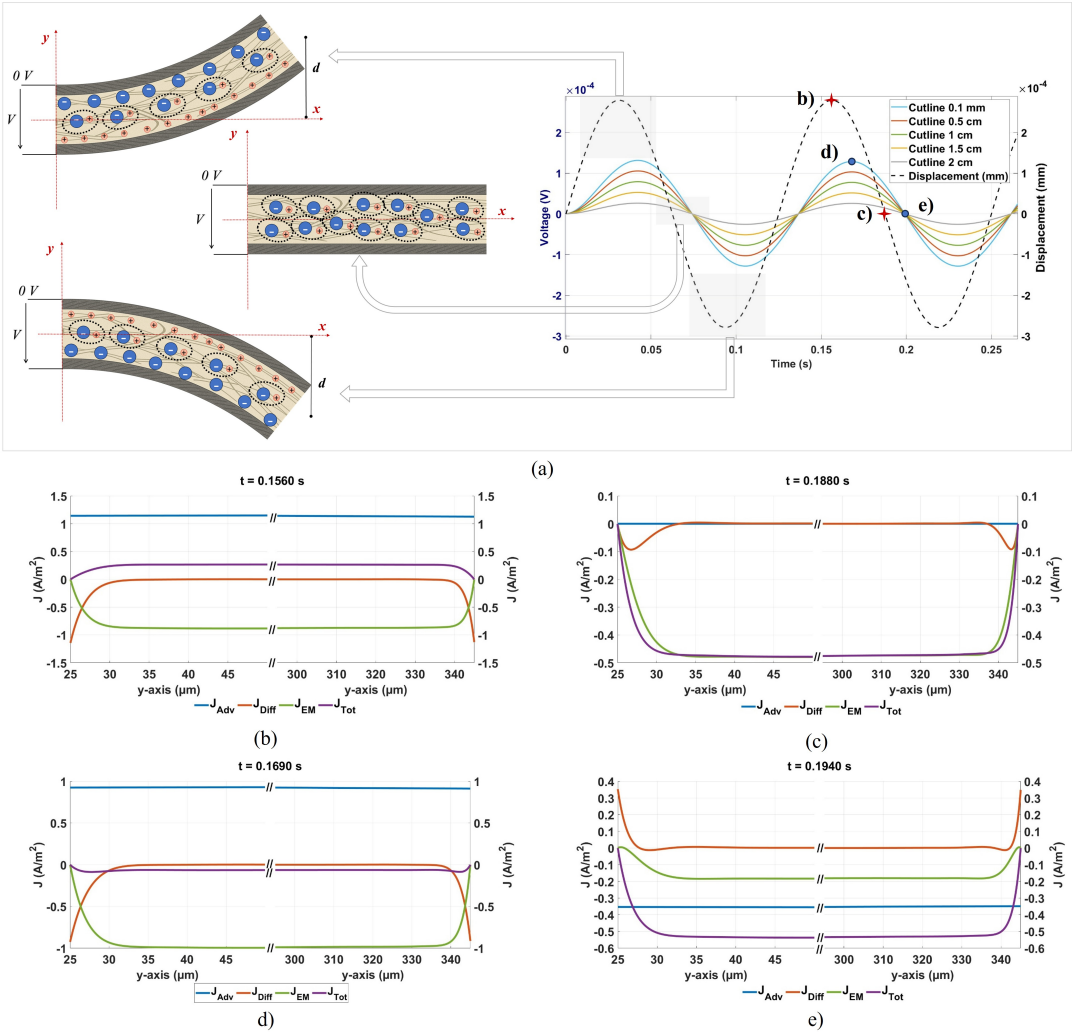


FIGURE 3.7: Simulation results for the BC-IL sensor under dynamic excitation. (a) Simulated voltage responses at various positions along the beam length, $CL_{0.1mm}$, $CL_{0.5cm}$, CL_{1cm}, and CL_{2cm}, illustrating the spatial dependence of the electrical output. The panel also includes a schematic illustration of the dual-ion sensing mechanism, highlighting the redistribution of cations and anions at positive displacement, negative displacement, and close to zero displacement. (b, c) Spatial profile on the y-axis $CL_{0.1mm}$ cutline of the total ionic current density and its components (diffusion, electromigration, and advection) at key points of the displacement cycle: (b) maximum displacement and (c) zero displacement during the falling edge. (d, e) Spatial profile on the y-axis $CL_{0.1mm}$ cutline of the ionic current density at characteristic points of the sensor voltage waveform: (d) voltage peak and (e) zero voltage during the falling edge.

gradients. In Figure 3.7(c), which shows the falling edge at zero displacement, the advection component disappears as the instantaneous velocity is zero. Under these conditions, electromigration becomes the dominant component of the total current, driven by the electric field resulting from the previous charge displacements. The diffusion current generally remains weak, but shows localized significance at the BC/PEDOT interface, while it is negligible in the bulk region. Overall, the diffusion and electromigration components are largely in phase, while the advection current is out of phase with them, yet still plays a crucial role as the primary initiator of the sensing mechanism. The phase delay between these three transport phenomena is responsible for the resulting phase shift between the measured sensor voltage and the mechanical excitation. The ionic current density distribution corresponding to the minimum displacement, shows dynamics analogous to Figure 3.7(b), but with the advection direction reversed. The current density corresponding to the zero displacement during the rising edge, matches the conditions of Figure 3.7(c), where advection is negligible and electromigration dominates near the interface. The second analysis, which is based on characteristic points of the sensor voltage, provides valuable insights into the electrochemical dynamics underlying the generation of the electrical signal. It illustrates how the phase shift between the ionic motion and the mechanical excitation contributes to the observed response of the sensor voltage. Together with the displacement-based analysis, these complementary perspectives provide a comprehensive understanding of the spatio-temporal coupling mechanisms governing mechanoelectric transduction in BC-IL sensors. In Figure 3.7(d), which corresponds to the voltage peak, the net ionic current can be assumed to be zero. A slightly negative value is reported due to the impossibility to identify the exact instant of the peak. At this point, the three components of the ionic current, advection, electromigration and diffusion, are approximately balanced. This configuration corresponds to the out-of-phase equilibrium between the mechanical input and the electrical output. In Figure 3.7(e), the current density becomes clearly negative, which is consistent with the sensor reaching the maximum negative slope of the voltage waveform.

This detailed analysis offers a comprehensive understanding of the sensing mechanism and provides design guidelines for IL selection tailored to application-specific requirements such as response speed and phase behavior. The investigation of the components of the ionic current at critical displacement and voltage instants reveals the complex interplay of advection, electromigration and diffusion and their collective influence on charge redistribution. In particular, the phase lag between mechanical deformation and ionic movement is identified as a key factor for the voltage response. Furthermore, the localized contribution of the diffusion current at the BC-PEDOT interface underlines the importance of interfacial phenomena for the overall electromechanical behavior of the sensor.

3.6 Data-Driven Models

We now complement the white-/grey-box analysis with data-driven identification. The goal is to obtain transparent surrogates for control (1-step-ahead) and robust long-horizon simulators (∞-step-ahead), benchmarked against the same experimental protocols.

Among the different data-driven grey and black-box identification techniques, both linear and nonlinear solutions previously adopted in modelling composite materials (Andò, Graziani, and Xibilia, 2019; Graziani et al., 2013) are considered

in this thesis. In particular, two linear models were used: FIR and ARX. Nonlinear models were also considered to assess the presence of nonlinear relationships in the system under study: NARX and NFIR). A deep learning model based on LSTM was also included in the analysis to evaluate the performance when recurrent networks are considered.

3.6.1 Data-Driven Grey-Box Models

i) Finite Impulse Response

An FIR model is a linear filter used in signal processing and control systems. It processes input signals using a finite number of past inputs without feedback, which makes it inherently stable (Oppenheim and Schafer, 1999).

$$y(k) = \sum_{i=1}^{n_b} b_i u(k-i) \tag{3.11}$$

where $y(k)$ is the output, $u(k)$ the input and b_i the coefficients of the FIR filter. This model is widely used for long-term prediction because it does not require past samples of the output variable.

ii) AutoRegressive with eXogenous Input

The ARX model is a parametric model used for system identification, incorporating both autoregressive (AR) and exogenous (X) inputs. It assumes a linear relationship between input and output signals with an additional white noise term (Ljung, 1999).

$$y(k) + \sum_{i=1}^{n_a} a_i y(k-i) = \sum_{j=1}^{n_b} b_j u(k-j) + e(k) \tag{3.12}$$

where n_a and n_b are the number of past output and input terms respectively and $e(k)$ is the noise term.

iii) Nonlinear Finite Impulse Response

An NFIR model extends the FIR model by including a non-linear function to map past inputs to the current output. This model is useful for capturing complex system dynamics (Billings, 2013).

$$y(k) = f(u(k-1), u(k-2), \ldots, u(k-n_b)) \tag{3.13}$$

where $f(\cdot)$ is a non-linear function that is applied to past inputs.

iv) Nonlinear AutoRegressive with eXogenous Input

A NARX model generalizes the ARX model by incorporating nonlinearities in the relationship between the predicted output and past values of both inputs and outputs. It is often used in system identification and time series prediction (Narendra and Parthasarathy, 1990).

$$y(k) = f(y(k-1), \ldots, y(k-n_a), u(k-1), \ldots, u(k-n_b)) + e(k) \tag{3.14}$$

where $f(\cdot)$ is a non-linear function. NARX model can be estimated using neural networks to model the nonlinear function. Tapped-delay lines for both input and output variables are used as input for the neural network.

3.6.2 Black-Box Models

i) Long Short-Term Memory

LSTMs are a type of recurrent neural network that can capture long-term dependencies using gating mechanisms. They are particularly effective for time series prediction and sequence learning tasks (Hochreiter and Schmidhuber, 1997).

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{3.15}$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3.16}$$
$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{3.17}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{3.18}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{3.19}$$
$$h_t = o_t \odot \tanh(c_t) \tag{3.20}$$

where x_t is the input, h_t is the hidden state, c_t is the cell state and $\sigma(\cdot)$ and $\tanh(\cdot)$ are activation functions.

3.6.3 Frequency Sweep Signal for System Identification

The data set used for the following analysis was acquired by applying a frequency sweep signal to the electromagnetic actuator in a range between 2 and 60 Hz. The magnets located near the BC tip are either attracted or repelled, resulting in mechanical stimulation of the device, which has a cantilever configuration. The measurements were performed for $60s$, with an acquisition frequency of $2kHz$. The time evolution of the BC input displacement detected by the laser and the output voltage signal measured at the Pedot electrodes are shown in Figure 3.8. The different amplitudes shown by the displacement signal are due to the presence of a mechanical resonance, due to the cantilever configuration of the device, which occurs when the excitation frequency is between 50 and 60 Hz.

The output signal was filtered with a low-pass filter with a cut-off frequency of 60 Hz, as shown in Figure 3.9. The available dataset was downsampled according to the Nyquist–Shannon criterion by reducing the sampling frequency to 500 Hz.

The dataset was then divided into training, validation (80%) and test (20%) to account for the relevant trends in each subset. The z-score normalization was also applied.

3.6.4 Results Analysis

The experimental results show the effectiveness of black-box models in the characterization of BC-based sensors. The FIR model is the simplest solution due to its linearity and the absence of output regressors. This makes it possible to perform simulations without measuring the past output values. Following a trial-and-error procedure guided by the cross-correlation analysis between the output and input variables, n_b=20 input regressors were used.

However, the performance achieved, which is reported in Tab. 3.5, shows that the results are not satisfactory. Specifically, the FIR model achieves an R^2=0.51 and an

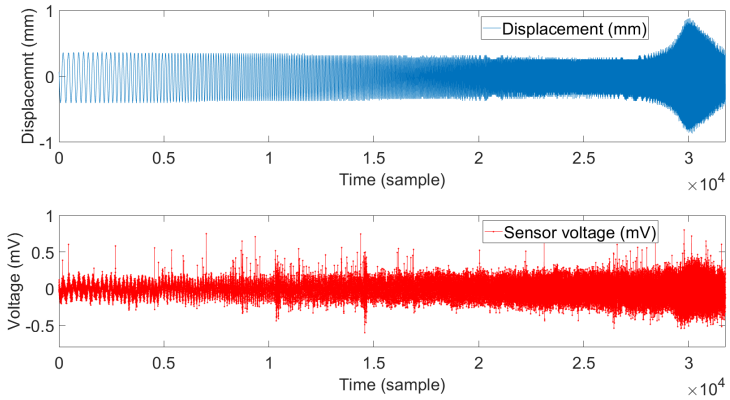


FIGURE 3.8: Time evolution of the tip displacement of the BC (input variable) and voltage acquired between the BC/pedot surfaces (output variable). The sampling frequency used for the modelling phase is 500 Hz.

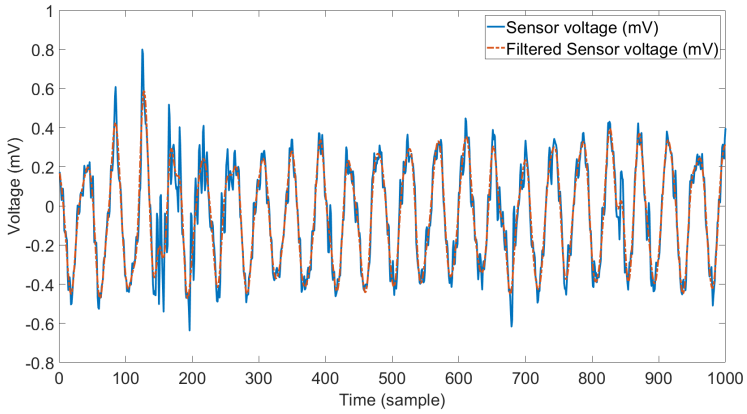


FIGURE 3.9: Output data and corresponding filtered version.

RMSE=0.88 on the test dataset. The regression plot for the FIR model evaluated with the test dataset is shown in Figure 3.10.

To improve the performance of the model, an ARX structure with n_a=10 and n_b=5 was considered. The ARX model exhibits significant improvements in 1-step-ahead predictions and achieves an R^2=0.98 for the test dataset. The ∞-step-ahead prediction, on the other hand, is not satisfactory as the R^2 deteriorates to 0.47 and the RMSE increases from 0.12 to 0.92. The regression plot for the ARX model, which was evaluated on the test set, is shown in Figure 3.11, where both the 1-step-ahead and the ∞-step-ahead predictions are shown.

The NFIR model was implemented using the same number of regressors of FIR ($n_b = 20$) and considering a multilayer perceptron (MLP) with sigmoidal activation functions. The best results were obtained with a single hidden layer of 10 neurons. As shown in Tab. 3.5, a significant improvement was achieved over both the FIR and ARX models and an R^2=0.77 for the ∞- step-ahead prediction was achieved. This is also illustrated by the regression plot in Figure 3.12.

Similarly, a NARX model based on an MLP with a single hidden layer of 10 neurons, $n_a = 5$ and $n_b = 3$ was considered which achieved an improvement in ∞-step-ahead prediction over the ARX model with an R^2=0.69. However, the performance achieved with the NFIR is still better than that of the NARX model. The

TABLE 3.5: Performance of the models in training and test for the 1-step and ∞-step-ahead prediction.

	1-step Prediction						∞-step Prediction					
	Train			Test			Train			Test		
	RMSE	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2
FIR	0.86	0.67	0.49	0.88	0.72	0.51	0.86	0.67	0.49	0.88	0.72	0.51
ARX	**0.15**	**0.10**	**0.98**	**0.12**	**0.09**	**0.99**	0.90	0.72	0.43	0.92	0.76	0.47
NFIR	0.61	0.48	0.74	0.63	0.50	0.77	**0.61**	**0.48**	**0.74**	**0.63**	**0.50**	**0.77**
NARX	0.17	0.13	**0.98**	0.16	0.11	0.95	0.77	0.59	0.50	0.77	0.62	0.69
LSTM	0.17	0.12	0.95	0.18	0.14	0.97	0.70	0.57	0.53	0.88	0.72	0.46

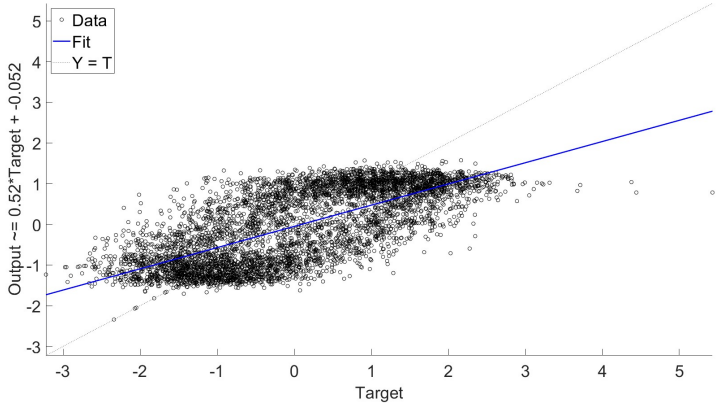


FIGURE 3.10: Regression plot of the FIR model for the test set

regression plot for the NARX model evaluated on the test set is shown in Figure 3.13, where both the 1-step-ahead and the ∞-step-ahead predictions are shown.

An LSTM model was also used for comparison. Good performance is achieved for the 1-step-ahead prediction reaching an R^2=0.97, while poorer results are obtained in the ∞-step-ahead prediction in both the training (R^2=0.53) and test datasets (R^2=0.46). The regression plot for the LSTM model evaluated on the test set is reported in Figure 3.14 where both the 1-step-ahead and the ∞-step-ahead predictions are shown.

The behaviour of the two best models, i.e. NFIR and NARX, is reported in Figures 3.15, 3.16 and 3.17, which show the comparison between the actual and predicted results in the training and two subsets of the test datasets.

The two test subsets were selected to show the behaviour of the models at different excitation frequencies. In particular, the NFIR shows similar performance to the NARX at low input frequencies, while the NARX shows better performance at high frequencies when the 1-step-ahead prediction is considered. For ∞-step prediction, the performance of NARX degrades significantly for both training and test data in the whole frequency range (see Figure 3.17).

The NFIR model has proven to be the best-performing model and shows superior long-term predictive ability relevant for simulation and control purposes. Further experimental campaigns are needed to evaluate the generalization capability of the model.

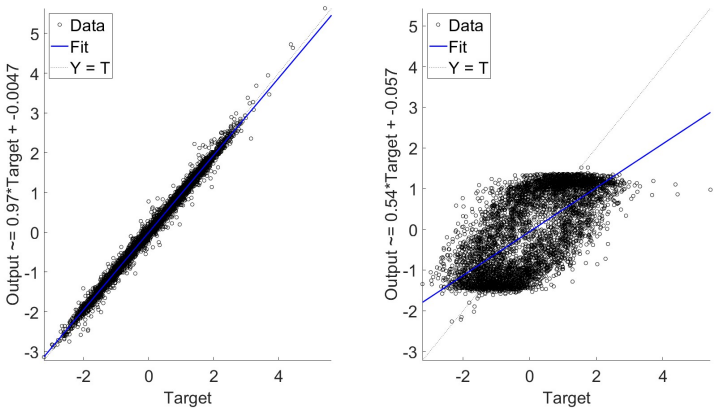


FIGURE 3.11: Regression plot of the ARX model for the test set: 1-step-ahead prediction (left panel), ∞-step-ahead prediction (right panel).

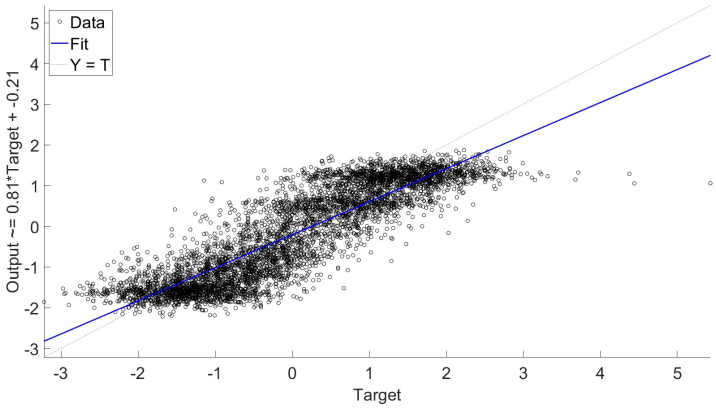


FIGURE 3.12: Regression plot of the NFIR model for the test set.

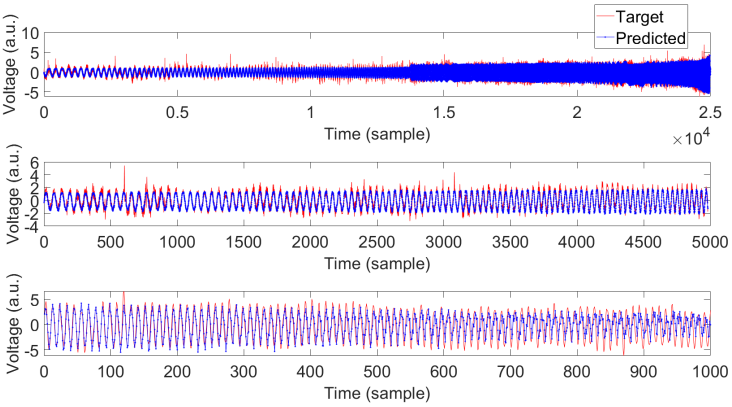


FIGURE 3.15: Comparison between the target and predicted output signals for the NFIR model: training data (first panel), test data (second and third panels).

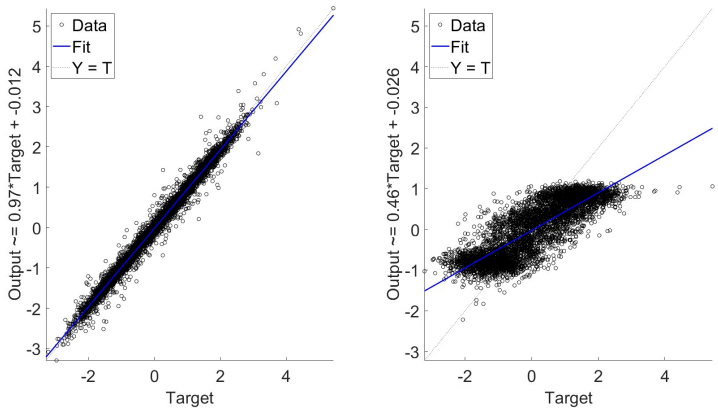


FIGURE 3.13: Regression plot of the NARX model for the test set: 1-step-ahead prediction (left panel), ∞-step-ahead prediction (right panel).

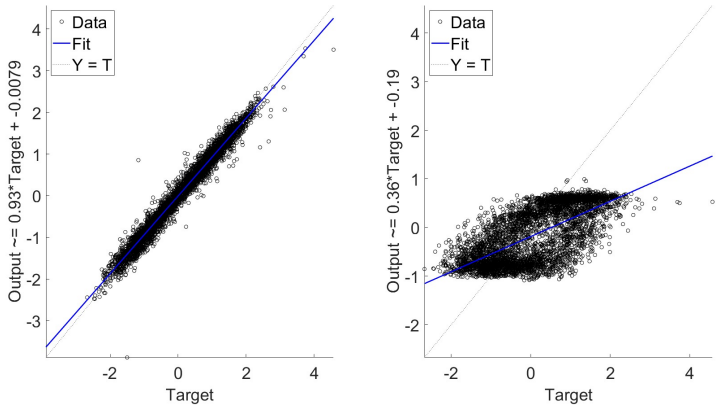


FIGURE 3.14: Regression plot of the LSTM model for the test set: 1-step-ahead prediction (left panel), ∞-step-ahead prediction (right panel) .

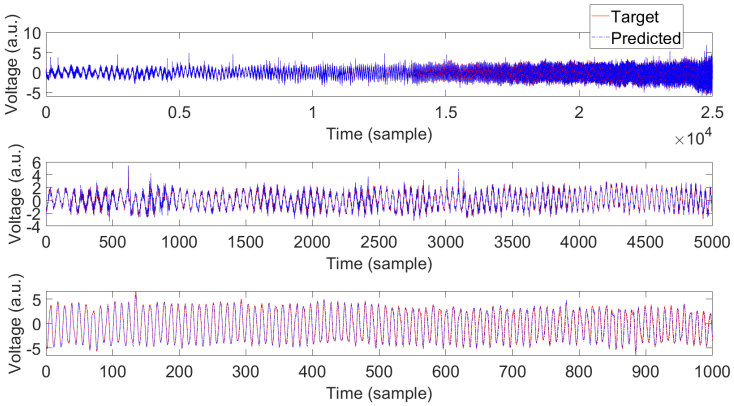


FIGURE 3.16: Comparison between the target and predicted output signals for the NARX model with a 1-step-ahead prediction: training data (first panel), test data (second and third panels).

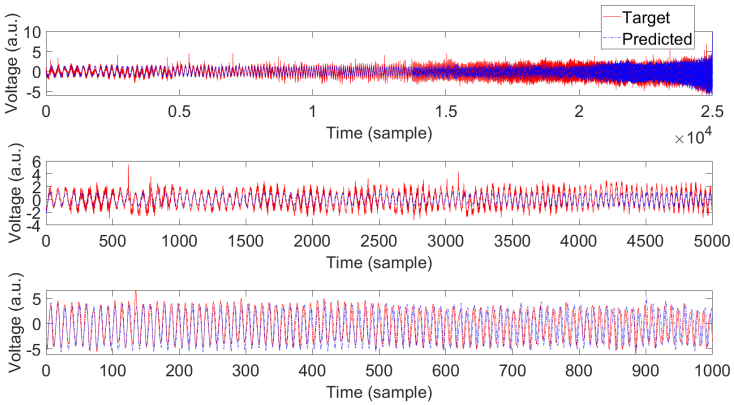


FIGURE 3.17: Comparison between the target and predicted output signals for the NARX model with a ∞-step-ahead prediction: training data (first panel), test data (second and third panels).

3.7 Discussion

This chapter has addressed the modeling of BC-based ionic sensors by combining physics-based and data-driven perspectives.

This work presents, for the first time, a first-principle white-box FEM for the characterization of dual-ion EAP sensors based on BC and ILs. BC is a low-cost, green, and fully biodegradable material that offers significant environmental advantages over synthetic or plant-derived alternatives, particularly in the context of sustainable electronics. It can be produced through bacterial fermentation using green feeding sources, including organic waste such as banana peels, further reducing its environmental impact and making its production process both sustainable and circular. Unlike conventional materials previously employed for similar applications, BC-IL composites require a dual-carrier ionic transport model, an aspect never addressed in the literature until now. By combining a 2D FEM mechanical model with a series of 1D electrochemical sub-models, we have developed a computationally efficient yet physically grounded framework suitable for rapid prototyping. The model accurately represents the mechanoelectrical transduction process through the coupling of pressure-induced ionic transport and potential generation and has been quantitatively validated using experimental data. The analysis shows that the sensor response is predominantly determined by the regions of highest curvature near the fixed clamp, where pressure gradients and advection effects are maximized. The dimensional decomposition significantly reduces computational cost while preserving essential physical behavior, enabling rapid exploration of material and geometric parameters.

The proposed dual-ion framework is general and material-agnostic: replacing material-specific parameters (e.g., elastic modulus, dielectric constant, ion diffusivities) enables direct transfer to other IL-infused polymer matrices (e.g., Nafion, chitosan, graphene-based ionomers). Within the taxonomy in Section 2.4, and how shown in Figure 3.18 the PDE formulation is *white-box* and its FEM realization *knowledge-driven grey*, offering mechanistic clarity but non-trivial computation; this is mitigated by the 2D–1D knowledge-based reduction that exploits the physics of polymer-pressure gradients to lower cost while retaining interpretability.

On the data side, data-driven grey- and black-box identification complements the physics track by delivering fast, accurate surrogates for system-level tasks: ARX

excels in 1-step prediction, NFIR provides the most reliable long-horizon simulations, while NARX and LSTM capture nonlinearities with higher data and training demands. Rather than substituting physics, these models operationalize it: the FEM pipeline reveals where and why signals arise (curvature-driven pressure gradients, advection–diffusion–electromigration interplay), guiding feature design, excitation protocols, and model structure. The data-driven models occupy the *Data-driven Grey* quadrant—linear regression (LR: FIR, ARX) and nonlinear regression (NLR: NFIR, NARX); and the LSTM baseline in the *Black* quadrant.

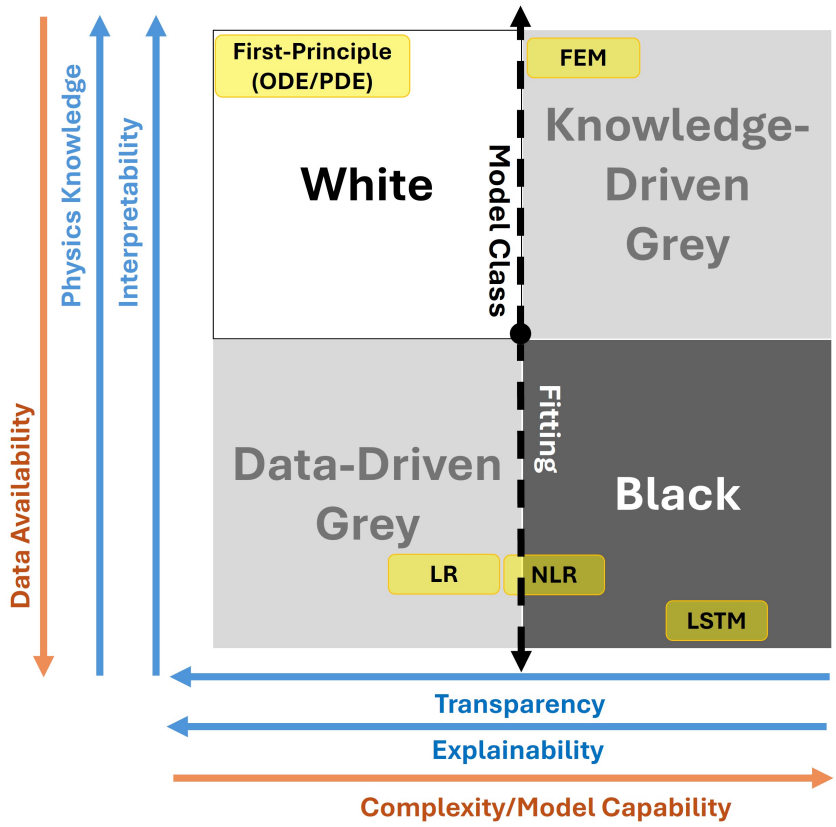


FIGURE 3.18: Placement of the BC-based sensors models on the dual-axis taxonomy (knowledge source vs. transparency). *White*: first-principles PDE of mechanoelectrical transduction. *Knowledge-driven Grey*: FEM implementation of the PDE. *Data-driven Grey*: linear regression (LR: FIR, ARX) and nonlinear regression (NLR: NFIR, NARX). *Black*: LSTM.

Future work will expand the materials and device design space and incrementally enrich the first-principles description. Emphasis will be placed on incorporating field-, rate-, and state-dependent constitutive properties and on modestly strengthening the coupling among mechanics, ion transport, and electrostatics, while preserving parsimony, identifiability, and computational tractability.

A natural next step is physics-informed machine learning. Learners constrained by conservation laws and boundary conditions, complemented by calibrated FEM for targeted synthetic data, will be developed to cover scarcely sampled regimes. This hybrid strategy aims to retain mechanistic transparency and reliability while achieving the speed and adaptability required for scalable design, control, and deployment of green BC–IL sensors.

Acknowledgements This work was funded by European Union - Next Generation EU, Missione 4 Componente 1 CUP J53D23003460006 Green SENSing systems based on Bacterial Cellulose (SENS-BC).

Chapter 4

Soft Sensors for Industrial Applications: Linear Regression, Dynamic Mode Decomposition and Symbolic Regression

4.1 Introduction

Industrial soft sensors (SSs) must deliver *accurate, fast, and interpretable* estimates under shifting operating conditions.

This chapter advances that goal by developing and evaluating two complementary families of models within a unified taxonomy: *Data-driven Grey* linear predictors—AutoRegressive with eXogenous inputs (ARX), Finite Impulse Response (FIR), and a Koopman operator–based Hankel Dynamic Mode Decomposition with control (HDMDc) equipped with Multi-Step-Ahead (MSA) optimization, hereafter *MSA-HDMDc*—and *interpretable symbolic* models based on Genetic Programming (GP)–driven Symbolic Regression (SR) and Sparse Identification of Nonlinear Dynamics (SINDy). We complement these models with post-hoc explanations via SHapley Additive exPlanations (SHAP).

We ground the study in two industrial benchmarks—a Sulfur Recovery Unit (SRU) and a Debutanizer Column (DC)—and a synthetic nonlinear benchmark (the Narendra–Li system), chosen to stress complementary aspects: field realism (SRU/DC) and ground-truth structural recovery (Narendra–Li) (Patanè, Sapuppo, and Xibilia, 2024; Calapristi et al., 2024a; Calapristi et al., 2024b).

The remainder of the chapter is organized as follows. We first outline the problem setting and taxonomy, then present the data-driven grey models and the MSA-HDMDc formulation, followed by the symbolic and deep-learning (DL)–aided SR pipeline. Next, we describe datasets and preprocessing, the experimental protocol and metrics, and results on SRU/DC and on the Narendra–Li benchmark with robustness and interpretability analyses via eXplainable Artificial Intelligence (XAI). We conclude with practical guidelines and implications for deployment.

4.2 Methods: Data-Driven Grey Linear Models

Following a state-of-the-art review of data-driven modeling for complex industrial dynamics—which motivates the MSA-HDMDc approach—this section presents the theoretical and mathematical foundations of the baseline linear models (ARX, FIR) and the HDMDc framework, and then details the MSA-HDMDc algorithm.

4.2.1 State of the Art

The main approach to identifying complex nonlinear dynamical systems originates with Poincare's studies and works on the geometry of subspaces of local linearizations around fixed points, periodic orbits and more general attractors (Guckenheimer and Holmes, 1983). This methodology has a deep theoretical foundation, such as the Hartman–Grobman theorem, which determines when and where it is possible to approximate a nonlinear system with linear dynamics.

On the one hand, such a geometric perspective enables the application of simple quantitative, locally linear models, such as autoregressive (ARX), principal component regression (PCR) and partial least-square regression (PLSR) models (Fortuna et al., 2007; Kadlec, Gabrys, and Strandt, 2009), and proper orthogonal decomposition (POD) (Babaei Pourkargar and Armaou, 2015), as well as the composition of multiple linear systems as components of more complex modeling techniques (Souza and Araujo, 2014; Chen et al., 2019).

In this scenario, the rich linear analytical framework can be used around such operating points and is therefore suitable for linear control strategies. On the other hand, the global analysis remains qualitative and is based on computational analysis, which is not suitable for predicting, estimating and controlling nonlinear systems far from fixed points and periodic orbits. Moreover, due to the complex theoretical environment, data-driven approaches are often used for SSs to support both linear and nonlinear methods (Chen et al., 2019; Liu et al., 2022; Xie et al., 2020; Xie et al., 2023; Dias et al., 2022; Patanè and Xibilia, 2021).

In this methodological and application scenario, the Koopman operator (Koopman, 1931) can provide a theoretical tool for obtaining a global linear representation that is valid for nonlinear systems, even far from fixed points and periodic orbits. A main motivation for the adoption of the Koopman framework is the possibility to simplify the dynamics by the eigenvalue decomposition of the Koopman operator (Mezić, 2005) and thus to represent a nonlinear dynamical system globally by an infinite-dimensional linear operator. It uses a Hilbert space of observable functions related to the state of the system to describe the space of all possible measurement state functions. It is linear and its spectral decomposition fully characterizes the behavior of a nonlinear system, without a direct relation to the operating points of the system.

The application of such a powerful tool to industrial problems, namely to obtain a finite-dimensional approximation of the Koopman operator, is a challenge of recent research (Brunton et al., 2022). Moreover, since the closed form of the Koopman operator is not always obtainable (Brunton et al., 2015), data-driven algorithms are needed. Koopman mode decomposition can be performed using data-driven approaches such as dynamic mode decomposition (DMD) (Brunton and Kutz, 2022).

Applications of DMD can be found in the literature in fluid dynamics (Song et al., 2013; Mezić, 2013), epidemiology (Proctor and Eckhoff, 2015), neuroscience (Alfatlawi and Srivastava, 2019), plasma physics (Taylor et al., 2018; Kaptanoglu et al., 2020), robotics (Bruder et al., 2019), power grid instabilities (Susuki and Mezic, 2012) and renewable energy prediction (Jones and Utyuzhnikov, 2022). DMD represents a method for approximating the Koopman operator that provides a best-fit linear model for one-step-ahead prediction. Such an approximation might not be rich enough to describe nonlinear dynamics. To overcome this limitation and apply DMD to nonlinear industrial processes, it is possible to extend DMD with different strategies based on either nonlinear functions or delayed measurements. Extended DMD (Williams, Kevrekidis, and Rowley, 2015) and sparse identification

nonlinear dynamics (SINDy) (Brunton, Proctor, and Kutz, 2016) belong to the first category. The second category, on which this chapter focuses, is based on the use of delayed state variables obtained by the Hankel operator. Such approaches overcome the limitations of standard DMD, which cannot accurately describe systems where the number of variables is smaller than the spectral complexity. Therefore, in Hankel-based DMD, the number of variables is increased by considering time-delayed vectors in addition to the current state vector. There are a few variants of the Hankel approach for the Koopman operator: Hankel DMD (HDMD) (Arbabi and Mezić, 2016), high-order DMD (HODMD) (Clainche and Vega, 2017) and Hankel alternative view of Koopman (HAVOK) (Brunton et al., 2017). Thanks to the state variable augmentation, these methods are more robust and accurate than classical DMD and are therefore suitable for the identification of nonlinear dynamical systems and offer robust noise filtering (Wu, Brunton, and Revzen, 2021; Clainche, Han, and Ferrer, 2019; Clainche, 2019).

With the aim of applying DMD-based approaches to industrial processes with exogenous inputs, the DMD with control (DMDc) approach has been proposed in the literature (Proctor, Brunton, and Kutz, 2016). This is a modified version of DMD that considers both system measurements and exogenous control inputs to identify input–output relationships and the underlying dynamics. Hankel DMD with control (HDMDc) has recently been introduced to handle both time-delayed state variables and control inputs (Mustavee et al., 2022; Shabab et al., 2021). Applications of Koopman theory for quasiperiodically driven systems have also been presented in (Das et al., 2023).

In this chapter, we propose an extension of the HDMDc approach to multi-step-ahead (MSA) prediction (hereafter referred to as MSA-HDMDc) in the SSs design domain. This solution leverages and exploits the intrinsic HDMDc capability for forecasting (Shabab et al., 2021), operating and, in addition, iterative multi-step-ahead model optimization and output prediction. This makes SSs suitable for the application of model-based online control strategies that are widely used in industrial processes, such as model predictive control (MPC) (Wolfram and Meurer, 2023; Liu et al., 2023; Zhang et al., 2022; Narasingam and Kwon, 2017). To evaluate the potential of the MSA-HDMDc approach in industry, and to test the robustness and reliability of the method in real-world industrial environments considering noise and uncertainty (Lin et al., 2024; Zhang et al., 2020), two widely used benchmarks in the SS field are considered: the sulfur recovery unit (SRU) (Fortuna et al., 2007; Patanè and Xibilia, 2021; Chen et al., 2019; Yuan et al., 2020; Bidar et al., 2018) and the debutanizer column (DC) (Fortuna, Graziani, and Xibilia, 2005; Tian-hao, Yuan-yuan, and Shao-yuan, 2024). Multi-step-ahead prediction of the output variables is evaluated on such datasets, and a comparison with currently adopted linear model identification techniques is performed.

4.2.2 Linear Regression Models

An ARX model set is determined by two polynomials whose degrees are n_a and n_b, respectively:

$$\begin{aligned} A(z^{-1},\theta) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a} \\ B(z^{-1},\theta) &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b} \end{aligned} \tag{4.1}$$

where z^{-1} represents the time delay operator and θ is the set of parameters:

$$\theta := [a_1\ a_2\ \cdots\ a_{n_a}\ b_1\ b_2\ \cdots\ b_{n_b}]^T \tag{4.2}$$

The acronym ARX can be explained in the model equation form for the calculation of $y(t)$, the predicted output at the time instant t:

$$y(t) = \frac{B(z^{-1},\theta)}{A(z^{-1},\theta)}u(t) + \frac{1}{A(z^{-1},\theta)}e(t) \tag{4.3}$$

or equivalently:

$$A(z^{-1},\theta)y(t) = B(z^{-1},\theta)u(t) + e(t) \tag{4.4}$$

where $e(t)$ is a zero-mean white noise process and $u(t)$ is the exogenous input vector.

AR refers to the autoregressive part $A(z^{-1},\theta)y(t)$ in the model, while X refers to the exogenous term $B(z^{-1},\theta)u(t)$. The model set is completely determined once the integers n_a, n_b and the parameter set θ have been specified.

A more general expression including an input/output delay is represented by:

$$y(t) = z^{-n_k}\frac{B(z^{-1},\theta)}{A(z^{-1},\theta)}u(t) + \frac{1}{A(z^{-1},\theta)}e(t) \tag{4.5}$$

where n_k is the number of the input–output delay samples.

When an SS is designed to replace the hardware sensors, the output regressors are not always available. In these cases, an infinite-step prediction should be performed, using as output regressors the past estimated values. As an alternative, it is preferred to not involve output regressors in the system dynamics description, and FIR models can be adopted. FIR is a special case of Equation (4.1), with $n_a = 0$.

4.2.3 Hankel Dynamic Mode Decomposition with Control

The algorithm produces a discrete state-space model, hence the notation for discrete instances, x_k, of the continuous time variable, $x(t)$, is used, where $x_k = x(kT_s)$ and T_s is the sampling time of the model. Delay coordinates (i.e., x_{k-1},x_{k-2}, etc.) are also included in the state-space model to account for state delay in the system. This procedure allows the creation of the augmented state space relevant to model nonlinear phenomena, as discussed in Section 4.1. Therefore, we define a state delay vector as:

$$x_{dk} = [x_{k-1}x_{k-2}\cdots x_{k-q+1}]^T, \tag{4.6}$$

where q is the number of delay coordinates (including the current time step) of the state, with $x_{dk} \in \mathbb{R}^{(q-1)n_x}$ and n_x the number of state variables.

The input delay vector is defined as:

$$u_{dk} = [u_{k-1}\ u_{k-2}\cdots u_{k-q_u+1}]^T, \tag{4.7}$$

where q_u is the number of delay coordinates (including the current time step) of the inputs, with $u_{dk} \in \mathbb{R}^{(q_u-1)n_u}$ and n_u the number of the exogenous input variables.

The discrete state-space function is defined as:

$$x_{k+1} = Ax_k + A_d x_{dk} + Bu_k + B_d u_{dk}, \tag{4.8}$$

where $A \in \mathbb{R}^{n_x \times n_x}$ is the state matrix, $A_d \in \mathbb{R}^{n_x \times (q-1)n_x}$ is the state delay system matrix, $B \in \mathbb{R}^{n_x \times n_u}$ is the input matrix and $B_d \in \mathbb{R}^{n_x \times (q_u-1)n_u}$ is the delay input

matrix. The system output is assumed to be equal to the state, i.e., the output matrix is assumed to be the identity matrix. When dealing with system identification in which only an input/output time series is available, this assumption implies that n_x should be chosen as the size of the process output vector. The training time series consists of discrete measurements of the outputs (i.e., $y_k = x_k$) and corresponding inputs (i.e., u_k).

The training data exploring the augmented state space, thanks to the delay shifts, are organized in the following matrices:

$$X = \begin{bmatrix} x_q & x_{q+1} & x_{q+2} & \cdots & x_{(q-1)+w} \end{bmatrix} \tag{4.9}$$

$$X' = \begin{bmatrix} x_{q+1} & x_{q+2} & x_{q+3} & \cdots & x_{q+w} \end{bmatrix} \tag{4.10}$$

$$X_d = \begin{bmatrix} x_{q-1} & x_q & x_{q+1} & \cdots & x_{(q-1)+w-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_{w+1} \\ x_1 & x_2 & x_3 & \cdots & x_w \end{bmatrix} \tag{4.11}$$

$$X'_d = \begin{bmatrix} x_q & x_{q+1} & x_{q+2} & \cdots & x_{(q-1)+w} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_3 & x_4 & x_5 & \cdots & x_{w+2} \\ x_2 & x_3 & x_4 & \cdots & x_{w+1} \end{bmatrix} \tag{4.12}$$

$$\Gamma = \begin{bmatrix} u_q & u_{q+1} & u_{q+2} & \cdots & u_{(q-1)+w} \end{bmatrix} \tag{4.13}$$

$$\Gamma_d = \begin{bmatrix} u_{q-1} & u_{q+0} & u_{q+1} & \cdots & u_{(q-1)+w-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ u_{(q-q_u)+2} & u_{(q-q_u)+3} & u_{(q-q_u)+4} & \cdots & x_{(q-q_u)+w+1} \\ u_{(q-q_u)+1} & x_{(q-q_u)+2} & u_{(q-q_u)+3} & \cdots & u_{(q-q_u)+w} \end{bmatrix} \tag{4.14}$$

where w represents the time snapshots and is the number of columns in the matrices, X' is the matrix X shifted forward by one time step, X_d is the matrix with delay states and Γ is the matrix of inputs. Moreover, to incorporate the dynamic effect of control inputs, an extended matrix of the exogenous inputs with time shifts (i.e., Γ_d) is created and included in the model. Equation (4.8) can now be combined with the matrices in Equations (4.9)–(4.14) to produce:

$$\begin{bmatrix} X' \\ X'_d \end{bmatrix} = AX + A_d X_d + B\Gamma + B_d \Gamma_d \tag{4.15}$$

Note that the primary objective of HDMDc is to determine the best-fit model matrices, A, A_d, B and B_d, given the data in X', X, X_d, Γ and Γ_d (Proctor, Brunton, and Kutz, 2016). Considering the definition of the Hankel matrix, H, for a generic single measurement time series, h_k, and applying a d time shift:

$$H = \begin{bmatrix} h_d & h_{d+1} & h_{d+2} & \cdots & h_{(d-1)+w} \\ h_{d-1} & h_d & h_{d+1} & \cdots & h_{(d-2)+w} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_1 & h_2 & h_3 & \cdots & h_w \end{bmatrix} \tag{4.16}$$

we can introduce the synoptic notation:

$$X_H = \begin{bmatrix} X \\ X_d \end{bmatrix}, X'_H = \begin{bmatrix} X' \\ X'_d \end{bmatrix}, \Gamma_H = \begin{bmatrix} \Gamma \\ \Gamma_d \end{bmatrix}, \\ A_H = \begin{bmatrix} A & A_d \end{bmatrix}, B_H = \begin{bmatrix} B & B_d \end{bmatrix} \tag{4.17}$$

with $X_H \in \mathbb{R}^{qn_x \times w}$ and $\Gamma_H \in \mathbb{R}^{q_u n_u \times w}$ the Hankel matrices for the time series x_k and u_k, respectively. A_H and B_H are the transformation matrices for the augmented state and inputs, with $A_H \in \mathbb{R}^{qn_x \times qn_x}$ and $B_H \in \mathbb{R}^{qn_x \times q_u n_u}$.

Considering the matrix $\Omega \in \mathbb{R}^{(qn_x + q_u n_u) \times w}$ as the composition of the delayed inputs and outputs, and G as the global transformation matrix described in Equation (4.18).

$$\Omega = \begin{bmatrix} X_H \\ \Gamma_H \end{bmatrix}, \quad G = \begin{bmatrix} A_H & B_H \end{bmatrix}, \tag{4.18}$$

we obtain:

$$X'_H = G\Omega \tag{4.19}$$

A truncated Singular Value Decomposition (SVD) of the Ω matrix results in the following approximation:

$$\Omega \approx \tilde{U}_p \tilde{\Sigma}_p \tilde{V}_p^T \tag{4.20}$$

where the notation $\tilde{}$ represents rank-p truncation of the corresponding matrix, $\tilde{U} \in \mathbb{R}^{(qn_x + q_u n_u) \times p}$, $\tilde{\Sigma} \in \mathbb{R}^{p \times p}$, and $\tilde{V} \in \mathbb{R}^{w \times p}$. Then the approximation of G can be computed as:

$$G \approx X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_p^T \tag{4.21}$$

For reconstructing the approximate state matrices $\tilde{A}_H$ and $\tilde{B}_H$, the matrix $\tilde{U}_p$ can be split in two separate components: $\tilde{U}_{p1}$, related to the state, and $\tilde{U}_{p2}$, related to the exogenous inputs:

$$\tilde{U}_p^T = \begin{bmatrix} \tilde{U}_{p1}^T & \tilde{U}_{p2}^T \end{bmatrix} \tag{4.22}$$

where $\tilde{U}_{p1} \in \mathbb{R}^{qn_x \times p}$ and $\tilde{U}_{p2} \in \mathbb{R}^{q_u n_u \times p}$.

The complete G matrix can be therefore split in:

$$G \approx \begin{bmatrix} \bar{A}_H & \bar{B}_H \end{bmatrix} = \begin{bmatrix} X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p1}^T & X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p2}^T \end{bmatrix} \tag{4.23}$$

Due to the high dimension of the matrices and to obtain further optimization in the computation of the reconstructed system, a truncated SVD of the X'_H matrix results in the following approximation:

$$X'_H \approx \hat{U}_r \hat{\Sigma}_r \hat{V}_r^T \tag{4.24}$$

where the notation $\hat{}$ represents rank-r truncation, $\hat{U}_r \in \mathbb{R}^{qn_x \times r}$, $\hat{\Sigma}_r \in \mathbb{R}^{r \times r}$, and $\hat{V}_r \in \mathbb{R}^{w \times r}$, and typically we consider $r < p$. Considering the projection of the operators $\bar{A}_H$ and $\bar{B}_H$ on the low-dimensional space we obtain:

$$\tilde{A}_H = \hat{U}_r^T \bar{A}_H \hat{U}_r = \hat{U}_r^T X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p1}^T \hat{U}_r \tag{4.25}$$

$$\tilde{B}_H = \hat{U}_r^T \bar{B}_H = \hat{U}_r^T X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p2}^T \tag{4.26}$$

with $\tilde{A}_H \in \mathbb{R}^{r \times r}$ and $\tilde{B}_H \in \mathbb{R}^{r \times q_u n_u}$ The approximated discrete-time system based on

the Hankel transformation of the original time series (i.e., $\tilde{x}_k^H$ and u_k^H) can be therefore represented as:

$$\tilde{x}_{k+1}^H = \tilde{A}_H \tilde{x}_k^H + \tilde{B}_H u_k^H \tag{4.27}$$

with $x_k^H = \hat{U}_r \tilde{x}_k^H$. The original time series x_k is then extracted from x_k^H considering only the rows with index $i = n \cdot q + 1$, where $n = 0, 1, \ldots, (n_x - 1)$.

Figure 4.1 clarifies the HDMDc procedure at a higher level. The input and output measurement data from the historical dataset of an industrial process are fed into the HDMDc block that performs in sequence the Hankel transformation of the input/output variables, the merging of the state and input matrices, which integrates the control signals (DMDc), and then the model identification procedure (DMD). The DMD performs a space transformation, based on the SVD, and returns the reduced estimated state-space system representation used for the output multi-step-ahead prediction.

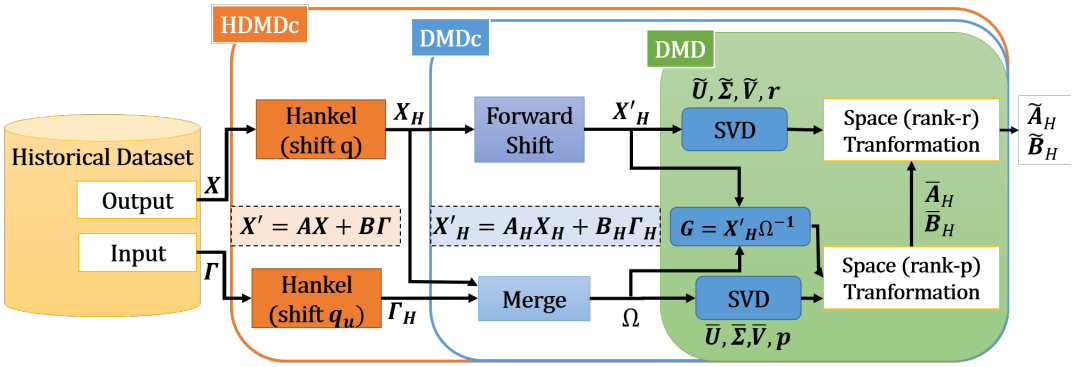


FIGURE 4.1: HDMDc block scheme.

4.2.4 Multi-Step-Ahead Hankel Dynamic Mode Decomposition with Control

In this section, the proposed MSA-HDMDc method is described and the procedure is presented in Algorithm 1. It performs a model optimization and multi-step-ahead prediction of the process output on the basis of a model identified using HDMDc. The model optimization is based on a cost function depending on a combination of key performance indicators (KPIs) such as the mean average percentage error ($MAPE_{\%}$) and the coefficient of determination (R^2), adopted for the comparison of the estimated output to the measured target.

Moreover, to compare a set of models to a chosen baseline (bl) one, the performance improvement ($PI\%$) index is defined for each KPI as:

$$KPI_{PI\%} = \frac{(T - KPI_{bl}) - (T - KPI_{newmodel})}{(T - KPI_{bl})}\% \tag{4.28}$$

with $T = 0$ for $MAPE_{\%}$ and $T = 1$ for R^2.

As a preliminary step, the optimal delay shifts q^{opt} and q_u^{opt}, for the state and the input, respectively, should be determined in sequence. The selection is performed by applying the HDMDc algorithm without order reduction, adopting an optimization algorithm (e.g., grid search strategy), and comparing the prediction performances in terms of $MAPE_{\%}$ and R^2, as will be shown in Section 4.5.1.

The MSA-HDMDc algorithm is described in the following. The acquired input/output data samples are required to perform the model identification procedure.

The training dataset is used to create the X and Γ matrices with the output and inputs, respectively. The state and input augmentation (X_H and Γ_H) are performed by applying the Hankel operator to the original measurements, and the extended state matrix, Ω, is obtained by appending the X_H and Γ_H matrices. The core SVD algorithm is performed on the Ω and X'_H matrices. The iteration for model optimization on multi-step-ahead prediction is then performed by determining, in sequence, the optimal reduction for the Ω and X'_H matrices. The core of the model reduction and reconstruction is performed in the function *Reconstruct* in Algorithm 2. It performs four operations:

- The state matrices order truncation (p,r) for model reduction;
- The determination of the HDMDc operators $\tilde{A}_H$ and $\tilde{B}_H$ as the state space representation of the identified reduced model;
- The iterative reconstruction of the multi-step-ahead estimated state in the reduced state space ($\tilde{X}_H$), for each selected time horizon within K_{max};
- The remapping of the reduced state variables to the original augmented state space ($\hat{X}_H$);
- The extraction of the original state variables from $\hat{X}_H$, selecting the rows related to the first time shift or each state variable;
- The evaluation of the $MAPE_{\%}$ and R^2, comparing the model predictions with the target output.

The model reduction is performed in two steps. In the first step, the Ω matrix is reduced by adopting different order reductions, $p \in p_{range}$, selected on a grid with maximum value $p_{max} = n_x q^{opt} + n_u q_u^{opt}$. In this phase, the matrix X'_H is kept to the full-order $r = r_{max} = n_x q$. The optimal reduction order, p^{opt}, is determined by maximizing the cost function, f_p, over the p_{range}. The cost function consists of a linear combination of the adopted KPIs ($MAPE_{\%}$ and R^2) evaluated on the maximum prediction time step, K_{max}. The optimization is performed using the validation data contained in the training dataset. In the second phase, the matrix X'_H is reduced by adopting different order reductions $r \in r_{range}$, selected on a grid with maximum value $r_{max} = min(n_x q^{opt}, p)$. The reduced matrix, Ω, with order $p = p^{opt}$ is here considered. The optimal reduction order, r^{opt}, is determined by maximizing the cost function, f_r, over the r_{range}. The multi-step-ahead prediction can be performed by reconstructing the output dynamics of the optimal identified model.

4.3 Methods: Symbolic & XAI

This section introduces three complementary strands of interpretable modeling. First, we present SR driven by Genetic Programming (GP) — covering Operon's GP–nonlinear–least-squares variant and PySR—to discover closed-form equations directly from data. Second, we review Sparse Identification of Nonlinear Dynamics (SINDy), which infers governing equations by selecting a parsimonious subset from a library of candidate terms. Third, we describe a Deep Learning (DL) - SR pipeline that learns compact latent encodings with DL and then fits transparent SR models on those encodings. Finally, we apply post-hoc XAI to probe both black-box components and discovered formulas. Throughout the synoptic results tables, we report the evaluation metrics defined in the methodological framework chapter (Section ii)) to ensure consistent interpretation across methods.

Algorithm 1 MSA-HDMDc Algorithm

Given the training datasets $X \in \mathbb{R}^{n_x \times w}$, $\Gamma \in \mathbb{R}^{n_u \times w}$;
$X = [x_1\ x_2\ \cdots\ x_{m-1}]$
$X' = [x_2\ x_3\ \cdots\ x_m]$
$\Gamma = [u_1\ u_2 \cdots\ u_{m-1}]$
Set K_{max} as the maximum prediction time step value
$p_{max} = n_x q^{opt} + n_u q_u^{opt}$
$r_{max} = min(n_x q^{opt}, p)$
Set p_{range} and r_{range} with $r < p$ and $r < r_{max}$
X_H=Hankel(X, q^{opt});
X'_H=Hankel(X', q^{opt})
Γ_H=Hankel(Γ, q_u^{opt})
$\Omega = [X_H^T\ \Gamma_H^T]^T$
Compute the SVD of $\Omega = \tilde{U}\tilde{\Sigma}\tilde{V}^T$
Compute the SVD of $X'_H = \hat{U}\hat{\Sigma}\hat{V}^T$
for $p \in p_{range}$ **do**
  $[MAPE_{\%(p,r_{max})}, R^2_{(p,r_{max})}]$=*Reconstruct* ($p$,$r_{max}$,$K_{max}$)
end for
$[MAPE_{\%bl_p}, R^2_{bl_p}]$=*Reconstruct* ($p_{max}$,$r_{max}$,$K_{max}$)
$f_p(MAPE_{\%(p,r_{max})}, R^2_{(p,r_{max})}, MAPE_{\%bl_p}, R^2_{bl_p}) =$
$MAPE_{PI\%} + R^2_{PI\%}$
$p^{opt} = \underset{p \in p_{range}}{\arg\max} f_p(p, r_{max})$
for $r \in r_{range}$ **do**
  $[MAPE_{\%(p^{opt},r)}, R^2_{(p^{opt},r)}]$= *Reconstruct* ($p^{opt}$,$r$, K_{max})
end for
$[MAPE_{\%bl_r}, R^2_{bl_r}]$=*Reconstruct* ($p^{opt}$,$r_{max}$,$K_{max}$)
$f_r(MAPE_{\%(p^{opt},r_{max})}, R^2_{(p^{opt},r_{max})}, MAPE_{\%bl_r}, R^2_{bl_r}) = MAPE_{PI\%} + R^2_{PI\%}$
$r^{opt} = \underset{r \in r_{range}}{\arg\max} f_r(p^{opt}, r)$
$[MAPE_{\%(p^{opt},r^{opt})}, R^2_{(p^{opt},r^{opt})}]=$
Reconstruct (p^{opt}, r^{opt}, K_{max})

Algorithm 2 Reduction and Reconstruction Algorithm

function [$MAPE_{\%}$, R^2] =*Reconstruct (p,r,K)*
Truncate the SVD matrices at order p: $\Omega \approx \tilde{U}_p \tilde{\Sigma}_p \tilde{V}_p^T$
Truncate the SVD matrices at order r: $X'_H \approx \hat{U}_r \hat{\Sigma}_r \hat{V}_r^T$
Compute the HDMDc operators $\tilde{A}_H = \hat{U}_r^T X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p1}^T \hat{U}_r$
and $\tilde{B}_H = \hat{U}_r^T X'_H \tilde{V}_p \tilde{\Sigma}_p^{-1} \tilde{U}_{p2}^T$
for $j = 1, 2, \cdots, w - K_{max}$ **do**
  Get x_j^H, u_j^H from the datasets X_H and Γ_H
  $\tilde{x}_j^H = \hat{U}_r^T x_j^H$
  for $k = 0, 1, \cdots, K_{max} - 1$ **do**
    $\tilde{x}_{j+k+1}^H = \tilde{A}_H \tilde{x}_{j+k}^H + \tilde{B}_H u_{j+k}^H$
    $\hat{x}_{j+k+1}^H = \hat{U} \tilde{x}_{j+k+1}^H$
  end for
  $\hat{X}_H = [\hat{x}_1^H \ \hat{x}_2^H \ \cdots \ \hat{x}_{w-K_{max}}^H]$
  Select $\hat{X}_H$ rows for the first time shift for each state variable
  Compute $MAPE_{\%}$ and R^2
end for
Return $MAPE_{\%}$ and R^2 for the prediction step K

4.3.1 GP-based SR

Genetic Programming (GP) is an evolutionary algorithm that aims to obtain accurate candidate solutions by starting from a population of unsuitable, randomly generated solutions and improving the individuals in the population through the processes of reproduction, mutation and selection in a Darwinian manner. The evolutionary process usually takes place in small tournaments between the individuals of each generation, where each equation is evaluated with a fitness function and the fittest individual of the tournament is used for the genetic operations, with the results being passed on to the next generation of individuals. The members of later generations are on average better than their immediate parents and thus slowly converge to the best candidate solution. Termination usually occurs when a certain fitness value is reached or all remaining individuals converge to a single optimal solution. However, premature termination is possible if the offspring reach a local optimum. To avoid this, the mutation operator introduces random changes in the equation tree that diversify the search space and thus increase the chances of reaching a global optimum. An improved variant of the classical GP algorithm is the GP-nonlinear least square (GP-NLS), which introduces a local search mechanism for the tree-based equation that transfers the burden of searching for suitable numerical coefficients to an already established equation. This approach utilises the search capabilities of GP as well as the computational efficiency of iterative nonlinear least squares algorithms such as the Levenberg-Marquardt algorithm (Levenberg, 1944) in finding appropriate coefficients for each of the tree leaves, which improves both the computational time and the accuracy of the proposed solutions.

i) Operon

The Operon framework implements a variation of the original GP algorithm called GP-NLS that uses NLS algorithms (Kommenda et al., 2020). As with the original GP implementation, the algorithm can be divided into crossover, mutation and selection

phases, with the addition of a NLS problem in the computation. The NLS introduces a suitable local search mechanism for the tree-based equations, which searches for suitable numerical coefficients for an already established equation. The algorithm requires that an offset and scaling term is added at the root of the generated tree and an additional coefficient is assigned to each feature node. Then all newly added free parameters are adjusted using a nonlinear least squares method, that can be solved iteratively using the Levenberg–Marquardt algorithm.

ii) PySR

An improved version of the GP-NLS algorithm is contained in the PySR framework (Cranmer, 2023), in which the search process has been modified. Specifically, the mutation step introduces the possibility of rejecting a mutation with a certain probability. The diversity of individuals can either be discouraged or encouraged so that both search phases and fine-tuning phases are possible. In addition, a *simplification* phase has been introduced in the loop of the algorithm to reduce the number of elements in the tree. The removal of unsuitable elements is already handled by the classical GP approaches but can lead to undesirable results during the mutation phases. Moreover, an *adaptive parsimony* mechanism has been introduced when evaluating the complexity of a solution. Simpler solutions are usually preferred in symbolic regression as they are generally easier to interpret for a human user. They are traditionally enforced in many GP algorithms with a fitness value that takes into account both the loss and the number of nodes in the tree. In this way, the algorithm can return a set of equations along with a score that takes into account the improvement in loss with respect to the increase in complexity.

4.3.2 SINDy

Sparse Identification of Nonlinear Dynamics (SINDy) is a popular alternative to GP, especially for the study of PDEs and general physical dynamical systems (Fasel et al., 2021). The idea behind SINDy is that an interpretable form of nonlinear dynamics can be expressed by the sparse linear combination of terms in a library of candidate functions. This approach is based on the assumption that real physical systems are characterized by only a few dominant terms that determine the system dynamics and influence the data and the relative rate of change. A system can then be analyzed by plotting its trajectories in a time series matrix $\boldsymbol{X}$, together with the associated derivative matrix $\dot{\mathbf{X}}$, which can be calculated using a numerically appropriate method. A sparse representation of the dynamical system can then be expressed as follows:

$$\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi} \tag{4.29}$$

where $\boldsymbol{\Theta}$ is a library of candidate functions and $\boldsymbol{\Xi}$ is a sparse coefficient matrix. The sparsity can be promoted using *l0*, *l1*, or *l2* normalization with the addition of a threshold in the sequential thresholded least squares (STLSQ) algorithm. The λ coefficient in the STLSQ algorithm serves as a sparsity-promoting threshold, defining a minimum value for coefficients in the weight vector. Coefficients with values below this threshold are set to 0, thereby influencing the regularization process.

This model can be further generalized by including control inputs in the calculation and discretizing the algorithm itself. The SINDy algorithm was developed to predict the derived terms for the system of interest. This may be a suitable approach for systems with continuous dynamics, but cannot be applied to discrete systems or

those with physical boundaries that do not allow an appropriate sampling rate. For these reasons, the equation 4.29 can be rewritten as:

$$\boldsymbol{X}_{n+1} = \boldsymbol{\Theta}(\boldsymbol{X}_n, \boldsymbol{U}_n)\boldsymbol{\Xi} \tag{4.30}$$

This generalization allows the formulation of a one-step predictor that can be transposed to other established models in the field of industrial modelling by carefully evaluating the variables $\boldsymbol{X}$ and $\boldsymbol{U}$.

4.3.3 Deep Learning aided SR

The use of Deep Learning (DL) in combination with SR becomes extremely valuable when a problem is either too complex to be tackled with pure SR or the feature space is too large to obtain a meaningful equation with the regression method. In the context of the development of SS for dynamical systems, each input feature expands the feature space by a factor n, where n is the number of input regressors considered. This extension significantly increases the dimensionality of the problem. The chosen strategy to cope with this complexity is to create a relatively simple Deep Neural Network structure that can reduce the feature space. This is achieved by partitioning the input features and encoding each partition into intermediate features, effectively decomposing the problem into sub-problems of lower complexity that can be treated individually. Although different network architectures can be explored, a successful approach consists in creating a subnetwork for each input and its associated regressors. These subnetworks process their respective inputs and autonomously generate a dynamic encoding for each input feature. These encoded variables are then fed into another network that combines the encodings to produce the final output prediction. This structure introduces sufficient input sparsity enabling the SR model to accurately match both the encodings learned by the DL model and the final output. Once the encoding equations are learned, the intermediate predictions can be recalculated using these equations and then used as training data for the next regression step. This step has been shown to be very effective as it mitigates residual errors that would be amplified in the subsequent prediction, while exploiting the difference in the expressive power of SR and NN to generalize the generated encodings. The proposed algorithm is illustrated in a flowchart in Figure 4.2. This approach has proven to be more effective than traditional SS development methods such as the least squares method, as deeper levels of complexity and nonlinearity are achieved. It also offers better interpretability compared to simple DL black box models. The integration of DL enables data-driven sparsification, leading to a two-stage SR process that learns both the encodings and the final equation, which can be reordered as a composite function of the input features. However, it should be noted that while this method can handle greater complexity compared to traditional SS approaches, the resulting equations can also become very complex. Depending on the NN architecture and application domain, complex equations may not provide useful insights to experts in the field. Therefore, mastering the final complexity should be one of the main goals when developing the NN architecture, while reducing the length and depth of the equations in the SR phase.

4.3.4 eXplainable Artificial Intelligence: SHAP

Shapley Additive Explanations (SHAP) (Lundberg and Lee, 2017a) is a method of explaining individual predictions based on the game-theory concept of Shapley values, originally introduced by Lloyd Shapley as a method of dividing the payoff

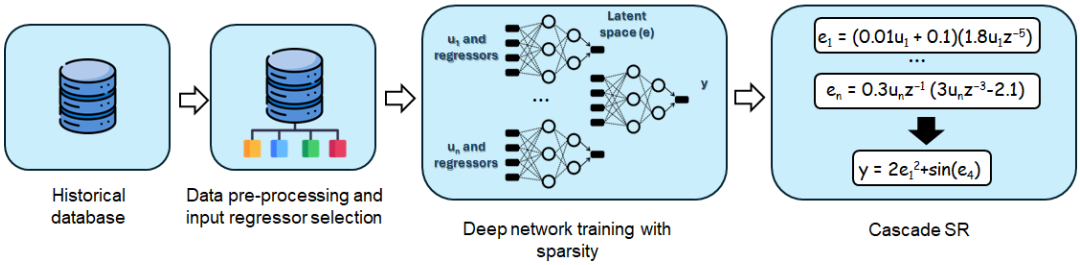


FIGURE 4.2: Deep Learning aided Symbolic Regression flowchart.

between players in coalition games so that each payoff is proportional to how much each player has contributed to the coalition. The idea behind SHAP is to explain the prediction of an instance x by calculating how much each feature increased or decreased the final prediction. This is achieved by representing the explanatory values as an additive feature assignment method. This means that the Shapley values, as proposed in LIME (Local Interpretable Model-Agnostic Explanations) (Ribeiro, Singh, and Guestrin, 2016), are computed on a linear surrogate model that is specifically trained to approximate the local prediction of the underlying model and express the explanatory values as a linear regression task. Furthermore, a surrogate model can be based on simplified inputs x', which requires the introduction of a mapping function that transforms the input into a simplified version of itself such that $x = h_x(x')$. Each explanation can be expressed as the sum of the average prediction and the contributions of the features, which must add up to the difference between the prediction for the instance x and the average prediction.

$$f(x) = g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j x'_j = E_x(\hat{f}(X)) + \sum_{j=1}^{M} \phi_j \tag{4.31}$$

where $E_x(\hat{f}(X))$ represents the average prediction, while ϕ_j represents the Shapley value for the input feature j.

4.3.5 Interpretability metrics.

We report the following interpretability metrics used later in Results and defined in Section 2.5:

- **Stability (S) as in Equation 2.13**: can be defined as the degree to which the local explanation for a given point changes compared to its neighbours. A high value of stability indicates that a small change in the explained feature causes a significant change in the explainer, which means that the explanation is not reliable because it is not locally stable.

- **Infidelity (INFD) as in Equation 2.14** : the measure of explanation infidelity focuses on the idea that given the subset of most relevant feature explanations, the explainer should ascribe high values even after a significant perturbation is applied.

- **Jaccard stability ((J_1, J_2, J_3))** as in Equation 2.15: it expresses the similarity between two sets. Formally, it is defined as the cardinality of the intersection over the cardinality of the union. This metric can be extended in terms of explainability by considering the k most important measured features from the

explanation, ordered by their relevance, and registering their change within the neighbourhood.

- **Adherence (Adh)** as in Equation 2.16: it represents the difference between the original nodes coefficients and the regressed ones for each common node. The applicability of the metric is limited to situations where the underlying equation is either already known or where there is some prior knowledge of the system that facilitates the formulation of a ground truth equation.

4.4 Case Studies

This section contains the description of two case studies that are widely used in the field of SSs. Both are from the petrochemical sector and are complex systems with exogenous inputs and strong nonlinearities.

4.4.1 The Sulfur Recovery Unit (SRU)

The SRU desulfurization unit considered here is located in a refinery in Sicily (Italy), as described in (Fortuna et al., 2003). SRUs in refineries are used to recover elemental sulfur from gaseous hydrogen sulfide (H_2S) contained in by-product gasses produced during the refining of crude oil and other industrial processes. Since H_2S is a hazardous environmental pollutant, such a process is of fundamental importance.

The inlets of each SRU line receive two acid gases: MEA gas, rich in H_2S, and SWS (Sour Water Stripping) gas, rich in H_2S and ammonia (NH_3). These input gases are combusted in two separate chamber reactors fed by a suitable airflow supply for combustion control. The output gas stream contains residues of H_2S and sulfur dioxide (SO_2). Normally, the ratio of H_2S to SO_2 in the tail gas must be maintained, which is specified by a setpoint. An additional secondary airflow is used as an input to improve process control. This variable is the output of a feedback control system and is used to reduce the peak values of H_2S and SO_2.

Figure 4.3 represents a working scheme for an SRU line. The application of SSs is therefore necessary to estimate such concentrations.

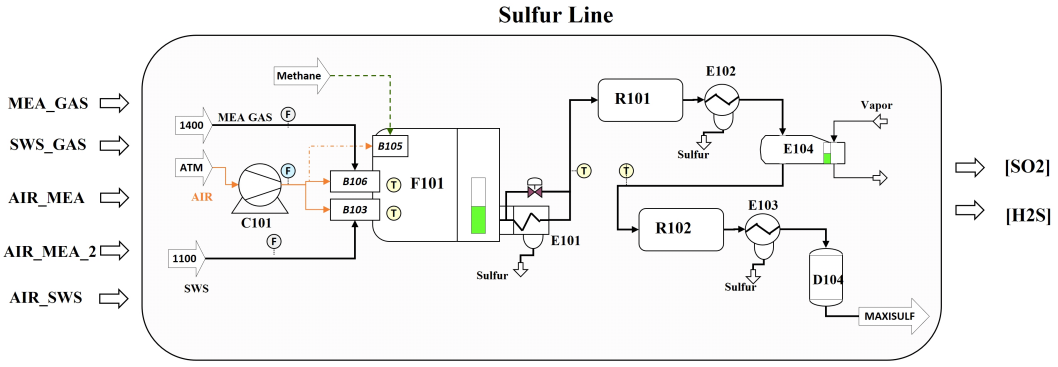


FIGURE 4.3: SRU line working scheme.

The input and output variables available in the SRU historical dataset are listed in Table 4.1. MSA-HDMDc is used to estimate the H_2S concentration (i.e., y_1 output).

TABLE 4.1: Input and output variables of the SRU models.

Variable	Description
$u^{(1)} = MEA\_GAS$	gas flow in the MEA chamber (NM^3/h)
$u^{(2)} = AIR\_MEA$	airflow in the MEA chamber (NM^3/h)
$u^{(3)} = SWS\_GAS$	total gas flow in the SWS chamber (NM^3/h)
$u^{(4)} = AIR\_SWS$	total airflow in the SWS chamber (NM^3/h)
$u^{(5)} = AIR\_MEA_2$	secondary air flow (NM^3/h)
$y_1 = [H_2S]$	H_2S concentration (output 1) (mol%)
$y_2 = [SO_2]$	SO_2 concentration (output 2) (mol%)

4.4.2 The Debutanizer Column (DC)

The column is located in ERG Raffineria Mediterranea s.r.l. (ERGMED) in Syracuse, Italy, and is an integral part of a desulphurization and naphtha splitting plant. In the DC, propane (C3) and butane (C4) are extracted from the naphtha stream as overheads (Fortuna, Graziani, and Xibilia, 2005).

The DC is required to:

- Ensure sufficient fractionation in the debutanizer;
- Maximize the C5 content (stabilized gasoline) in the debutanizer overhead (LP gas splitter feed) while complying with the legally prescribed limit;
- Minimize the C4 (butane) content in the bottom of the debutanizer (feed to the naphtha splitter).

A detailed schematic of the debutanizer column is shown in Figure 4.4. It includes the following components:

- E150B heat exchanger;
- E107AB overhead condenser;
- E108AB bottom reboiler;
- P102AB head reflux pump;
- P103AB feed pump to the LPG splitter;
- D104 reflux accumulator.

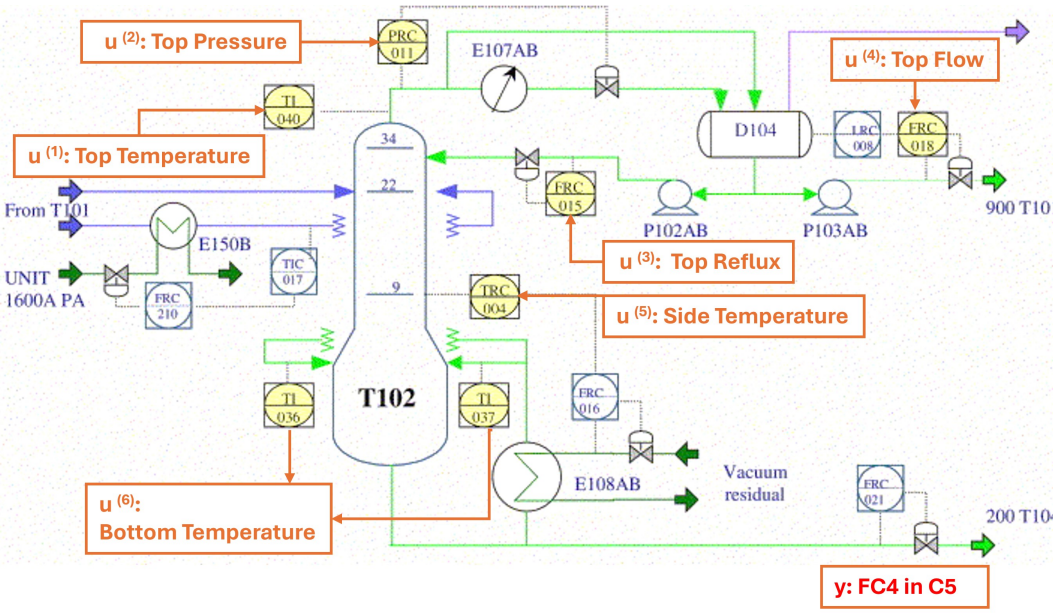


FIGURE 4.4: Schematic representation of the debutanizer column (DC) with indication of the location of the hardware measuring devices, the model exogenous input, u, and soft sensor model output, y.

A number of hardware sensors are installed in the plant to monitor product quality. The subset of sensors that are relevant for the described application are listed in Table 4.2.

TABLE 4.2: Input and output variables of the DC models.

Variable	Description
$u^{(1)} = T040$	top temperature (°C)
$u^{(2)} = P011$	top pressure (Kg/cm^2)
$u^{(3)} = F015$	top reflux (m^3/h)
$u^{(4)} = F018$	top flow (m^3/h)
$u^{(5)} = T004$	side temperature (°C)
$u^{(6)} = (T036 + T037)/2$	$T036$ and $T037$ bottom temperatures (°C)
$y = F_{C4}$	C4 concentration in the bottom flow (%)

The C4 concentration in the bottom flow is estimated as the output of the designed SS. It is not detected on the bottom stream, but at the overheads of the deisopentanization column. The C4 content in the C5 depends solely on the operating conditions of the debutanizer: it can be assumed that the C4 detected in the C5 stream is that which flows from the bottom of the debutanizer. Due to the location of the analyzer, the concentration values are determined with a long delay, which is not exactly known but is constant and probably in the range of 30 min. To improve the control quality of the DC, real-time estimation of both C4 content and C5 content is required. For this purpose, a virtual sensor is needed, which is described in the following sections.

4.4.3 Narendra-Li Synthetic Benchmark

In the field of dynamical systems, the Narendra-Li system is a remarkable example often used as a testbed (Gedon et al., 2021). The equations of the system, discussed in

(Narendra and Li, 2013), represent a complex nonlinear discrete system defined as follows:

$$\begin{aligned} x_{1,n+1} &= \left(\frac{x_{1,n}}{1 + x_{1,n}^2} + p_1 \right) \sin(x_{2,n}) \\ x_{2,n+1} &= x_{2,n} \cos(x_{2,n}) + x_{1,n} \exp\left\{ \left(-\frac{x_{1,n}^2 + x_{2,n}^2}{p_2} \right) \right\} \\ &\quad + \frac{u_n^3}{1 + u_n^2 + p_3 \cos(x_{1,n} + x_{2,n})} \\ y_n &= \frac{x_{1,n}}{1 + p_4 \sin(x_{2,n}) + p_5 \sin(x_{1,n})} \end{aligned} \tag{4.32}$$

In this representation, $x_{1,n}$ and $x_{2,n}$ denote the states, u_n represents the input signal, y_n is the output signal and p is a vector of 5 parameters specifically chosen for this study as $p = [2.2, 1.4, 1, 0.5, 0.2]$. The deliberately complex and uniquely nonlinear nature of this second-order system ensures that traditional linear system identification and control techniques are not applicable. The SINDy library was designed to include function operators such as $+, -, *, /, sin, cos, exp$ among the interactions of the regressors up to the second order, as well as a bias term. In contrast, in the GP-based regressors, the same operators were included in the operator pool, with the exception of the *cos* function, which could still be derived from the *sin* function during evolution to avoid operator redundancy. In addition, the nesting of sine and exponential functions was explicitly forbidden in the PySR regressor. Furthermore, the *sin* and *exp* operators were assigned complexity values of 3 and 6 respectively whereas the other operators present a value of 1. These restrictions are aimed at narrowing down the search space, which leads to a solution that is easier to interpret. The maximum equation length for both GP regressors was set to 50, slightly more than the desired length, to give the regressor additional leeway for searching and fine-tuning the result. Finally, white Gaussian noise was chosen as the system input u, as is common in classical system identification. The reason for this choice lies in the ability of white noise to effectively excite all frequency components of the system.

4.5 Results: Data-Driven Grey Linear Models

4.5.1 SRU Prediction Analysis

In this section, the proposed MSA-HDMDc is applied to the SRU case study and the results are compared with the baseline models. The available dataset consists of about 14,000 samples, with a sampling period of one minute. In addition, 70% of the dataset is used for model training, 15% for the validation of the hyperparameters and model order reduction and, finally, the testing is performed on the remaining 15% of the dataset.

i) Hyperparameter Tuning

An MSA prediction is performed for the output, y_1. To validate the performance of the procedure, a time horizon of $K_{max} = 30$ steps is selected. To better show the effect of the MSA prediction for different time horizons, values in the interval $K \in \{1, 5, 10, 15, 20, 25, 30\}$ are considered.

Linear Regression Models Two linear models were considered: ARX and FIR. The optimal model order was selected based on the minimum description length (MDL) criterion (Ljung, 1986). In particular, the ARX structure was identified to have eight common poles ($n_a = 8$), two zeros ($n_b = 3$), and no delay for all input variables ($n_k = 0$), while the FIR order parameters resulted to be $n_b = 10$ and $n_k = 0$.

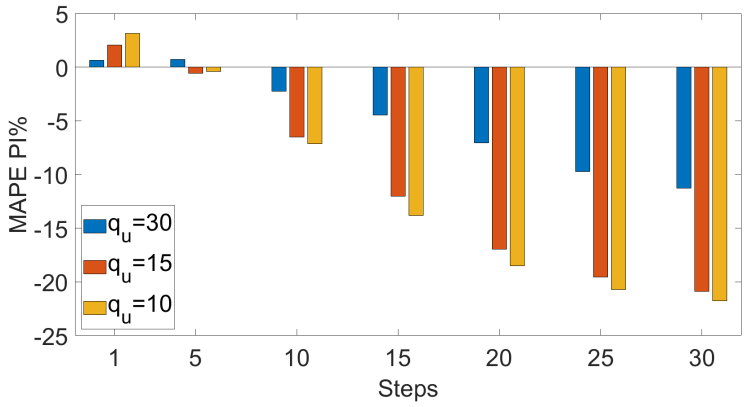
MSA-HDMDc Model In a preliminary phase of the iterative MSA-HDMDc procedure, a parametric study was performed based on model performance in terms of $MAPE_{\%}$ and R^2. Such a procedure allowed the definition of the hyperparameters related to the delay shifts q and q_u applied to the state and inputs, respectively.

The first step was to determine the optimal state variable delay shift, q. A grid search strategy was applied and, for each q that lies between $q = 20$ and $q = 60$ with a step of 10, the full-rank HDMDc model was identified. The estimated output reconstructed at the maximum prediction step (i.e., 30 steps) was compared with the measured output, y_1. For statistical analysis, the validation dataset was divided into 20 subsets of 100 samples. $MAPE_{\%}$ and R^2 were evaluated for each subset, and the corresponding distribution was determined. In particular, the median value of $MAPE_{\%}$ was used to select the optimal hyperparameter, q.

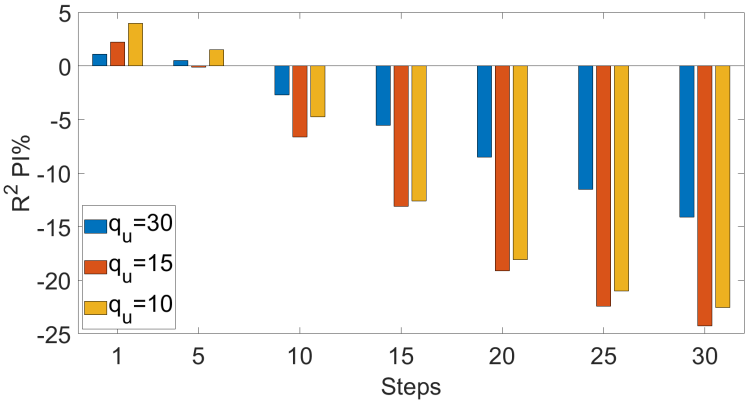
Table 4.3 shows the mean value of $MAPE_{\%}$ over the 20 trials considering a 30-step-ahead prediction for the selected $q \in \{20, 30, 40, 50, 60\}$. The best performing model corresponds to $q^{opt} = 40$, as shown by the reported $PI\%$. In a second step, assuming that the optimal value $q^{opt} = 40$ is fixed, a further parametric optimization was performed by varying the Hankel shift, q_u, applied to the exogenous control inputs in the Γ_H matrix. Since $q_u \leq q^{opt}$, for the system to be causal, the model was identified for each q_u belonging to the set $q_u \in \{10, 15, 30, 40\}$. The estimated output, reconstructed at each considered prediction step, was compared with the measured output, y_1. Considering the model with $q_u = q^{opt} = 40$, corresponding to the maximum allowable value, as the baseline one, the barplot in Figure 4.5 shows how the system performance changes ($PI\%$), at different prediction steps, by decreasing the value of q_u. It can be noticed how the reduction of the Hankel shift, q_u, applied to the input variables causes negative $PI\%$ and, thus, the decaying of the model performance, both in terms of $MAPE_{\%}$ and R^2 for prediction steps higher than 5. This led to the selection of $q_u^{opt} = q^{opt} = 40$ as the optimal number of Hankel shifts for inputs and state variables.

TABLE 4.3: SRU case study: performance comparison for the selection of the q^{opt}. The mean value over 20 subsets of data of the $MAPE_{\%}$ is reported for different state time shifts, q. The KPI is evaluated for a 30-step-ahead prediction. The $PI\%$ is reported considering $q = 40$ as the reference value.

State Time-Shift Optimization					
q	**20**	**30**	**40**	**50**	**60**
MAPE$_{\%}$	5.53	5.22	5.19	5.28	5.46
PI%	−6.66	−0.58	0	−1.79	−5.13



(a) $PI\%$ for $MAPE_{\%}$



(b) $PI\%$ for R^2

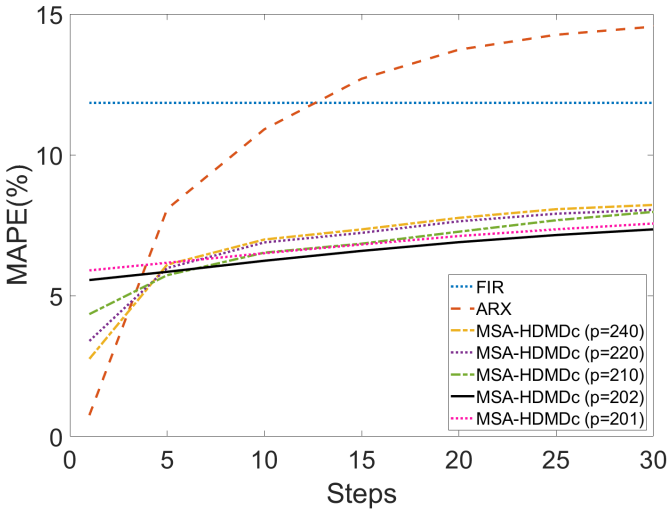
FIGURE 4.5: SRU case study: percentage performance improvement $PI\%$ for (**a**) $MAPE_{\%}$ and (**b**) R^2 at each prediction step, varying the input delay shifts, q_u, in the MSA-HDMDc algorithm. The $PI\%$ was calculated for each of the identified models with respect to the baseline model with $q_u = q = 40$.

ii) Model Order Reduction

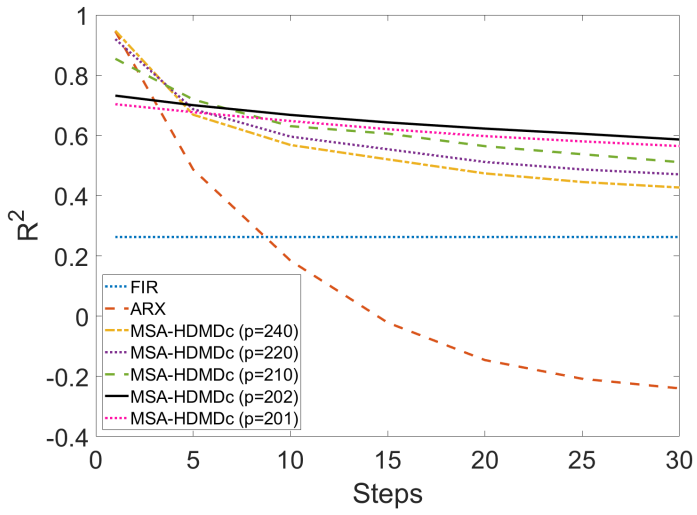
To evaluate the importance of the model reduction phase, as proposed in MSA-HDMDc, the results obtained during the model order optimization are shown for different prediction horizons. To better assess the performance of the model, both $MAPE_{\%}$ and R^2 are shown.

According to Algorithm 1, the analysis was performed considering two optimization phases, first for p and then for r, each representing the SVD truncation for the matrices Ω (Equation (4.20)) and X'_H (Equation (4.24)). In a first step, the Ω matrix was truncated from the full-order $p_{max} = n_x q^{opt} + n_u q_u^{opt} = 240$ (with $q^{opt} = 40$, $n_x = 1$, $q_u^{opt} = 40$, and $n_u = 5$) to the reduced order p. Models with different p-reductions from the set $p_{range} \in \{201, 202, 210, 220, 240\}$ and with the full-order matrix X'_H with order $r_{max} = min(n_x q^{opt}, p) = 40$ are shown in Figure 4.6, where the indices $MAPE_{\%}$ and R^2 are given for different prediction steps. The global performance of HDMDc on the validation dataset was compared with the performance of the baseline methods (i.e., ARX and FIR). It can be noticed that the MSA-HDMDc model outperforms the FIR model for all prediction steps and the ARX model for prediction steps greater than

or equal to 5, regardless of the model order, p. The comparison of the MSA-HDMDc model for the different p also shows that the reduction of the Ω matrix to order p has a positive effect on the performance of the model for prediction steps greater than 5, and the optimal configuration is achieved for order $p = 202$.



(a) $MAPE_{\%}$

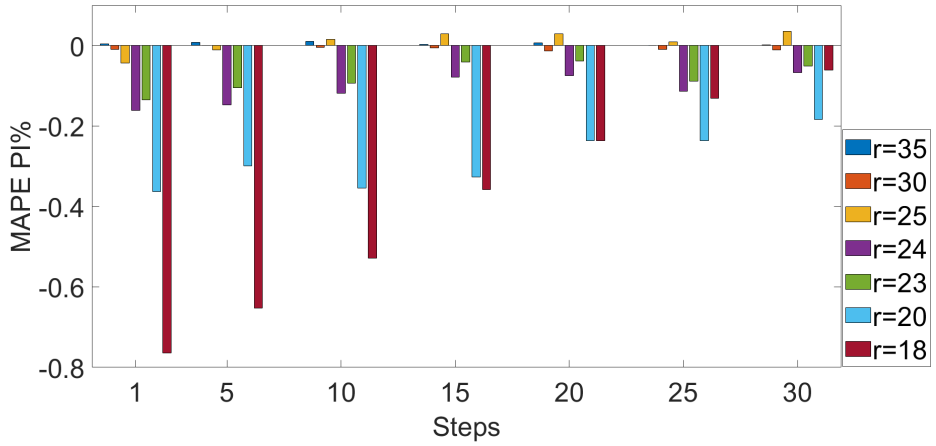


(b) R^2

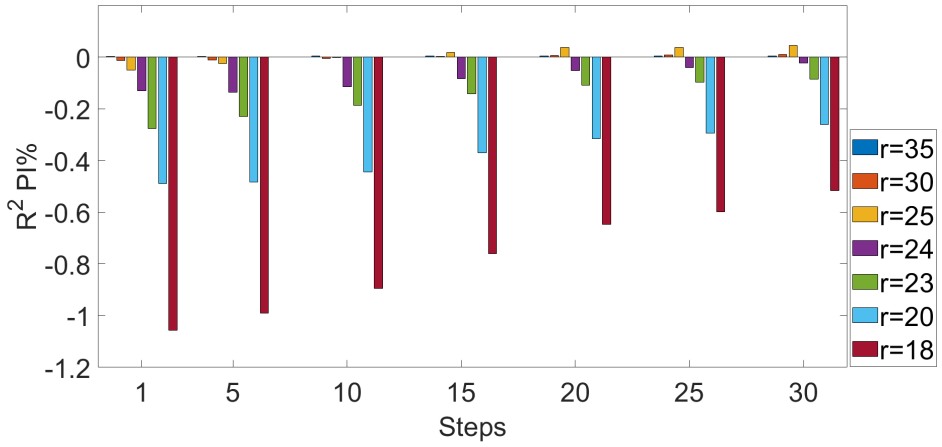
FIGURE 4.6: SRU case study: MSA model performances: (**a**) $MAPE_{\%}$, (**b**) R^2 for ARX, FIR, and MSA-HDMDc models by varying the reduced order, p, of the Ω matrix in the $p_{range} \in \{201, 202, 210, 220, 240\}$ and considering the matrix X'_H at full-order $r_{max} = 40$.

In a second step, while maintaining the optimal Ω order $p^{opt} = 202$, the X'_H matrix was truncated from the full-order $r_{max} = min(n_x q^{opt}, p^{opt}) = 40$ to the reduced order, r. Models with different r reductions belonging to the set $r \in \{18, 20, 23, 25, 30, 35\}$ were identified. The global performance comparison on the validation dataset is shown in Figure 4.7, where the $PI\%$ is given for both $MAPE_{\%}$ and R^2 at different prediction steps. The reference model for the $PI\%$ is MSA-HDMDc with $p^{opt} = 202$ and X'_H full-order $r_{max} = 40$. It can be noticed how the MSA-HDMDc model with a reduction of X'_H to order $r = 25$ outperforms the full-order model for prediction

steps greater than 5. These results confirm that the optimized MSA-HDMDc order reduction allows identification of the dominant dynamics and thus introduces robustness and noise rejection features to the reduced model. These properties thus improve the long-term prediction performance compared with the full-order system. Finally, the optimal order for the MSA-HDMDc model was determined to be $p^{opt} = 202$ and $r^{opt} = 25$.



(a) $PI\%$ for $MAPE_{\%}$



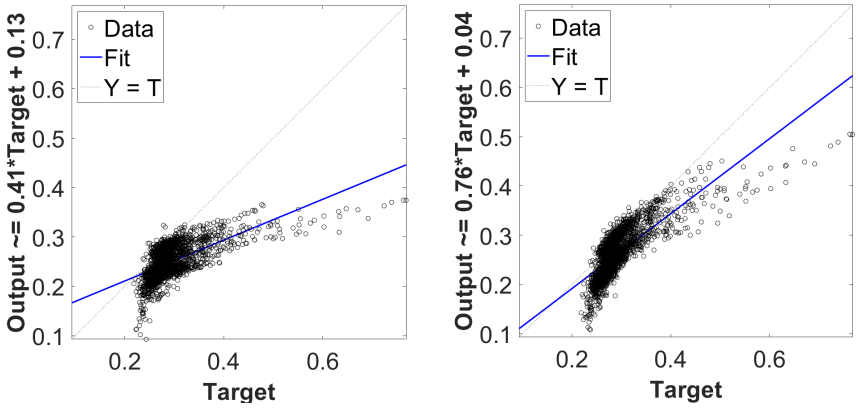
(b) $PI\%$ for R^2

FIGURE 4.7: SRU case study: barplot of $PI\%$ for (**a**) $MAPE_{\%}$ and (**b**) R^2 with Ω matrix order reduction $p^{opt} = 202$ and varying the X' matrix reduction order in the range $r_{range} \in \{18, 20, 23, 25, 30, 35\}$. The $PI\%$ was calculated for each of the identified models with respect to the reference model with $p^{opt} = 202$ and $r = r_{max} = 40$.

iii) Model Comparisons and Discussion

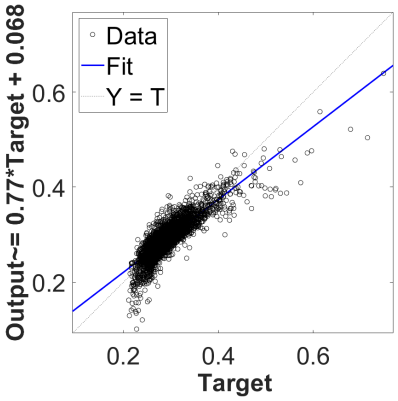
In this section, the performance of the MSA-HDMDc reduced-order model is evaluated. In particular, the results for the maximum step prediction (i.e., 30 steps), are here presented. The regression plots on the test dataset, for the baseline and the optimal MSA-HDMDc models, are reported in Figure 4.8. The ARX model regression plot presents a slope of 0.41 and a bias of 0.13; the global performance over the test dataset is not considered acceptable. The FIR regression plot presents a slope of 0.76

with a bias of 0.04. MSA-HDMDc outperforms both the baseline models with a slope of 0.77 and a bias of 0.068.



(a) ARX

(b) FIR



(c) MSA-HDMDc

FIGURE 4.8: SRU case study: regression plots of predicted output at 30 steps versus the target measured output, y_1: (**a**) ARX model, (**b**) FIR model, (**c**) MSA-HDMDc model with optimal parameters $q^{opt} = 40$, $q_u^{opt} = 40$, $p^{opt} = 202$, $r^{opt} = 25$.

The time plot in Figure 4.9 shows a comparison between the measured output, y_1, and the 30-step-ahead predicted output for the baseline and the MSA-HDMDc models for a subset of the test dataset.

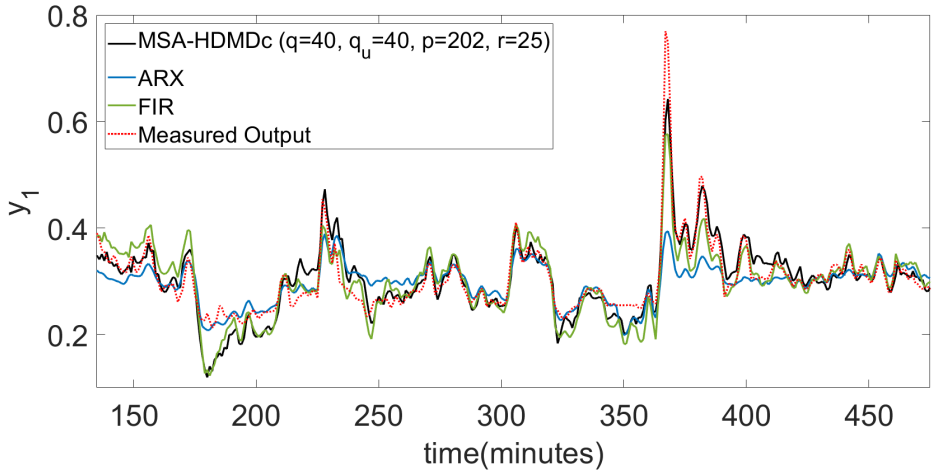


FIGURE 4.9: SRU case study: comparison of the measured output (y_1) with the predicted ones at 30-step-ahead for the baseline and the MSA-HDMDc models with optimal parameters $q^{opt} = 40$, $q_u^{opt} = 40$, $p^{opt} = 202$, $r^{opt} = 25$.

As described above, the main objective of the SRU is to remove H_2S from the gas flow. Therefore, the estimation of the output peaks is of greatest interest for the designed model. Figure 4.9 shows the prediction performance of the proposed model, which outperforms the results obtained with the baseline approaches, especially with respect to the peak events.

Tables 4.4 and 4.5 reports the $MAPE_{\%}$ and R^2 for the test dataset considering different prediction steps for the considered models. Both KPIs are in agreement and show the superiority of the MSA-HDMDc model for large prediction horizons.

TABLE 4.4: SRU case study: $MAPE_{\%}$ values at different prediction steps obtained for the considered models: ARX, FIR, MSA-HDMDc ($q^{opt} = 40$, $q_u^{opt} = 40$, $p^{opt} = 202$, $r^{opt} = 25$).

	MAPE$_{\%}$						
Steps	**1**	**5**	**10**	**15**	**20**	**25**	**30**
ARX	**2.75**	8.08	10.91	12.70	13.74	14.26	14.55
FIR	11.84	11.84	11.84	11.84	11.84	11.84	11.84
MSA-HDMDc	5.56	**5.85**	**6.24**	**6.59**	**6.90**	**7.15**	**7.35**

Bold values represent the best performance for the specific prediction steps column.

TABLE 4.5: SRU case study: R^2 values at different prediction steps obtained for the considered models: ARX, FIR, MSA-HDMDc ($q^{opt} = 40$, $q_u^{opt} = 40$, $p^{opt} = 202$, $r^{opt} = 25$).

	R^2						
Steps	**1**	**5**	**10**	**15**	**20**	**25**	**30**
ARX	**0.95**	0.49	0.18	−0.02	−0.15	−0.21	−0.24
FIR	0.26	0.26	0.26	0.26	0.26	0.26	0.26
MSA-HDMDc	0.73	**0.70**	**0.67**	**0.64**	**0.62**	**0.61**	**0.59**

Bold values represent the best performance for the specific prediction steps column.

As mentioned in Section 4.1, the strength of the Koopman operator, which forms the basis of MSA-HDMDc, lies in the identification of a global state space model that is valid, even far from specific working points and/or attractors. It differs from standard linear models that either exploit linearization around specific working points or extend the linear approximation to the entire domain. With this in mind, the exogenous inputs in the entire test interval were clustered using the k-means algorithm with squared Euclidean distance, which is commonly used in pattern recognition (Sahbudin, Scarpa, and Serrano, 2019), classification and predictive modeling (Cohn and Holm, 2021). Such a method aims to identify different operating points contained in the input dynamics. To select the optimal number of clusters, the silhouette score distribution (Rousseeuw, 1987) was used, obtaining three distinct clusters. The results of such a procedure allowed us to divide the test time series into sub-intervals, each of which is associated with a cluster identifying a working point.

The results of the clustering are shown in Figure 4.10. The first panel shows the time evolution of the exogenous inputs as named and described in Table 4.1. The second field shows the different operating points, defined as clusters, identified by the k-means algorithm in the analyzed time window. The third field contains the $MAPE_{\%}$ time evolution, which was analyzed in time batches of 100 min for a 30-step-ahead prediction. It can be observed that, in the first and last time interval, belonging to $Cluster1$, which is representative of the majority of the dataset, all three models predict the output with good performance. For data belonging to the second and third clusters, only MSA-HDMDc guarantees a performance similar to those obtained in the previous cluster. This confirms the suitability of MSA-HDMDc to identify global models.

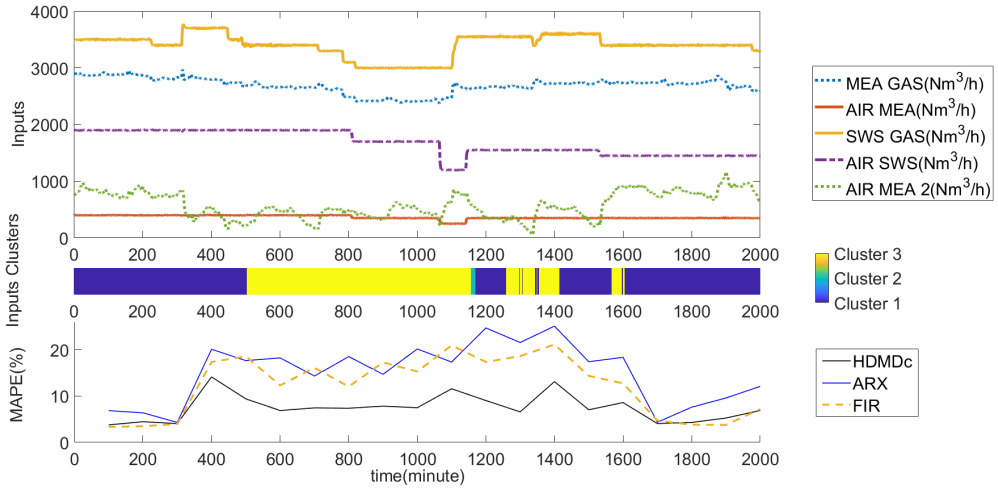


FIGURE 4.10: SRU case study: analysis of $MAPE_{\%}$ computed using time batches of 100 samples for a 30-step-ahead prediction on a selected interval of the test dataset. The corresponding normalized input signals and associated clusters are also included. 1st panel: time evolution of the inputs, 2nd panel: input clusters, 3rd panel: time evolution of $MAPE_{\%}$.

4.5.2 DC Prediction Analysis

In this section, the proposed MSA-HDMDc is applied to the DC case study and the results are compared with the baseline models. The available dataset consists of 4 months of data, i.e., March, May, July and September 2004, with a sampling period of 6 min. Here, 50% of the dataset (March and May) is used for model training, 25% (July) for the validation of the hyperparameters and model order reduction and, finally, the testing is performed on the final 25% of the dataset (September).

i) Hyperparameter Tuning

An MSA prediction on the y output is performed. To validate the performance of the procedure, a time horizon of $K_{max} = 20$ steps corresponding to 120 min is selected. To better show the effect of MSA prediction for different time horizons, values in the interval $K \in \{2, 5, 10, 20\}$ steps are considered.

Linear Regression Models Two linear models were considered: ARX and FIR. The optimal model order was selected based on the minimum description length (MDL) criterion (Ljung, 1986). In particular, the ARX structure was identified to have three common poles ($n_a = 3$), eight zeros ($n_b = 8$) and no delay for all input variables ($n_k = 0$), while the FIR order parameters resulted to be $n_b = 8$ and $n_k = 0$.

MSA-HDMDc Model In a preliminary stage of the iterative MSA-HDMDc method, the HDMDc algorithm was used without order reduction. A grid search strategy was applied to find the optimal q in the range from $q = 10$ to $q = 30$. The estimated output reconstructed for the prediction step, K_{max}, was compared with the measured output, y. For the statistical analysis, the validation dataset was divided into 70 subsets of 100 samples. $MAPE_{\%}$ and R^2 were evaluated for each subset and the corresponding distribution was determined. In particular, the median value of $MAPE_{\%}$ was taken into account when selecting the optimal q hyperparameters.

Table 4.6 shows the mean value over the 70 trials of the $MAPE_{\%}$, considering a 20-steps-ahead prediction. The considered state delay shifts are $q \in \{10, 12, 15, 17, 20, 30\}$. The best-performing model corresponds to $q^{opt} = 12$, as shown by the $PI\%$ reported.

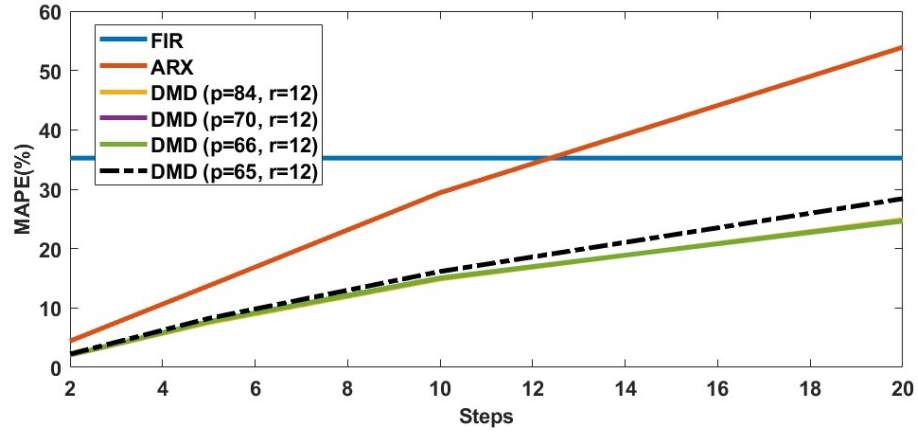
As a second step, considering the optimal value $q^{opt} = 12$, further parametric optimization was carried out by varying the Hankel shift, q_u, applied to the exogenous control inputs in the Γ_H matrix. Being $q_u \leq q^{opt}$, for the system to be causal, the model was identified for each q_u belonging to the set $q_u \in \{5, 10, 12\}$. The estimated output, reconstructed at each considered prediction step, was compared with the measured output, y. It was found that the reduction of the Hankel shift, q_u, applied to the input variables causes decaying of the model performance, both in terms of $MAPE_{\%}$ and R^2. This led to the selection of $q_u^{opt} = q^{opt} = 12$ as the optimal number of Hankel shifts for inputs and state variables.

TABLE 4.6: DC case study: performance comparison for the selection of the q^{opt}. The mean value over 70 subsets of 100 of data samples of the $MAPE_{\%}$ is reported for different state time shifts, q. The KPI is evaluated for a 20-step-ahead prediction. The $PI\%$ is reported considering $q = 12$ as the reference value.

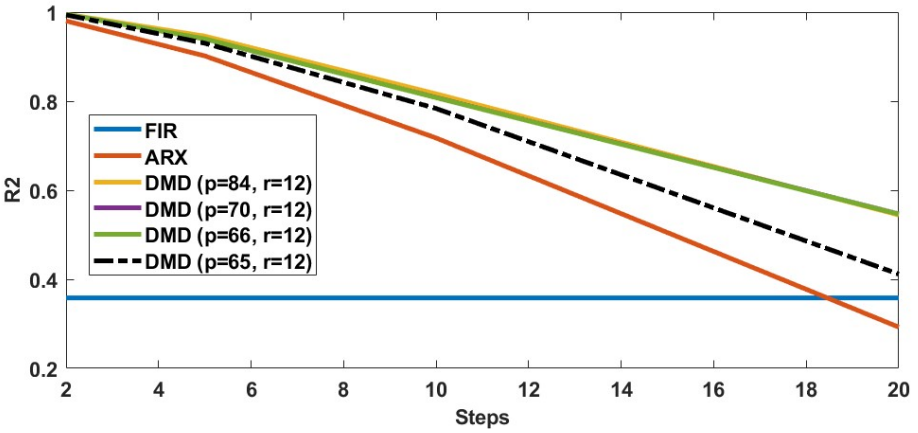
State Time-Shift Optimization						
q	**10**	**12**	**15**	**17**	**20**	**30**
MAPE$_{\%}$	25.73	24.87	25.32	26.43	27.52	30.19
PI$\%$	-3.43	0	-1.81	-6.28	-10.65	-17.20

ii) Model Order Reduction

In a first step, the Ω matrix was truncated from the full-order $p_{max} = n_x q^{opt} + n_u q_u^{opt} = 84$ (where $q^{opt} = 12$, $n_x = 1$, $q_u^{opt} = 12$ and $n_u = 6$) to the reduced order p. Models with different p reductions in the set $p_{range} \in \{65, 66, 70, 84\}$ and with the full-order matrix X'_H with $r_{max} = min(n_x q^{opt}, p) = 12$ are shown in Figure 4.11, where the indices $MAPE_{\%}$ and R^2 are given for different prediction steps. The global performance on the validation dataset was compared with the baseline methods (i.e., ARX and FIR). It can be seen that the MSA-HDMDc model outperforms both FIR and ARX for all prediction steps, regardless of the model order, p. When comparing MSA-HDMDc for the different p, it is also noted that reducing the Ω matrix to order p leaves the performance unchanged, as it decreases from $p = 84$ to $p = 66$, but starts to deteriorate at $p = 65$. This leads to the conclusion that $p^{opt} = 66$ holds for all prediction steps.



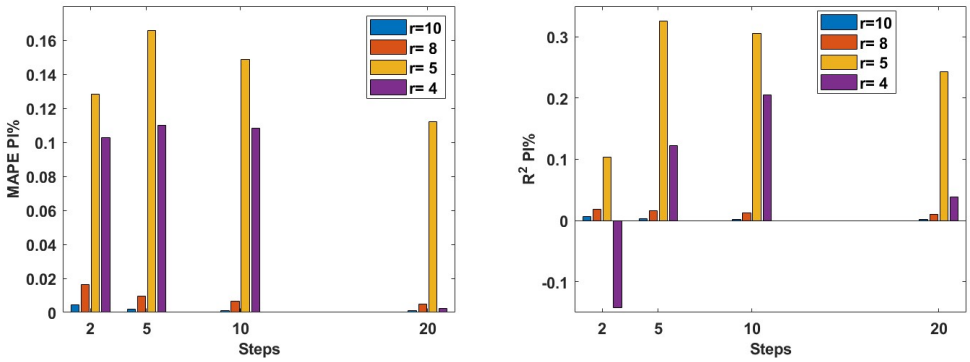
(a) $MAPE_{\%}$



(b) R^2

FIGURE 4.11: DC case study: MSA model performances in terms of (**a**) $MAPE_{\%}$, (**b**) R^2 for ARX, FIR and MSA-HDMDc models by varying the reduced order, p, of the Ω matrix in the $p_{range} \in \{65, 66, 70, 84\}$ and considering the matrix X'_H at full-order $r_{max} = 12$.

In a second step, while maintaining the optimal Ω order $p^{opt} = 66$, the X'_H matrix was truncated from the full-order $r_{max} = min(n_x q^{opt}, p^{opt}) = 12$ to the reduced order r. Models with different r reductions belonging to the set $r \in \{4, 5, 8, 12\}$ were identified. The global performance comparison on the validation dataset is shown in Figure 4.12, where the $PI\%$ is given with respect to MSA-HDMDc, with $p^{opt} = 66$ and $r_{max} = 40$ for both $MAPE_{\%}$ and R^2 at different prediction steps. It can be seen that the MSA-HDMDc model with a reduction of X'_H to order $r = 5$ outperforms the full-order model for all selected prediction steps. These results confirm that the optimized MSA-HDMDc order reduction allows to identify the dominant dynamics and thus introduces robustness and noise rejection features to the reduced model. These properties thus improve the long-term prediction performance compared with the full-order system. Finally, the optimal order for the MSA-HDMDc model was determined to be $p^{opt} = 66$ and $r^{opt} = 5$.

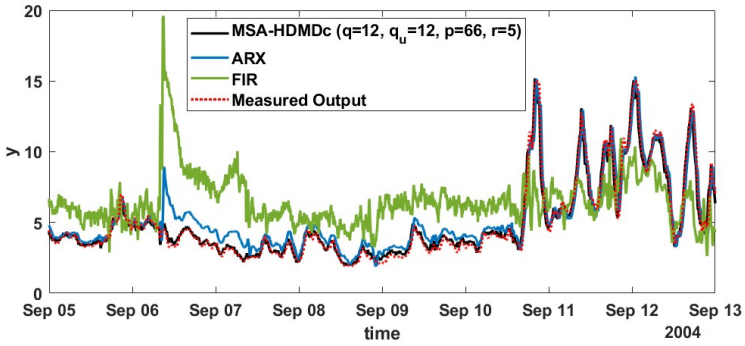


(a) $PI\%$ for $MAPE_{\%}$ (b) $PI\%$ for R^2

FIGURE 4.12: DC case study: barplot of $PI\%$ for (**a**) $MAPE_{\%}$ and (**b**) R^2 with Ω matrix order reduction $p^{opt} = 66$ and varying the X' matrix reduction order in the range $r_{range} \in \{4, 5, 8, 12\}$. The $PI\%$ was calculated for each of the identified models with respect to the reference MSA-HDMDc model with $p^{opt} = 66$ and $r = r_{max} = 12$.

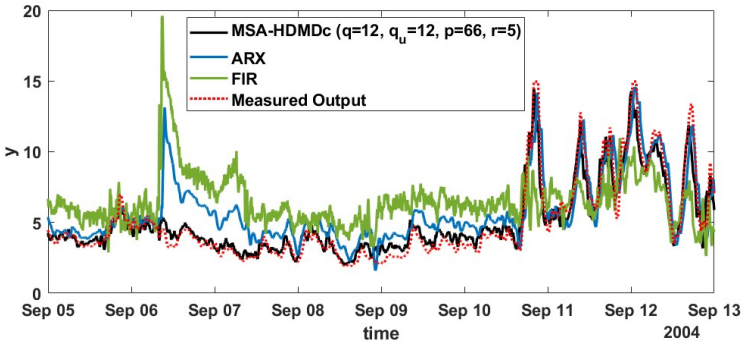
iii) Model Comparison and Discussion

In this section, the performance of the reduced-order MSA-HDMDc model is further evaluated in terms of the predicted time series and robustness to the variability of the operating points. The time plot in Figure 4.13 shows a comparison between the measured output, y, and the 5- and 10-step-ahead predicted outputs for the baseline and MSA-HDMDc models for a subset of the test dataset. It shows that the prediction performance of the proposed model outperforms the results obtained with the baseline approaches, especially for the 10-step-ahead prediction, and confirms that the difference between the models becomes more pronounced at longer prediction intervals.



(**a**) 5-step-ahead (30 min)

FIGURE 4.13: *Cont.*



(**b**) 10-step-ahead (60 min)

FIGURE 4.13: DC case study: comparison of the measured output (y) with the predicted one at (**a**) 5-step-ahead (30 min) and (**b**) 10-step-ahead (60 min) for the baseline and the MSA-HDMDc models with the optimal parameters $q^{opt} = 12$, $q_u^{opt} = 12$, $p^{opt} = 66$, $r^{opt} = 5$ on a selected interval of the test dataset.

TABLE 4.7: DC case study: $MAPE_{\%}$ values at different prediction steps obtained for the considered models: ARX, FIR, MSA-HDMDc ($q^{opt} = 12$, $q_u^{opt} = 12$, $p^{opt} = 66$, $r^{opt} = 5$) in the test dataset.

	MAPE$_{\%}$			
Steps	**2**	**5**	**10**	**20**
ARX	4.44	13.75	29.40	53.91
FIR	57.42	57.42	57.42	57.42
MSA-HDMDc	**1.66**	**6.01**	**12.93**	**23.13**

Bold values represent the best performance for the specific prediction steps column.

TABLE 4.8: DC case study: R^2 values at different prediction steps obtained for the considered models: ARX, FIR, MSA-HDMDc ($q^{opt} = 12$, $q_u^{opt} = 12$, $p^{opt} = 66$, $r^{opt} = 5$) in the test dataset.

	R^2			
Steps	**2**	**5**	**10**	**20**
ARX	0.983	0.854	0.347	−1.20
FIR	−0.99	−0.99	−0.99	−0.99
MSA-HDMDc	**0.998**	**0.974**	**0.890**	**0.661**

Bold values represent the best performance for the specific prediction steps column.

Tables 4.7 and 4.8 reports the $MAPE_{\%}$ and R^2 for the test dataset considering different prediction steps for the considered models. Both KPIs are in agreement and show the superiority of the MSA-HDMDc model for all the prediction horizons.

The analysis of the model performance was performed by identifying different operating points through exogenous inputs clustering. The k-means algorithm (Sahbudin, Scarpa, and Serrano, 2019; Cohn and Holm, 2021) was applied with squared Euclidean distance. The optimal number of clusters was found to be two by using the silhouette score distribution (Rousseeuw, 1987). The clustering results are reported

in Figure 4.14. In the first panel, the time evolution of the normalized exogenous inputs, as named and described in Table 4.2, is shown. In the second panel, the clusters identified by the k-means algorithm are reported in the analyzed time window showing the operating points with different colors. The third panel contains the $MAPE_{\%}$ time evolution computed in time batches of 100 samples for a 5-step-ahead prediction. It can be seen that, for data belonging to the operating point labeled as Cluster 1 between 1400 and 2000 samples, only the MSA-HDMDc model maintains consistent performance across different system operating points. In contrast, the performance of the ARX and FIR models degrades significantly and shows high variability with changes in operating conditions. This highlights the sensitivity of the ARX and FIR models to variations in operating points and confirms the suitability of MSA-HDMDc for identifying global models that provide robust performance under different operating conditions.

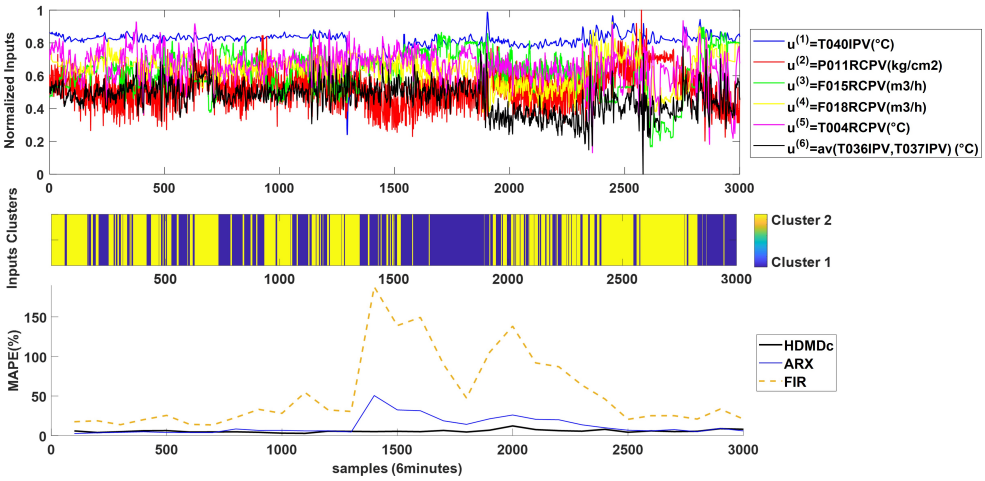


FIGURE 4.14: DC case study: analysis of $MAPE_{\%}$ computed using time batches of 100 samples for a 5-step-ahead prediction on a selected interval of the test dataset. The corresponding normalized input signals and associated clusters are also included. 1st panel: time evolution of the inputs, 2nd panel: input clusters, 3rd panel: time evolution of $MAPE_{\%}$.

4.6 Results: DL/Symbolic & XAI

This section evaluates *interpretable* data-driven pipelines that combine Deep Learning (DL) and Symbolic Regression (SR), and contrasts them with purely symbolic approaches. We target dynamical soft–sensor modeling under nonlinear FIR (NFIR)/ nonliner ARX (NARX) assumptions by constructing standardized lagged–regressor spaces, then training and comparing three families: GP–based SR (Operon, PySR), SINDy with STLSQ, and a DL–aided SR pipeline where compact neural encoders compress the lag space and a symbolic head yields closed–form outputs. Beyond accuracy (root men-squared error - RMSE, R^2) and model complexity, we quantify explainability with SHAP–based metrics (stability, infidelity, Jaccard overlap), and assess *adherence* to known physics when available.

We use two complementary testbenches. **(1) Narendra–Li synthetic system** provides a ground truth set of equations, enabling a stringent check of structural recovery (operator/library sufficiency, coefficient fidelity) alongside predictive error.

(2) SRU industrial dataset stresses robustness to noise, delays, and regime shifts in a real refinery unit, focusing on closed–form soft sensors for H_2S and SO_2 suitable for deployment and auditing. Together, these cases probe both ends of the spectrum: exact identifiability vs. field realism.

We first report the synthetic benchmark results—method tuning and comparative metrics—then present the SRU study, detailing the DL–aided SR encoders, discovered equations, prediction performance, and SHAP analyses.

4.6.1 Narendra-Li System Analysis

The GP-based SR approaches (i.e. Operon and PySR) and the Sindy methods have been evaluated on the Narendra-Li system and compared as regards both performance and interopretability. The identification process has been performed on a dataset of 10000 samples divided into 70% training, 10% validation, and 20% test. The STLSQ algorithm in the SINDy approach requires optimization through the selection of a suitable λ coefficient, striking a compromise between the accuracy and complexity of the final equation. This process is illustrated in Figure 4.15, where the loss function evaluated in terms of RMSE and the number of coefficients for different λ values are shown. The tuning process indicates that the loss remains relatively stable until the threshold value reaches 0.23. Beyond this point, some crucial information is likely to be omitted from the model, leading to a loss increase. Therefore, a reasonable compromise between loss and the number of components in the equation was found by setting $\lambda = 0.2$. The equations derived from the SINDy algorithm are reported in the following:

$$
\begin{aligned}
x_{1,n+1} &= 2.2\sin(x_{2,n}) + 1.3\sin(x_{2,n}) + 0.48\sin(x_{1,n} + x_{2,n})x_{1,n} \\
&\quad - 0.48\sin(x_{1,n} - x_{2,n})x_{1,n} + 0.77\cos(x_{1,n} + x_{2,n}) \\
&\quad - 0.76\cos(x_{1,n} - x_{2,n}) + 0.27\cos(x_{1,n} - x_{2,n}) \\
x_{2,n+1} &= 2.03x_{1,n} - 1.69x_{2,n} + 1.02u_n - 1.81\sin(x_{1,n})x_{1,n} \\
&\quad + 5.51\sin(x_{2,n}) - 0.86\sin(x_{2,n})u_n^2 \\
&\quad - 1.324\sin(x_{2,n} + u_n) + 1.40\cos(x_{1,n})x_{1,n} \\
&\quad + 0.94\cos(x_{1,n})x_{1,n}^2 - 0.61\cos(u_n)x_{2,n} + 0.39 \\
&\quad + \cos(x_{2,n} + u_n)u_n - 1.25\cos(x_{2,n} - u_n)x_{2,n} \\
&\quad + 1.42\exp(x_{1,n})x_{1,n} - 0.36\exp(x_{1,n})x_{1,n}^2 \\
y_n &= 2.67\sin(x_{2,n}) - 0.90\sin\left(x_{2,n}^2\right) - 0.19\cos(x_{1,n} + x_{2,n})
\end{aligned}
\tag{4.33}
$$

The resulting model does not contain important terms and operators such as fractions and exponentials, which are present in the original system. This is reflected in the performance indices, which show a low value of R^2 (see table 4.9). The outcome of the Operon method is here reported:

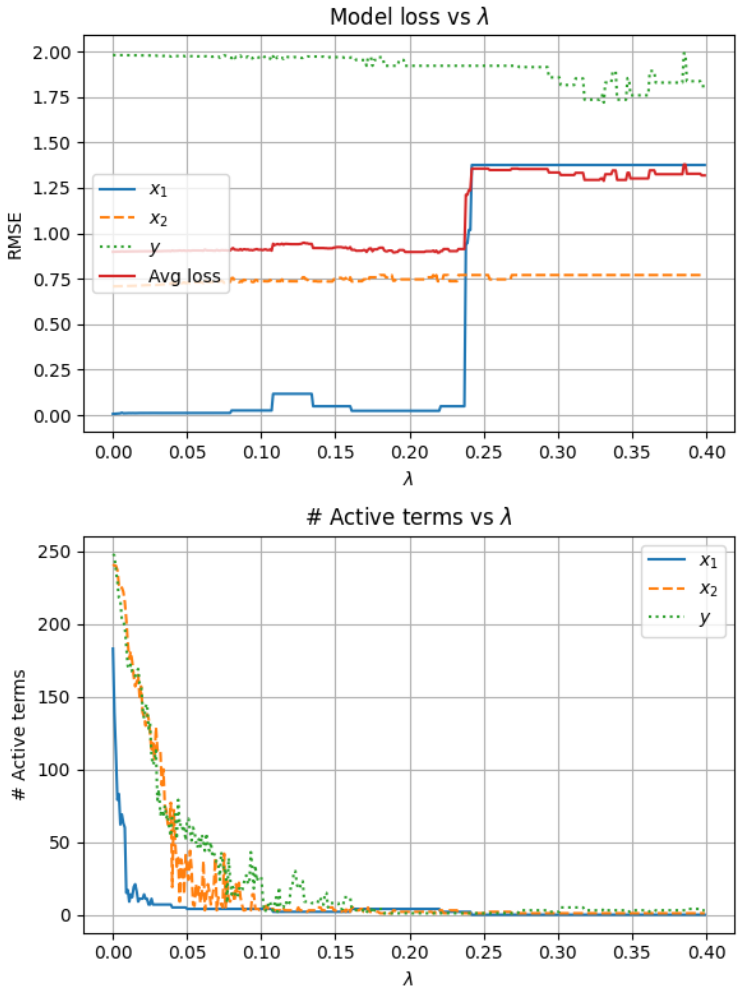


FIGURE 4.15: SINDy model loss and complexity for the parameter λ in the range [0 0.4].

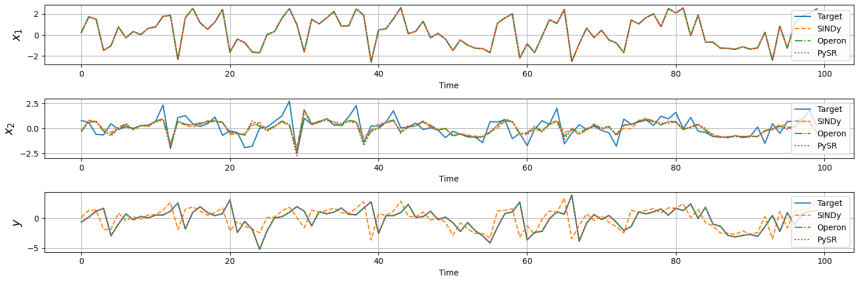


FIGURE 4.16: State and output predictions for the three methods (a) SINDy, (b) Operon and (c) PySR.

$$
\begin{aligned}
x_{1,n+1} &= \left(-0.02x_1 + \frac{(-6.1x_1)\sin(0.228x_1)}{-7.8x_1 + 14\sin(0.526x_1)} + 8.9\right)\frac{\sin(x_2)}{4} \\
x_{2,n+1} &= -0.3x_1 - (-0.027x_1 - 0.063)\sin(18.36x_1) - \\
&\quad (-0.034u - 0.009x_2 + 0.056)\sin(18.089x_1) \\
&\quad (0.42u - 0.09x_1 - 0.99x_2 + (-1.8u - \sin(1.967x_1)) \\
&\quad (0.015x_1 + 0.22))\sin(0.03x_1 + 0.994x_2 + 1.588) + \\
&\quad \sin(0.43x_1) + 0.089\sin(2.437x_1) + 0.03 \\
y_n &= -\frac{5.0x_{1,n}}{-\sin(x_{1,n}) - 2.5\sin(x_{2,n}) - 5.0}
\end{aligned}
\tag{4.34}
$$

It could be observed that the model identified by the Operon approach has a reduced number of terms when compared with the SINDy model. Moreover, the RMSE and R^2 are greatly improved. Finally, the PySR model is illustrated below:

$$
\begin{aligned}
x_{1,n+1} &= \left(2.2 + \frac{1}{x_{1,n} + \frac{1}{x_{1,n}}}\right)\sin(x_{2,n}) \\
x_{2,n+1} &= \left(\frac{x_{1,n}}{x_{2,n}^2 + e^{0.71x_{1,n}^2}} + x_{2,n}\right)\sin(x_{2,n} + 1.58) \\
y_n &= x_{1,n} + \frac{x_{1,n}}{-1.0 + \frac{1}{-0.2\sin(x_{1,n}) - 0.5\sin(x_{2,n})}}
\end{aligned}
\tag{4.35}
$$

The model obtained has a worse error performance with respect to the Operon equations. The identification procedure, however, failed to include terms related to the input in the x_2 equation. Further investigations to include constraints on the use of feature-related terms in the equation identification should be performed when process knowledge is available. A more detailed comparison can be found in Table 4.9. The comparisons between the model predictions and the target signal are shown in Figure 4.16.

To evaluate the Adherence metrics (*Adh*), eq. 4.32 has been used for comparison. As can be seen from the identified equations and the *Adh* metric, the PySR approach has achieved a perfect identification of the x_1 and y equations as they can be easily rearranged into the expected ones. The x_2 equation, on the other hand, yielded only a partial fit of the sine and exponential terms and does not contain terms related to the input. It is worth noting that both the Operon and SINDy systems

	SINDy	Operon	PySR
$RMSE$	0.898 ± 0.013	$\mathbf{0.244 \pm 0.003}$	0.321 ± 0.021
R^2	0.472 ± 0.014	$\mathbf{0.855 \pm 0.008}$	0.749 ± 0.005
S	0.016 ± 0.003	$\mathbf{0.014 \pm 0.002}$	0.018 ± 0.005
$INFD$	0.0003 ± 0.0003	0.0004 ± 0.0003	$\mathbf{4.31e{-6} \pm 5.23e{-7}}$
J_1	$\mathbf{1.0 \pm 0.0}$	$\mathbf{1.0 \pm 0.0}$	$\mathbf{1.0 \pm 0.0}$
J_2	$\mathbf{1.0 \pm 0.0}$	$\mathbf{1.0 \pm 0.0}$	0.884 ± 0.002
J_3	0.907 ± 0.012	$\mathbf{1.0 \pm 0.0}$	$\mathbf{1.0 \pm 0.0}$
Adh	0.185 ± 0.108	0.363 ± 0.004	$\mathbf{0.719 \pm 0.003}$

TABLE 4.9: Loss and interpretability metrics for the three approaches considering the test dataset.

successfully identified the dominant term $2.2 \sin(x_{2,n})$ in the x_1 equation and that in the Operon case, the y equation could be transformed into the expected form by simple mathematical steps. The other components in the regressed equations approximate the missing terms. The SHAP Explainer has been used to create a summary map for the contribution of each feature. It has been applied to visualize the distribution of feature weights across the entire test dataset. Moreover, the Explainer has been used to compute interpretability metrics, which are listed along with the loss metrics in Table 4.9.

The Operon algorithm achieved the best results in the Stability and Jaccard metrics. PySR shows the best performance for the infidelity and adherence metrics. To visualize the results obtained, the global explanations of all the three identified equations $x_1(n)$, $x_2(n)$, $u(n)$ are shown in Figure 4.17 as well as the ground truth values. The relative importance of each variable for the identified equation can be analyzed showing a better matching for the results obtained using the Operon method.

4.6.2 SRU Prediction Analysis

Historical dataset available for the Sulfur line 4 as described in Section 4.4.1 was chosen as a testbed for this model. The available data consists of 17,264 samples obtained with a sampling time of one minute. These were divided into training (70%), validation (15%) and test sets (15%), taking into account temporal dependencies within the process when moving the data and normalizing each set using z-score normalization. Figure 4.18 shows a detailed representation of the NN architecture used and its activation functions. The activation function *tanh* was selected due to its compatibility with the SR stage and its better interpretability in the application domain. In addition, the ReLU activation function provided suboptimal results, indicating the presence of a dying ReLU problem (Lu, 2020). Using the given NN architecture, two different networks were developed and trained independently to predict the two concentrations. Each layer in the architecture is fully connected and the model was trained using the Adam optimizer, also implementing an early stopping strategy and an adaptive learning rate to effectively minimize the loss.

A small number of neurons were used in the hidden encoding layers to simplify the structure. The predictions of the encoding layers on the training dataset were recorded and used as target values for the PySR framework, resulting in five different models, one for each encoded input. The operators considered in the fitting process are $+, -, *, /, sin, exp, sqrt, tan, sinh, tanh$ and real constants, with the maximum length set to 30. Such a specified length allows the genetic algorithm to explore the search space more thoroughly and provide a more accurate overall result. Choosing

an appropriate maximum equation length is crucial, even if a shorter equation is desired, to reduce the complexity of the final output. The PySR algorithm performs migration between the best offspring in each population, so further exploration of the search space greatly affects the overall evolutionary process. Each operator was assigned a weight of 1 and some combinations of operators were prevented as they would be either redundant or uninterpretable, such as $sin(tanh(x))$ or $sin(sin(x))$. Assigning different weights to each operator can be useful to incentivize some operators and penalize others by increasing their complexity. However, due to the inclusion of DL in the target data, no prior assumptions were made about the possible operators in the equations. Both the H_2S and SO_2 input encodings are given in Eq. 4.36 and 4.37 respectively.

$$
\begin{aligned}
e_{1,1} &= (0.01u_1 + 0.1)(2.5u_1 + 1.8u_1z^{-5} - 5.3u_1z^{-9}) - 0.04 \\
e_{2,1} &= 0.1u_2 + 0.1u_2z^{-9} + 0.1\sin\left(u_2z^{-3} - 0.4(0.6 - u_2z^{-1})^2\right) \\
e_{3,1} &= 0.36\sin\left(u_3z^{-7} - u_3z^{-5} + u_3z^{-9} + \tan\left(u_3 - 1.2u_3z^{-5}\right)\right) - 0.14 \\
e_{4,1} &= -0.36\tanh\left(\frac{0.11}{u_4}\right)\tanh\left(\left(\frac{0.1}{u_4z^{-7}} - u_4 - u_4z^{-1}\right)^2 - u_4\right) \\
&\quad (u_4z^{-}3 + 1.16) \\
e_{5,1} &= 0.11 - \tanh\left(u_5z^{-3} - u_5 + \sin\left(0.5u_5z^{-7} + 0.5u_5z^{-9} - u_5\right) + 1.2\right)
\end{aligned}
\tag{4.36}
$$

$$
\begin{aligned}
e_{1,2} &= 0.27\,u_1z^{-9} - 0.27\,u_1z^{-1} - 0.02\,\sin(2.15\,u_1z^{-9} - 0.66) \\
&\quad + \tanh(u_1 + u_1z^{-5} - 2u_1z^{-9} + 0.13) \\
e_{2,2} &= 0.21\,\tanh\left(0.47 - \sin\left(u_2z^{-5} + \frac{u_2z^{-9}(u_2z^{-3} - u_2z^{-7})}{u_2z^{-3}}\right)\right) \\
e_{3,2} &= \tanh(\sinh(u_3 - u_3z^{-9} + (u_3z^{-1} - u_3z^{-7})^2 - \exp\left(-u_3 + 1.7\,u_3z^{-3}\right))) \\
e_{4,2} &= 0.08\,\sin((u_4z^{-9})^2) + 0.2\,\sin\left(1.5\,u_4z^{-7} - \frac{0.2}{u_4}\right) \\
&\quad - \tanh(u_4 - u_4z^{-3} - u_4z^{-5} + u_4z^{-9}) \\
e_{5,2} &= \tanh(0.67\,u_5 - 0.38\,u_5z^{-5} - 0.29\,u_5z^{-9} + 0.45) + 0.17\,u_5z^{-5} - 0.17\,u_5z^{-3}
\end{aligned}
\tag{4.37}
$$

To achieve a more compact notation, the temporal shifts are indicated using the unary delay operator z^{-1}, which means that $u_i(t)z^{-k} = u_i(t-k)$, where $k \in [0, 1, 3, 5, 7, 9]$ to reflect the considered time shifts.

The obtained encoding functions show that the two networks produce slightly different equations, which is an indication that the input dynamics affect the two output concentrations differently. The equations are expressed in terms of the same input u_i, where $i \in [1, 5]$, at different time shifts. Each equation was chosen from a *hall of fame* of equations, ordered through a complexity index that depends on the number of terms. To give preference to simpler equations for better interpretability, a criterion that maximizes the fractional decrease in MSE over the increase in complexity of previous equations while still preserving the loss was selected. This criterion, denoted as s and defined in Eq. 4.38, is based on the approach of identifying drops in the loss-complexity curve (Schmidt and Lipson, 2009). Following this principle, the equation with the highest score is selected among the expressions whose loss is better than at least 1.5 times the most accurate model in the hall of fame. This led to the selection of a local maximum score weighted by the model loss at that complexity level, as shown in Figure 4.19, where the loss-complexity and score-complexity plots

for the five encodings are shown.

$$s = \frac{-\Delta \log(MSE)}{\Delta c} \tag{4.38}$$

In Eq. 4.38, c is the complexity of the candidate solution, which can be expressed as $\boldsymbol{O} \cdot \boldsymbol{w}$, where $\boldsymbol{O}$ is the set of operators involved and $\boldsymbol{w}$ is the cost associated with each operator. The output relationships were discovered using the same set of functions and hyperparameters, with the only change being the maximum length, which was set to 50 to allow further exploration of the mathematical search space. The equations for the H_2S and SO_2 concentration are shown in Eq. 4.39, where the terms e_i in the equations refer to the relative encoded product of the NN for the five input signals, respectively. The output relationships were discovered using the same function set and hyperparameters, with the only change being that the maximum length was set to 50 to allow further exploration of the mathematical search space.

$$\begin{aligned}
[H_2S] &= 0.51 \left(\exp\left(\frac{0.4eu_3}{1 - eu_5} - 0.8eu_1 + 0.4eu_5 + \tan(eu_5) \right) \right)^2 \\
[SO_2] &= 0.04sinh(eu_1 + 0.46) + 0.04tanh^2(eu_3) + 0.3 - 0.46(eu_5)
\end{aligned} \tag{4.39}$$

It is worth noting that among the equations selected based on the aforementioned criteria, the encoded versions of *SWS_GAS* and *AIR_MEA_2* are missing due to their lower correlation with the outputs. In fact, the GP performs an inherent feature selection through its evolution process and discards features that do not significantly improve the equation tree. Nevertheless, the absence of these two inputs may lead to concerns about the robustness of the obtained model. For this reason, two additional equations containing all encoded inputs are presented in Eq. 4.40.

$$\begin{aligned}
[H_2S] &= \frac{\sqrt{\exp\left(0.37eu_5 + \tan(eu_5) - 0.74eu_1 + \frac{0.37(eu_2 + eu_3 + eu_4)}{1 - eu_5}\right)}}{2} \\
[SO_2] &= 0.04eu_1^2 eu_3 + 0.06eu_3(eu_2 + eu_5(1.2eu_3 + eu_4)) \\
&\quad - 0.46\sin(eu_5) + 0.04\tan(eu_1) + 0.32
\end{aligned} \tag{4.40}$$

To properly compare the NN-supported SR and the plain SR approach, two additional SSs were developed with the same function set, the same hyperparameters and a maximum length of 50. The previously introduced criterion was also applied to the SR, this time performed with the full 30 input features. The equations obtained from the plain SR are in Eq. 4.41, while a comparison of the predictions of the three approaches is shown in Figure 4.20 together with the relative metrics in Table 4.10.

$$\begin{aligned}
[H_2S] &= \tanh(0.09\exp(u_{5,k} - u_{5,k-9})\exp(2u_{5,k} - 2u_{5,k-1})) + 0.2 \\
[SO_2] &= (0.14 - 0.11\tanh(u_{5,k} - u_{5,k-9}))\exp(-\sin(u_{5,k} - u_{5,k-3}))
\end{aligned} \tag{4.41}$$

The plain SR equations show that the input *AIR_SWS* as a feedback gas has a strong correlation with the outputs, which overshadows the other inputs in the output equations and limits the interpretability with regard to their dynamics. As shown in Table 4.10 and Figure 4.20, both the NN and NN-supported SR methods show superior performance. It can be seen, especially in the presence of peaks, which are

	R^2	$RMSE$
NN	$\mathbf{(0.89, 0.91)}$	$\mathbf{(0.014, 0.0002)}$
SR	$(0.77, 0.86)$	$(0.020, 0.017)$
NN-SR	$(0.89, 0.90)$	$(0.014, 0.014)$

TABLE 4.10: Loss metrics for the three considered approaches on the test dataset.

	SINDy	Operon	PySR
$RMSE$	0.6758 ± 0.003	$\mathbf{0.6 \pm 0.046}$	0.781 ± 0.004
R^2	0.499 ± 0.004	$\mathbf{0.596 \pm 0.001}$	0.372 ± 0.018
S	0.064 ± 0.045	$\mathbf{0.048 \pm 0.001}$	$\mathbf{0.048 \pm 0.001}$
$INFD$	$\mathbf{0.0001 \pm 0.0002}$	0.0006 ± 0.0001	$\mathbf{0.0001 \pm 0.0001}$
J_1	$\mathbf{0.982 \pm 0.04}$	$\mathbf{0.992 \pm 0.01}$	$\mathbf{0.944 \pm 0.01}$
J_2	$\mathbf{0.982 \pm 0.04}$	$\mathbf{0.916 \pm 0.09}$	$\mathbf{0.999 \pm 0.01}$
J_3	$\mathbf{0.982 \pm 0.04}$	$\mathbf{0.998 \pm 0.002}$	$\mathbf{0.999 \pm 0.01}$

TABLE 4.11: SRU Case Study: Loss and interpretability metrics for the three approaches considering the test dataset.

of greater importance for controlling the process the temporal evolution of the $S0_2$ reported in Figure 4.20 (b) shows a better approximation of the NN-SR in the peak estimation compared to the plain SR. Moreover, the NN and NN-SR predictions show similarities due to the partially shared dynamic encodings.

The SHAP Explainer has been used to create a summary map for the contribution of each feature. It has been applied to visualize the distribution of feature weights across the entire test dataset. Moreover, the Explainer has been used to compute interpretability metrics, which are listed along with the loss metrics in Table 4.11.

The Operon algorithm achieved the best results in the Stability and Jaccard metrics. PySR shows the best performance for the infidelity and adherence metrics. To visualize the results obtained, the global explanations of all the two identified equations $u_{i,k}(n)$ are shown in Figure 4.21. The relative importance of each variable for the identified equation can be analyzed showing a better matching for the results obtained using the Operon method.

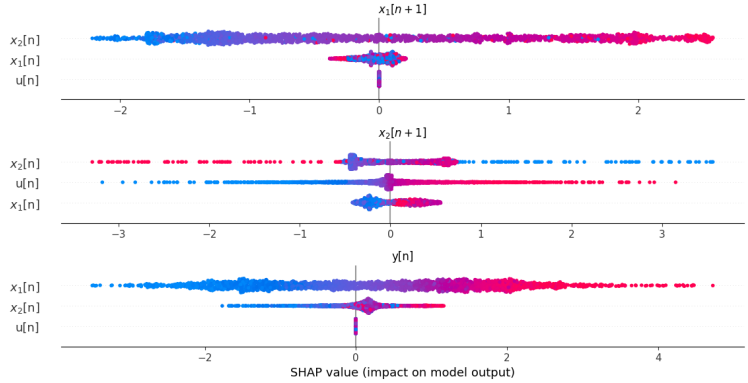
4.7 Discussion

Two complementary tracks—MSA-HDMDc on SRU/DC and interpretable SR (with/without DL encoders) on both Narendra–Li and SRU—to cover regime-robust industrial forecasting and ground-truth symbolic recovery were evalated.

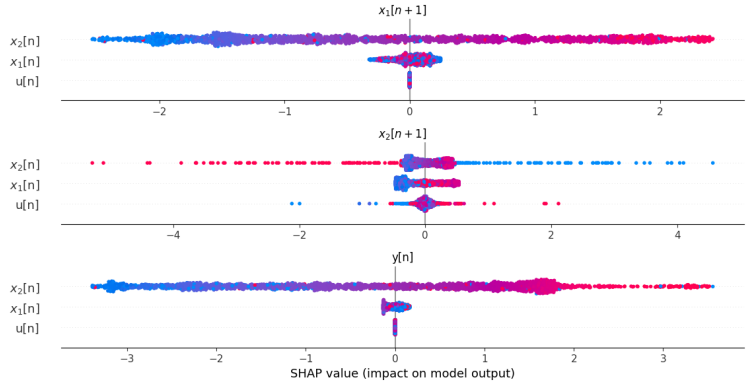
The MSA–HDMDc is proposed as a regime–robust soft–sensor design for industrial time series (SRU, DC) (Patanè, Sapuppo, and Xibilia, 2024). The approach augments the state with Hankel delays to capture nonlinear dynamics in a *linear* surrogate, and then optimizes multi–step prediction via a two–stage SVD truncation on the augmented data spaces (orders p, r). Delay selections (q, q_u) and reductions (p, r) were tuned against $MAPE_{\%}$ and R^2, yielding reduced models that (i) *consistently* outperform ARX/FIR as the horizon increases and (ii) retain accuracy across operating–regime clusters (k–means), evidencing reduced sensitivity to working–point shifts. In the taxonomy, ARX/FIR and HDMDc/MSA–HDMDc all lie in the

Data–driven Grey quadrant; however, MSA–HDMDc offers a *global* Koopman–style linearization with *auditable* structure and superior long–horizon fidelity. The observed gains arise from: (a) delay embeddings that preserve dominant spectral content; (b) SVD–based order reduction that filters noise while retaining predictive observables; and (c) explicit multi–step optimization. Practically, this delivers fast, deployable surrogates for monitoring and control. Looking ahead, transfer to parallel lines can mitigate data scarcity, while coupling with multi–resolution DMD will target slow/fast scale separation in processes exhibiting nonstationary spectra.

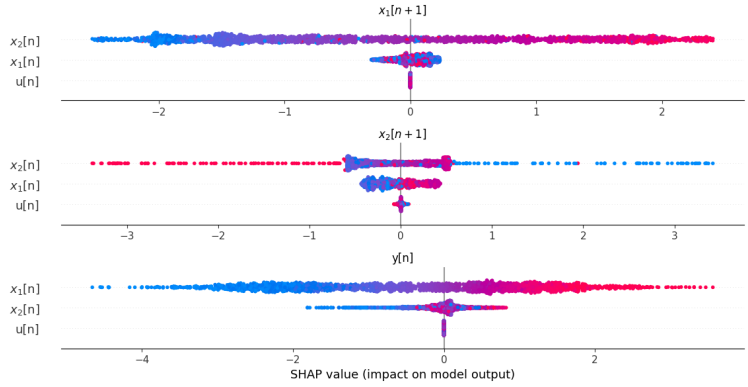
Within the taxonomy, *SINDy* and *GP–based SR* (Operon, PySR) sits at the boundary between **Data–driven Grey** and interpretable Black, depending on how strongly prior structure (lags/operators) constrains discovery; *neural encoders* used to compress lagged inputs are **Black**, but the full DL–aided SR pipeline ends in a closed–form head that restores transparency. On the Narendra–Li benchmark, GP methods achieved the best identification accuracy (Operon lowest RMSE/highest R^2), while PySR showed stronger *adherence* to ground–truth functional forms (despite missing some input–dependent terms); SINDy underperformed in accuracy with the chosen library but remained structurally interpretable. On industrial data, DL–aided SR (small encoders per input $\rightarrow$ symbolic head) scaled symbolic discovery to high–dimensional lag spaces, producing compact equations that matched NN accuracy closely and exposed salient interactions; SHAP–based metrics confirmed stable, low–infidelity attributions. This establishes a practical compromise: use *Black* encoders only for dynamic compression, then enforce *closed–form* outputs for auditability and hand–off to operations. Limitations and paths forward include: imposing domain constraints (dimensional consistency, monotonicities, saturations) during SR; adding control–aware libraries in SINDy; and extracting *grey–box* structure from encoders (e.g., Sparse/Physics–guided Autoencoders) to further restrict the symbolic search space. These steps move the symbolic stack toward *physics–informed ML*, tightening alignment with process physics while preserving interpretability.



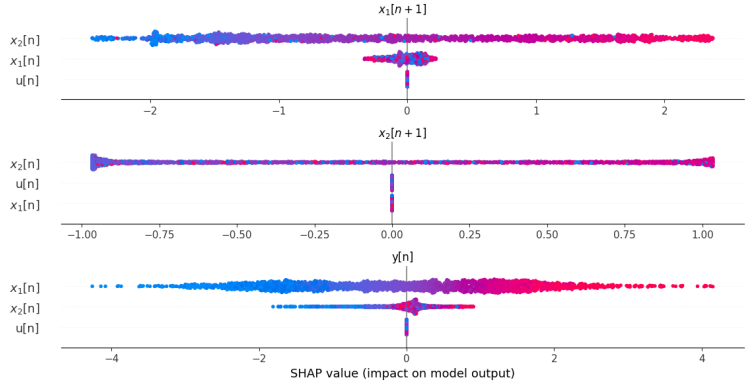
(a)



(b)



(c)



(d)

FIGURE 4.17: Global explanations of the Narendra-Li system on a test dataset of 2000 elements for ground truth and developed models: (a) Ground truth (b) SINDy, (c) Operon, (d) PySR.

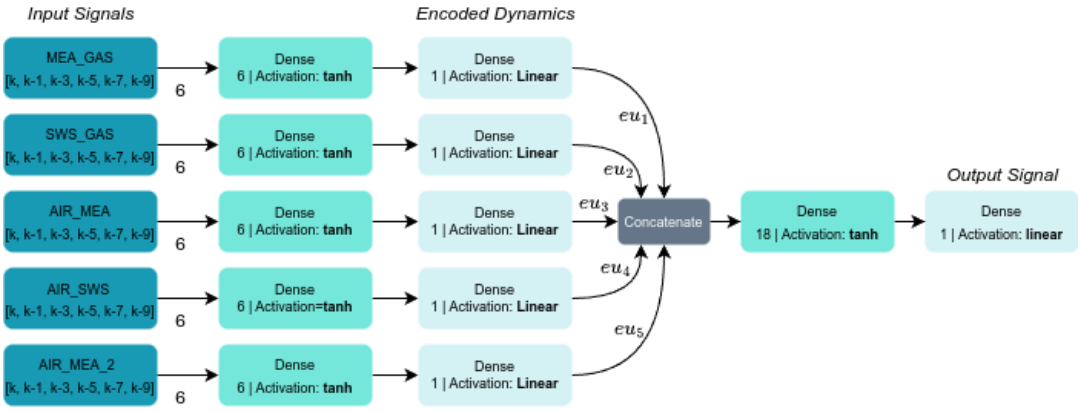


FIGURE 4.18: Block diagram for the utilized NN architecture.

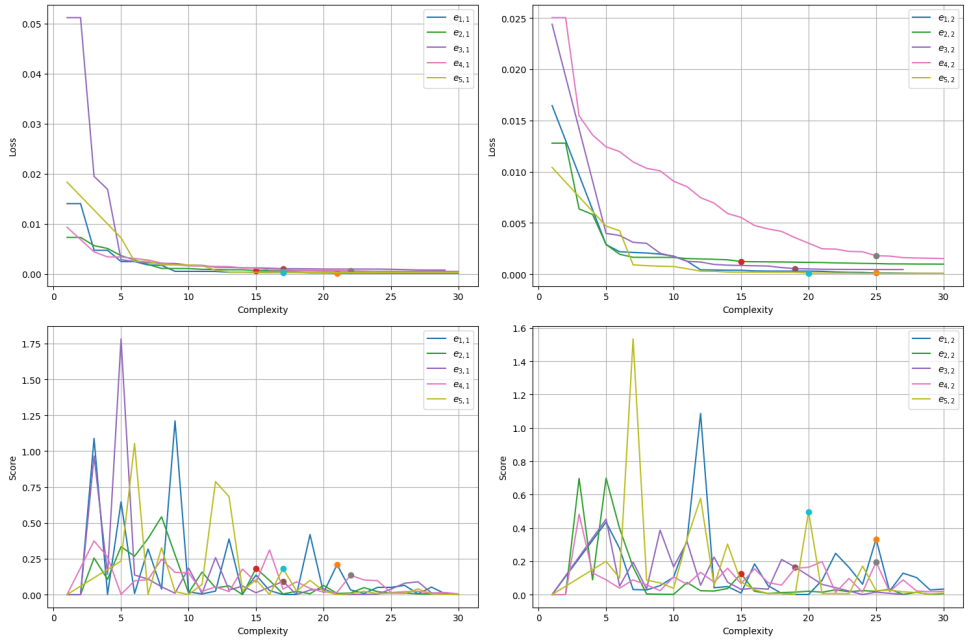


FIGURE 4.19: Loss-complexity and score-complexity curves of the encoded inputs for $[H_2S]$ and $[SO_2]$ respectively.

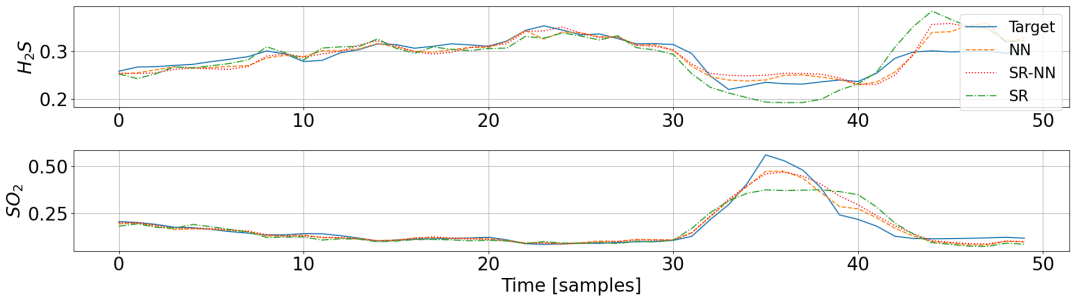
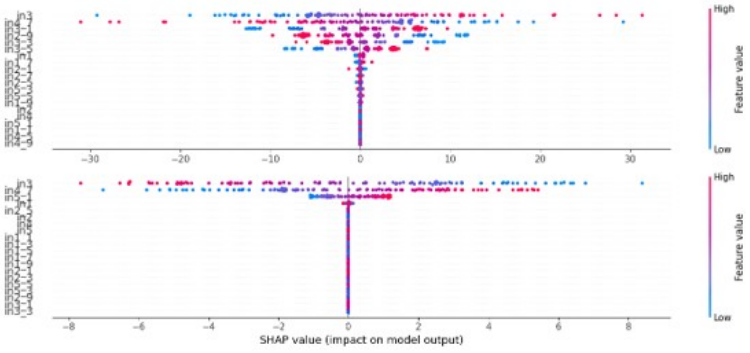
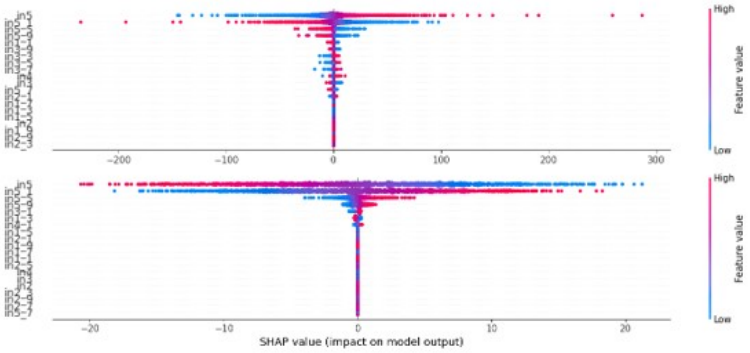


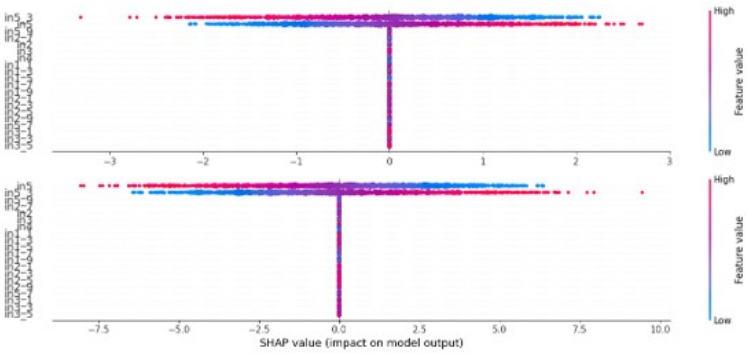
FIGURE 4.20: Predictions of NN, SR, and NN-aided SR on a portion of the test dataset.



(a)



(b)



(c)

FIGURE 4.21: Global explanations of the SRU system on a test dataset of 2000 elements: (a) SINDy, (b) Operon, (c) PySR.

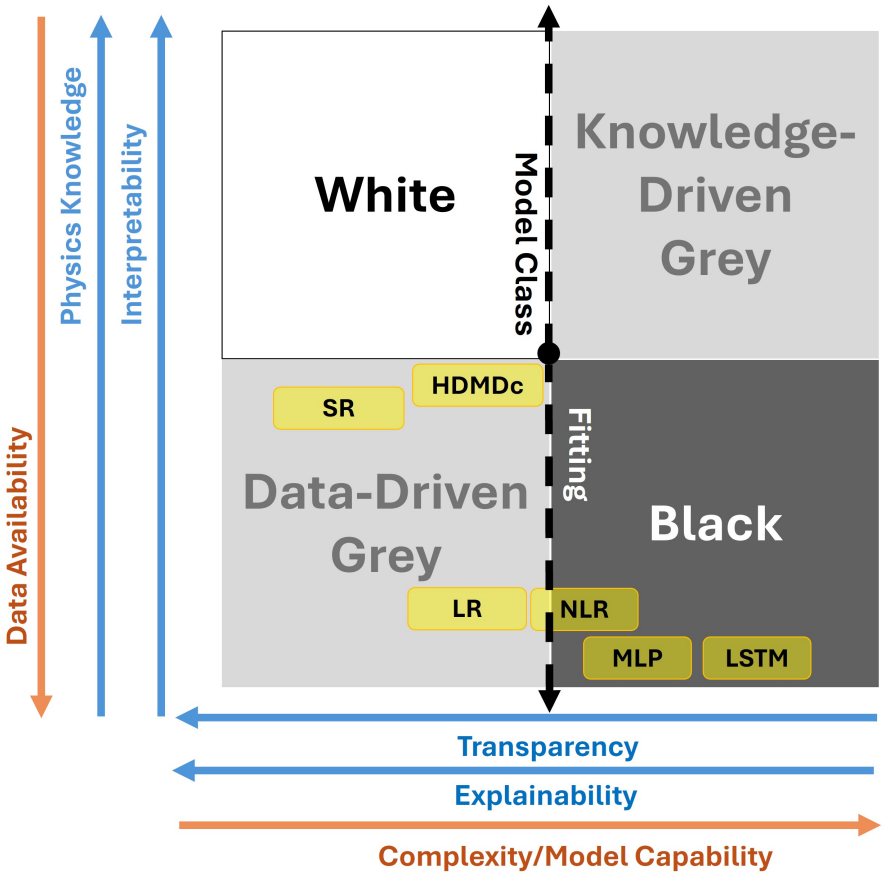


FIGURE 4.22: Taxonomy placement of presented models. ARX/FIR and HDMDc/MSA–HDMDc: **Data–driven Grey**; SR: boundary between **Data–driven Grey** and explainable (XAI) Black (depending on priors); neural encoders based on Multilayer Perceptron (MLP): **Black**.

Chapter 5

Vehicle-to-Grid Availability Prediction: From Data-Driven Grey State-Space to Explainable AI

5.1 Introduction

With global electric vehicle (EV) penetration projected to reach 60% by 2030 (Agency, 2022; Agency, 2021), integrating EVs into vehicle-to-grid (V2G) systems and smart grids (SGs) is increasingly crucial. V2G enables bidirectional energy exchange, turning EVs into distributed, mobile storage units within Industry 4.0 energy frameworks that couple cyber–physical systems (CPS), automation, and real-time data to coordinate renewables, buildings, and mobility (Kamble, Vadirajacharya, and Patil, 2019; Ding et al., 2024). In smart multi-energy management, V2G can help address uncertainties from variable renewables and loads—key challenges in multi-energy microgrids—as well as energy price uncertainty (Zhang et al., 2024b; Zhang et al., 2024a).

As single EVs have negligible grid impact, fleets are organized by *aggregators* (utilities, fleet operators, OEMs, or independent entities) that interface with system operators and markets (Afentoulis et al., 2022; Barbero et al., 2020). Market minimum-capacity thresholds and charger limits make aggregator coordination essential for participation and dispatch. Uncoordinated charging can exacerbate peak loads and local constraints (Lillebo et al., 2019; Deb et al., 2018); conversely, well-planned, coordinated V2G can deliver frequency/voltage regulation, peak shaving, load balancing, and resilience services (Grasel, Baptista, and Tragner, 2024; Fachrizal et al., 2024; Karmaker et al., 2024; Paine, 2019).

Operating these portfolios requires accurate forecasts of *aggregated available capacity* (AAC)—the energy that an EV fleet can provide when connected via aggregator hubs—across day-ahead and short-term horizons. Day-ahead AAC supports wholesale bidding; short-term AAC underpins balancing service bids and intraday operations (e.g., half-hour settlements) (Afentoulis et al., 2022; Barbero et al., 2020). AAC depends on driver behavior, infrastructure siting (Dixon et al., 2022), and vehicle attributes.

With IoT-enabled sensing across floating car data (FCD) and V2G hubs, real-time status/location and grid conditions can be exploited for predictive control (Comi et al., 2021; Li et al., 2024). In addition, exogenous features—meteorology, calendar effects (weekends/holidays) and traffic—also shape availability by modulating mobility patterns (Bakhshi et al., 2022; Gim, 2018; Napoli et al., 2024).

Methodologically, we study a state-space alternative to prevalent sequence models (e.g., long short-term memory, LSTM (P et al., 2022; Curreri, Patanè, and Xibilia, 2021; Shipman et al., 2021a; Shipman et al., 2021b)): dynamic mode decomposition

with control (DMDc) (Proctor, Brunton, and Kutz, 2016), and its Hankel extension (HDMDc) that augments the state with time-delayed outputs/inputs. Grounded in Koopman theory, HDMDc has shown promise on quasi-periodic, controlled systems (e.g., traffic corridors) (Mustavee et al., 2022; Shabab et al., 2021; Das et al., 2023). We apply HDMDc and LSTM to aggregated FCD enriched with meteorology and a fuzzy calendar index (weekends/holidays) or measured traffic to deliver both accuracy and interpretability. Moreover, Explainable AI (XAI) techniques (Lundberg and Lee, 2017b; Ribeiro, Singh, and Guestrin, 2016) are also adopted to meet modern transparency and interpretability requirements, ensuring that predictions align with the decision-making needs of energy providers (Xu et al., 2022).

This chapter presents an integrated analysis of three complementary V2G datasets, VED, Padua, and Rome, through the lens of interpretable modeling. Each dataset is analyzed using models positioned along the modern two-dimensional taxonomy of white-box, black-box, knowledge-driven grey, and data-driven grey models. The VED dataset is studied using Hankel Dynamic Mode Decomposition with control (HDMDc), a grey-box state-space model, alongside deep multi-layer LSTM models. The Padua dataset focuses on ARX, FIR, nonlinear NARX modeling, and transferability of parameters across different scenarios. The Rome dataset emphasizes interpretability of models, leveraging XAI techniques. Together, these results are unified to illustrate the role of transparency, interpretability, and explainability across varying model complexities and knowledge integration.

The remainder of the chapter is organized as follows. Section 5.2 reviews related work. Sections 5.4, 5.5, and 5.6 present dataset-specific methods and results, followed by a cross-dataset synthesis and discussion.

5.2 Related Works

This section introduces a comprehensive literature review of existing models for predicting electrical quantities in the V2G domain. In particular, Table 5.1 summarizes previous research, classifying the works in terms of model and model class, prediction horizon, target variable, data source, and exogenous inputs—factors external to the internal system model but affecting the prediction.

Various models have been used in the literature to predict different V2G-related target variables: AAC (Graham and Teng, 2023; Napoli et al., 2024; Xu et al., 2024; Shipman et al., 2021a; Shipman et al., 2021b; Nogay, 2022), energy demand (Perry, Wang, and Ho, 2021), schedulable energy capacity (SEC) (Li et al., 2021a; Mao et al., 2019) including charging and discharging load, load forecast for energy price determination (Gautam, Verma, and Srivastava, 2019), occupancy and energy charging load at V2G hubs (Amara-Ouali et al., 2022), frequency containment reserve (FCR) participation (Jahromi et al., 2024), energy supply (ES) and peak demand (PD) (Schläpfer et al., 2021), as well as drivers' habits and preferences when connecting to V2G hubs (Zeng, Moura, and Zhou, 2023).

These diverse applications all relate to the integration of V2G into smart grids, enabling operators to optimize the scheduling of EV participation in ancillary services and to meet demand in various markets. Such models are used for price determination in the day-ahead market, intraday power trading, and scheduling power sources at different times of the day to compensate for fluctuations in renewable energy and high-demand intervals. Accordingly, different time scales and prediction horizons are involved in modeling: the short-term scale of hours with a minimum settlement time of half an hour (Graham and Teng, 2023; Napoli et al., 2024; Xu et al., 2024; Perry,

Wang, and Ho, 2021; Gautam, Verma, and Srivastava, 2019; Li et al., 2021a; Mao et al., 2019) and the one-day-ahead forecast scale (Li et al., 2021a; Mao et al., 2019; Shipman et al., 2021a; Shipman et al., 2021b; Amara-Ouali et al., 2022). The one-day-ahead forecast is determined offline for 24 h in advance and serves the day-ahead energy market. However, such long-term predictions are subject to significant uncertainty (Mao et al., 2019). To mitigate this uncertainty, rolling predictions on the order of hours are introduced to meet the needs of short-term ancillary services.

These models are applied to various historical data sources: GPS localization and battery management system data from EVs fleets with a limited number of vehicles (Xu et al., 2024; Li et al., 2021a; Shipman et al., 2021a; Shipman et al., 2021b; Nogay, 2022; Jahromi et al., 2024; Zeng, Moura, and Zhou, 2023); charging and discharging session information based on V2G infrastructures, here referred to as hubs (Graham and Teng, 2023; Perry, Wang, and Ho, 2021; Amara-Ouali et al., 2022); simulated EV data (Gautam, Verma, and Srivastava, 2019; Jahromi et al., 2024); and real-world extensive FCD mobility data and vehicle information (Napoli et al., 2024; Mao et al., 2019; Schläpfer et al., 2021). Most models are trained using historical data of the same target variable, while some also incorporate features that are uncorrelated with the V2G system. These additional features, referred to as exogenous inputs, include calendar information (Graham and Teng, 2023; Napoli et al., 2024; Perry, Wang, and Ho, 2021; Gautam, Verma, and Srivastava, 2019; Amara-Ouali et al., 2022), some including weekends (Graham and Teng, 2023; Napoli et al., 2024; Perry, Wang, and Ho, 2021) and holidays (Napoli et al., 2024), meteorological (Napoli et al., 2024; Gautam, Verma, and Srivastava, 2019) and energy market events (Shipman et al., 2021a; Shipman et al., 2021b; Nogay, 2022), or price (Zeng, Moura, and Zhou, 2023).

Another distinction between the models used to predict V2G variables lies in their classification. Previous research predominantly employs data-driven models, whereas deterministic models are seldom used for comparative predictions, as noted in (Graham and Teng, 2023), due to the energy supplier's aversion to risk in predictions. Deterministic models are also utilized for static analyses rather than for making predictions, particularly in the analysis of mobility data to support V2G systems (Zeng, Moura, and Zhou, 2023; Schläpfer et al., 2021). Dynamic nonlinear black-box models have been used extensively compared to linear regression models. These models account for nonlinear dynamics and are more effective in capturing the behavior of V2G systems compared to dynamic linear regression models. These black-box models include a number of techniques such as neural networks (NN) (Graham and Teng, 2023; Napoli et al., 2024; Nogay, 2022; Li et al., 2021a), long short-term memory networks (LSTM) (Napoli et al., 2024; Shipman et al., 2021b) potentiated by K-means clustering and federated learning in (Perry, Wang, and Ho, 2021) or by convolutional neural networks in (CNN)(Shipman et al., 2021a), random forest (RF) (Amara-Ouali et al., 2022), gradient-boosted decision tree (GBDT) (Mao et al., 2019), and extreme gradient boosting (XGBoost) (Jahromi et al., 2024). The few applications of dynamic linear autoregressive models are presented in (Graham and Teng, 2023; Amara-Ouali et al., 2022).

In this highly complex context of predictive modeling, our work introduces innovations in multiple aspects of the methodological framework, as shown in the summary Table 5.1. The method is applied to short-term predictions covering intervals from one to four hours, aiming to enhance performance for periods exceeding one hour. Unlike most approaches, it utilizes generic FCD that can be easily obtained from insurance companies rather than relying on mobile phone GPS data. Additionally, for the first time, we have included exogenous factors such as weather data and fuzzified weekend and holiday rates, which have never been used for predictions exceeding

half an hour. These features were initially analyzed by the authors in (Napoli et al., 2024) for half-hour predictions as a proof of concept, demonstrating their effectiveness in improving predictive model performance. Notably, weather information has never been used in predictive models applied to real-world case studies, having only been utilized with simulated EV data (Gautam, Verma, and Srivastava, 2019).

The main methodological innovation of our approach is the use of a data-driven, dynamic linear state-space model (HDMDc). Unlike black-box models, HDMDc identifies a single global model that operates across varying conditions (Patanè, Sapuppo, and Xibilia, 2024). It also makes explicit how exogenous factors influence the target variable, improving the interpretability of predictions for energy providers. In addition, the state-space formulation enables model-order reduction, which distills dominant dynamics and yields further insight into predictions. Complementary studies assessed transferability—including traffic as an exogenous input—(Patanè et al., 2024a) and provided post-hoc explainability on the comprehensive dataset (Sapuppo et al., 2025).

5.3 Vehicle Data Preprocessing: From FCD to AAC

The data collection and pre-processing stages were fundamental components of this research. Emphasis was placed on working with data obtained in real-world conditions: vehicle tracking FCD data in the domains of traffic analysis, logistics, and mobility. All quantities in what follows are aggregated on a parametric half-hour time interval, denoted by $T_{hh} = 30$ min; we index samples by k, each corresponding to one T_{hh}.

5.3.1 Stop Maps and Hubs Selection

The first step is to identify V2G points of interest (i.e., candidate hubs) as V2G services rely on the identification of suitable geographical zones based on vehicle data analysis. The movement from one location (origin) to another (destination) (O–D) to perform one or more tasks is called a *trip*. Access to telemetry data enables the collection of trip information with FCD, which enables continuous vehicle tracking over time (Comi et al., 2021). The vehicle position (a sequence of geographic coordinates) and status (travelling or stopping states) are also given. Two consecutive data points of a particular vehicle can be used to identify the start and end of a trip and to detect major changes in vehicle position based on the corresponding status information. A *journey* or *trip chain* refers to a series of trips linked such that the destination of one trip coincides with the starting point of the next. Consequently, the activity stops are determined using the fine-grained FCD. The proposed process is based on measurements of a vehicle's speed and engine state. The process evaluates these measurements to determine whether the vehicle is traveling very slowly or has come to a complete stop. Observations where the vehicle has stopped at a bottleneck but appears to be parked are the main cause of inaccuracies in identifying stops from vehicle data. The method guarantees that only *active* stops (e.g., longer than a predefined threshold and far away from intermediate stops/service points such as gas stations) are classified as such in the result by evaluating both the speed during the previous time interval and the geographical coordinate data as well as the engine status. An initial visualization of the pre-processed data is made by plotting the stops on the global geographical area to identify points suitable as V2G hubs. A hub Hub is associated with a circular catchment Hub^r of radius r.

TABLE 5.1: State of the art in predictive models for V2G-related variables.

Model	Prediction	Data	Exogenous Inputs	Model Class	Target
Persistence model, Generalized linear model, NN (Graham and Teng, 2023)	Half-hour-ahead	Hub	Calendar, **Weekends**	Deterministic, Data-driven Dynamic Linear, Dynamic Nonlinear Black-Box	**AAC**
NN, LSTM (Napoli et al., 2024)	Half-hour-ahead	**Generic FCD Data**	**Meteo, Fuzzy Weekend and Holiday rate**	Dynamic Nonlinear Black-Box	**AAC**
MAML-CNN-LSTM-Attention Algorithm (Xu et al., 2024)	**Hour-ahead**	EVs limited fleet (Rental Car Fleet)	-	Dynamic Nonlinear Black-Box	**AAC**
K-Means clustering, LSTM using federated learning (Perry, Wang, and Ho, 2021)	**Hour-ahead**	Hub	Calendar, **Weekends**	Dynamic Nonlinear Black-Box	Energy demand
Multilayer perceptron (MLP) (Gautam, Verma, and Srivastava, 2019)	**Hour-ahead**	Simulated EVs and Consumer preferences	Calendar, **Meteo**	Dynamic Nonlinear Black-Box	Load forecast for electricity price determination
LSTM (Li et al., 2021a)	Offline (day-ahead) **Rolling (hour-ahead)**	EVs fleet	-	Dynamic Nonlinear Black-Box	SEC
GBDT (Mao et al., 2019)	Offline (Day-ahead) **Rolling (hour-ahead)**	**Generic FCD Data**	-	Dynamic Nonlinear Black-Box	SEC
CNN-LSTM (Shipman et al., 2021a; Shipman et al., 2021b)	Day-ahead	EVs Limited fleet	None/Market event	Dynamic Nonlinear Black-Box	**AAC**
LSTM, NAR (Nogay, 2022)	Day-ahead	EVs Limited fleet	Market event simulation	Dynamic Nonlinear Black-Box	**AAC**
RF, SARIMA (Amara-Ouali et al., 2022)	Day-ahead	Hub	Calendar	Dynamic Nonlinear Black-Box, Data-Driven Dynamic Linear	Occupancy and charging load for single EV
XGBoost (Jahromi et al., 2024)	Yearly	EVs limited fleet/Simulated Data	-	Dynamic Nonlinear Black-Box	FCR participation
Analytical: Vehicle contribution sum (Schläpfer et al., 2021)	-	**Generic FCD Data** (Mobile Phone GPS)	-	Static Deterministic	Daily Aggregated V2G ES and PD
Analytical (Zeng, Moura, and Zhou, 2023)	-	EVs limited fleet (Shared Mobility on Demand)	Energy price	Static Deterministic	Driver preference for V2G or mobility
HDMDc—This study (Patanè et al., 2024b)	**Rolling 1 to 4 hour-ahead**	**Generic FCD Data**	**Meteo, Fuzzy Weekend and Holiday rate**	**Data-Driven Dynamic Linear State Space**	**AAC**
LR, NLR + Transfer—This study (Patanè et al., 2024a)	**Rolling 1 to 4 hour-ahead**	**Generic FCD Data**	**Traffic**	**Data-Driven Dynamic Linear and Nonlinear Regression, Transferability**	**AAC**
ML+XAI—This study (Sapuppo et al., 2025)	**Half-hour-ahead**	**Generic FCD Data**	**Meteo, Fuzzy Weekend and Holiday rate**	**Dynamic Nonlinear Black-Box + Explainability**	**AAC**

The features in bold represent those shared with the HDMDc methods presented in this chapter.

5.3.2 Virtual Electrification

Following the integration of dynamic mobility data with static vehicle category data, the entire dataset is processed to *virtually electrify* each vehicle entry. This virtual electrification process entails linking each car model in the dataset to a battery-electric equivalent (BEV) and assigning an energy value, measured in kWh, to its battery pack BC_v and a segment-specific energy consumption (kWh/km). The activity is performed by:

- a classification often used by manufacturers (ACEA-like segments),
- a market research on the characteristics of battery packs used in EVs (60 BEV models on the Italian market; when multiple packs exist, the largest is selected),
- the assignment of an average energy value in relation to the class of the vehicle reported in the database, and the assignment of a typical consumption per segment.

Segment	ACEA classification	Description	Average energy [kWh]
A	Small	City cars	28
B	Small	Small cars	30
C	Lower Medium	Medium cars	70
D	Upper Medium	Large cars	74
E	Luxury	Executive cars	71
F	Luxury	Luxury cars	86
S	–	Sport coupes	101
J	SUV	Sport utility cars	74
M	MPV	Multi purpose cars	66

TABLE 5.2: Passenger car classification and average battery energy per segment.

5.3.3 State of Charge (SoC) Simulation

A second pre-processing phase consists of calculating the state of charge $SoC_v[k] \in [0,1]$ (or $SoC\%$ when expressed in percent) for non-electric vehicles mapped to BEVs and, when available, refining electric vehicles' SoC from their trajectories. Two closely related sets of assumptions are used in our case studies:

i) Common elements

- **Initialization:** maximum charge at the start of the simulation or of each day: $SoC_v = 1$ (i.e., 100%).
- **Safety reserve:** a minimum state of charge must be maintained to cover the remaining part of the travel chain. We denote this reserve by $SoC_{\min} \in [0,1]$ and enforce $SoC_v[k] \geq SoC_{\min}$ at all times.
- **Driving discharge:** energy used in interval k is $E_v^{\text{drive}}[k] = c_v\, d_v[k]$, where $d_v[k]$ is the distance traveled and c_v is the segment-specific consumption. A representative value used in our experiments is $c_v = 0.2$ kWh/km. The SoC decrement is $\Delta SoC_v^{\text{drive}}[k] = E_v^{\text{drive}}[k]/BC_v$.
- **Availability condition:** a vehicle is considered available to supply energy to the grid when it is stopped inside Hub^r and $SoC_v[k] > SoC_{\min}$.

ii) VED setting

- **Reserve:** $SoC_{\min} = 0.50$ (50%).
- **When stopped and connected:** daytime DC fast charging power $P^{\text{chg}}_{\text{day}} = 50\,\text{kW}$ (07:00–19:00), nighttime slow charging $P^{\text{chg}}_{\text{night}} = 6\,\text{kW}$ (19:00–07:00), charging efficiency $\eta = 0.90$.
- **V2G export:** when providing grid services, a power rating $P^{\text{exp}} = 50\,\text{kW}$ is assumed. The SoC change over one T_{hh} interval accounts for charging/discharging energy and efficiency and is saturated to $[0, 1]$.

iii) Padua/Rome setting

- **Reserve:** $SoC_{\min} = \max(0.30,$ residual energy needed to complete the remaining trip chain$)$.
- **When stopped and connected:** for simplification, conventional grid-to-vehicle charging while connected is neglected.
- **V2G export:** a fixed export power $P^{\text{exp}} = 22\,\text{kW}$ is used; SoC is decreased accordingly over each T_{hh}.

5.3.4 Space and Time Data Aggregation

The next step involves mapping trips and stops by spatially aggregating data to create a subset for each candidate V2G hub and aggregating in time with the parametric interval T_{hh}, generating a time series that serves as the target for the dynamic prediction model. The calculation of the AAC time series for a specific V2G hub leverages the concept of the percentage state of charge $SoC\%$, which can be determined as an indirect measure of both distance traveled and V2G stop intervals.

We denote by

$$\mathcal{A}^r[k] \;=\; \Big\{ v \;:\; v \text{ parked within radius } r \text{ of } Hub \text{ during interval } k,\;\; SoC_v[k] > SoC_{\min} \Big\}$$

the set of *available* vehicles in interval k. For each vehicle $v \in \mathcal{A}^r[k]$, the available capacity in one half-hour is

$$AC_v^{T_{hh}}[k] \;=\; \max\big(SoC_v[k-1] - SoC_{\min},\, 0\big) \;\cdot\; BC_v. \tag{5.1}$$

The *aggregated available capacity* (AAC) over the hub catchment is then

$$AAC_{Hub^r}^{T_{hh}}[k] \;=\; \sum_{v \in \mathcal{A}^r[k]} AC_v^{T_{hh}}[k]. \tag{5.2}$$

Equations (5.1)–(5.2) yield the half-hourly target time series $AAC_{Hub^r}^{T_{hh}}[k]$, which is fed into the predictive model identification.

5.4 VED Dataset: HDMDc vs. LSTM

The proposed prediction model framework consists of several parts, as shown in Figure 5.1. The two blocks on the left refer to the data collection and pre-processing for the vehicle dataset and the extraction of the exogenous inputs. The first block

comprises the acquisition of the FCD data and its description in terms of origin–destination (OD) with the aim of extracting the trips and stops and thus determining the state of charge (*SoC*%) for each individual vehicle. A spatio-temporal aggregation is then performed to obtain the AAC time series, which is used as the target variable for the prediction models. In particular, the spatial aggregation was performed in a geographical area within a radius r from the selected hub, while the temporal aggregation was performed in half-hour (T_{hh}) intervals to match the time scale of the intraday energy market as in most literature examples Graham and Teng, 2023; Napoli et al., 2024; Shipman et al., 2021a; Shipman et al., 2021b. The second block is dedicated to extracting exogenous inputs, aiming to obtain continuous time series with *hh* sampling time. This includes fuzzy data for national holidays and weekends as well as meteorological data on precipitation, temperature, and wind speed. The core block consists of predictive models based on HDMDc and LSTM, enabling rolling predictions with different time horizons. A summary of the main variables involved is provided in Table 5.3.

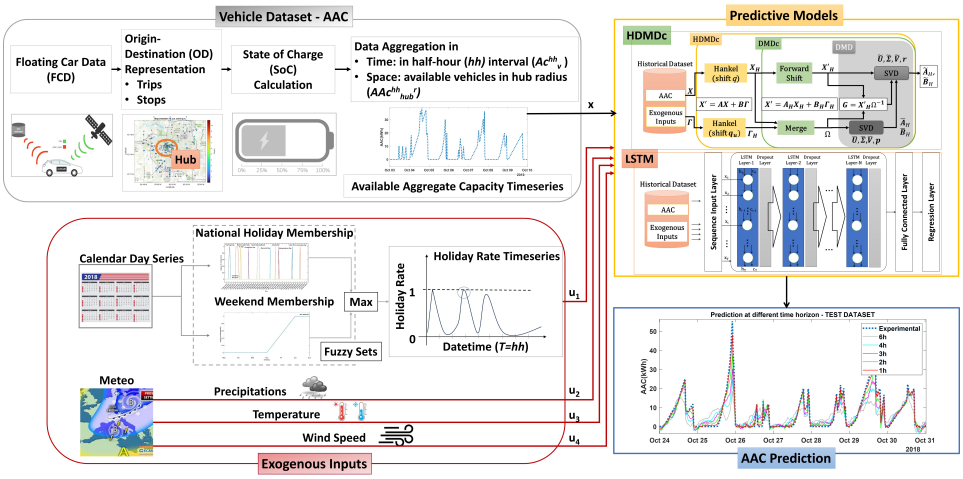


FIGURE 5.1: AAC prediction model framework **(VED dataset)**.

TABLE 5.3: Predictive model framework variables **(VED dataset)**.

Variable	Description	Units
u_1 Fuzzy holiday rate	Calendar exogenous input	–
u_2 Precipitation	Meteorological input	mm
u_3 Temperature	Meteorological input	°C
u_4 Wind speed	Meteorological input	km/h
Hub	V2G infrastructure (hub)	–
r	Hub catchment radius	km
SoC	State of charge	%
$SoC_{\min}$	Minimum state of charge constraint	%
$AC_v^{T_{hh}}$	Vehicle available capacity over T_{hh}	kWh
Hub^r	Area within radius r from Hub	–
$x_k = AAC_{Hub^r}^{T_{hh}}[k]$	Target AAC time series (per Eq. 5.2)	kWh

5.4.1 Dataset Description

The data collection and pre-processing is here described fo vehicle tracking FCD data in the domains of traffic analysis, logistics, and mobility; weather data available both as historical archives and forecasts; and well-known data such as calendar information and holidays. These data have undergone pre-processing to create a standardized format suitable for applying the algorithms under evaluation.

i) Vehicle Dataset

The vehicle energy dataset (VED) (Oh, Leblanc, and Peng, 2022) is an open-access dataset of fuel- and energy-related information collected from 383 individual vehicles in Ann Arbor, MI, USA. It includes GPS records of vehicle routes and time-series data on fuel consumption, energy consumption, speed, and auxiliary energy use. The dataset covers a wide range of vehicles: 264 internal combustion engines (ICEs), 92 hybrid electric vehicles (HEVs), and 27 plug-in hybrid electric vehicles/electric vehicles (PHEV/EVs), operating in real-world conditions for one year from November 2017 to November 2018. Specifically, the following dynamic features can be selected: date, vehicle identifier, trip identifier, duration, latitude, and longitude. In addition, the HV battery SoC% is provided for the PHEVs and EVs. In order to create a benchmark dataset and considering the future projection for the EVs market, an assumption was made: the ICEs and HEVs were assumed to be EVs contributing to the V2G logic. For simplicity, they were assumed to be EVs of the same make and model (Nissan Leaf with a 40 kWh battery capacity). SoC for ICEs and HEVs was calculated as an indirect measure of distance traveled and charging stop intervals as

described in Section 5.3.3. We adopt the conventions of Section 5.3: $T_{hh} = 30$ min, $x_k := AAC^{T_{hh}}_{Hub^r}[k]$ (Eq. 5.2), and $AC^{T_{hh}}_v$ as in Eq. 5.1.

Figure 5.2 shows the density of stops during different daily intervals (0–6 AM, 6 AM–12 AM, 12 AM–6 PM, and 6 PM–12 PM) and integrated over the entire data time interval. The color scale represents the duration of the stops in hours, assuming a minimum stop duration of 30 min. A candidate point of interest was selected and here referred to as Hub_1, thanks to the high vehicle stop density and the location in the city center and university area, as shown in Figure 5.3.

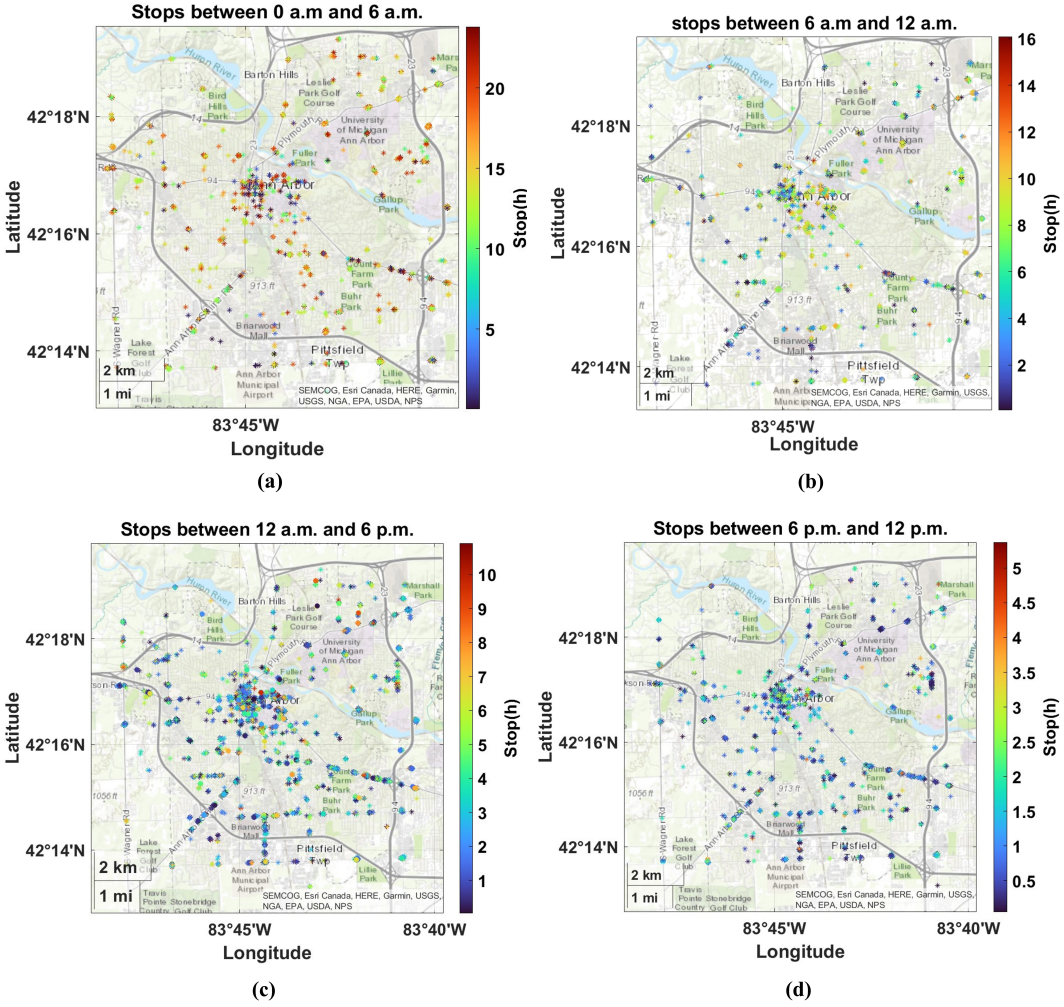


FIGURE 5.2: Stop maps in different time intervals of the day integrated over the entire data time interval **(VED dataset)**. The color bar shows the duration of the stops. The stop events started in the following time windows: (**a**) from 0 AM to 6 AM, (**b**) from 6 AM to 12 AM, (**c**) from 12 AM to 6 PM, and (**d**) from 6 PM to 12 PM.

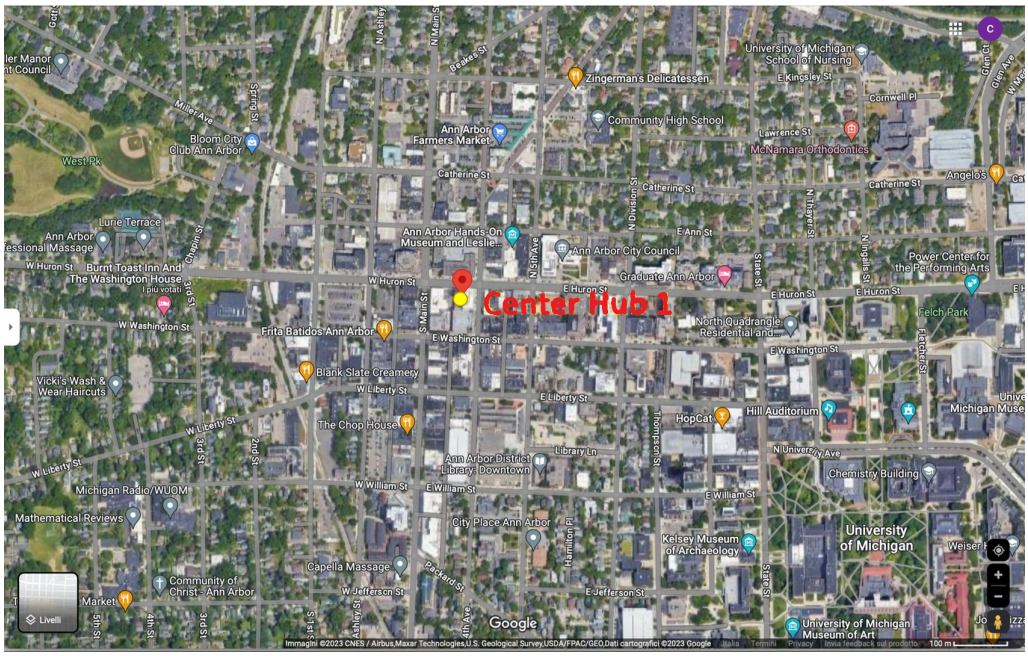


FIGURE 5.3: Selection of the aggregation hub in the Ann Arbor Area: satellite view of Hub_1 area in the city center and university zone **(VED dataset)**.

ii) Meteorological Dataset

The meteorological data can be extracted from the MeteoStat database using the Python API (*Meteo Stat Phyton API* n.d.) based on the GPS coordinates of the geographical area under study. The information on precipitation in mm, temperature in °C, and wind speed in km/h can be extracted on an hourly basis for the period under investigation. The pre-processing of the meteorological information involved the imputing of the missing data, which were replaced with the average of the signal in the same week, and the resampling to the *hh* sampling interval.

iii) National Holidays Dataset

State office closings for state holidays are considered additional information to be integrated into the input dataset. They are regulated by the Michigan Department of Civil Service Regulation 5a.08. Public Act 124 of 1865 is the Michigan law governing official state holidays (*Ann Arbour School Breaks* n.d.; *Michigan ORS Non-Business Days* n.d.). Non-business days, as in Table 5.4, were considered in conjunction with weekend information to obtain a comprehensive holiday rate to be incorporated into the model.

TABLE 5.4: Office of Retirement Services (ORS) non-business days **(VED dataset)**.

Non-Business Day	**Date**
Weekends	Saturdays and Sundays
Thanksgiving Day	23 November 2017
Day after Thanksgiving	24 November 2017
Christmas (Eve and Day)	24–25 December 2017
New Year (Eve and Day)	31 December 2017–1 January 2018
Martin Luther King, Jr. Day	15 January 2018
President's Day	19 February 2018
Memorial Day	28 May 2018
Juneteenth	19 June 2018
Independence Day	4 July 2018
Labor Day	3 September 2018
Columbus Day	8 October 2018
Veterans Day	12 November 2018

A fuzzy model is applied in order to obtain a unified holiday rate resulting from the fuzzification of national holidays and weekend information for each day of the year (Napoli et al., 2024).

The information about holidays and weekends is represented by a discontinuous time series, which is not suitable for the identification of dynamic models. The goal of fuzzifying such inputs is to obtain a continuous time series, with *hh* time interval, that contains both weekend and holiday information. Fuzzy membership functions were developed to account for the effects that weekends and holidays might have on drivers' habits on the days before and after these periods. The weekend and holiday membership functions, shown graphically in Figure 5.4a,b, respectively, were applied to the time series of calendar dates. When the weekend membership function is applied to a calendar day, it outputs a degree of truth, ranging from 0 to 1, indicating the likelihood of the day being a weekend. Similarly, applying the holiday membership function to a calendar day results in a continuous value between 0 and 1, representing the degree of truth for the day being a holiday. To create a single continuous dynamic feature that integrates both weekend and holiday information, the maximum value of these membership functions was taken.

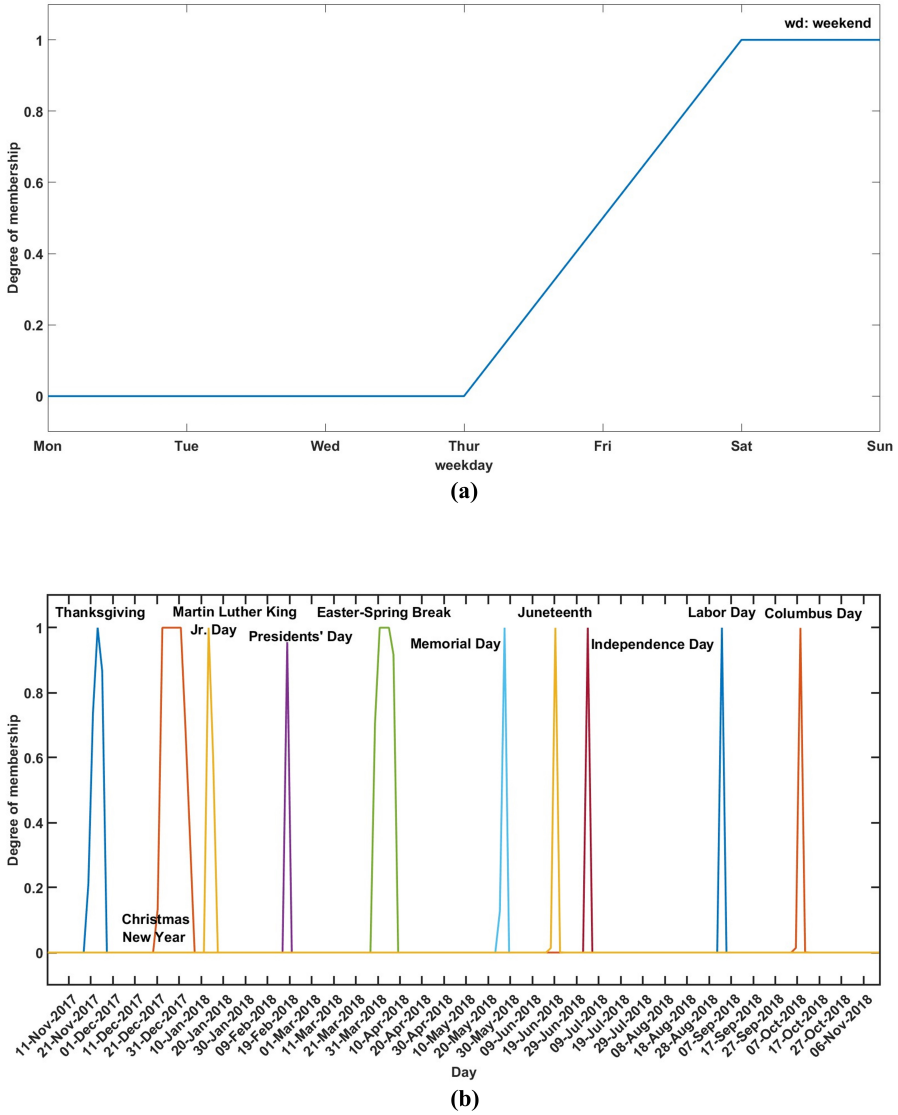


FIGURE 5.4: Membership function for the fuzzification of the holiday rate: (**a**) weekend membership; (**b**) national holiday membership functions **(VED dataset)**.

5.4.2 Methods: HDMDc and LSTM Models

HDMDc is classified as a *knowledge-driven grey* model: grounded in the state-space representation (physics-inspired structure) but with parameters fitted from data. The LSTM model represents a *data-driven grey* approach when equipped with interpretability constraints or post-hoc explanation tools.

Considering the prediction steps k_p, the multi-step-ahead prediction is performed by iterating the one-step-ahead prediction k_p times in a closed loop. This involves using the previously predicted value as an autoregressive term and feeding the inputs from the previous step into the model at each iteration, as shown in Equation (5.3), since the meteorological forecast and calendrical information are considered available within this time interval. During each iteration, any negative predicted values are

replaced with zero and then fed into the model for the subsequent iteration.

$$x_{k+k_p} = f(\tilde{x}_k, \tilde{x}_{k+1}, \cdots, \tilde{x}_{k+k_p-1}, u_k, u_{k+1}, \cdots, u_{k+k_p-1}), \tag{5.3}$$

where f is a general representation of the relationship between the rolling prediction at time k_p-step-ahead and the previously predicted samples and the previous samples of the exogenous inputs.

i) HDMDc

The one-step-ahead predicted discrete time-series based on the Hankel transformation of the original time series (i.e., $\tilde{x}_k^H$ and u_k^H) and based on the identified matrices as defined in Section 4.2.3, and these are represented as in Equation (5.4):

$$\tilde{x}_{k+1}^H = \tilde{A}_H \tilde{x}_k^H + \tilde{B}_H u_k^H, \tag{5.4}$$

with $x_k^H = \hat{U}_r \tilde{x}_k^H$. The original time series x_k is then extracted from x_k^H considering only the rows with index $i = n \cdot q + 1$ where $n = 0, 1, .., (n_x - 1)$.

ii) LSTM

The one-step-ahead time-series based on the LSTM transformation is based on a optimized structure of the network and can be represented in Equation (5.5) as

$$x_{k+1} = f_{LSTM}(x_k, u_k), \tag{5.5}$$

with f_{LSTM} being the simplified nonlinear representation of the input–output relationship in LSTM.

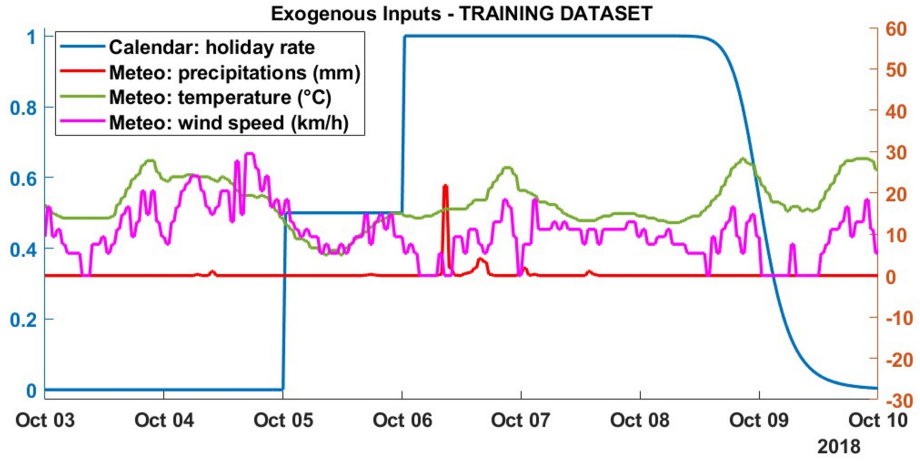
5.4.3 Results and Discussion

In this section, we first describe the models' learning procedures, detailing the identification and prediction processes, and outlining the test cases. These test cases are subsequently elaborated upon in the results subsections for both HDMDc and LSTM.

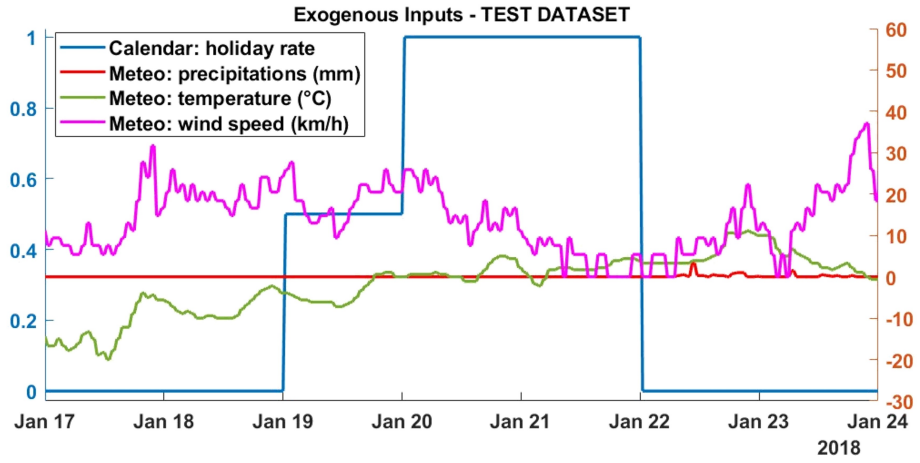
i) Learning Procedure

The data were divided into training (50%), validation (25%), and test data (25%). Specifically, in order to avoid seasonal bias and unbalanced datasets, the first two weeks per month were selected as training, then the following two weeks were divided, one for validation and one as test data. Such a choice ensures that all seasons, weekends, and national holidays are included in both the training and test phases, providing a thorough representation of diverse meteorological conditions and capturing these temporal variations across all datasets. As described in Section 5.3, the exogenous inputs are the fuzzified holiday rate and the meteorological information (precipitation, temperature, and wind speed). Figure 5.5 shows such inputs for two selected weeks: (a) from Wednesday, 3rd October to Tuesday, 10th October 2018 from the training dataset and (b) from 17 to 24 January 2018 from the test dataset. The time series AAC_{Hub^r} refers to the selected Hub_1 with a radius $r = 1$, as in Section i) in Figure 5.3. It is provided as target and autoregressive term in the model identification process. In particular, the training and validation sessions were performed for one-step-ahead prediction, i.e., at each time step in the input sequences, the models learn to predict the value of the subsequent time step.

Parameters choice, training, and test cases are here discussed for the two different predictive methods, HDMDc and LSTM, in Sections 5.4.3-ii) and iii), respectively. Both models' performances in the training and test dataset are shown for different prediction steps (1 h ($k_p = 2$), 2 h ($k_p = 4$), 3 h ($k_p = 6$), 4 h ($k_p = 8$)) and compared in terms of KPIs, i.e., mean square error (MAE), root mean square error ($RMSE$) and correlation coefficient (R), predicted time series and, specifically for the HDMDc, also regression plots.



(a)



(b)

FIGURE 5.5: Exogenous inputs: left y-axis–holiday rate (blue); right y-axis—precipitation in mm (red), temperature in °C (green), and wind speed in km/h (magenta). (**a**) Selection of a training set week - Wednesday 3 to Tuesday 10 October 2018. (**b**) Selection of a test set week - Wednesday 17 to Tuesday 24 January 2018 **(VED dataset)**.

ii) HDMDc

As a first step, the state and input variables have been extended by adding a series of delay coordinates to the state and inputs as described in Section 4.2.3. These delays, q and q_u, as in the Equations (4.6) and (4.7), were both chosen to be equal to each other (Patanè, Sapuppo, and Xibilia, 2024), with a value of 48 corresponding to 24 h. This choice was made based on considerations of the daily periodicity of the data

for both the state and the inputs. The system order n_x therefore increased from 1, given by the single state variable AAC, to $qn_x = 48$, as in Equation (4.17). The number of exogenous inputs increased from $n_u = 4$ to $q_u n_u = 192$. As a second step, once the model structure was determined, the HDMDc model was identified using the training dataset, and the state-space model matrices were obtained as in the Equations (4.2.3) and (4.2.3). Rolling k_p-step-ahead prediction of the AAC signal was therefore performed by iterating Equation (5.4) for both the training and test datasets. The chosen set of prediction steps was $k_p = 2, 4, 6, 8$, corresponding to 1, 2, 3, and 4 h.

The key performance indicators are listed in Table 5.5 for both the training and test datasets. HDMDc performs well on both datasets for predictions up to 4 hours. For long prediction horizons, the HDMDc shows some limitations: It underestimates the target AAC but maintains a good correlation (>0.859) to the target. This observation, which can be derived from the macroscopic key performance indicators, is confirmed by the comparison between the predicted AAC time series and the experimental target value, as shown in Figure 5.6 for two selected weekly examples: (a) the training dataset and (b) the test datasets. The rising/falling slopes and peak shapes are also maintained for long prediction horizons, albeit on a lower scale. The same content, extended to the entire test dataset, is visible in the regression plots in Figure 5.7c,d, where the scatter is closely related to the regression line, even if underestimated. This could not be a critical issue from the energy provider's perspective, since the primary interest is predicting the minimum AAC to sell to the energy market to avoid penalties.

These detailed observations provide valuable insights: the training dataset is well-designed, covering multiple system behaviors and indicating that the exogenous input features are relevant for predicting AAC (Napoli et al., 2024). The HDMDc effectively captures the global dynamics of the system without overfitting, consistent with Koopman theory. This theory allows for identifying a global model that operates properly across multiple system operating points, highlighting the advantage of the data-driven linear state-space model over nonlinear black-box models, as described in Section 5.2. These characteristics make the integration of exogenous input data sources and HDMDc a reliable candidate for automated prediction processes, supporting decision-making for energy providers and managing multiple energy sources in smart grids.

TABLE 5.5: HDMDc performances in the Training and Test Dataset **(VED dataset)**.

	Train			**Test**		
Prediction	**MAE**	**RMSE**	**R**	**MAE**	**RMSE**	**R**
1 h	0.75	1.14	0.995	0.82	1.24	0.995
2 h	1.78	2.65	0.971	1.89	2.86	0.972
3 h	2.55	3.79	0.929	2.69	4.07	0.934
4 h	3.27	4.82	0.859	3.47	5.20	0.869

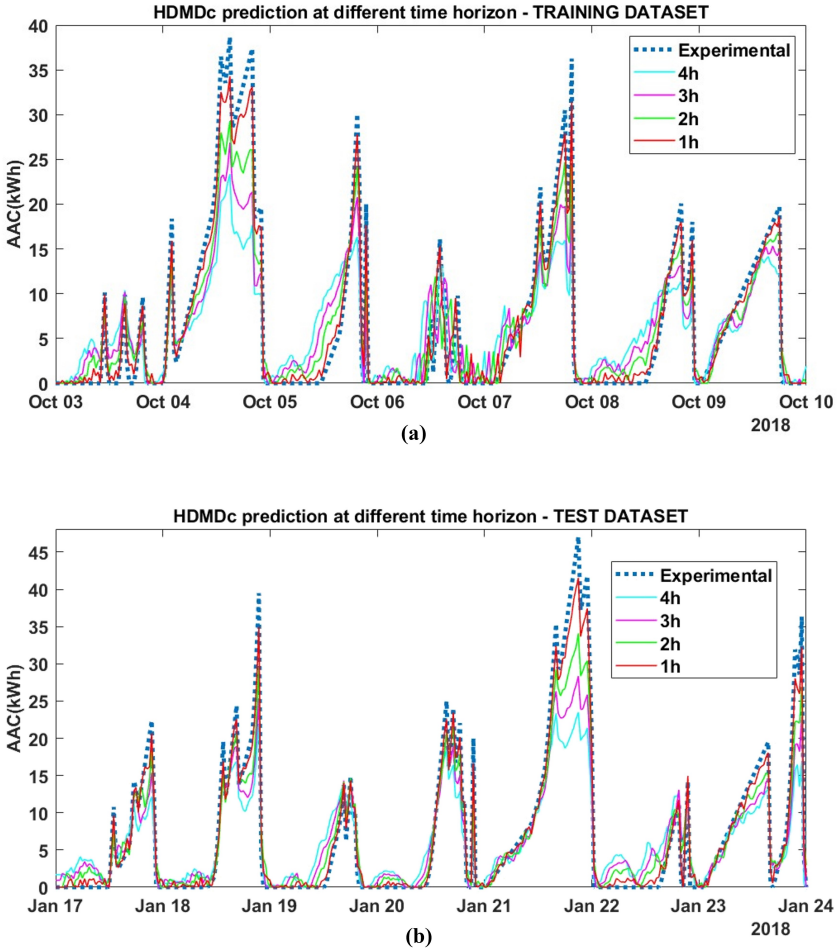


FIGURE 5.6: HDMDc time series prediction with different time horizons: 1 h, 2 h, 3 h, and 4 h. (**a**) Selection of a training set week (Wednesday 3 to Tuesday 10 October 2018) **(VED dataset)**. (**b**) Selection of a test set week (Wednesday 17 to Tuesday 24 January 2018).

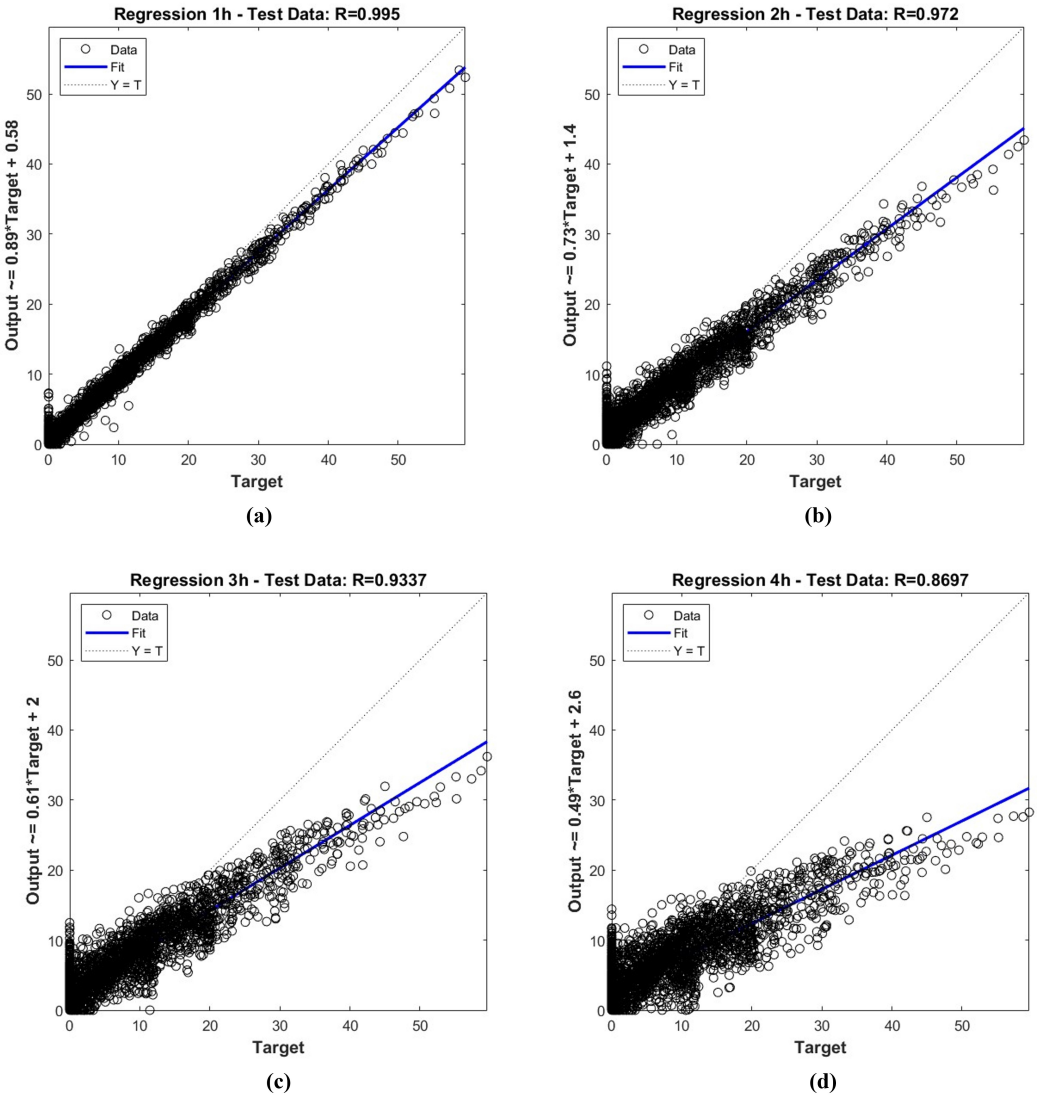


FIGURE 5.7: HDMDc regression plots for prediction with different time horizons of the test dataset: (**a**) 1 h, (**b**) 2 h, (**c**) 3 h, and (**d**) 4 h **(VED dataset)**.

iii) LSTM

In a first step, model optimization to determine the hyperparameters was performed utilizing a Bayesian algorithm using the training and validation datasets. The optimization was based on the minimization of the correlation coefficient (R) as a metric. The hyperparameter ranges were set to

- the LSTM depth between 1 and 3
- the number of hidden units between 50 and 350
- the dropout probability between 0.1 and 0.7
- initial learn rate between 0.01 and 0.1

The hyperparameters optimized in terms of R for the validation datasets were found to be 1 LSTM layers, 176 hidden units per layer, a 0.56 dropout layer, and a 0.034 initial learn rate.

In a second step, after the model structure was determined and identified by the optimization procedure, the LSTM model was used to reconstruct the AAC signal for the k_p-step-ahead prediction by closed loop iteration for both the training and test datasets. As mentioned above, the negative values of the prediction were replaced by zero and used as an autoregressive sample for the subsequent prediction step.

Finally, the performance of LSTM prediction over the different selected time horizons was evaluated and compared. The key performance indicators are listed in Table 5.6 for the training and test datasets.The LSTM model performs acceptably on the training datasets for predictions up to 2 h and maintains a correlation of $R = 0.852$. However, its performance deteriorates rapidly for longer prediction horizons. In the test dataset, the model fails to achieve satisfactory performance and shows a poor correlation coefficient ($R = 0.647$) for a prediction horizon of 1 hour, with a rapid decline for longer horizons. This observation derived from the macroscopic KPI is confirmed by the comparison between the predicted AAC time series and the experimental target values. Figure 5.8 show the predicted timeseries for two selected weekly examples: (a) the training dataset and (b) the test dataset. Among the nonlinear black-box models, the LSTM shows capability in managing integrated data sources and provides good performance for short-term predictions, as the authors have shown in a previous work (Napoli et al., 2024), but as a limitation, it loses accuracy for longer predictions.

TABLE 5.6: LSTM (1 layers, 176 hidden units, 0.56 dropout layer, 0.034 initial learn rate) performances on the training and test dataset **(VED dataset)**.

	Train			**Test**		
Prediction	**MAE**	**RMSE**	**R**	**MAE**	**RMSE**	**R**
1 h	2.28	4.28	0.852	3.67	6.53	0.647
2 h	3.21	4.93	0.80	4.63	7.29	0.54
3 h	4.20	5.79	0.739	5.68	8.06	0.456
4 h	5.10	6.65	0.677	6.57	8.78	0.385

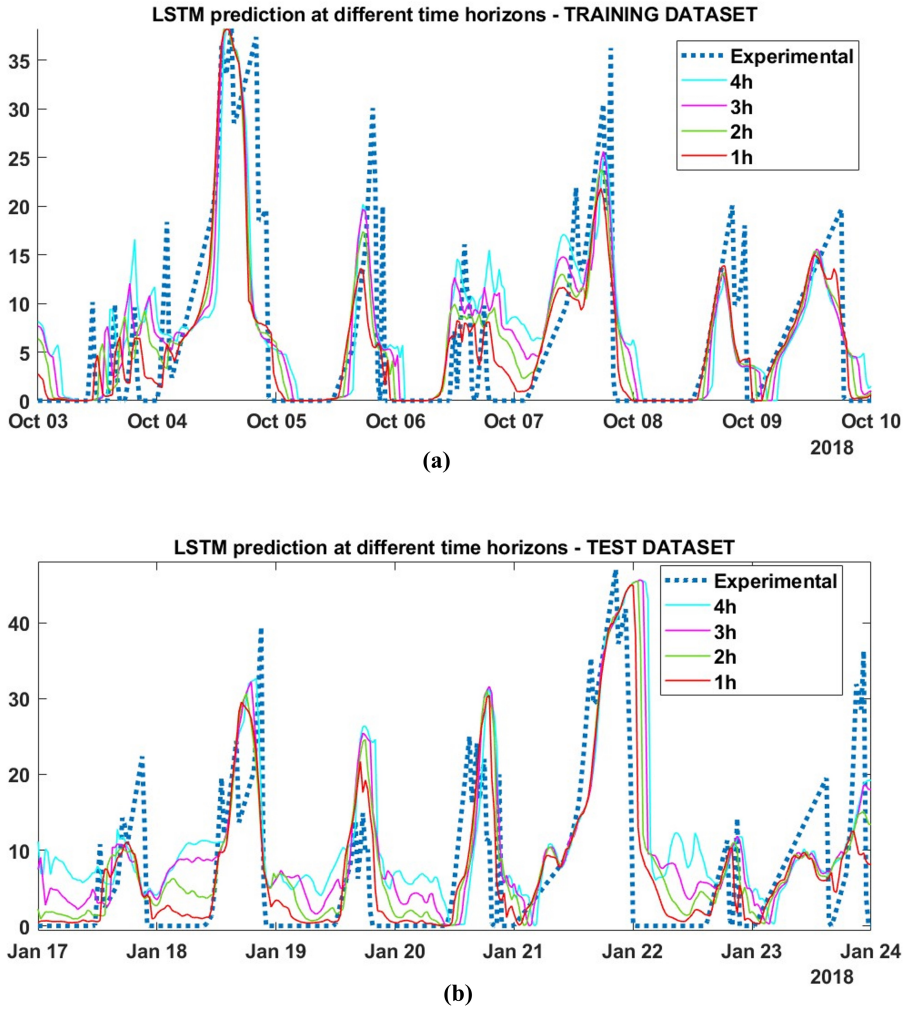


FIGURE 5.8: LSTM time series prediction with different time horizons: 1 h, 2 h, 3 h, and 4 h. (**a**) Selection of a training set week (Wednesday 3 to Tuesday 10 October 2018). (**b**) Selection of a test set week (Wednesday 17 to Tuesday 24 January 2018) **(VED dataset)**.

5.5 Padua Dataset: ARX/FIR/NARX and Transferability

5.5.1 Dataset Description

The available vehicle dataset includes static data on vehicle categories and dynamic data including the geographical coordinates of vehicle routes. Basic vehicle information, including vehicle class, brand, registration year, type, fuel type and gross weight is included in the information form for each vehicle considered in the sample and used for virtual electrification of the vehicles. The vehicle identifier, date (the date on which the record was recorded), timestamp (the time at which the record was recorded), coordinates (the geographical location: longitude and latitude), instantaneous speed, road type (urban, extra-urban, highway) and directional angle are included in the daily trip logs, which contain all trips made by the sampled vehicle in chronological order. However, the dataset does not include information on the purpose of the registered trips (e.g., work or study) or the type of activity the vehicle was engaged in.

The zoning process aligns with the study area data, identifying potential locations for V2G service implementation, such as parking lots near cinemas, retail centers, and workplaces.

An additional dataset comprising traffic flow information, i.e. specifically vehicle counts, is integrated to capture traffic dynamics on motorways and main roads near the designated V2G hub zones. This data provides insights into traffic density and movement patterns, enhancing the analysis by reflecting real-time conditions in areas proximate to potential V2G hubs. The level of the vehicle flows as well as the expansion to the universe of investigation can exploit the information coming from the traffic counts, i.e., counts of users (vehicle) flows, on some elements (links - roads) of the transportation supply system (network). Such info can be easy to obtain, often automatically through sensors or cameras located in designed road sections. In particular, count locations should be designed with respect to their information content and their use, e.g. origin-destination demand estimation.

The extraction and pre-processing structure as described in Section 5.3 are applied to extract the AAC and the traffic data on the different aggregation points for the whole observation period. Considering the large dataset needed by AI approaches, a data augmentation procedure has been finally applied, consisting of increasing the available data by adding uniform noise with a maximum amplitude of 10%, increasing the amount of available data tenfold.

i) Vehicle Dataset

The methodology for estimating the potential energy that can be fed into the electrical grid was applied to Padua, a city in the Veneto region (northern Italy), the third largest region in Italy where EVs are registered. The city of Padua and thirteen smaller neighbouring municipalities form the study area. The total population is around 370,000 inhabitants, mainly concentrated in the centre of Padua, where the residential density is over 58 inhabitants per hectare. Furthermore, the study area has a total of less than 7,000 stores with about 24,000 retail employees, mainly located in the historic city centre and in the small town of Abano Terme. There are about 5,000 warehouses with about 18,000 employees, mainly located west of the city centre, close to the border with Noventa Padovana, where a large warehouse centre is located. Finally, the study area was divided into 56 different zones (Figure 5.9(a)) taking into account the characteristics of land use (e.g. inhabitants, employees, stores) and emphasising the homogeneity of functions (e.g. residential area, shopping area, industrial area). Based on the identified zones and the driving patterns resulting from the analysis of the studied vehicles (i.e. using FCD), some points of interest for the location of the stations for V2G services could be identified. 45 possible points of interest were identified, including shopping malls, hospitals, train stations, universities, large parking lots in high traffic areas (e.g. city centre), cinemas, stadiums and some supermarkets. It was also checked whether these were areas of high attraction by comparing them with the aggregated points related to the destinations of the trips recorded in the study area on the days studied. Of these 45 points, 2 more relevant locations were selected to which the final energy assessment study was applied. These are a peripheral area with medium traffic volume and medium stops (i.e. zone 56) and an area with high traffic volume and medium-short stops (i.e. zone 24).

The available dataset consists of observations collected over five working days during the autumn period (October–November 2018) on selected weekdays:

- Day1: Monday 15/10/2018

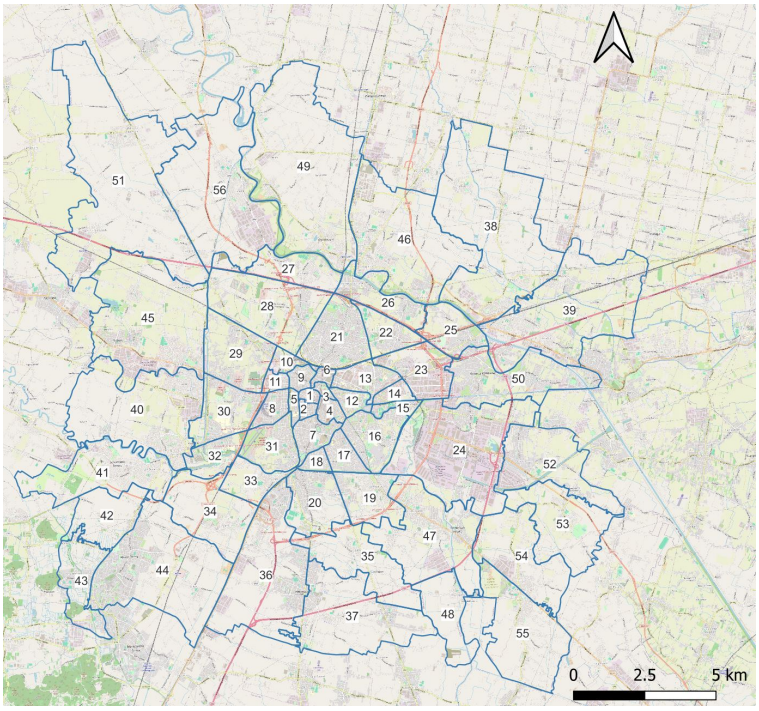
- Day2: Monday 22/10/2018
- Day3: Wednesday 07/11/2018
- Day4: Thursday 15/10/2018
- Day5: Friday 23/11/2018

The selected working days were analyzed for variability in meteorological conditions, and no significant differences were found, as all days had favorable weather according to historical databases. Consequently, no further investigation into the impact of weather, weekends or holidays (Napoli et al., 2024; Patanè et al., 2024b) on the predictive models was conducted. The case study has been developed for testing the proposed methodology in terms of both using easy-to-obtain data that describe well the traffic conditions and outcomes provided. The used data refer to 2018 given that a large dataset was available covering 5 full working days of a large sample of vehicle that during these days of surveys drove in a road of the study area. For such days of survey, several traffic counts within and outside the study area were available. These data allowed to validate the flows estimated. Besides, the goodness and robustness of these data was the objective of further studies which pointed out the robustness of traffic forecast. See for example, (Comi et al., 2021). Therefore, to use such data allowed to really test our proposed methodology to well representative case study.

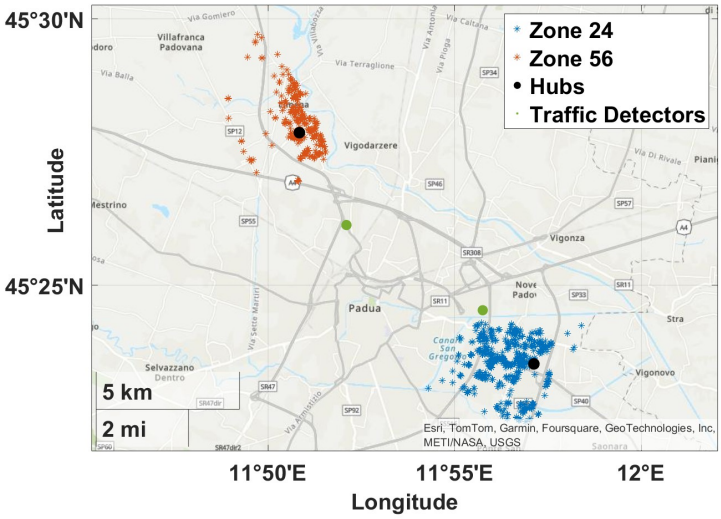
After a comprehensive data-cleaning process, which involved discarding records with incomplete entries, the dataset was analyzed to investigate trip patterns and durations. In total, 29,158 vehicles were examined, resulting in approximately 70,000 recorded trip chains over the course of the five-day observation period. In the land-use analysis, zoning is performed using data from the study area to identify optimal locations for implementing V2G services. Figure 5.9(b) provides an initial analysis of mobility data from the Veneto region, illustrating the distribution of vehicle stops within the study area for a sample day. This visualization enables the identification of zones and potential hubs suitable for V2G services. Figure 5.9(b) shows the location of two traffic detectors monitoring traffic flow toward the identified zones, further supporting hub placement decisions. The vehicles parked refer mainly to systematic travels and the methodology refers to a planning horizon as well as to not only a specific parking lots, but to parking zone. Then, according to specific traffic situations, the users could not reach the specific locations but the area should remain the same given that it is assumed that the traffic jam can push drivers to change their path but not their destination and mode for traveling.

ii) Traffic Dataset

A large sample of automated traffic counts was available in the Veneto region, comprising traffic counts on several motorways and main roads. This dataset enabled the characterization of traffic flows in terms of vehicle counts over five days corresponding to the mobility data sample days. To reconstruct the road network flows and to identify the sampling rate in the study area, it was necessary to acquire a series of information relating to traffic counts on the ANAS and motorway networks (CAV operators, Autovie Venete, Autostrade per l'Italia). Furthermore, in five road sections among those of the ANAS network, manual counts were also carried out, confirming the reliability of the data from the automatic ANAS surveys. From the analysis of the hourly data on the ANAS network, classified by vehicle type, the average hourly profiles for cars and commercial vehicles were identified. It has been revealed that on average, in all the surveyed sections, the hourly profiles are similar, showing two



(a)



(b)

FIGURE 5.9: Padua map for V2G hub selection: (a) Padua area zoning and (b) stop map over a sample day under study for the two selected Hubs in Zone 24 (blue), Zone 56 (red), Hubs position (black points), and related traffic detectors (green points) are also indicated **(Padua dataset)**

peak time slots (a morning one between 6:00 and 8:00 and an afternoon one between 16:00 and 18:00). Between 8:00 and 16:00, the flow of cars remains almost constant, settling between 4% and 7% of the total daily value measured in correspondence with the surveyed section. A given similarity between the hourly distribution of cars and light commercial vehicles (less than 3.5 t p.t.t.) has been identified. From the analysis of the temporal profile of the equivalent flow, it was possible to identify the 7:00 – 8:00 time slot as the morning peak with an average equivalent flow of 1,252 vehicles/hour (i.e. 7.2% of the average daily flow passing through the relevant sections) and the 17:00 – 18:00 time slot as the afternoon peak with an equivalent flow of 1,301 vehicles/hour (i.e. 7.5% of the average daily traffic in the relevant sections).

5.5.2 Methods: ARX, FIR, NARX and Transferability

The Padua dataset focuses on identifying linear and nonlinear structures:

- **ARX** and **FIR** models are *data-driven grey* depending on parametrization.
- **Nonlinear NARX** models often approach *data-driven grey* when interpretability measures are included.
- Transferability analysis examines how model parameters generalize across different scenarios.

In this chapter, we compare the performance and the transferability of different classes of models implemented with state-of-the-art methodologies. Both linear and nonlinear data-driven models have been considered: ARX models, external dynamics nonlinear models implemented with MLP and internal dynamics nonlinear models implemented with LSTM networks.

The ARX model is given by:

$$AAC(t) = \sum_{k=1}^{n} a_k \cdot AAC(t-k) + \sum_{k=1}^{m} b_k \cdot T(t-k) \tag{5.6}$$

where $AAC(t-k)$ for $k = 1, \cdots, n$ are the past samples of the output variable, T is the input vector, a_k and b_k are the model coefficients identified using the least square method through the experimental dataset, n (model order) and m (number of input regressors) have been determined using optimization criteria. The described model is designed to perform one-step-ahead prediction. The prediction over a longer time horizon is obtained by iterating Eq. 5.6, using estimated values of the output.

In the nonlinear ARX model implemented through MLP the following relation is considered:

$$AAC(t) = f(AAC(t-1), AAC(t-2), \cdots, AAC(t-n), T(t-1), T(t-2), \cdots, T(t-m)) \tag{5.7}$$

where f is a nonlinear function implemented by the MLP through multiple hidden layers constituted by neurons with a nonlinear activation function. The n and m parameters were selected equal to the ones identified for the ARX model. The hyperparameters to be optimized are the number of fully connected layers, the layer size and the activation function.

An LSTM network belongs to the recurrent neural network class that processes input data by iterating over time steps and updating the internal state that retains information from the previous time steps. A sequence-to-sequence LSTM neural

network can predict future values in a time series or sequence based on preceding time steps as input (Van Houdt, Mosquera, and Napoles, 2020). An LSTM architecture consists of a sequence input layer, with size depending on the number of input data features, LSTM hidden layers with ReLU activation and dropout implementation to avoid overfitting, a fully connected layer and a regression layer. The hyperparameters to be optimized are the number of LSTM layers and units, and the dropout probability.

The different models were compared in terms of key performance indicators as the root mean squared error (RMSE) and the coefficient of determination (R^2) (Chicco, Warrens, and Jurman, 2021) as defined in Section 2.5.

5.5.3 Results and Discussion

In this section, the results for the one-step- and the three-step-ahead prediction of AAC are presented. The linear ARX model, the nonlinear external dynamic neural model implemented with MLP and the nonlinear internal dynamic model implemented with LSTM are compared. In the first step, a preliminary analysis was performed to select the relevant model inputs.

For this purpose, four different configurations of model inputs were compared:

- **Model A:** pure autoregressive model;
- **Model B:** autoregressive model with the daytime (D) as exogenous input;
- **Model C:** autoregressive model with traffic (Tr) as exogenous input;
- **Model D:** autoregressive model with D and Tr as exogenous inputs.

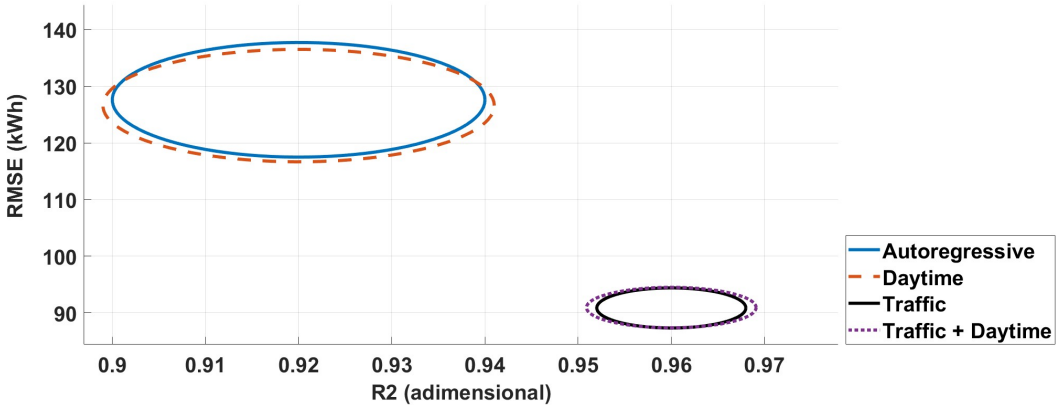
The Akaike's Information Criterion (AIC) was used to select the optimal model order and finite delay in a grid search procedure.

Due to the data scarcity besides the data augmentation procedure mentioned in Section 5.5.1, to improve the statistical significance of the reported analysis, a k-fold cross-validation procedure was considered. In particular, the five days of data available were split into four (training and validation) and one (testing) following a 5-fold cross-validation method. The results obtained are shown in Fig. 5.10, where the four input configurations are compared when the one-step- and three-step-ahead predictions are considered. The figure refers to the ARX models trained and tested on hub 24. Each ellipse shows the distribution of the performance indices used for the test dataset: the RMSE and the R^2.

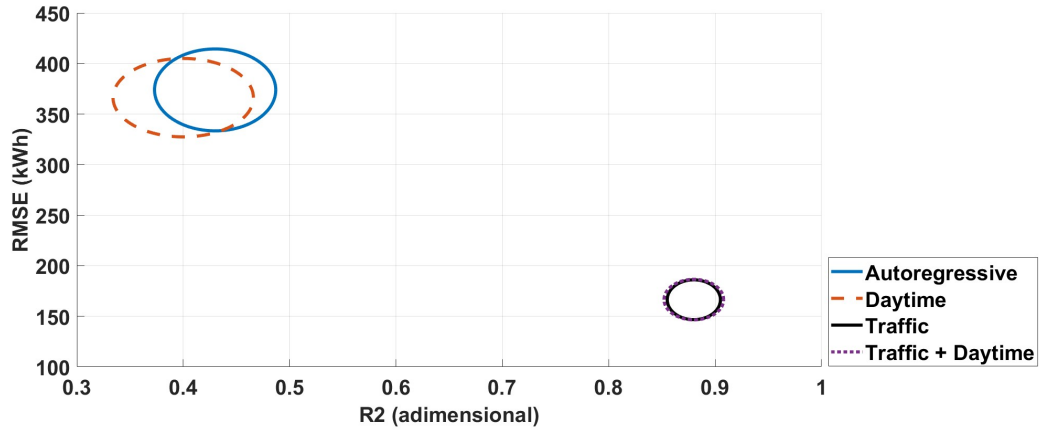
The center of each ellipse corresponds to the mean value over the 5-fold cross-validation, while the dimensions of the axes correspond to the obtained standard deviation.

It can be seen from the figure that the best performance is achieved when traffic is included as an input variable (Model C) while adding the daytime to the input vector is not relevant. Specifically, for the one-step-ahead prediction (Fig, 5.10 (a)), the R^2 index improves from about 0.92 to 0.94 and the RMSE decreases from 127 to 90. An even more significant improvement results for the three-step-ahead prediction (Fig, 5.10 (b)): in this case, from Model A to Model C, the R^2 increases from 0.43 to 0.88, while the RMSE decreases from 373 to 163.

The time evolution of the output variable for the ARX models is shown in Fig. 5.11 for both the one-step- and the three-step-ahead prediction. It can be observed that models that include traffic as an input maintain their performance when a three-step-ahead prediction is calculated. This does not apply to models that contain only past output values.

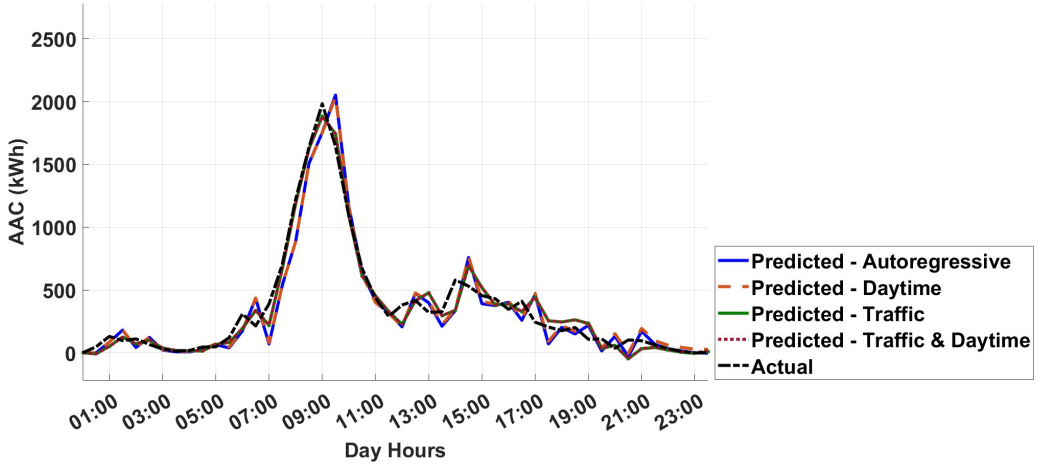


(a)

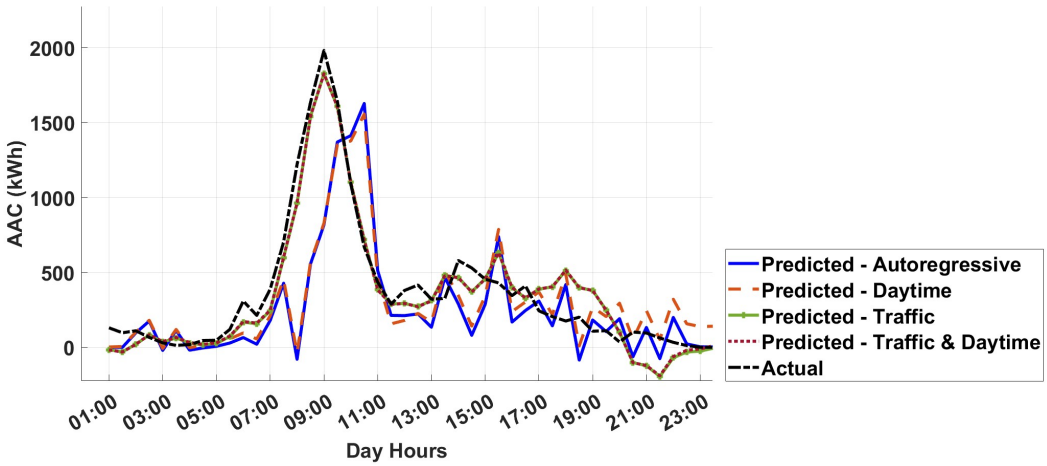


(b)

FIGURE 5.10: Comparison between different ARX models trained and tested on Hub 24 for (a) one-step-ahead prediction and (b) three-step-ahead prediction. The ellipses represent the performance of the ARX model identified with different exogenous inputs: the blue ellipse corresponds to the model based solely on autoregressive input; the orange dashed ellipse includes daytime as an exogenous input; the black solid-line ellipse incorporates traffic as an exogenous input; and the red dotted-line ellipse combines traffic and daytime as exogenous inputs. Each ellipse represents the distribution of performance indices, RMSE and R^2, for the test dataset obtained through a 5-fold cross-validation procedure. The center of each ellipse corresponds to the mean value over the 5-fold cross-validation, while the dimensions of the axes denote the standard deviation, providing insights into the statistical robustness of the models **(Padua dataset)**.

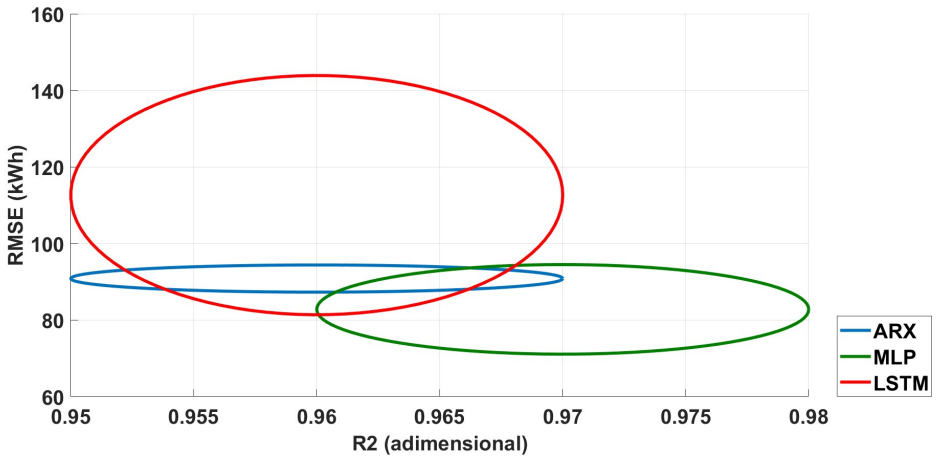


(a)

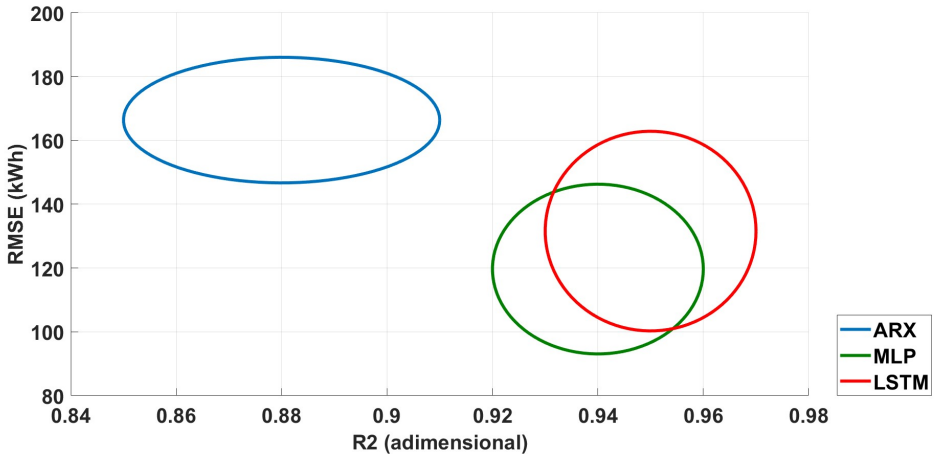


(b)

FIGURE 5.11: Time evolution related to a complete day used to test the different ARX models trained and tested on the Hub 24: (a) one-step-ahead prediction and (b) three-step-ahead prediction. The solid blue line represents the predicted AAC based solely on autoregressive input. The orange dashed line corresponds to the predictions incorporating daytime as an exogenous input. The green line shows the model predictions using traffic data as an exogenous input, while the red dash-dotted line represents predictions combining both traffic and daytime as exogenous inputs. The black dashed line indicates the actual AAC values observed, providing a benchmark for evaluating the model's performance **(Padua dataset)**.



(a)



(b)

FIGURE 5.12: Comparison between ARX (blue), MLP (green) and LSTM (red) models trained and tested on the hub 24: (a) one-step-ahead prediction and (b) three-step-ahead prediction. Each ellipse represents the distribution of performance indices, RMSE and R^2, for the test dataset obtained through a 5-fold cross-validation procedure. The center of each ellipse corresponds to the mean value over the 5-fold cross-validation, while the dimensions of the axes denote the standard deviation, providing insights into the statistical robustness of the models **(Padua dataset)**.

	One-step-ahead prediction				Three-step-ahead prediction	
	Training		Test		Test	
	RMSE	R^2	RMSE	R^2	RMSE	R^2
no input	126.48±9.63	0.92±0.02	127.61±10.12	0.92±0.02	373.90±40.51	0.43±0.06
daytime	125.52±9.50	0.92±0.02	126.59±9.94	0.92±0.02	366.22±38.79	0.41±0.07
traffic	112.53±10.71	0.94±0.02	90.85±3.55	0.96±0.01	166.38±19.68	0.88±0.03
traffic & daytime	113.11±10.57	0.93±0.001	90.92±3.58	0.96±0.01	166.5±20.0	0.88±0.03

TABLE 5.7: Performance analysis of the different ARX models for the hub 24. The mean and standard deviation obtained during the 5-fold procedure are reported for the training and test data **(Padua dataset)**.

	One-step-ahead prediction				Three-step-ahead prediction	
	Training		Test		Test	
	RMSE	R^2	RMSE	R^2	RMSE	R^2
ARX	112.53±10.71	0.94±0.02	90.85±3.55	0.96±0.01	166.38±19.68	0.88±0.03
MLP	47.64±4.15	0.99±0.002	82.77± 11.74	0.97 ± 0.01	119.91 ± 26.62	0.94± 0.02
LSTM	79.62 ± 10.15	0.98± 0.01	112.64 ± 31.25	0.96 ± 0.01	131.58 ± 31.29	0.95 ± 0.02

TABLE 5.8: Performance analysis of the linear and nonlinear models including the traffic as input, for the hub 24. The mean and standard deviation obtained during the k-fold procedure are reported for the training and test data both in the case of one-step- and three-step-ahead prediction **(Padua dataset)**.

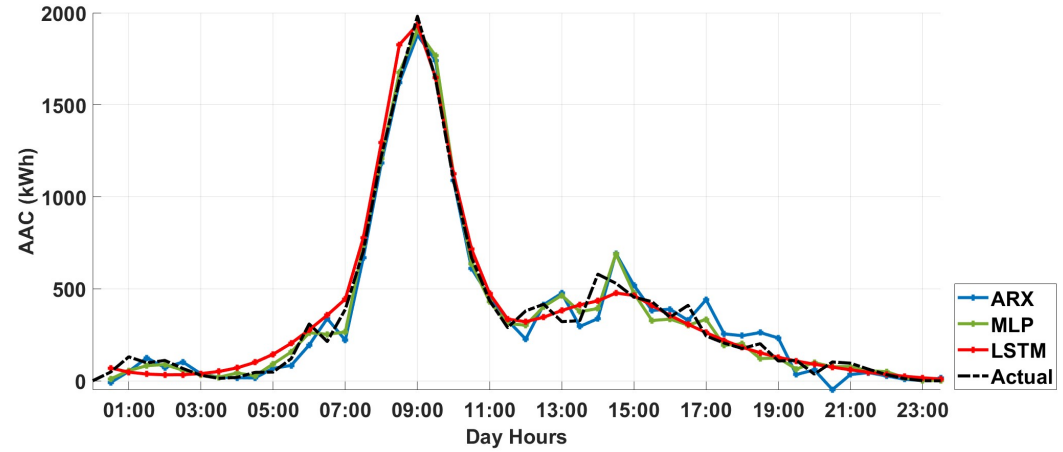
Table 5.7 shows the performance of the best model for each input configuration. A statistical analysis was performed on the five different folds considered for hub 24.

The same comparison was also carried out with nonlinear models. In the case of both MLP and LSTM, the optimal choice of parameters involves traffic among the inputs. The same regressor structure used for the ARX model was also considered for the MLP model. The MLP hyperparameters consist of the number of hidden layers, the neurons per layer and the activation function. A grid search optimization procedure was applied, resulting in the following configuration: three hidden layers with ten neurons each using the ReLU activation function. Optimization of the hyperparameters of the LSTM network resulted in a structure consisting of three layers with five neurons each.

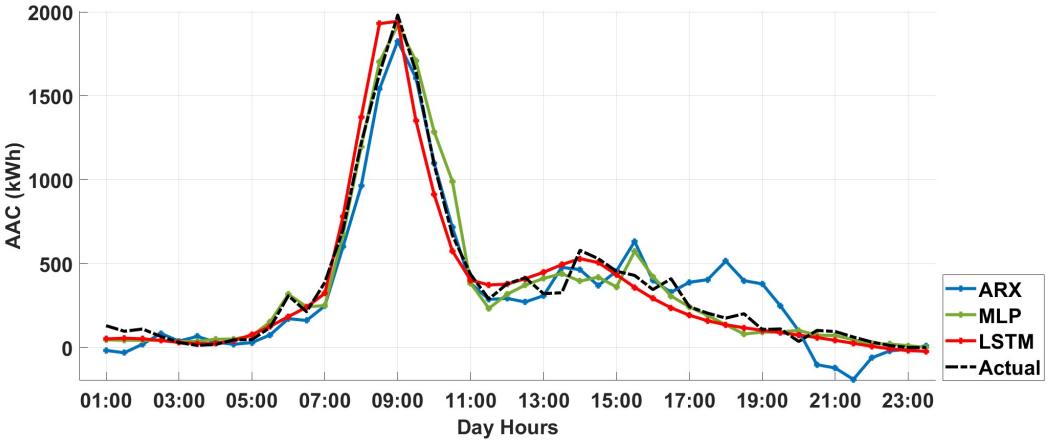
From Tables 5.8 and Fig. 5.12, it can be seen that the MLP shows the best performance in terms of both RMSE and R^2 for one-step-ahead prediction. The ARX shows a lower variance in terms of RMSE compared to the other methods. The LSTM shows performance similar to the MLP. The results for the three-step-ahead prediction (Fig. 5.12 (b)) show a significant drop in performance for the ARX, while the nonlinear models work consistently and show again similar performance. A value of R^2=0.94 and R^2=0.95 is obtained for MLP and LSTM, respectively, while R^2=0.88 is achieved for the ARX model.

Time evolution comparison over one day, which is part of the test set, for the linear and nonlinear models under analysis is reported in Fig. 5.13 where both one-step- and three-step-ahead predictions are shown. The results confirm the superiority of the MLP model, although the other models also provide suitable results, especially in correspondence to the AAC peak.

Similar results were obtained for hub 56, as summarized in Table 5.9 for the best-performing model that has traffic as input.



(a)



(b)

FIGURE 5.13: Time evolution comparison on a test dataset: (a) one-step-ahead prediction and (b) three-step-ahead prediction. The lines in the figure correspond to different models and their predictions. The blue solid line represents the predictions from the ARX model. The green solid line corresponds to the predictions from the MLP model, while the red solid line indicates the predictions from the LSTM model. The black dash-dotted line represents the actual observed AAC values, serving as a reference for evaluating the predictive performance of the models **(Padua dataset)**.

	One-step-ahead prediction				**Three-step-ahead prediction**	
	Training		**Test**		**Test**	
	RMSE	R^2	**RMSE**	R^2	**RMSE**	R^2
ARX	64.08±7.30	0.88±0.02	61.46±9.74	0.89±0.02	98.58±11.89	0.72±0.04
MLP	40.45±6.36	0.95±0.01	55.04±6.77	0.91±0.02	69.58±9.90	0.87±0.04
LSTM	42.58 ± 4.3	0.95± 0.01	70.37 ± 16.62	0.88 ± 0.04	73.77± 15.08	0.87± 0.04

TABLE 5.9: Performance analysis of the linear and nonlinear models including the traffic as input, for the hub 56 **(Padua dataset)**.

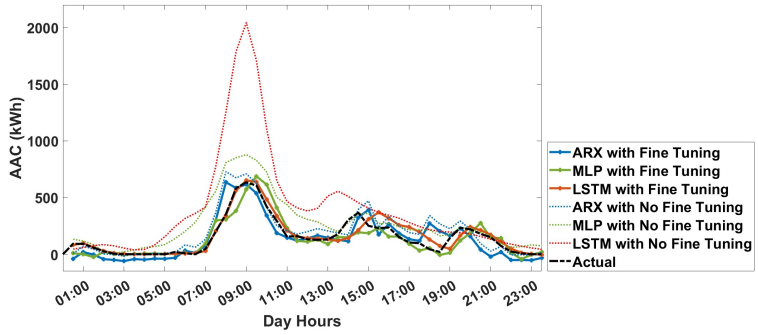
i) Model transferability analysis

In this section, the transferability of the models obtained from one hub to another is analyzed. Two different scenarios are considered. In the first scenario, it is assumed that no data is available for the target domain. This means that the models derived for the source zone are simulated directly with the inputs of the target zone to calculate the predictions. In the second scenario, it is assumed that only a limited amount of data is available from the target area. In this case, the model trained based on the source data is fine-tuned using only one day of data collected in the target area (20% of available data). In the case of MLP and LSTM, this means a few training epochs with the new dataset. In the case of the ARX model, a scaling of the target domain dataset is performed to match it to the source domain dataset. The models obtained from hub 24 were simulated with the input data from hub 56 and vice versa, to implement the first scenario. The results are shown in Tab. 5.10.

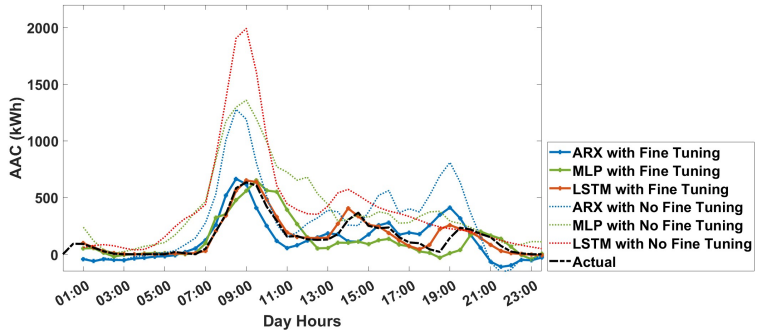
The performance of the models obtained for hub 24 and transferred to hub 56 is not satisfactory. For example, the R^2 value of the ARX model decreases from 0.88 (see Tab. 5.9) to 0.67 for one-step-ahead prediction and from 0.72 (see Tab. 5.8) to 0.04 for three-step-ahead prediction, making the model unsuitable. Better results were obtained by transferring the model from hub 56 to 24. In this case, the ARX model decreased from 0.94 to 0.87 for the one-step-ahead prediction and from 0.88 to 0.55 for the three-step-ahead prediction. The performance degradation indices ΔRMSE and ΔR^2 are also included in the table. It should be noted that, in some cases, the R^2 is negative, so in these cases, the chosen model fits worse than using the mean value as an estimator. Similar results are obtained for the other models. It can be concluded that none of the models achieves an adequate level of performance using the direct transfer method considered in the first scenario. The performance drop can be related to some differences in cluster characteristics. The hubs analyzed are a peripheral area with medium traffic volume and medium stops (zone 56) and an area with high traffic volume and medium-short stops (zone 24). The different characteristics of the two hubs are the main cause of the drop in performance when the models are transferred directly from one hub to the other. This is overcome by using a fine-tuning strategy.

The results obtained for the second scenario are shown in Tab. 5.11. In this case, the ARX shows better performance with respect to the first scenario, when both the one-step-ahead and three-step-ahead prediction are considered (R^2=0.69 and R^2=0.41, respectively). However, the results are still not satisfactory. The nonlinear methods show, instead, quite good performance. Specifically, in the case of transfer from hub 24 to 56, the MLP decreases from R^2=0.95 (see Tab. 5.9) to R^2=0.80 for the one-step-ahead prediction and from R^2=0.95 (see Tab. 5.8) to R^2=0.68 for the three-step-ahead prediction. The LSTM decreases from R^2=0.87 (see Tab, 5.9) to R^2=0.81 for the one-step-ahead prediction and from R^2=0.87 (see Tab. 5.8) to R^2=0.80 for the three-step-ahead prediction, revealing to be the best method as regards transferability. Even better results are obtained for the transfer from hub 56 to 24. The global analysis of the model performance leads to the conclusion that the LSTM performs better than the other approaches considered, as also shown by the time evolutions reported in Figure 5.14.

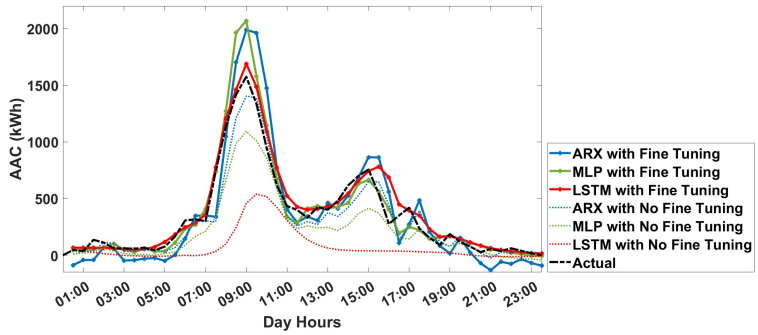
Although the proposed analysis was only tested on two hubs, in the absence of further experimental data, the transfer learning approach is a promising method for handling the prediction of AAC in new hubs, reducing the need for data collection. In the presence of multiple hubs, the ability to cluster the hubs, based on their characteristics, can further improve the transfer procedure.



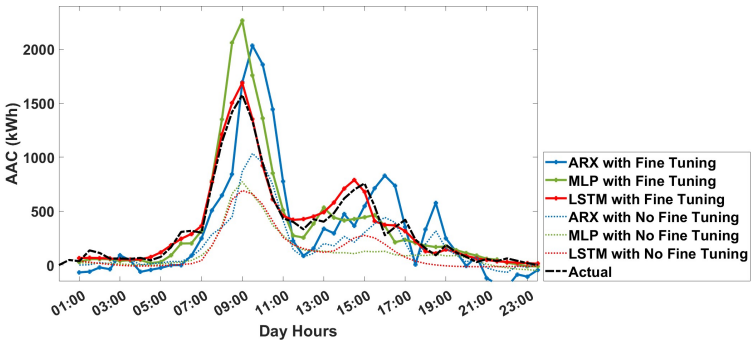
(a)



(b)



(c)



(d)

FIGURE 5.14: Comparison between the time evolution of the analyzed predictive models for the test data when the transfer approaches are adopted: direct transfer and fine-tuning from hub 24 to 56 for the (a) one-step-ahead and (b) one-step-ahead prediction and, similarly, from hub 56 to 24 (c) one-step-ahead and (d) one-step-ahead prediction. The predictive models under transfer learning scenarios are represented as follows: the ARX model (blue) with fine-tuning (solid) and without fine-tuning (dotted); the MLP model (green) with fine-tuning (solid) and without fine-tuning (dotted); and the LSTM model (red) with fine-tuning (solid) and without fine-tuning (dotted). The black dashed line indicates the actual AAC values, serving as a reference **(Padua dataset)**.

		One-step-ahead prediction				Three-step-ahead prediction			
		RMSE	R^2	**ΔRMSE**	ΔR^2	**RMSE**	R^2	**ΔRMSE**	ΔR^2
Transfer Hub 24 ->Hub 56	**ARX**	118.63 ± 9.38	0.67 ±0.06	-48%	-25%	299.43 ±20.94	0.04 ± 0.80	-67%	-94%
	MLP	159.25 ± 20.82	0.52 ±0.07	-66%	-43%	317.02 ± 28.78	0.20±0.09	-78%	-77%
	LSTM	362.25 ±34.51	<0	-81%	<-100%	385.39 ± 26.198	<0	-81%	<-100%
Transfer Hub 56 ->Hub 24	**ARX**	140.84 ± 12.17	0.87 ±0.07	-35%	-9%	327.70 ±30.02	0.55 ± 0.13	-49%	-38%
	MLP	238.87 ± 38.61	0.22 ±0.64	-65%	-77%	419.03 ± 69.70	<0	-71%	<-100%
	LSTM	423.87 ± 51.51	0.20 ±0.15	-75%	-79%	384.95 ± 51.59	0.35 ±0.03	-66%	-63%

TABLE 5.10: Performance analysis for the transfer of the model derived for Hub 24 to Hub 56 and viceversa **(Padua dataset)**.

		One-step-ahead prediction				Three-step-ahead prediction			
		RMSE	R^2	**ΔRMSE**	ΔR^2	**RMSE**	R^2	**ΔRMSE**	ΔR^2
Transfer Hub 24 ->Hub 56	**ARX**	99.90 ± 7.11	0.69 ± 0.03	-38%	-22%	139.92±14.26	0.41±0.09	-30%	-43%
	MLP	81,22±11.59	0.80±0.03	-32%	-12%	105.13±13.70	0.68±0.06	-34%	-22%
	LSTM	80,98±11.81	0.81±0.07	-14%	-8%	83.85±8.90	0.80±0.06	-13%	-8%
Transfer Hub 56 ->Hub 24	**ARX**	156.43 ± 25.15	0.87 ±0.07	-42%	-9%	239.26±33.72	0.58±0.13	-43%	-34%
	MLP	116.33±14.11	0.93±0.02	-29%	-4%	148.95±37.86	0.88±0.07	-19%	-6%
	LSTM	148.31±35.54	0.89±0.09	-28%	-7%	148.03±45.23	0.89±0.09	-12%	-6%

TABLE 5.11: Performance analysis for the transfer with finetuning of the model derived for Hub 24 to Hub 56 and viceversa **(Padua dataset)**.

5.6 Rome Dataset: XAI for Black-Box Models

5.6.1 Data Collection and Analysis

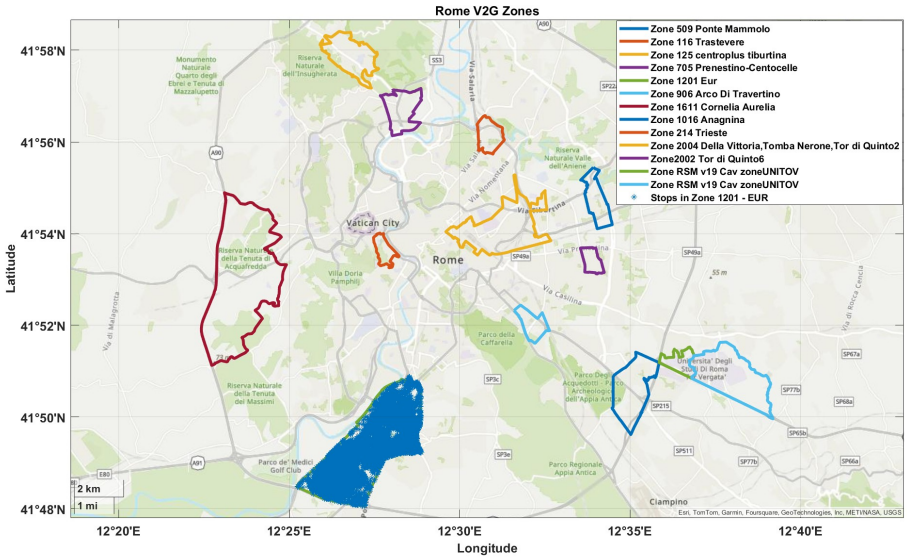
Data pre-processing integrates multiple sources, standardizes formats, and applies cleaning, normalization, aggregation, and feature engineering for AAC prediction.

The data consist of information on car journeys performed by a sample of vehicles driving within the Latium region (that is, at least one survey data within the region on the day of the survey) from the first to the last trip made during the whole day. The data have been analysed to identify travel patterns, thus obtaining indications on the trips performed. The information form includes basic vehicle data for each sampled vehicle, including vehicle class, brand, year, type, fuel type, and gross weight. The vehicle identifier, date (the day the record is logged), timestamp (the time the record is logged), coordinates (the geographic location: latitude and longitude), instantaneous speed, type of road (urban, extra-urban, freeway), and direction angle are all included in the daily car operation logs, which list all trips the surveyed vehicle made in chronological order. There is no information available regarding the nature of the activity performed or the trip purpose (e.g., work or study) of the surveyed cars. A subset of 661 internal combustion engine vehicles, active under real-world conditions during 2023 and circulating in a specific area of interest (EUR, Rome), was analyzed. Vehicles were classified following established procedures (n.d.[b]) and virtually electrified by assigning the determined values for battery pack energy and energy consumption. This enabled the analysis of virtual Battery Electric Vehicles (BEV) equivalents. The available database consists of 58 days spread over four time intervals (February 15–28, May 31–June 15, July 12–25, and September 27–October 10) capturing seasonal and calendar-related variations.

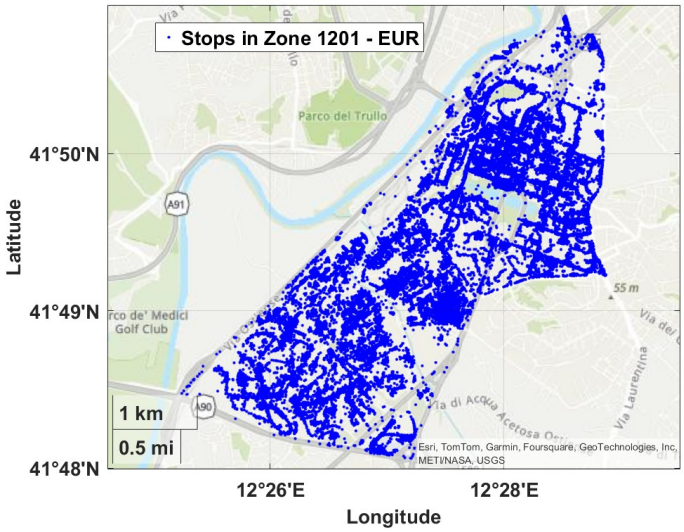
i) Metropolitan Area Zoning and Hubs Selection

An initial analysis of the mobility data was carried out by plotting the stops in the selected time intervals on the global map to identify geographical points suitable as V2G hubs. Figure 5.15 shows a preliminary analysis of the mobility data and the density of stops. The classification of areas of interest was based on their primary

functions or designated land use. The EUR zone in Rome was selected: it is primarily designated as a business district and public office area, with commercial land use being the predominant category.



(a) Rome Zones



(b) Rome Zones

FIGURE 5.15: Rome V2G zone candidates: (a) entire metropolitan area, (b) stops in the EUR zone in Rome.

ii) Trip Chain and Stops Extraction

The car trip detection phase identifies activity stops for each sampled vehicle based on predefined guidelines, particularly for BEV equivalents. It extracts origin-destination (O-D) trips for resident and non-resident vehicles, including travel times, O-D locations, battery charge at stops, and subsequent trips. GPS coordinates and vehicle

status (traveling or stopped) determine trip start and end points by detecting significant position changes. A trip chain is defined as a sequence of trips where each destination aligns with the next trip's origin. This fine-grained FCD analysis enables determining the battery levels and subsequent vehicle activities after each stop.

iii) V2G Activity and State of Charge (SoC) Simulation

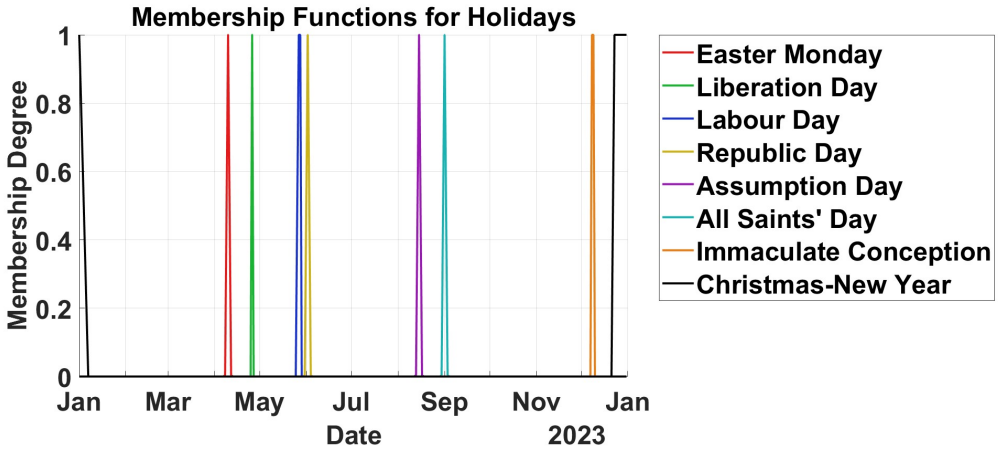
Based on the virtual electrification of the fleet, SoC is estimated on distance traveled and charging stops, assuming an initial 100% SoC and a minimum threshold $SoC_{\min}$, based on a constant (30%) or on the charge required to complete the trip chain (Shipman et al., 2021a). Key parameters, provided by the virtual electrification process, include energy consumption per kilometer, V2G discharge rates, charging efficiency, and power export ratings, with DC fast charging during the day and slow charging at night.

iv) Spatial and Temporal Aggregation

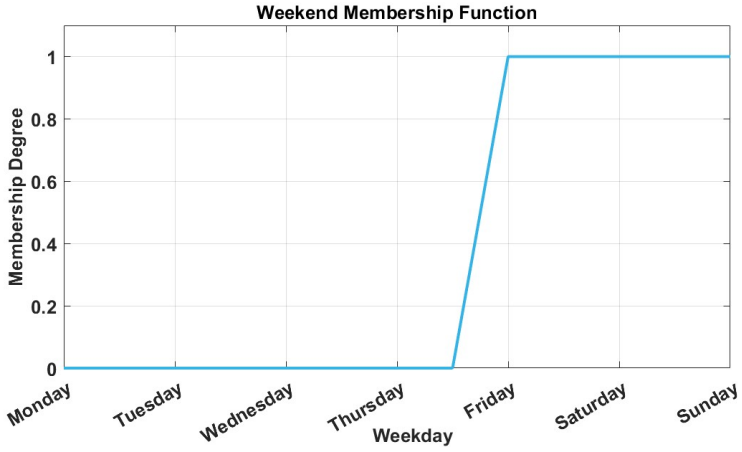
Data Trips and stops are mapped to generate a spatiotemporal dataset for V2G aggregation, using a parametric Th interval to create time series for dynamic prediction. The available capacity of each vehicle (AC_v) in a half-hour period is derived from real or simulated SoC_v. The available capacity per vehicle over T_{hh} and the aggregated available capacity over the hub zone follow Eqs. 5.1 and 5.2, respectively, with the Rome hub area denoted by Hub^r.

v) Meteorological Dataset

The meteorological data was extracted from the Visual Crossing Weather Service database (*Visual Crossing Weather Data Service* n.d.). The information on precipitation, precipitation probability, temperature, feels-like temperature, humidity, sea level pressure, visibility, cloud cover, conditions, wind speed and wind direction was extracted on an hourly basis for the period under investigation. The pre-processing involved the imputing of the missing data and the resampling to half hour period. The missing data were replaced using a moving average filter.



(a) Holiday



(b) Weekend

FIGURE 5.16: Membership function for the fuzzification of the holiday rate. (a) Weekend membership. (b) National holiday membership functions **(Rome dataset)**.

vi) National Holidays Fuzzy Set

The information about Italian national holidays and weekends is represented by a discontinuous time series unsuitable for dynamic models. The objective of the fuzzification of such inputs is to obtain a continuous time series including both weekend and national holiday information. Fuzzy membership functions were generated by considering the effect that holidays and weekends could have on drivers' habits in the previous and successive days. The membership functions for the holidays and weekends, Figures 5.16(a) and (b) respectively, were both applied to the original dates, and the maximum value was taken to obtain a single continuous dynamic feature comprehensive for both information.

5.6.2 Explorative Data Analysis

An initial feature selection was conducted by eliminating exogenous input variables with low variability, high correlation to the other input variables or low correlation to the predicted variable. Specifically, feels-like temperature and humidity were excluded due to their high correlation with temperature, sea level pressure was

removed due to its weak correlation with AAC, and visibility was discarded because of its minimal variability in the Latium area.

i) Zero-Lag Correlation

Figure 5.17 presents the correlation coefficients between the input features and the predicted AAC variable, offering a static overview of their relationships.

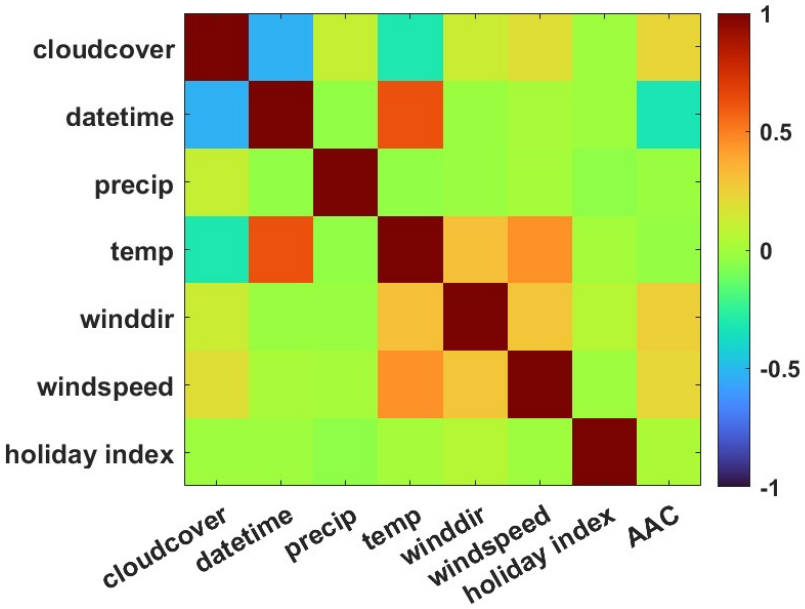


FIGURE 5.17: Correlation Matrix **(Rome dataset)**

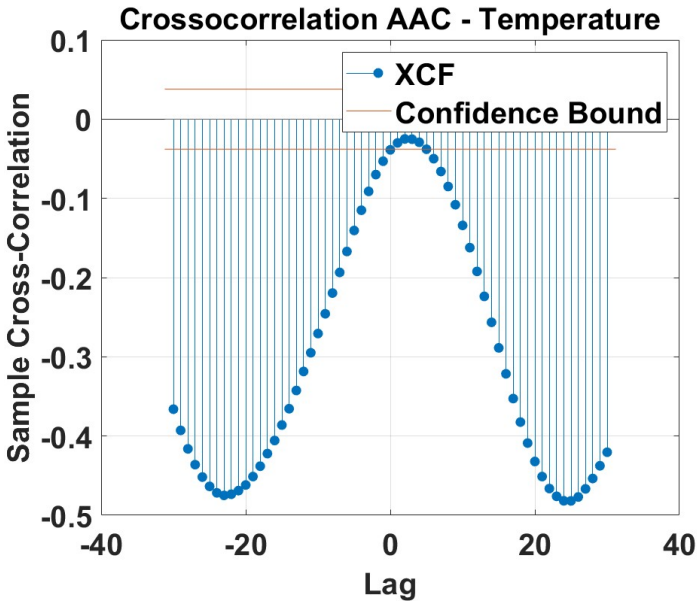


FIGURE 5.18: Correlation curve between the AAC target variable and the exogenous inputs **(Rome dataset)**.

ii) Dynamic Model Order Selection

Given the dynamic nature of the system, the model order was determined through correlation curve analysis at various lags between each selected input and the predicted AAC variable. This analysis guided the selection of input and output regressors, ensuring that relevant temporal dependencies were captured. AAC demonstrated correlations with past temperature values, as depicted in Figure 5.18, justifying the inclusion of lagged temperature values as regressors. A lag of 24 samples (equivalent to 12 hours) was selected based on the observed persistence in temperature, wind speed, and wind direction correlations, which also appeared reasonable for the holiday index. Conversely, variables exhibiting the highest zero-lag correlation (e.g., cloud cover and datetime) were included without additional regressors. The 24-sample lag was also applied to the autoregressive components to maintain consistency in capturing temporal dependencies.

5.6.3 Methods: ARX, NARX and XAI

Data-driven modeling techniques were employed, with a focus on interpretability and explainability to support energy providers in V2G applications. Linear models offer inherent interpretability, as they are represented by equations where coefficients indicate the relative importance of system variables in the prediction. In contrast, ML-based nonlinear models necessitates post-hoc explanation methods.

Modeling approaches for time series prediction depend on the selection of the appropriate model class, whether linear or nonlinear. Models can incorporate only autoregressive components, leveraging past values of the predicted variable (AR for linear and NAR for nonlinear models), only exogenous inputs, capturing external influencing factors (FIR for linear and NFIR for nonlinear models), or a combination of both autoregressive and exogenous inputs (ARX for linear and NARX for nonlinear models).

i) Linear Models

A linear ARX model set is determined by two polynomials whose degrees are n_a and n_b, respectively:

$$\begin{aligned} A(z^{-1},\theta) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a} \\ B(z^{-1},\theta) &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b} \end{aligned} \tag{5.8}$$

where z^{-1} represents the time delay operator and θ is the set of parameters:

$$\theta := [a_1 \;\; a_2 \;\; \cdots \;\; a_{n_a} \;\; b_1 \;\; b_2 \;\; \cdots \;\; b_{n_b}]^T \tag{5.9}$$

The acronym ARX can be explained in the model equation form for the calculation of $y(t)$, the predicted output at the time instant t:

$$A(z^{-1},\theta)y(t) = B(z^{-1},\theta)u(t) + e(t) \tag{5.10}$$

where $e(t)$ is a zero-mean white noise process and $u(t)$ is the exogenous input vector (Ljung, 1986).

AR refers to the AutoRegressive part $A(z^{-1},\theta)y(t)$ in the model, while X refers to an eXogenous term $B(z^{-1},\theta)u(t)$. The model set is completely determined once the integers n_a, n_b, and the parameter set θ have been specified. The AR model is a special case of Eq. (5.8) with $n_b = 0$. Alternatively, Finite Impulse Response (FIR) models can

be used when output regressors are omitted. FIR is a special case of Eq. (5.8) with $n_a = 0$.

ii) Nonlinear Models

Different ML-based nonlinear models (Lindholm et al., 2022) were applied with different input sets.

- Tree Ensemble (TE): Ensemble models combine results from many weak learners into one high-quality ensemble model.The hyperparameters to be optimized are ensemble method (Bag, LSBoost), Number of learners (10-500), Learning rate (0.001-1), Minimum leaf size (1-1080), Number of predictors to sample (1-146)
- Gaussian process regression (GPR): The hyperparameters to be optimized are: Basis function (zero, constant, linear), Kernel function (Rational Quadratic, Squared Exponential, Matern 5/2, Matern 3/2, and Exponential), Sigma.

iii) Model Interpretation and Explaination

The explanation of the model is crucial for two main reasons. Firstly, it aids in the design and validation of the predictive model by providing insights into input selection. Secondly, it ensures reliability for the end user, specifically the energy provider, by justifying the model's predictions. Black-box nonlinear models require explainability techniques such as SHAP (Lundberg and Lee, 2017b) to interpret their decision-making process. It applies cooperative game theory principles to attribute feature contributions in predictive models. Each model input feature is treated as a game player, and the model function defines the game rules. The computed Shapley values quantify each feature's contribution to the model's prediction by assessing its impact across all possible coalitions. Given a model $f(x)$ and an input instance x, the Shapley value for each feature i is computed as:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[f(S \cup \{i\}) - f(S) \right] \tag{5.11}$$

where S represents all possible feature subsets excluding i, and N is the total set of features. SHAP ensures a fair feature attribution by considering all possible feature coalitions. The sum of all Shapley values satisfies:

$$f(x) = \sum_{i=1}^{N} \phi_i \tag{5.12}$$

which decomposes the model output into additive feature contributions. This property enables SHAP to perform feature importance analysis, thereby enhancing trust in AI-driven decision-making.

5.6.4 Results and Discussion

Different model identification and prediction were performed for the selected model class, input set and the regressors:

- *FIR/NFIR*
- *ARX/NARX*

Data was divided into training and validation (85%) and testing (15%) datasets. In particular, to avoid seasonal bias and unbalanced datasets, each period of the year for which data is extracted was divided in training/validation and test, and holiday weeks were included both in the training and test phases. A k-fold validation technique with $k = 5$ was applied to avoid overfitting. Data was standardized using z-score normalization. The model optimization for hyperparameter determination was implemented via a Bayesian algorithm on the training/validation data for each input set. The optimization metric was based on the minimization of the root mean square error (RMSE), and global performances were compared using also mean absolute error (MAE) and determination coefficient (R^2).

i) FIR/NFIR Identification

The optimized hyperparameters for the nonlinear ML-based models were determined as follows:

- TE: ensemble method (LSBoost), Number of learners (484), Learning rate (0.17), Minimum leaf size (29), Number of predictors to sample (6)
- Gaussian process regression (GPR): The hyperparameters to be optimized are: Basis function (constant), Kernel function (Exponential), Sigma (Automatic).

ii) ARX/NARX Identification

The optimized hyperparameters for the nonlinear ML-based models were determined as follows:

- TE: ensemble method (LSBoost), Number of learners (181), Learning rate (0.12), Minimum leaf size (9), Number of predictors to sample (34)
- Gaussian process regression (GPR): The hyperparameters to be optimized are: Basis function (constant), Kernel function (Exponential), Sigma (Automatic).

iii) Model Comparison

The models were applied for a 30-minute (1 sample) ahead prediction and compared based on global key performance indicators (KPIs) and their time prediction response. Table 5.12 highlights that ARX/NARX models generally outperform FIR/NFIR models, indicating that exogenous inputs alone do not sufficiently capture the system dynamics without incorporating historical information of the target AAC variable. This suggests that autoregressive components play a fundamental role in capturing temporal dependencies in AAC prediction.

Within the FIR/NFIR models, the Exponential GPR exhibits slightly superior performance in the training/validation phase, whereas the TE achieves the best results in the test dataset. This suggests that exogenous inputs exhibit a nonlinear relationships with the involved variables, making nonlinear models more suitable for capturing their influence. A similar pattern is observed in ARX/NARX models: the Exponential GPR performs better in training/validation, while the TE achieves the best correlation with actual AAC values in the test set. The superior generalization of the TE in the test phase suggests its robustness to unseen data.

To further investigate model performance, the time response analysis is illustrated in Figure 5.19. Both the Exponential GPR and the TE exhibit strong alignment with

TABLE 5.12: Performance metrics for different model types grouped by model class **(Rome dataset)**.

Model Class	Model Type	RMSE (Valid.)	R^2 (Valid.)	MAE (Valid.)	RMSE (Test)	R^2 (Test)	MAE (Test)
FIR/NFIR	**Linear Regression**	0.62	0.41	0.46	0.59	0.20	0.45
	Exponential GPR	0.38	0.77	0.27	0.61	0.14	0.47
	Tree Ensemble	0.32	0.84	0.23	0.58	0.23	0.46
ARX/NARX	**Linear Regression**	0.31	0.85	0.23	**0.29**	0.71	0.23
	Exponential GPR	**0.30**	**0.86**	**0.21**	0.34	0.74	0.26
	Tree Ensemble	0.31	0.85	0.22	0.32	**0.77**	**0.23**

the true AAC dynamics, accurately capturing peak variations, particularly around 12:00 AM, which is crucial for V2G ancillary service provision.

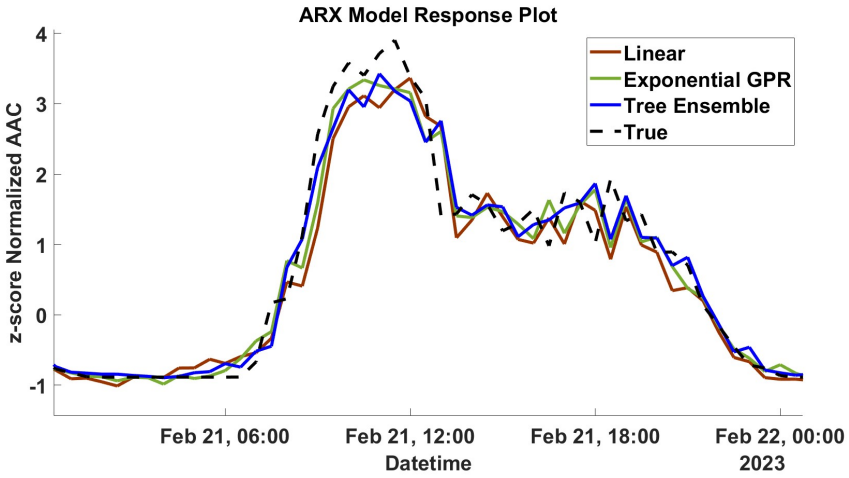
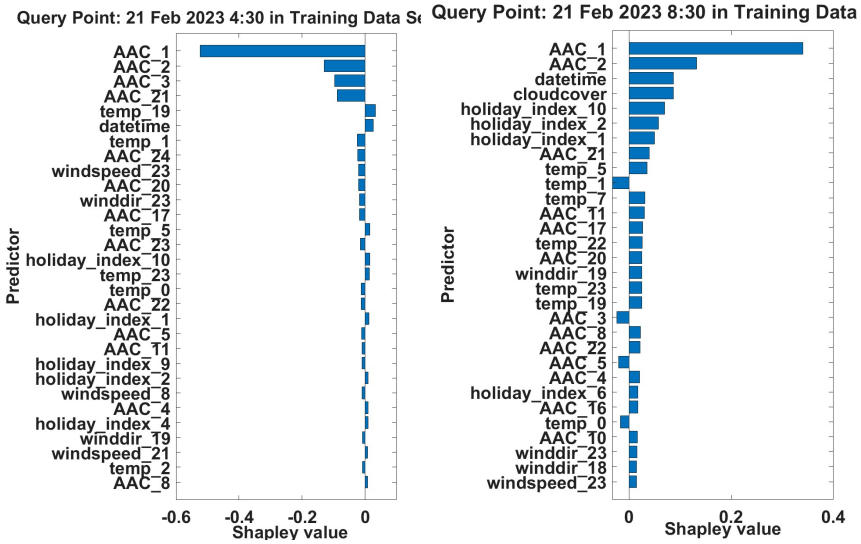
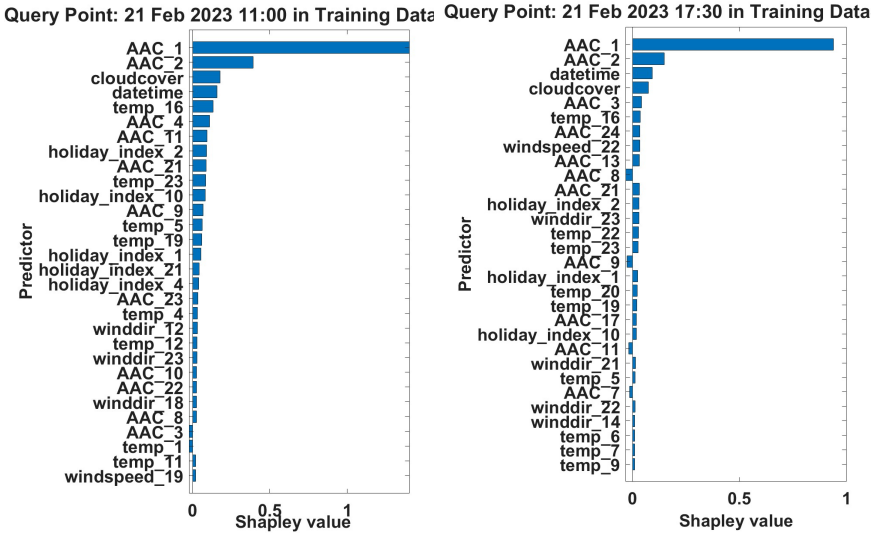


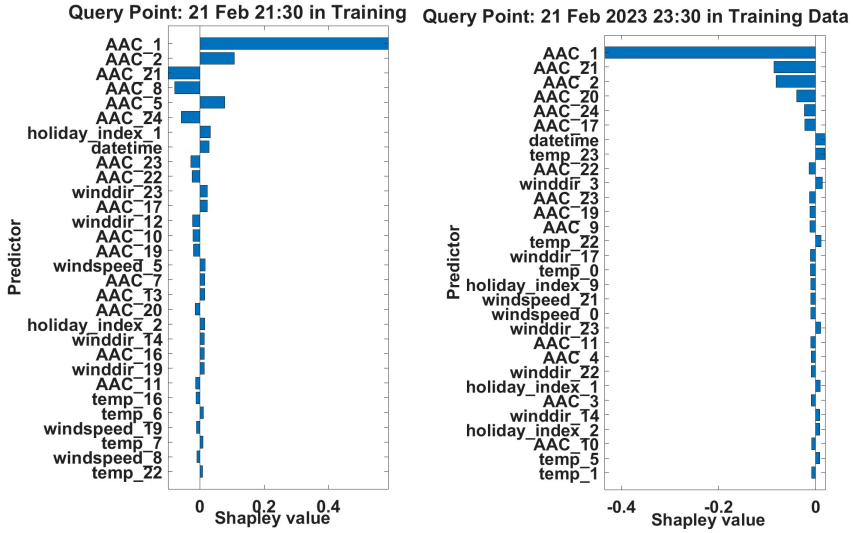
FIGURE 5.19: 30-minute ahead prediction for AAC_{EUR} using different Tree Ensamble NARX for a period validation dataset time interval **(Rome dataset)**.



(a) 21 Feb 2023, 4:30 (b) 21 Feb 2023, 8:30



(c) 21 Feb 2023, 11:00 (d) 21 Feb 2023, 17:30



(e) 21 Feb 2023, 21:30 (f) 21 Feb 2023, 23:30

FIGURE 5.20: Local SHAP-based prediction explanation for the NARX Tree Ensemble model. Explanation for different query points on February 21, belonging to the training/validation dataset: (a) 04:30, (b) 08:30, (c) 11:00, (d) 17:30, (e) 21:30, (f) 23:30 **(Rome dataset)**.

iv) Model Explanation

A more detailed analysis of model performance and prediction behavior is conducted through local SHAP-based explanations for a selected day (February 15, 2023) from the training and validation dataset at different time intervals. The first query point analyzed is at 4:30 AM, when the AAC is close to zero. The local explanation indicates that this low prediction is primarily driven by autoregressive components with negative Shapley values, while the previous day's temperature and datetime provide a slight positive contribution. This pattern likely reflects overnight parking trends in the EUR commercial district, with early departures influenced by residual temperature effects from the previous day.

The second query point, 8:30 AM, falls within the rising phase of the AAC curve, where drivers' parking decisions may be affected by meteorological and calendar-related factors. The SHAP analysis reveals that datetime, cloud cover, holiday index, and temperature, as well as their past values, positively contribute to AAC predictions, suggesting that these variables play a significant role in morning parking dynamics in the EUR zone.

The subsequent query points at 11:00 AM and 5:30 PM occur near AAC peaks and are predominantly influenced by the most recent autoregressive components (lags 1 and 2), combined with meteorological and holiday information. These factors indicate that both short-term historical patterns and external conditions strongly impact peak-hour parking availability.

The later query points at 9:30 PM and 11:30 PM show a stronger dependence on daytime effects and autoregressive components, likely reflecting habitual driver behaviors related to evening commutes and overnight parking. The holiday index has only a marginal influence, suggesting that special calendar events might slightly alter these recurring patterns.

5.7 Discussion

This chapter brought together three complementary perspectives on V2G availability prediction, moving from a data-driven grey state-space view (VED), through linear and nonlinear regression with transferability (Padua), to black-box modeling with explainability (Rome). Our intent was not only to compare accuracy, but to clarify when and why each modeling family adds value to an energy provider's decision workflow.

On the **VED dataset**, HDMDc consistently delivered strong multi-hour forecasts up to 4 h, with high correlations ($R \geq 0.86$) while preserving a compact linear state-space structure. This balance between parsimony and fidelity is central to day-ahead bidding and short-term balancing, where operators benefit from a single global model that is stable under quasi-periodic driving and explicitly parameterizes exogenous effects. In contrast, LSTM achieved reasonable 1–2 h performance but degraded quickly with horizon, reinforcing a practical trade-off: flexible nonlinear learners can fit short windows well, yet their long-horizon generalization can suffer without additional structure or constraints.

On the **Padua dataset**, we stressed input design and portability. Traffic emerged as the most informative exogenous signal: adding counts near candidate hubs tightened both one-step and three-step predictions for ARX and for nonlinear models. Among model classes, MLP and LSTM matched or exceeded ARX over three steps, while ARX retained lower variance and clearer interpretability. Direct transfer across hubs proved fragile—performance dropped when zone function and dwell-time patterns

diverged—but a light *fine-tuning* stage (one day of local data) largely restored accuracy. In practice, this suggests a scalable rollout path: train on a reference hub, cluster similar zones, then fine-tune per site with minimal data collection.

On the **Rome dataset**, we emphasized transparency. Linear ARX offered immediately interpretable coefficients, whereas NARX based on black-box models required post-hoc explanation. SHAP-based analyses aligned with intuition from the exploratory study: calendar/holiday intensity and a small set of meteorological drivers (e.g., temperature and wind signals at modest lags) contribute materially to AAC, but their influence is time- and context-dependent. These explanations are not merely diagnostic; they guide feature governance and help justify bids and curtailments to operations teams and regulators.

Drawing conclusions: (i) When horizons extend beyond 1–2 h and bid stability matters, grey state-space via HDMDc is a robust default, offering accuracy, order reduction, and clear exogenous pathways. (ii) Where rich local signals exist (traffic, site telemetry) and horizons are short, lightweight nonlinear models can lift accuracy—provided we manage overfitting and retain a path to explanation. (iii) For network-wide deployment, portability hinges on modest fine-tuning; naive zero-shot transfer across heterogeneous hubs is rarely sufficient. (iv) Explainability closes the loop between modeling and operations, translating feature effects into actionable levers for scheduling and market participation.

Figure 5.21 visualizes this taxonomy mapping across datasets and tasks, making the unifying framework actionable for future deployments and extensions.

As future work on the deployment side, we envisage hub clustering with automatic fine-tuning budgets, and routine XAI audits to track feature drift. Together, these steps advance a reliable, interpretable, and transferable stack for AAC forecasting in real V2G programs, and will be integrated into a multi-energy Energy Management System (EMS) for decision support. In particular, AAC forecasts will be able to feed stochastic/robust MPC and day-ahead–intraday co-optimization to jointly schedule electricity, heat, and gas assets—coordinating V2G charging/discharging with PV, battery/thermal storage, HVAC/thermal networks, and CHP—so as to minimize cost and emissions under uncertainty while respecting grid and comfort constraints. This multi-energy integration will also enable bilevel bidding for energy and ancillary services, returning actionable set-points and confidence bands to operators.

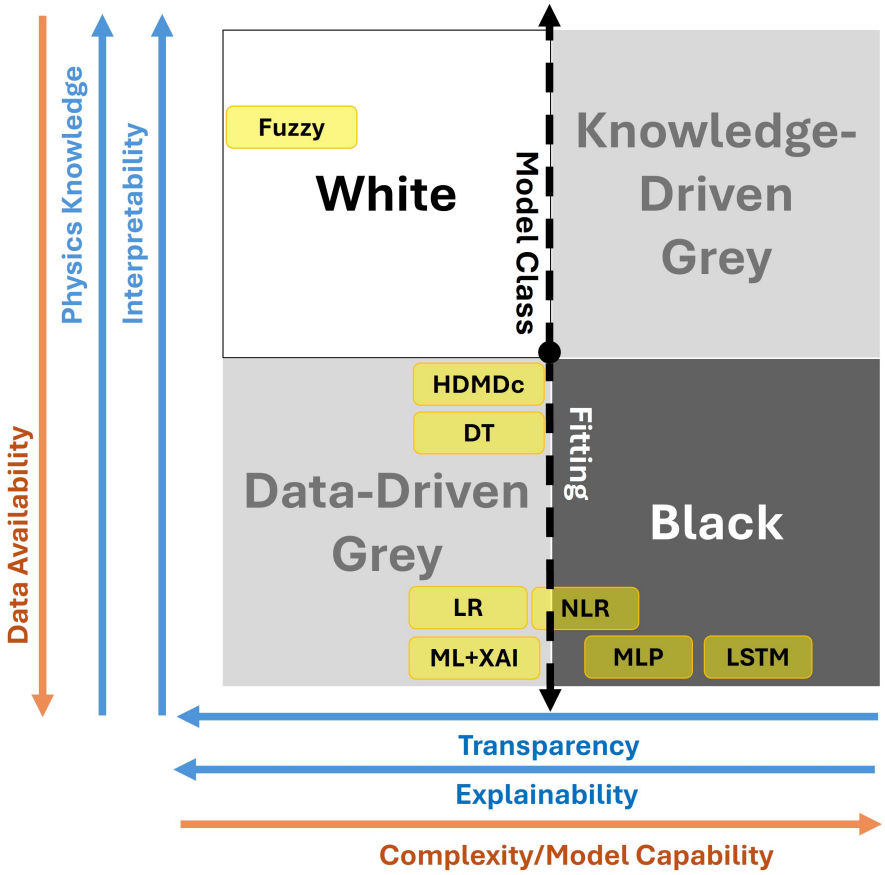


FIGURE 5.21: Taxonomy placement of presented models. ARX/FIR and HDMDc/MSA–HDMDc: **Data–driven Grey**; SR: boundary between **Data–driven Grey** and explainable (XAI) Black (depending on priors); neural encoders based on Multilayer Perceptron (MLP): **Black**; Fuzzy: **White**.

Chapter 6

Conclusions

This thesis addressed the challenge of developing interpretable modeling frameworks for dynamical systems across three application domains: (i) smart sensors based on green materials, (ii) soft sensors for industrial processes, and (iii) predictive and explainable models for Vehicle-to-Grid (V2G) applications. A unifying methodological objective was pursued, namely to balance modeling capability with transparency so that predictive power does not come at the expense of interpretability.

6.1 Summary of Contributions

The main contributions are as follows:

- *Smart sensors with green materials*: multiphysics Finite Element (FEM) modeling was integrated with system identification to describe mechanoelectric transduction in bacterial cellulose devices, linking material physics to system-level response.
- *Industrial soft sensors*: classical linear models (ARX, FIR) and their nonlinear counterparts (NARX, NFIR) were benchmarked against Koopman-based state-space formulations (HDMDc), showing how grey-box approaches can unify interpretability and predictive accuracy in the Sulfur Recovery Unit and Debutanizer Column case studies.
- *V2G forecasting*: data-driven methods, including LSTM and machine-learning pipelines, were coupled with explainable AI (XAI) tools such as SHAP to obtain accurate forecasts of Aggregated Available Capacity and actionable insights into exogenous drivers, for example weather and calendar features.

6.2 Positioning in the Modeling Taxonomy

A two-axis taxonomy was introduced to organize modeling approaches, with a vertical axis spanning interpretability and physics knowledge at the top to fitting-oriented data dependency at the bottom, and a horizontal axis ranging from transparency and explainability on the left to modeling capability and complexity on the right. Figure 6.1 places the implemented models within four quadrants: Quadrant 1 (White), Quadrant 2 (Knowledge-Driven Grey), Quadrant 3 (Black), and Quadrant 4 (Data-Driven Grey). The placements are:

First-principles ODE/PDE models - White. Governing ordinary or partial differential equations derived from conservation laws and constitutive relations reside in the extreme north–west of Quadrant 1. They rely purely on domain knowledge and

are intrinsically interpretable, with negligible dependence on data beyond nominal parameter specification.

Finite Element Implementation (FEM) - Knowledge-Driven Grey. The FEM implementation of the first-principles PDE model is placed in Quadrant 2, close to Quadrant 1. The physics-first structure and state variables remain transparent, yet credible predictions depend on expert modeling choices and empirical closures, for example boundary and initial conditions, constitutive parameters, mesh topology and resolution, element order, stabilization, and time integration. Targeted calibration and validation against data are typically required. When operated primarily as an inverse data-calibrated digital twin or augmented by data-driven surrogates, the placement drifts toward Quadrant 4.

Fuzzy and Adaptive Neurofuzzy Inference System (ANFIS) - White and Knowledge-Driven Grey. Rule-based fuzzy systems encode expert knowledge in linguistic rules and membership functions, hence Quadrant 1. When rule bases or memberships are learned or refined from data, as in neuro-fuzzy schemes, the placement moves toward Quadrant 2, maintaining explicit domain priors while fitting structure and parameters.

Linear regression models (LR) as ARX, FIR - Data-Driven Grey. ARX and FIR are data-driven linear regressors in Quadrant 4. Parameters are identified during fitting, but the mapping between regressors and predictions remains explicit and auditable. Although linear regression is often labeled white-box due to inherent interpretability, increasing data volume, feature dimensionality, or model order can diminish practical transparency. Classifying such high-dimensional linear models as black-box is misleading relative to deep neural architectures. The Data-Driven Grey designation, namely data-driven yet still transparent and under expert insights interpretable, is therefore most appropriate.

HDMDc state-space models - Data-Driven Grey bordering Knowledge-Driven Grey. Hankel DMD with control is identified entirely from data, which places it in Quadrant 4. Its state-space structure, modal decomposition, and input–output maps support analysis and engineering interpretation, locating it near the Quadrant 2 boundary. When model order becomes large, or when singular-value-based lifting and reduction are used, transparency can erode because observables are represented in latent coordinates with weakened physical semantics.

Decision trees (DT), Data-Driven Grey with a path to Knowledge-Driven Grey. Induced from data via impurity minimization, trees yield path-wise transparent piecewise rules, hence Quadrant 4. Depth and node proliferation reduce global comprehensibility. Physics-aware feature engineering, monotonicity or safety constraints, or rule templates shift the placement toward Quadrant 2. Ensembles increase capability but reduce transparency unless complemented by rule extraction or XAI.

Nonlinear Regression (NLR) as NARX and NFIR, Data-Driven Grey with increasing complexity. NARX and NFIR are learned from data and belong to Quadrant 4. Their architectures preserve inspectable regressors and mapping functions, supporting traceability between delayed inputs, delayed outputs, and predictions. Increasing

dimensionality, for example deeper networks or more lags, reduces practical transparency and can move the placement toward Quadrant 3. In long-horizon simulation, NFIR often retains stability due to the absence of output feedback, while NARX offers greater expressivity at the cost of stronger regularization and sensitivity analyses.

LSTM and deep recurrent networks, Black. Long Short-Term Memory and related deep recurrent architectures are positioned in Quadrant 3. They deliver high modeling capability for nonlinear dynamics and long-range dependencies, but large parameter spaces and latent representations limit intrinsic transparency, motivating auxiliary explanation tools.

Symbolic Regression (SR), Data-Driven Grey, with upward shift when physics priors are enforced. When the search space and constraints encode physics-informed priors, for example dimensional consistency, invariants, or conservation terms, Symbolic Regression (SR) moves upward toward Knowledge-Driven Grey; without such priors it remains Data-Driven Grey. In both regimes, explicit closed-form models support causal scrutiny, though unchecked symbolic complexity can hinder readability and maintenance.

Machine Learning (ML) pipelines with XAI, Data-Driven Grey depiction; base models remain Black. Nonlinear ML models and ensembles alone reside in Quadrant 3 (e.g. MLP). Post hoc explanation, for example SHAP and LIME, increase practical transparency; accordingly, the figure labels "ML+XAI" in Data-Driven Grey. Because explanations do not alter model complexity, validation against physical constraints remains essential.

Graded Placement The axes and placements proposed here are *graded* rather than absolute. Several indicators admit *quantitative* metrics, whereas others are inherently *qualitative* (e.g., design transparency, semantic alignment of states and parameters). Where quantitative surrogates for non-functional properties exist—such as sparsity or monotonicity constraints as indicators for interpretability, or rule comprehensibility scores for transparency—they remain *fuzzy* and context dependent. Consequently, models should be viewed as occupying *regions* (with uncertainty bands) rather than fixed points: design choices (architecture, priors, regularization), data practices (calibration breadth, operating-regime coverage), and tooling (physics constraints, XAI) can shift a model across adjacent quadrants over its lifecycle.

6.3 Final Remarks

By systematically positioning and applying models across a spectrum of interpretability, physics knowledge, data availability, and modeling capability, the thesis demonstrates that transparency and explainability need not be sacrificed for performance. The proposed taxonomy refines the conventional white, grey, and black categories in three ways. It provides a more precise distinction between baseline model classes while enabling fuzzy classification for approaches sharing properties across categories. It clarifies the effect of state-of-the-art enhancements, such as XAI and physics-informed ML, which shift black-box models toward more interpretable Data-Driven Grey or Knowledge-Driven Grey classes. It offers practical guidance, allowing researchers to select interventions that improve either capability or interpretability under resource constraints. Taken together, these contributions embed the

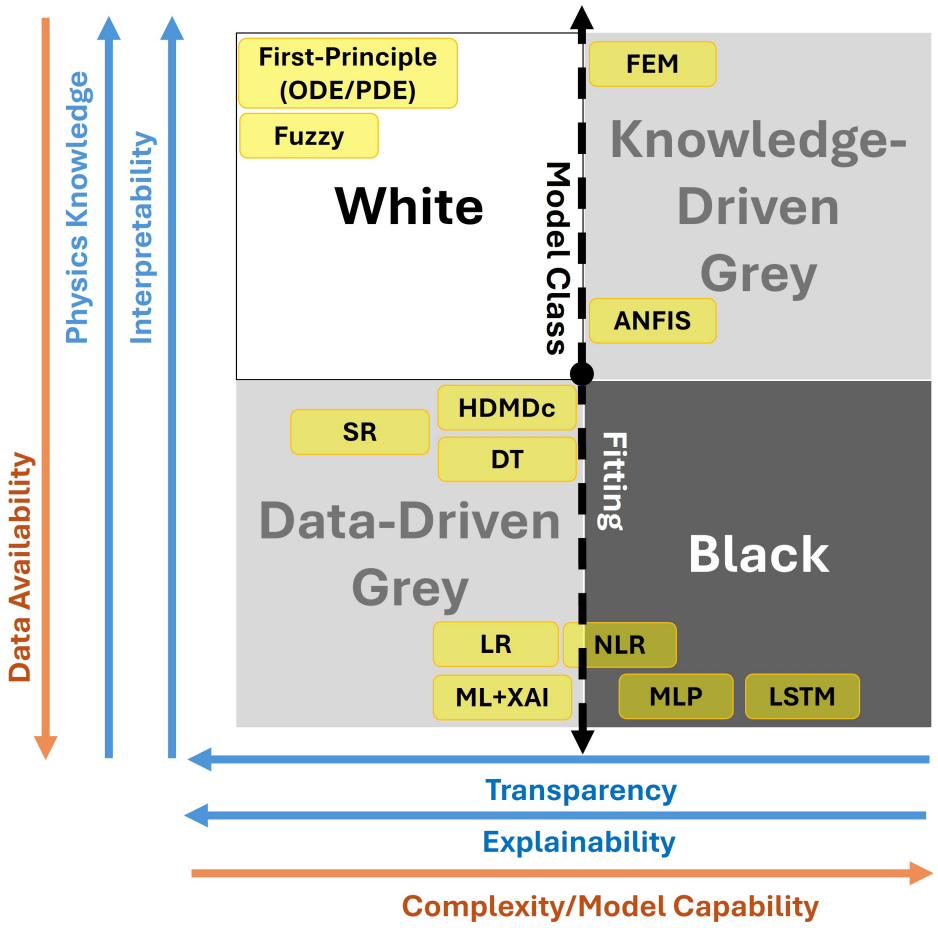


FIGURE 6.1: Placement of the implemented models within the dual-axis taxonomy.

white/grey/black paradigm within a richer framework grounded in transparency, interpretability, and explainability, and translate that framework into actionable practice across green sensing, industrial soft sensing, and V2G forecasting.

6.4 Perspectives

The results motivate several research directions focused on enhancing interpretability while preserving capability in both methodological and application-specific domains.

6.4.1 Methodological directions

Physics–data hybridization (FEM → ML and PINNs/PI-ML): extend the coupling of FEM with ML toward physically informed neural models, using FEM fields as priors/synthetic supervision to retain mechanistic transparency while reducing computational burden.

Quantitative metrics and graded taxonomy: the axes and placements proposed here are graded rather than absolute. Several indicators admit quantitative metrics, whereas others are inherently qualitative (e.g., design transparency, semantic alignment of states and parameters). Where quantitative surrogates for non-functional

properties exist—such as sparsity or monotonicity constraints as indicators for interpretability, or rule-comprehensibility scores for transparency—they remain fuzzy and context-dependent. Consequently, models should be viewed as occupying regions (with uncertainty bands) rather than fixed points: design choices (architecture, priors, regularization), data practices (calibration breadth, operating-regime coverage), and tooling (physics constraints, XAI) can shift a model across adjacent quadrants over its lifecycle.

Taxonomy as a prescriptive design compass: Beyond its current use as an *a posteriori* classification framework, the proposed taxonomy can evolve into a prescriptive tool that proactively guides model design. By integrating the requirements defined in Section 2.3 (e.g., explainability, computational constraints, domain knowledge availability, and transferability) with the evaluation metrics of Section 2.5, the taxonomy can serve as a decision-making scaffold for model developers and system owners. In practice, this would enable a structured workflow: (i) elicit application requirements collaboratively to identify a target region on the taxonomy map; (ii) pre-select model classes compatible with the desired interpretability–capability balance (for instance, Knowledge-Driven Grey suggesting PINNs or Grey-box models, Data-Driven Grey suggesting ARX, HDMDc, or constrained Symbolic Regression); and (iii) use interpretability and complexity metrics as feedback signals to keep the model within the intended quadrant during design. This prescriptive interpretation turns the taxonomy from a map of where models have been positioned into a compass for guiding future modeling choices—providing a shared language for negotiating trade-offs between accuracy, transparency, and computational effort before implementation.

Uncertainty and transferability toolchain: incorporate set-membership and distributional uncertainty quantification (UQ) and propagate these through Koopman/HDMDc/ARX/NARX pipelines; formalize domain-adaptation and transfer-learning protocols with interpretable intermediate representations, confidence bands, and drift tests.

Interpretable neuro-fuzzy + XAI pipelines: develop rule-based/neuro-fuzzy forecasters with global–local explanations (e.g., SHAP/LIME), exposing rule salience, monotone constraints, and counterfactual checks for auditability.

Transparent optimization for decision support: couple forecasts to linear/MILP or MPC optimizers that produce binding-constraint/marginal-value explanations; propagate input UQ to schedule/bid confidence intervals.

6.4.2 Application-specific directions

Materials and first-principles refinement (BC sensors): refine parameter identification with field-dependent properties in dual-carrier PDE models; systematically study alternative ionic liquids and biodegradable conductive polymers; widen the model spectrum by injecting PDE knowledge into ML surrogates.

Industrial soft sensors (SRU lines): deploy UQ-enabled soft sensors; study cross-line transferability and maintenance via interpretable state embeddings and set-membership bounds.

V2G forecasting and deployment (VED/Padua/Rome): characterize models per infrastructure class (hub topology, charger technology) and assess transferability across residential/commercial/industrial areas; explore neuro-fuzzy formulations to expose rule-level logic and validate with XAI (SHAP/LIME).

Multi-energy EMS integration: embed V2G as a controllable flexibility/storage asset co-optimized with PV, heat pumps, thermal storage, and CHP under network/market constraints; use hierarchical, interpretable EMS layers (HDMDc/ARX for forecasting; rule-based or neuro-fuzzy dispatch; transparent linear/MILP or MPC optimization with binding-constraint and marginal-value explanations); ensure interoperability and propagate set-membership uncertainty on AAC/prices to yield auditable confidence intervals on schedules and bids.

Bibliography

(N.d.[b]). Regulation (EEC) No 4064/89 Merger Procedure. URL: https://ec.europa.eu/competition/mergers/cases/decisions/m1572_en.pdf.

Afentoulis, Konstantinos D. et al. (2022). "Smart charging business model framework for electric vehicle aggregators". In: *Applied Energy* 328, p. 120179. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2022.120179.

Agbaoye, R.O. et al. (2017). "Elastic constants and mechanical properties of PEDOT from first principles calculations". In: *Computational Materials Science* 139, pp. 234–242. ISSN: 0927-0256. DOI: 10.1016/j.commatsci.2017.07.042.

Agency, International Energy (2021). "Net Zero by 2050. A Roadmap for the Global Energy Sector." In: *IEA, Paris*. DOI: https://www.iea.org/reports/net-zero-by-2050.

— (2022). "By 2030 EVs represent more than 60% of vehicles sold globally, and require an adequate surge in chargers installed in buildings". In: *IEA, Paris*. DOI: https://www.iea.org/reports/by-2030-evs-represent-more-than-60-of-vehicles-sold-globally-and-require-an-adequate-surge-in-chargers-installed-in-buildings.

Aiosa, Grazia V., Maurizio Palesi, and Francesca Sapuppo (2023). "EXplainable AI for Decision Support to Obesity Comorbidities Diagnosis". In: *IEEE Access* 11, pp. 107767–107782. DOI: 10.1109/ACCESS.2023.3320057.

Akaike, Hirotugu (1974). "A new look at the statistical model identification". In: *IEEE Transactions on Automatic Control* 19.6, pp. 716–723. DOI: 10.1109/TAC.1974.1100705.

Aldeia, Guilherme and Fabricio De França (May 2022). "Interpretability in symbolic regression: a benchmark of explanatory methods using the Feynman data set". In: *Genetic Programming and Evolvable Machines* 23. DOI: 10.1007/s10710-022-09435-x.

Alfatlawi, Mustaffa and Vaibhav Srivastava (Aug. 2019). "An Incremental Approach to Online Dynamic Mode Decomposition for Time-Varying Systems with Applications to EEG Data Modeling". In: *Journal of Computational Dynamics*.

Alizadeh, Reza, Janet K. Allen, and Farrokh Mistree (Apr. 2020). "Managing computational complexity using surrogate models: a critical review". In: *Research in Engineering Design* 31.3, 275–298. ISSN: 1435-6066. DOI: 10.1007/s00163-020-00336-7. URL: http://dx.doi.org/10.1007/s00163-020-00336-7.

Amara-Ouali, Yvenn et al. (2022). "A benchmark of electric vehicle load and occupancy models for day-ahead forecasting on open charging session data". In: *Proceedings of the Thirteenth ACM International Conference on Future Energy Systems*. e-Energy '22. Virtual Event: Association for Computing Machinery, 193–207. ISBN: 9781450393973. DOI: 10.1145/3538637.3538850. URL: https://doi.org/10.1145/3538637.3538850.

Andò, Bruno, Salvatore Graziani, and Maria Gabriella Xibilia (2019). "Low-order Nonlinear Finite-Impulse Response Soft Sensors for Ionic Electroactive Actuators Based on Deep Learning". In: *IEEE Transactions on Instrumentation and Measurement* 68.5, pp. 1637–1646. DOI: 10.1109/TIM.2018.2884450.

Angelopoulos, Anastasios N. and Stephen Bates (2023). "A gentle introduction to conformal prediction and distribution-free uncertainty quantification". In: *Proceedings of the National Academy of Sciences* 120.8, e2204233119. DOI: doi.org/10.48550/arXiv.2107.07511.

Ann Arbour School Breaks (n.d.). https://annarborwithkids.com/articles/winter-spring-break-schedules-2022/.

Arbabi, Hassan and Igor Mezić (Nov. 2016). "Ergodic theory, Dynamic Mode Decomposition and Computation of Spectral Properties of the Koopman operator". In: *SIAM Journal on Applied Dynamical Systems*. DOI: 10.1137/17M1125236.

Auenhammer, Robert M. et al. (Mar. 2024). "X-ray scattering tensor tomography based finite element modelling of heterogeneous materials". In: *npj Computational Materials* 10.1. ISSN: 2057-3960. DOI: 10.1038/s41524-024-01234-5. URL: http://dx.doi.org/10.1038/s41524-024-01234-5.

Babaei Pourkargar, Davood and Antonios Armaou (2015). "Control of spatially distributed processes with unknown transport-reaction parameters via two layer system adaptations". In: *AIChE Journal* 61.8, pp. 2497–2507.

Bakhshi, Vahid et al. (Mar. 2022). "Evaluating Rainy Weather Effects on Driving Behaviour Dimensions of Driving Behaviour Questionnaire". In: *Journal of Advanced Transportation* 2022, pp. 1–10. ISSN: 2042-3195. DOI: 10.1155/2022/6000715.

Barbero, Mattia et al. (2020). "Critical evaluation of European balancing markets to enable the participation of Demand Aggregators". In: *Applied Energy* 264, p. 114707. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2020.114707.

Bartlett, Peter L and Shahar Mendelson (2003). "Rademacher and Gaussian complexities: Risk bounds and structural results". In: *Machine Learning* 52, pp. 95–113. DOI: 10.1007/3-540-44581-1_15.

Bidar, Bahareh et al. (June 2018). "Soft Sensor Modeling Based on Multi-State-Dependent Parameter Models and Application for Quality Monitoring in Industrial Sulfur Recovery Process". In: *IEEE Sensors Journal* 18 (11), pp. 4583–4591. ISSN: 1530-437X. DOI: 10.1109/JSEN.2018.2818886.

Billings, Stephen A. (2013). *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*. Wiley. ISBN: 978-0470746664.

Bohlin, Torsten, ed. (2006). *Practical Grey-Box Process Identification: Theory and Applications*. Advances in Industrial Control. London: Springer London. DOI: 10.1007/1-84628-403-1.

Boos, Eugen et al. (2023). "Improving a Deep Learning Temperature-Forecasting Model of a 3-Axis Precision Machine with Domain Randomized Thermal Simulation Data". In: *Production at the Leading Edge of Technology*. Lecture Notes in Production Engineering. Springer International Publishing, pp. 574–584. DOI: 10.1007/978-3-031-18318-8_58.

Brier, Glenn W. (1950). "Verification of forecasts expressed in terms of probability". In: *Monthly Weather Review* 78.1, pp. 1–3. DOI: doi:10.1175/1520-0493(1950)078%3C0001:VOFEIT%3E2.0.CO;2.

Bruder, Daniel et al. (Feb. 2019). "Modeling and Control of Soft Robots Using the Koopman Operator and Model Predictive Control". In: *Robotics: Science and Systems*.

Brunton, Steven L. and J. Nathan Kutz (May 2022). *Data-Driven Science and Engineering*. Cambridge University Press. ISBN: 9781009089517. DOI: 10.1017/9781009089517.

Brunton, Steven L., Joshua L. Proctor, and J. Nathan Kutz (Apr. 2016). "Discovering governing equations from data by sparse identification of nonlinear dynamical

systems". In: *Proceedings of the National Academy of Sciences* 113 (15), pp. 3932–3937. ISSN: 0027-8424. DOI: 10.1073/pnas.1517384113.

Brunton, Steven L. et al. (Oct. 2015). "Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control". In: *PLoS ONE* 11. DOI: 10.1371/journal.pone.0150171. URL: http://arxiv.org/abs/1510.03007http://dx.doi.org/10.1371/journal.pone.0150171.

Brunton, Steven L. et al. (May 2017). "Chaos as an intermittently forced linear system". In: *Nature Communications* 8 (1), p. 19. ISSN: 2041-1723. DOI: 10.1038/s41467-017-00030-8.

Brunton, Steven L et al. (2022). "Modern Koopman Theory for Dynamical Systems". In: *SIAM Review* 64 (2), pp. 229–340. DOI: 10.1137/21M1401243. URL: https://doi.org/10.1137/21M1401243.

Böttcher, Maria et al. (2021a). "Efficient Utilization of Surrogate Models for Uncertainty Quantification". In: *PAMM* 20.1, e202000210. DOI: 10.1002/pamm.202000210.

Böttcher, Maria et al. (2021b). "ELSA: An Efficient, Adaptive Ensemble Learning-Based Sampling Approach". In: *Advances in Engineering Software* 154, p. 102974. DOI: 10.1016/j.advengsoft.2021.102974.

Calapristi, Marco et al. (2024a). "Interpretability analysis of Symbolic Regression models for dynamical systems". In: *2024 International Conference on Control, Automation and Diagnosis (ICCAD)*, pp. 1–6. DOI: 10.1109/ICCAD60883.2024.10553801.

Calapristi, Marco et al. (2024b). "Symbolic Regression for Industrial Applications: An NN-Based Approach". In: *2024 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*, pp. 618–623. DOI: 10.1109/MetroXRAINE62247.2024.10797142.

Caponetto, R. et al. (2013a). "An Enhanced Fractional Order Model of Ionic Polymer-Metal Composites Actuator". In: *Advances in Mathematical Physics* 2013, pp. 1–6. DOI: 10.1155/2013/717659.

Caponetto, R et al. (2013b). "An optimized frequency-dependent multiphysics model for an ionic polymer metal composite actuator with ethylene glycol as the solvent". In: 22.12, p. 125016. DOI: 10.1088/0964-1726/22/12/125016.

Caponetto, R. et al. (2014a). "IPMC frequency dependent multiphysics model considering electrodes high surface and fractional effects". In: *2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pp. 1529–1532. DOI: 10.1109/I2MTC.2014.6861001.

Caponetto, R. et al. (2023a). "Modeling of bacterial cellulose-based composite". In: *Electronics* 12.21, p. 4530. DOI: 10.3390/electronics12214530.

Caponetto, Riccardo et al. (2014b). "A Multiphysics Frequency-Dependent Model of an IP^2C Actuator". In: *IEEE Transactions on Instrumentation and Measurement* 63.5, pp. 1347–1355. DOI: 10.1109/TIM.2014.2298172.

Caponetto, Riccardo et al. (2014c). "Identification of IPMC nonlinear model via single and multi-objective optimization algorithms". In: *ISA Transactions* 53.2, pp. 481–488. ISSN: 0019-0578. DOI: doi.org/10.1016/j.isatra.2013.11.012.

Caponetto, Riccardo et al. (2023b). "Modeling of Bacterial Cellulose-Based Composite". In: *Electronics* 12.21. ISSN: 2079-9292. DOI: 10.3390/electronics12214530.

Chen, Xiaolong et al. (Dec. 2019). "Ensemble regularized local finite impulse response models and soft sensor application in nonlinear dynamic industrial processes". In: *Applied Soft Computing* 85, p. 105806. ISSN: 15684946. DOI: 10.1016/j.asoc.2019.105806.

Cheng, Zhihan et al. (2025). "A Comprehensive Review of Explainable Artificial Intelligence (XAI) in Computer Vision". In: *Sensors* 25.13. ISSN: 1424-8220. DOI: 10.3390/s25134166. URL: https://www.mdpi.com/1424-8220/25/13/4166.

Chicco, Davide, Matthijs J. Warrens, and Giuseppe Jurman (2021). "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation". In: *PeerJ Computer Science* 7, 1 – 24. DOI: 10.7717/PEERJ-CS.623.

Clainche, Soledad Le (Apr. 2019). "Prediction of the Optimal Vortex in Synthetic Jets". In: *Energies* 12 (9), p. 1635. ISSN: 1996-1073. DOI: 10.3390/en12091635.

Clainche, Soledad Le, Zhong-Hua Han, and Esteban Ferrer (Sept. 2019). "An alternative method to study cross-flow instabilities based on high order dynamic mode decomposition". In: *Physics of Fluids* 31 (9), p. 094101. ISSN: 1070-6631. DOI: 10.1063/1.5110697.

Clainche, Soledad Le and José M. Vega (Jan. 2017). "Higher Order Dynamic Mode Decomposition". In: *SIAM Journal on Applied Dynamical Systems* 16 (2), pp. 882–925. ISSN: 1536-0040. DOI: 10.1137/15M1054924. URL: https://epubs.siam.org/doi/10.1137/15M1054924.

Cohn, Ryan and Elizabeth Holm (June 2021). "Unsupervised Machine Learning Via Transfer Learning and k-Means Clustering to Classify Materials Image Data". In: *Integrating Materials and Manufacturing Innovation* 10 (2), pp. 231–244. ISSN: 2193-9764. DOI: 10.1007/s40192-021-00205-8.

Comi, Antonio et al. (2021). "Private Car O-D Flow Estimation Based on Automated Vehicle Monitoring Data: Theoretical Issues and Empirical Evidence". In: *Information* 12.12. ISSN: 2078-2489. DOI: 10.3390/info12120493.

Cranmer, Miles (2023). *Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl*. arXiv: 2305.01582 [astro-ph.IM].

Crispin, X. et al. (2006). "The Origin of the High Conductivity of Poly(3,4-ethylenedioxythiophene)-Poly(styrenesulfonate) (PEDOT-PSS) Plastic Electrodes". In: *Chemistry of Materials* 18.18, pp. 4354–4360. DOI: 10.1021/cm061032+.

Cui, Huanqing et al. (Aug. 2020). "Intelligent Polymer-Based Bioinspired Actuators: From Monofunction to Multifunction". In: *Advanced Intelligent Systems* 2.11. ISSN: 2640-4567. DOI: 10.1002/aisy.202000138. URL: http://dx.doi.org/10.1002/aisy.202000138.

Curreri, Francesco, Luca Patanè, and Maria Gabriella Xibilia (2021). "RNN- and LSTM-Based Soft Sensors Transferability for an Industrial Process". In: *Sensors* 21.3. ISSN: 1424-8220. DOI: 10.3390/s21030823.

Das, Suddhasattwa et al. (2023). "Koopman-Theoretic Modeling of Quasiperiodically Driven Systems: Example of Signalized Traffic Corridor". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 53.7, pp. 4466–4476. DOI: 10.1109/TSMC.2023.3253077.

Dayhoff, Judith E. and James M. DeLeo (2001). "Artificial Neural Networks". In: *Cancer* 91.S8, pp. 1615–1635. DOI: 10.1002/1097-0142(20010415)91:8+%3C1615::AID-CNCR1175%3E3.0.CO;2-L.

Deb, Sanchari et al. (2018). "Impact of Electric Vehicle Charging Station Load on Distribution Network". In: *Energies* 11.1. ISSN: 1996-1073. DOI: 10.3390/en11010178.

Dias, Tiago et al. (May 2022). "Linear and Non-Linear Soft Sensors for Predicting the Research Octane Number (RON) through Integrated Synchronization, Resolution Selection and Modelling". In: *Sensors* 22 (10), p. 3734. ISSN: 1424-8220. DOI: 10.3390/s22103734.

Ding, Bing et al. (2024). "A CCP-based distributed cooperative operation strategy for multi-agent energy systems integrated with wind, solar, and buildings". In:

Applied Energy 365, p. 123275. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2024.123275. URL: https://www.sciencedirect.com/science/article/pii/S0306261924006585.

Dinov, Ivo D. (2018). "Black Box Machine-Learning Methods: Neural Networks and Support Vector Machines". In: *Data Science and Predictive Analytics: Biomedical and Health Applications using R*. Springer International Publishing, pp. 383–422. DOI: 10.1007/978-3-319-72347-1_11.

Dixon, James et al. (2022). "Vehicle to grid: driver plug-in patterns, their impact on the cost and carbon of charging, and implications for system flexibility". In: *eTransportation* 13, p. 100180. ISSN: 2590-1168. DOI: https://doi.org/10.1016/j.etran.2022.100180.

El-Atab, Nazek et al. (2020). "Soft Actuators for Soft Robotic Applications: A Review". In: *Advanced Intelligent Systems* 2.10, p. 2000128. DOI: https://doi.org/10.1002/aisy.202000128. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202000128. URL: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000128.

Esa, Faezah, Siti Tasirin, and Norliza Abd.Rahman (Dec. 2014). "Overview of Bacterial Cellulose Production and Application". In: *Agriculture and Agricultural Science Procedia* 2, pp. 113–119. DOI: 10.1016/j.aaspro.2014.11.017.

Fachrizal, Reza et al. (2024). "Urban-scale energy matching optimization with smart EV charging and V2G in a net-zero energy city powered by wind and solar energy". In: *eTransportation* 20, p. 100314. ISSN: 2590-1168. DOI: https://doi.org/10.1016/j.etran.2024.100314.

Fasel, Urban et al. (2021). *SINDy with Control: A Tutorial*. arXiv: 2108.13404 [math.OC].

Fortuna, L., S. Graziani, and M.G. Xibilia (2005). "Soft sensors for product quality monitoring in debutanizer distillation columns". In: *Control Engineering Practice* 13.4, pp. 499–508. ISSN: 0967-0661. DOI: https://doi.org/10.1016/j.conengprac.2004.04.013. URL: https://www.sciencedirect.com/science/article/pii/S0967066104000899.

Fortuna, L. et al. (2003). "Soft analyzers for a sulfur recovery unit". In: *Control Engineering Practice* 11.12. Award winning applications-2002 IFAC World Congress, pp. 1491 –1500. ISSN: 0967-0661. DOI: https://doi.org/10.1016/S0967-0661(03)00079-0. URL: http://www.sciencedirect.com/science/article/pii/S0967066103000790.

Fortuna, Luigi et al. (Jan. 2007). *Soft Sensors for Monitoring and Control of Industrial Processes*. Springer-Verlag London. ISBN: 978-1-84628-479-3. DOI: 10.1007/978-1-84628-480-9.

Fujimoto, Richard et al., eds. (2017). *Research Challenges in Modeling and Simulation for Engineering Complex Systems*. Cham: Springer International Publishing.

G. Montalban, Mercedes et al. (May 2015). "Effect of Temperature, Anion, and Alkyl Chain Length on the Density and Refractive Index of 1-Alkyl-3-methylimidazolium-Based Ionic Liquids". In: *Journal of Chemical And Engineering Data* 60. DOI: 10.1021/je501091q.

Gao, Hanyao et al. (Sept. 2024). "Machine learning in business and finance: a literature review and research opportunities". en. In: *Financ. Innov.* 10.1.

Gautam, Akash, Arun Kumar Verma, and Manaswi Srivastava (2019). "A Novel Algorithm for Scheduling of Electric Vehicle Using Adaptive Load Forecasting with Vehicle-to-Grid Integration". In: *2019 8th International Conference on Power Systems (ICPS)*, pp. 1–6. DOI: 10.1109/ICPS48983.2019.9067702.

Gedon, Daniel et al. (2021). "Deep State Space Models for Nonlinear System Identification". In: *IFAC-PapersOnLine SYSID 2021* 54.7, pp. 481–486. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2021.08.406.

Gim, Tae-Hyoung Tommy (Mar. 2018). "SEM application to the household travel survey on weekends versus weekdays: the case of Seoul, South Korea". In: *European Transport Research Review* 10 (1), p. 11. ISSN: 1867-0717. DOI: 10.1007/s12544-018-0285-9.

Graham, Jemima and Fei Teng (2023). "Vehicle-to-grid plug-in forecasting for participation in ancillary services markets". In: *2023 IEEE Belgrade PowerTech*. IEEE. DOI: 10.1109/powertech55446.2023.10202778.

Grasel, Bernhard, Josè Baptista, and Manfred Tragner (2024). "The Impact of V2G Charging Stations (Active Power Electronics) to the Higher Frequency Grid Impedance". In: *Sustainable Energy, Grids and Networks* 38. DOI: https://doi.org/10.1016/j.segan.2024.101306.

Graziani, S. et al. (2013). "Multi-input identification of IP2C actuators". In: *2013 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1147–1151. DOI: 10.1109/I2MTC.2013.6555593.

Guckenheimer, John and Philip Holmes (1983). *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Vol. 42. Springer New York. ISBN: 978-1-4612-7020-1. DOI: 10.1007/978-1-4612-1140-2.

Guidotti, Riccardo et al. (2018). "A Survey of Methods for Explaining Black Box Models". In: *ACM Computing Surveys* 51.5, 93:1–93:42. DOI: 10.1145/3236009.

Hashemi, Ali, Javad Beheshti, and Mahdieh Mohammadi (2025). "Physics-Based AI-Driven Surrogate Modeling for Structural Displacement Prediction in Mechanical Systems With Limited Sensor Data". In: *IEEE Access* 13, pp. 130585–130602. DOI: 10.1109/ACCESS.2025.3590664.

Hayamizu, Kikuko et al. (Dec. 2004). "Ionic Conduction and Ion Diffusion in Binary Room-Temperature Ionic Liquids Composed of [emim][BF4] and LiBF4". In: *J.Phys.Chem.B* 108. DOI: 10.1021/jp0476601.

He, Q. et al. (2019a). "Mechanoelectric transduction of ionic polymer-graphene composite sensor with ionic liquid as electrolyte". In: *Sensors and Actuators A: Physical* 286, pp. 68–77.

He, Qingsong et al. (2019b). "Mechanoelectric transduction of ionic polymer-graphene composite sensor with ionic liquid as electrolyte". In: *Sensors and Actuators A: Physical* 286, pp. 68–77.

He, Qingsong et al. (2019c). "Mechanoelectric transduction of ionic polymer-graphene composite sensor with ionic liquid as electrolyte". In: *Sensors and Actuators A: Physical* 286, pp. 68–77. ISSN: 0924-4247. DOI: https://doi.org/10.1016/j.sna.2018.12.014.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Hofmann, Thomas, Bernhard Schölkopf, and Alexander J. Smola (2008). "Kernel Methods in Machine Learning". In: *The Annals of Statistics* 36.3, pp. 1171–1220. DOI: 10.1214/009053607000000677.

Hohman, Fred et al. (2019). "Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers". In: *IEEE Transactions on Visualization and Computer Graphics* 25.8, pp. 2674–2693. DOI: 10.1109/TVCG.2018.2843369.

Huang, J. et al. (2022). "Recent advances in functional bacterial cellulose for wearable physical sensing applications". In: *Advanced Materials Technologies* 7.1, p. 2100617. DOI: 10.1002/admt.202100617.

Jahromi, Saeed Naghdizadegan et al. (2024). "A comprehensive framework for predicting electric vehicle's participation in ancillary service markets". In: *IET Smart Grid*.

Jiang, Jingang et al. (Dec. 2023). "Application-Oriented Modeling of Soft Actuator Ionic Polymer–Metal Composites: A Review". In: *Advanced Intelligent Systems* 6.3. ISSN: 2640-4567. DOI: 10.1002/aisy.202300568. URL: http://dx.doi.org/10.1002/aisy.202300568.

Jones, C.N.S. and S.V. Utyuzhnikov (June 2022). "Application of higher order dynamic mode decomposition to modal analysis and prediction of power systems with renewable sources of energy". In: *International Journal of Electrical Power & Energy Systems* 138, p. 107925. ISSN: 01420615. DOI: 10.1016/j.ijepes.2021.107925.

Kadlec, Petr, Bogdan Gabrys, and Sibylle Strandt (2009). "Data-driven Soft Sensors in the process industry". In: *Computers & Chemical Engineering* 33.4, pp. 795 – 814. ISSN: 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2008.12.012. URL: http://www.sciencedirect.com/science/article/pii/S0098135409000076.

Kamble, Sachin Gorakh, Kinhal Vadirajacharya, and Udaykumar Vasudeo Patil (2019). "Decision Making in Power Distribution System Reconfiguration by Blended Biased and Unbiased Weightage Method". In: *Journal of Sensor and Actuator Networks* 8.2. ISSN: 2224-2708. DOI: 10.3390/jsan8020020. URL: https://www.mdpi.com/2224-2708/8/2/20.

Kanjilal, Baishali et al. (Feb. 2023). "Bioionic Liquids: Enabling a Paradigm Shift Toward Advanced and Smart Biomedical Applications". In: *Advanced Intelligent Systems* 5.5. ISSN: 2640-4567. DOI: 10.1002/aisy.202200306. URL: http://dx.doi.org/10.1002/aisy.202200306.

Kaptanoglu, A. A. et al. (Mar. 2020). "Characterizing magnetized plasmas with dynamic mode decomposition". In: *Physics of Plasmas* 27 (3), p. 032108. ISSN: 1070-664X. DOI: 10.1063/1.5138932.

Karmaker, Ashish Kumar et al. (2024). "Electric vehicle hosting capacity analysis: Challenges and solutions". In: *Renewable and Sustainable Energy Reviews* 189, p. 113916. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2023.113916.

Karniadakis, George Em et al. (2021). "Physics-informed machine learning". In: *Nature Reviews Physics* 3.6, pp. 422–440. DOI: 10.1038/s42254-021-00314-5.

Kommenda, Michael et al. (Sept. 2020). "Parameter identification for symbolic regression using nonlinear least squares". In: *Genetic Programming and Evolvable Machines* 21. DOI: 10.1007/s10710-019-09371-3.

Koopman, B. O. (May 1931). "Hamiltonian Systems and Transformation in Hilbert Space". In: *Proceedings of the National Academy of Sciences* 17 (5), pp. 315–318. ISSN: 0027-8424. DOI: 10.1073/pnas.17.5.315.

König, Gunnar et al. (2021). "Relative Feature Importance". In: *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 9318–9325. DOI: 10.1109/ICPR48806.2021.9413090.

Levenberg, Kenneth (1944). "A method for the solution of certain non-linear problems in least squares". en. In: *Quart. Appl. Math.* 2.2, pp. 164–168.

Li, Mince et al. (2024). "Toward Efficient Smart Management: A Review of Modeling and Optimization Approaches in Electric Vehicle-Transportation Network-Grid Integration". In: *Green Energy and Intelligent Transportation*, p. 100181. ISSN: 2773-1537. DOI: https://doi.org/10.1016/j.geits.2024.100181. URL: https://www.sciencedirect.com/science/article/pii/S2773153724000331.

Li, Shuangqi et al. (2021a). "Boosting Grid Efficiency and Resiliency by Releasing V2G Potentiality Through a Novel Rolling Prediction-Decision Framework and

Deep-LSTM Algorithm". In: *IEEE Systems Journal* 15.2, pp. 2562–2570. DOI: 10.1109/JSYST.2020.3001630.

Li, Weichen and Xiaojia Shelly Zhang (June 2024). "Computational morphogenesis for liquid crystal elastomer metamaterial". In: *npj Computational Materials* 10.1. ISSN: 2057-3960. DOI: 10.1038/s41524-024-01300-y. URL: http://dx.doi.org/10.1038/s41524-024-01300-y.

Li, Xudong et al. (2021b). "Central moment discrepancy based domain adaptation for intelligent bearing fault diagnosis". In: *Neurocomputing* 429, pp. 12–24.

Li, Yanfei et al. (2021c). "Grey-Box Modeling and Application for Building Energy Simulations – A Critical Review". In: *Renewable and Sustainable Energy Reviews* 146, p. 111174. DOI: 10.1016/j.rser.2021.111174.

Lillebo, Martin et al. (July 2019). "Impact of large-scale EV integration and fast chargers in a Norwegian LV grid". In: *The Journal of Engineering* 2019 (18), pp. 5104–5108. ISSN: 2051-3305. DOI: 10.1049/joe.2018.9318.

Lin, Sheng et al. (2024). "Dynamic Low-Rank and Sparse Priors Constrained Deep Autoencoders for Hyperspectral Anomaly Detection". In: *IEEE Transactions on Instrumentation and Measurement* 73, pp. 1–18. DOI: 10.1109/TIM.2023.3323997.

Lindholm, Andreas et al. (2022). *Machine Learning: A First Course for Engineers and Scientists*. Cambridge University Press.

Liu, Jiahang et al. (2023). "Receding Horizon Actor–Critic Learning Control for Nonlinear Time-Delay Systems With Unknown Dynamics". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 53.8, pp. 4980–4993. DOI: 10.1109/TSMC.2023.3254911.

Liu, Jinping et al. (2022). "Frame-Dilated Convolutional Fusion Network and GRU-Based Self-Attention Dual-Channel Network for Soft-Sensor Modeling of Industrial Process Quality Indexes". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52.9, pp. 5989–6002. DOI: 10.1109/TSMC.2021.3130232.

Ljung, Lennart (1986). *System Identification: Theory for the User*. USA: Prentice-Hall, Inc. ISBN: 0138816409.

— (1999). *System Identification: Theory for the User*. 2th. Upper Saddle River, NJ: Prentice-Hall PTR.

Lopez-Sanchez, P. et al. (2014). "Micromechanics and Poroelasticity of Hydrated Cellulose Networks". In: *Biomacromolecules* 15.6. PMID: 24784575, pp. 2274–2284. DOI: 10.1021/bm500405h.

Loyola-González, Octavio (2019). "Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View". In: *IEEE Access* 7, pp. 154096–154113. DOI: 10.1109/ACCESS.2019.2949286.

Lu, Lu (June 2020). "Dying ReLU and Initialization: Theory and Numerical Examples". In: *Communications in Computational Physics* 28.5, 1671–1706. ISSN: 1991-7120. DOI: 10.4208/cicp.oa-2020-0165. URL: http://dx.doi.org/10.4208/cicp.OA-2020-0165.

Lundberg, Scott and Su-In Lee (2017a). *A Unified Approach to Interpreting Model Predictions*. arXiv: 1705.07874 [cs.AI].

Lundberg, Scott M. and Su-In Lee (2017b). "A Unified Approach to Interpreting Model Predictions". In: 4765–4774.

Mackay, Calum and David Nowell (Apr. 2023). "Informed machine learning methods for application in engineering: A review". In: *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 237, p. 095440622311645. DOI: 10.1177/09544062231164575.

Mao, Meiqin et al. (2019). "Schedulable capacity forecasting for electric vehicles based on big data analysis". In: *Journal of Modern Power Systems and Clean Energy* 7.6, pp. 1651–1662. DOI: 10.1007/s40565-019-00573-3.

Marcinkevičs, Ričards and Julia E. Vogt (2023). *Interpretability and Explainability: A Machine Learning Zoo Mini-tour*. arXiv: 2012.01805 [cs.LG].

Meteo Stat Phyton API (n.d.). https://dev.meteostat.net/python/.

Mezić, Igor (Aug. 2005). "Spectral Properties of Dynamical Systems, Model Reduction and Decompositions". In: *Nonlinear Dynamics* 41 (1-3), pp. 309–325. ISSN: 0924-090X. DOI: 10.1007/s11071-005-2824-x.

— (Jan. 2013). "Analysis of Fluid Flows via Spectral Properties of the Koopman Operator". In: *Annual Review of Fluid Mechanics* 45 (1). doi: 10.1146/annurev-fluid-011212-140652, pp. 357–378. ISSN: 0066-4189. DOI: 10.1146/annurev-fluid-011212-140652. URL: https://doi.org/10.1146/annurev-fluid-011212-140652.

Michigan ORS Non-Business Days (n.d.). https://www.michigan.gov/psru/ors-non-business-days.

Mittal, Saurabh and Andreas Tolk (2019). *Complexity Challenges in Cyber Physical Systems: Using Modeling and Simulation (M&S) to Support Intelligence, Adaptation and Autonomy*. Springer.

Moreno-Mateos, Miguel Angel and Paul Steinmann (Dec. 2024). "Crosslinking degree variations enable programming and controlling soft fracture via sideways cracking". In: *npj Computational Materials* 10.1. ISSN: 2057-3960. DOI: 10.1038/s41524-024-01489-y. URL: http://dx.doi.org/10.1038/s41524-024-01489-y.

Moreno-Mateos, Miguel Angel et al. (July 2022). "Hybrid magnetorheological elastomers enable versatile soft actuators". In: *npj Computational Materials* 8.1. ISSN: 2057-3960. DOI: 10.1038/s41524-022-00844-1. URL: http://dx.doi.org/10.1038/s41524-022-00844-1.

Morris, Max D. (1991). "Factorial sampling plans for preliminary computational experiments". In: *Technometrics* 33.2, pp. 161–174. DOI: 10.1080/00401706.1991.10484804.

Mostafavi, Saman et al. (2018). "Calibration of White-Box Whole-Building Energy Models Using a Systems-Identification Approach". In: *IECON 2018 – 44th Annual Conference of the IEEE Industrial Electronics Society*, pp. 795–800. DOI: 10.1109/IECON.2018.8591600.

Mustavee, Shakib et al. (July 2022). "A linear dynamical perspective on epidemiology: interplay between early COVID-19 outbreak and human mobility". In: *Nonlinear Dynamics* 109 (2), pp. 1233–1252. ISSN: 0924-090X. DOI: 10.1007/s11071-022-07469-5.

Napoli, Giuseppe et al. (2024). "A comprehensive data analysis for aggregate capacity forecasting in Vehicle-to-Grid applications". In: *8th International Conference on Control, Automation and Diagnosis (ICCAD'24)*.

Narasingam, Abhinav and Joseph Sang-Il Kwon (2017). "Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing". In: *Computers and Chemical Engineering* 106. ESCAPE-26, pp. 501–511. ISSN: 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2017.07.002. URL: https://www.sciencedirect.com/science/article/pii/S0098135417302739.

Narendra, Kumpati S and Sai-Ming Li (2013). "Mathematical Perspectives on Neural Networks". In: Psychology Press. Chap. Neural Networks in Control Systems, p. 347.

Narendra, Kumpati S. and Kannan Parthasarathy (1990). "Identification and Control of Dynamical Systems Using Neural Networks". In: *IEEE Transactions on Neural Networks* 1.1, pp. 4–27.

Nemat-Nasser, S. and Jiang Li (Apr. 2000). "Electromechanical response of ionic polymer-metal composites". In: *Journal of Applied Physics* 87, pp. 3321–3331. DOI: 10.1063/1.372343.

Newbury, Kenneth Matthew (2002). "Characterization, modeling, and control of ionic polymer transducers". PhD thesis. Virginia Polytechnic Institute and State University.

Newbury, KM and DJ Leo (Jan. 2002). "Electromechanical Modeling and characterization of ionic polymer benders". In: *Journal of Intelligent Material Systems and Structures* 13, pp. 51–60. DOI: 10.1106/104538902027978.

Niculescu-Mizil, Alexandru and Rich Caruana (2005). "Predicting Good Probabilities with Supervised Learning". In: *Proceedings of KDD*, pp. 625–632. DOI: 10.1145/1081870.1081899.

Nogay, H. Selcuk (2022). "Estimating the aggregated available capacity for vehicle to grid services using deep learning and Nonlinear Autoregressive Neural Network". In: *Sustainable Energy, Grids and Networks* 29, p. 100590. ISSN: 2352-4677. DOI: https://doi.org/10.1016/j.segan.2021.100590.

Oh, Geunseob, David J. Leblanc, and Huei Peng (2022). "Vehicle Energy Dataset (VED), A Large-Scale Dataset for Vehicle Energy Consumption Research". In: *IEEE Trans Intel Trans Sys* 23.4, pp. 3302–3312. DOI: 10.1109/TITS.2020.3035596.

Oppenheim, Alan V. and Ronald W. Schafer (1999). *Discrete-Time Signal Processing*. 2nd. Prentice Hall. ISBN: 978-0137549207.

Ouyang, Jianyong et al. (2004). "On the mechanism of conductivity enhancement in poly(3,4-ethylenedioxythiophene):poly(styrene sulfonate) film through solvent treatment". In: *Polymer* 45.25, pp. 8443–8450. ISSN: 0032-3861. DOI: 10.1016/j.polymer.2004.10.001.

P, Swathi Prathaa et al. (2022). "Deep Learning and Dynamic Mode Decomposition for Inflation and Interest rate Forecasting". In: *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–7. DOI: 10.1109/ICCCNT54827.2022.9984629.

Paine, Greg (2019). "Understanding the true value of V2G." In: *An analysis of the customers and value streams for V2G in the UK, Cenex*. URL: https://www.cenex.co.uk/app/uploads/2019/10/True-Value-of-V2G-Report.pdf.

Pan, Sinno Jialin and Qiang Yang (2010). "A Survey on Transfer Learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10, pp. 1345–1359. DOI: 10.1109/TKDE.2009.191.

Park, Si Won et al. (2022). "Recent Progress in Development and Applications of Ionic Polymer-Metal Composite". In: *Micromachines* 13.8. ISSN: 2072-666X. DOI: 10.3390/mi13081290.

Pasquale, G. Di et al. (2019a). "Geometrical analysis of a bacterial cellulose-based sensing element". In: pp. 225–228. DOI: 10.1109/I2MTC.2019.8826915.

Pasquale, G. Di et al. (2020). "Performance characterization of a biodegradable deformation sensor based on bacterial cellulose". In: *IEEE Transactions on Instrumentation and Measurement* 69.5, pp. 2561–2569. DOI: 10.1109/TIM.2019.2957159.

Pasquale, Giovanna Di et al. (2019b). "A Bacterial Cellulose Based Mass Sensor". In: *2019 IEEE International Symposium on Measurements and Networking (M and N)*, pp. 1–4. DOI: 10.1109/IWMN.2019.8805008.

Patanè, Luca et al. (2024a). "Model identification and transferability analysis for vehicle-to-grid aggregate available capacity prediction based on origin–destination mobility data". In: *Energies* 17.24.

Patanè, Luca et al. (2024b). "Predictive Models for Aggregate Available Capacity Prediction in Vehicle-to-Grid Applications". In: *J. of Sens. and Act. Net.* 13.5, p. 49.

Patanè, Luca, Francesca Sapuppo, and Maria Gabriella Xibilia (2024). "Soft Sensors for Industrial Processes Using Multi-Step-Ahead Hankel Dynamic Mode Decomposition with Control". In: *Electronics* 13.15. ISSN: 2079-9292. DOI: 10.3390/electronics13153047. URL: https://www.mdpi.com/2079-9292/13/15/3047.

Patanè, Luca and Maria Gabriella Xibilia (Aug. 2021). "Echo-state networks for soft sensor design in an SRU process". In: *Information Sciences* 566, pp. 195–214. ISSN: 00200255. DOI: 10.1016/j.ins.2021.03.013.

Patanè, Luca et al. (2024a). "A Multiphysics Framework for Bacterial Cellulose Sensor Modeling". In: *2024 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering*.

Patanè, Luca et al. (2024b). "Predictive Models for Aggregate Available Capacity Prediction in Vehicle-to-Grid Applications". In: *Journal of Sensor and Actuator Networks* 13.5. ISSN: 2224-2708. DOI: 10.3390/jsan13050049. URL: https://www.mdpi.com/2224-2708/13/5/49.

Pearl, Judea (2009). *Causality: Models, Reasoning, and Inference*. 2nd ed. Cambridge: Cambridge University Press. ISBN: 978-0-521-89560-6. DOI: 10.1017/CBO9780511803161.

Perry, Dylan, Ning Wang, and Shen-Shyang Ho (2021). "Energy Demand Prediction with Optimized Clustering-Based Federated Learning". In: *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6. DOI: 10.1109/GLOBECOM46510.2021.9685647.

Pintelas, Emmanuel, Ioannis E. Livieris, and Panagiotis Pintelas (2020). "A Grey-Box Ensemble Model Exploiting Black-Box Accuracy and White-Box Intrinsic Interpretability". In: *Algorithms* 13.1, p. 17. DOI: 10.3390/a13010017.

Proctor, Joshua L., Steven L. Brunton, and J. Nathan Kutz (Jan. 2016). "Dynamic Mode Decomposition with Control". In: *SIAM Journal on Applied Dynamical Systems* 15 (1), pp. 142–161. ISSN: 1536-0040. DOI: 10.1137/15M1013857.

Proctor, Joshua L and Philip A Eckhoff (Mar. 2015). "Discovering dynamic patterns from infectious disease data using dynamic mode decomposition". In: *International Health* 7 (2), pp. 139–145. ISSN: 1876-3413. DOI: 10.1093/inthealth/ihv009. URL: https://doi.org/10.1093/inthealth/ihv009.

Pugal, D. et al. (2010). "Ionic polymer-metal composite mechanoelectrical transduction: review and perspectives". In: *Polymer International* 59, pp. 279–289.

Pugal, D et al. (2013). "IPMC mechanoelectrical transduction: its scalability and optimization". In: *Smart Materials and Structures* 22.12, p. 125029. DOI: 10.1088/0964-1726/22/12/125029.

Pugal, Deivid (2012). *Physics based model of ionic polymer-metal composite electromechanical and mechanoelectrical transduction*. University of Nevada, Reno.

Punning, A et al. (2009). "A distributed model of ionomeric polymer metal composite". In: *Journal of Intelligent Material Systems and Structures* 20.14, pp. 1711–1724.

Quiñonero-Candela, Joaquín et al., eds. (2009). *Dataset Shift in Machine Learning*. Cambridge, MA: MIT Press. ISBN: 978-0-262-17005-5.

Raissi, M., P. Perdikaris, and G. E. Karniadakis (2019). "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems". In: *Journal of Computational Physics* 378, pp. 686–707. DOI: 10.1016/j.jcp.2018.10.045.

Ralph, Benjamin James et al. (2021). "Machine Learning Driven Prediction of Residual Stresses for the Shot Peening Process Using a Finite Element Based Grey-Box Model Approach". In: *Journal of Manufacturing and Materials Processing* 5.2, p. 39. DOI: 10.3390/jmmp5020039.

Rane, Nitin Liladhar and Mallikarjuna Paramesha (2024). "Explainable Artificial Intelligence (XAI) as a Foundation for Trustworthy Artificial Intelligence". In: *Trustworthy Artificial Intelligence in Industry and Society*. Deep Science Publishing.

Raue, Andreas et al. (2009). "Structural and practical identifiability analysis using profile likelihood". In: *Bioinformatics* 25.15, pp. 1923–1929. DOI: 10.1093/bioinformatics/btp358.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin (2016). ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, pp. 1135–1144. DOI: 10.1145/2939672.2939778.

Rissanen, Jorma (1978). "Modeling by shortest data description". In: *Automatica* 14.5, pp. 465–471. DOI: 10.1016/0005-1098(78)90005-5.

Roscher, Ribana et al. (2020). "Explainable machine learning for scientific insights and discoveries". In: *IEEE Access* 8, pp. 42200–42216.

Rousseeuw, Peter J. (Nov. 1987). "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". In: *Journal of Computational and Applied Mathematics* 20, pp. 53–65. ISSN: 03770427. DOI: 10.1016/0377-0427(87)90125-7.

Rudin, Cynthia (2019). "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead". In: *Nature Machine Intelligence* 1.5, pp. 206–215. DOI: 10.1038/s42256-019-0048-x.

Rueden, Laura von et al. (2021). "Informed Machine Learning – A Taxonomy and Survey of Integrating Knowledge into Learning Systems". In: *Machine Learning: Science and Technology*.

Sahbudin, Murtadha Arif Bin, Marco Scarpa, and Salvatore Serrano (Feb. 2019). "MongoDB Clustering using K-means for Real-Time Song Recognition". In: *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, pp. 350–354. ISBN: 978-1-5386-9223-3. DOI: 10.1109/ICCNC.2019.8685489.

Saltelli, Andrea et al. (2008). *Global Sensitivity Analysis: The Primer*. Wiley. DOI: 10.1002/9780470725184.

Sapuppo, Francesca et al. (2025). "Explainability Analysis of V2G Aggregated Capacity Prediction". In: *2025 International Conference on Control, Automation and Diagnosis (ICCAD)*, pp. 1–6. DOI: 10.1109/ICCAD64771.2025.11099476.

Sarker, Abhirup, Tamzid Ul Islam, and Md. Robiul Islam (Sept. 2024). "A Review on Recent Trends of Bioinspired Soft Robotics: Actuators, Control Methods, Materials Selection, Sensors, Challenges, and Future Prospects". In: *Advanced Intelligent Systems* 7.3. ISSN: 2640-4567. DOI: 10.1002/aisy.202400414. URL: http://dx.doi.org/10.1002/aisy.202400414.

Schläpfer, Markus et al. (2021). "Using Mobility Patterns for the Planning of Vehicle-to-Grid Infrastructures that Support Photovoltaics in Cities". In: *ArXiv* abs/2112.15006.

Schmidt, Michael and Hod Lipson (Apr. 2009). "Distilling free-form natural laws from experimental data". en. In: *Science* 324.5923, pp. 81–85.

Schwartz, Roy et al. (2020). "Green AI". In: *Communications of the ACM* 63.12, pp. 54–63. DOI: 10.1145/3381831.

Schwarz, Gideon (1978). "Estimating the dimension of a model". In: *The Annals of Statistics* 6.2, pp. 461–464. DOI: 10.1214/aos/1176344136.

Shabab, Kazi Redwan et al. (July 2021). "Exploring DMD-type Algorithms for Modeling Signalised Intersections". In: *arXiv:2107.06369v1*.

Shahcheraghian, Amir, Hatef Madani, and Adrian Ilinca (2024). "From White to Black-Box Models: A Review of Simulation Tools for Building Energy Management and Their Application in Consulting Practices". In: *Energies* 17.2, p. 376. DOI: 10.3390/en17020376.

Shakerin, Farhad and Gopal Gupta (2020). "White-Box Induction From SVM Models: Explainable AI with Logic Programming". In: *Theory and Practice of Logic Programming* 20.5, pp. 656–670. DOI: 10.1017/S1471068420000081.

Shipman, Rob et al. (2021a). "Online Machine Learning of Available Capacity for Vehicle-to-Grid Services during the Coronavirus Pandemic". In: *Energies* 14.21. ISSN: 1996-1073. DOI: 10.3390/en14217176.

Shipman, Rob et al. (2021b). "We got the power: Predicting available capacity for vehicle-to-grid services using a deep recurrent neural network". In: *Energy* 221, p. 119813. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2021.119813.

Sijabat, Edwin K et al. (2020). "Optimization on the synthesis of bacterial nano cellulose (BNC) from banana peel waste for water filter membrane applications". In: *Materials Research Express* 7.5, p. 055010. DOI: 10.1088/2053-1591/ab8df7. URL: https://dx.doi.org/10.1088/2053-1591/ab8df7.

Sohlberg, B. and E. W. Jacobsen (2008). "Grey Box Modelling – Branches and Experiences". In: *IFAC Proceedings Volumes*. Vol. 41. 2, pp. 11415–11420. DOI: 10.3182/20080624-3-RO-2918.00689.

Song, G. et al. (Dec. 2013). "Global and Koopman modes analysis of sound generation in mixing layers". In: *Physics of Fluids* 25 (12), p. 124101. ISSN: 1070-6631. DOI: 10.1063/1.4834438. URL: http://aip.scitation.org/doi/10.1063/1.4834438.

Souza, Francisco and Rui Araujo (2014). "Online mixture of univariate linear regression models for adaptive soft sensors". In: *IEEE Transactions on Industrial Informatics* 10.2. Cited by: 35; All Open Access, Green Open Access, 937 – 945. DOI: 10.1109/TII.2013.2283147.

Strubell, Emma, Ananya Ganesh, and Andrew McCallum (2019). "Energy and Policy Considerations for Deep Learning in NLP". In: *Proceedings of ACL*, pp. 3645–3650. DOI: 10.18653/v1/P19-1355.

Stöcker, Julien Philipp et al. (2023). "FE-NN: Efficient-Scale Transition for Heterogeneous Microstructures Using Neural Networks". In: *PAMM* 23.3, e202300011. DOI: 10.1002/pamm.202300011.

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan (2017). "Axiomatic Attribution for Deep Networks". In: *Proceedings of ICML*. arXiv:1703.01365.

Susuki, Yoshihiko and Igor Mezic (Aug. 2012). "Nonlinear Koopman Modes and a Precursor to Power System Swing Instabilities". In: *IEEE Transactions on Power Systems* 27 (3), pp. 1182–1191. ISSN: 0885-8950. DOI: 10.1109/TPWRS.2012.2183625.

Taylor, Roy et al. (May 2018). "Dynamic mode decomposition for plasma diagnostics and validation". In: *Review of Scientific Instruments* 89 (5), p. 053501. ISSN: 0034-6748. DOI: 10.1063/1.5027419. URL: http://aip.scitation.org/doi/10.1063/1.5027419.

Tian-hao, MOU, ZOU Yuan-yuan, and LI Shao-yuan (2024). "Enhancing graph convolutional network of knowledge-based co-evolution for industrial process key variable prediction." In: *Control Theory & Applications/Kongzhi Lilun Yu Yinyong* 41.3.

Tibshirani, Robert (1996). "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B* 58.1, pp. 267–288. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

Torralba, Antonio and Alexei A. Efros (2011). "Unbiased Look at Dataset Bias". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1521–1528. DOI: 10.1109/CVPR.2011.5995347.

Trigona, Carlo et al. (2021). "Towards Environmentally Friendly Accelerometers Based on Bacterial Cellulose". In: *Applied Sciences* 11.17. ISSN: 2076-3417. DOI: 10.3390/app11177903.

Udrescu, Silviu-Marian and Max Tegmark (2020). "AI Feynman: A physics-inspired method for symbolic regression". In: *Science Advances* 6.16, eaay2631. DOI: 10.1126/sciadv.aay2631.

Vahabi, Meisam et al. (2011). "Experimental identification of IPMC actuator parameters through incorporation of linear and nonlinear least squares methods". In: *Sensors and Actuators A: Physical* 168.1, pp. 140–148.

Van Houdt, Greg, Carlos Mosquera, and Gonzalo Napoles (Dec. 2020). "A Review on the Long Short-Term Memory Model". In: *Artificial Intelligence Review* 53. DOI: 10.1007/s10462-020-09838-1.

Vapnik, Vladimir (1995). *The Nature of Statistical Learning Theory*. Springer. DOI: 10.1007/978-1-4757-2440-0.

Visual Crossing Weather Data Service (n.d.). https://www.visualcrossing.com/weather/weather-data-services.

Wang, Jindong et al. (2022). "Generalizing to Unseen Domains: A Survey on Domain Generalization". In: *arXiv preprint arXiv:2103.03097*. Version updated in 2022.

Wang, Jing, Javad Tavakoli, and Youhong Tang (2019). "Bacterial cellulose production, properties and applications with different culture methods - A review". In: *Carbohydrate Polymers* 219, pp. 63–76. ISSN: 0144-8617. DOI: 10.1016/j.carbpol.2019.05.008.

Wang, Tiesheng et al. (Aug. 2016). "Electroactive polymers for sensing". en. In: *Interface Focus* 6.4, p. 20160026.

Weiss, Karl, Taghi M Khoshgoftaar, and DingDing Wang (2016). "A survey of transfer learning". In: *Journal of Big Data* 3, p. 9.

Wiemer, Hajo et al. (2023). "Need for UAI–Anatomy of the Paradigm of Usable Artificial Intelligence for Domain-Specific AI Applicability". In: *Multimodal Technologies and Interaction* 7.3, p. 27. DOI: 10.3390/mti7030027.

Williams, Matthew O., Ioannis G. Kevrekidis, and Clarence W. Rowley (Dec. 2015). "A Data–Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition". In: *Journal of Nonlinear Science* 25 (6), pp. 1307–1346. ISSN: 0938-8974. DOI: 10.1007/s00332-015-9258-5.

Wolfram, Dirk and Thomas Meurer (2023). "DMD-Based Model Predictive Control for a Coupled PDE-ODE System". In: *IFAC-PapersOnLine* 56.2. 22nd IFAC World Congress, pp. 4258–4263. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2023.10.1789. URL: https://www.sciencedirect.com/science/article/pii/S2405896323021985.

Wood, Nathan Gabriel (June 2024). "Explainable AI in the military domain". en. In: *Ethics Inf. Technol.* 26.2.

Wu, Ziyou, Steven L. Brunton, and Shai Revzen (Dec. 2021). "Challenges in dynamic mode decomposition". In: *Journal of The Royal Society Interface* 18 (185). ISSN: 1742-5662. DOI: 10.1098/rsif.2021.0686.

Xie, Shiwen et al. (2020). "Optimal Setting and Control for Iron Removal Process Based on Adaptive Neural Network Soft-Sensor". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50.7, pp. 2408–2420. DOI: 10.1109/TSMC.2018.2815580.

Xie, Yongfang et al. (2023). "Adversarial Training-Based Deep Layer-Wise Probabilistic Network for Enhancing Soft Sensor Modeling of Industrial Processes". In: *IEEE*

Transactions on Systems, Man, and Cybernetics: Systems, pp. 1–13. DOI: 10.1109/TSMC.2023.3322195.

Xu, Chongchong et al. (2022). "Review on Interpretable Machine Learning in Smart Grid". In: *Energies* 15.4427. DOI: 10.3390/en15124427.

Xu, Mingkuo et al. (2024). "On the V2G capacity of shared electric vehicles and its forecasting through MAML-CNN-LSTM-Attention algorithm". In: *IET Generation, Transmission & Distribution* 18.6, pp. 1158–1171. DOI: https://doi.org/10.1049/gtd2.12921. eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/gtd2.12921. URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/gtd2.12921.

Xu, Yanwen et al. (2023). "Physics-Informed Machine Learning for Reliability and Systems Safety Applications: State of the Art and Challenges". In: *Reliability Engineering & System Safety* 230, p. 108900. DOI: 10.1016/j.ress.2023.108900.

Yang, L., Y. Yang, and H. Wang (2023a). "Modeling and control of ionic polymer metal composite actuators: A review". In: *European Polymer Journal* 186, p. 111821. DOI: 10.1016/j.eurpolymj.2023.111821.

Yang, Liang, Yanning Yang, and Hong Wang (2023b). "Modeling and control of ionic polymer metal composite actuators: A review". In: *European Polymer Journal* 186, p. 111821. ISSN: 0014-3057. DOI: https://doi.org/10.1016/j.eurpolymj.2023.111821.

Yang, Zhuo et al. (2017). "Investigating Grey-Box Modeling for Predictive Analytics in Smart Manufacturing". In: *43rd Design Automation Conference, Volume 2B*. Proceedings of the ASME, V02BT03A024.

Yuan, Xiaofeng et al. (Sept. 2020). "Stacked isomorphic autoencoder based soft analyzer and its application to sulfur recovery unit". In: *Information Sciences* 534, pp. 72–84. ISSN: 00200255. DOI: 10.1016/j.ins.2020.03.018.

Yuen, Jonathan D. et al. (2020). "Microbial Nanocellulose Printed Circuit Boards for Medical Sensing". In: *Sensors* 20.7. ISSN: 1424-8220. DOI: 10.3390/s20072047. URL: https://www.mdpi.com/1424-8220/20/7/2047.

Zeng, Teng, Scott Moura, and Zhe Zhou (2023). "Joint Mobility and Vehicle-to-Grid Coordination in Rebalancing Shared Mobility-on-Demand Systems". In: *IFAC-PapersOnLine* 56.2. 22nd IFAC World Congress, pp. 6642–6647. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2023.10.365.

Zhang, Chunju et al. (2020). "Deep Feature Aggregation Network for Hyperspectral Remote Sensing Image Classification". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13, pp. 5314–5325. DOI: 10.1109/JSTARS.2020.3020733.

Zhang, Haopeng et al. (2024a). "A Stochastic Bi-level Optimal Allocation Approach of Intelligent Buildings Considering Energy Storage Sharing Services". In: *IEEE Transactions on Consumer Electronics*, pp. 1–1. DOI: 10.1109/TCE.2024.3412803.

Zhang, Rufeng et al. (2024b). "Two-stage robust operation of electricity-gas-heat integrated multi-energy microgrids considering heterogeneous uncertainties". In: *Applied Energy* 371, p. 123690. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2024.123690. URL: https://www.sciencedirect.com/science/article/pii/S0306261924010730.

Zhang, Xiaoshuang et al. (Nov. 2021). "PEDOT:PSS: From conductive polymers to sensors". In: *Nanotechnology and Precision Engineering* 4.4, p. 045004. ISSN: 1672-6030. DOI: 10.1063/10.0006866.

Zhang, Xinglong et al. (2022). "Robust Learning-Based Predictive Control for Discrete-Time Nonlinear Systems With Unknown Dynamics and State Constraints". In:

IEEE Transactions on Systems, Man, and Cybernetics: Systems 52.12, pp. 7314–7327. DOI: 10.1109/TSMC.2022.3146284.