Monitoring and Event Detection for Flexibility Management in Cyber-Physical Energy Systems

Müller, Nils

Link to article, DOI:
10.11581/DTU.00000289

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

DTU Wind and Energy Systems
Department of Wind and Energy Systems

Monitoring and Event Detection for Flexibility Management in Cyber-Physical Energy Systems

Ph.D. Thesis

Nils Müller

Risø, Denmark, July 2023

Monitoring and Event Detection for Flexibility Management in Cyber-Physical Energy Systems

Author
Nils Müller

Supervisors
Kai Heussen, Senior Researcher
Department of Wind and Energy Systems, Technical University of Denmark, Denmark

Henrik W. Bindner, Senior Researcher
Department of Wind and Energy Systems, Technical University of Denmark, Denmark

Dissertation Examination Committee:
Yi Zong, Senior Researcher (Internal examiner)
Department of Wind and Energy Systems, Technical University of Denmark, Denmark

Sami Repo, Professor
Faculty of Information Technology and Communication Sciences, Tampere University, Finland

Jay Johnson, Distinguished Member of Technical Staff
Renewable and Distributed Systems Integration Group, Sandia National Laboratories, United States

Division of Power and Energy Systems (PES)
DTU Wind and Energy Systems
Frederiksborgvej 399, DTU Risø Campus
4000 Roskilde, Denmark
https://windenergy.dtu.dk/english
Tel: (+45) 46 77 50 85
E-mail: communication@windenergy.dtu.dk

Release date: July 2023
Class: Public
Field: Wind and Energy Systems
Remarks: The dissertation is presented to the Department of Wind and Energy Systems of the Technical University of Denmark in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

DOI: https://doi.org/10.11581/DTU.00000289

Preface

This thesis was prepared at the Department of Wind and Energy Systems of the Technical University of Denmark (DTU) in partial fulfilment of the requirements for acquiring the degree of Doctor of Philosophy in Engineering. The author contributed to and was financially supported by the ERA-Net project *HONOR* under the Grant Agreement No. 91363.

The Ph.D. project was conducted between 15 July 2020 and 14 July 2023 under the supervision of Senior Researcher Kai Heussen and Senior Researcher Henrik W. Bindner. During this period, the author was employed as a Ph.D. student in the Section for Distributed Energy Systems in the Division of Power and Energy Systems at the Department of Wind and Energy Systems of the Technical University of Denmark.

This thesis provides a summary of the research work conducted by the author during the Ph.D. project. It encompasses a thesis framework and summary, organized in five chapters, and seven attached scientific articles. Six of these articles have been peer-reviewed and published, while the remaining one is currently under review at an international journal.

Copenhagen, 14 July 2023

Nils Müller

Acknowledgements

The support and guidance of numerous people played an indispensable role in making my Ph.D. journey over the past three years an exciting and rewarding reality. First and foremost, I am deeply grateful to my Ph.D. supervisors Kai Heussen and Henrik W. Bindner for their continuous and thoughtful mentorship, and for giving me the opportunity and trust to enter this exciting field of research, even though it was largely new to me. A special thanks to my main supervisor Kai for always helping and motivating me in his own unique way to explore new research directions that I certainly owe a few well-invested gray hairs.

One of the most thrilling facets of my Ph.D. project has been the chance to engage in international collaborations with fellow researchers. I would like to thank Jörg Matthes and Kaibin Bao for warmly welcoming me at the Institute for Automation and Applied Informatics at the Karlsruhe Institute of Technology. The fruitful discussions and access to experimental facilities have significantly shaped my work. I also would like to express my gratitude to my colleagues from the HONOR project, in particular Zeeshan Afzal and Mathias Ekstedt, for constructive collaborations across different scientific disciplines.

It has been exceptionally rewarding for me to be part of the awESOMe DES group at DTU. In particular, I thank Carsten Heinrich for supporting me in finding initial orientation in the myriad of research questions and achieving first results. A special thanks also to my secret third supervisor Haris Ziras for always being available for great discussions and collaborations which has been a key factor in keeping me convinced of what I am doing along the way.

My sincere thanks to all the passionate colleagues in Risø and Lyngby, who offered me a productive and friendly working environment. A heartfelt gratitude to Jan Martin Zepter for always assisting me with all kinds of organizational matters regarding the Ph.D, far beyond the official mentorship period. I am further indebted to Haris for proofreading my thesis, and to Henrik for finding and correcting all the mistakes in my Danish abstract.

Finally, a special thanks to my friends and family. There are no words to express my gratitude to my parents. Thank you for your unconditional love and support. Only this allowed me to get this far. I also thank my roommates Jan, Nele, and Nuni for offering me a place of retreat and ensuring my nutrition in stressful times. And last but not least, Nele, having you at my side even during every single little and big "Weltuntergang" within this three-year journey was the most important thing and means the world to me.

Summary

Growing concerns about climate change and dependence on finite fossil fuels from authoritarian regimes are motivating an accelerated transition towards sustainable energy systems. The development is characterized by increasing integration of distributed renewable generation and electrification of several sectors, which puts stress on electrical distribution grids. A cost- and resource-efficient strategy to relieve grid infrastructure is the flexible operation of distributed energy resources (DERs) such as solar plants, heat pumps, and electric vehicles. Planning, realization and verification of such local flexibility requires closed-loop integration of physical processes with digital systems on DER and distribution grid level. The associated increase of operational complexity, new events and failure causes, and emerging data and information needs set new situational awareness requirements on asset and grid level.

This thesis studies the deployment of machine learning (ML)-based predictive analytics for addressing operational awareness requirements of DERs and distribution grids that emerge from local flexibility and the associated digitalization. In this context, three research topics are addressed based on seven research articles.

The first topic addresses the question *"What are the operational challenges of local flexibility and the associated digitalization that set new requirements on the situational awareness for DERs and distribution grids?"*. Flexibility realization can be subject to weather- and consumer-induced uncertainty, computational limitations and data manipulation. Thus, forecasts for flexibility scheduling should be computationally lightweight and robust against manipulations, apart from being accurate. While digitalization enables a broad spectrum of flexibility mechanisms, the associated increasing load activity and emerging cyber threats require new real-time monitoring and event identification solutions: Low voltage (LV) state estimation should account for possible flexibility-induced stochasticity, which includes uncertainty quantification and incorporation of price signals and other data sources. To increase awareness of distribution system operators (DSOs) about flexibility activations from mechanisms without their active involvement, flexibility activation identification should be investigated. Finally, the risk of coordinated attacks against fleets of flexible resources requires advanced attack identification concepts for DERs, which includes integrated evaluation of cyber network traffic and physical process data.

Within the second research topic, the question *"How do weather- and consumer-induced stochasticity, computational constraints and cyber attacks impair flexibility realization, and how can data and modeling strategies based on predictive analytics enable or facilitate computationally efficient, cost-optimal and cyber-secure usage of*

DERs as flexible resources?" is addressed. A systematic evaluation of forecasting strategies for flexibility schedule optimization of a residential photovoltaic-battery system reveals that unpredictable weather- and consumer behavior reduces financial rewards for prosumers by 5-10 percentage points compared to a theoretical optimum. Moreover, it is shown that ML-based forecasting enables such near-optimal cost-savings while being computationally lightweight, applicable with almost no data history, and robust against weather-input manipulations. In contrast, the results demonstrate that scheduling flexible assets based on manipulated price signals can turn savings into additional costs and peak shaving- into peak-reinforcing behavior, which showcases potential consequences of cyber-physical attacks. As means for cyber-secure DER operation, cyber-physical event identification concepts are investigated. It is shown that the joint evaluation of cyber network and physical process data applying supervised ML improves the event identification performance and enables attack and fault detection in one holistic approach. To leverage these benefits while meeting practical requirements such as independence of scare historical event observations and human-verifiable predictions, the data-driven Cyber-Physical Event Reasoning System CyPhERS is developed. It is demonstrated that CyPhERS can generate informative and human-interpretable event signatures which can be evaluated to infer event information such as occurrence, type, affected devices, attacker location, and physical impact for both known and unknown attacks and faults.

The third research topic is concerned with the question *"How does local flexibility affect monitoring of distribution grids, and how can data and modeling strategies applying predictive analytics facilitate effective situational awareness for DSOs under high shares of flexible resources?"*. On the basis of a systematic evaluation of several flexibility scenarios it is demonstrated that local flexibility can introduce aleatoric uncertainty to data-driven LV state estimation. By applying a Bayesian neural network and adding real-time power and voltage readings from secondary substations to the inputs, flexibility-induced uncertainty can be reliably quantified and partly counteracted. To further support the operational awareness of DSOs, a data-driven flexibility activation identification concept is proposed. It is demonstrated that flexibility activation identification in aggregated load data is feasible while accounting for practical requirements such as real-time detection and handling of multiple and partly unknown load-altering event types by applying unsupervised anomaly detection and open-set classification.

The results of this thesis demonstrate how ML-based predictive analytics can be leveraged to provide solutions to operational challenges for DERs and distribution grids in the context of local flexibility and the associated digitalization, while respecting practical requirements such as computational efficiency, prediction verifiability and use of publicly available tools.

Resumé

Stigende bekymringer om klimaforandringer og afhængighed af begrænsede fossile brændstoffer, der ofte kommer fra autoritære regimer motiverer en accelereret overgang til bæredygtige energisystemer. Udviklingen er karakteriseret ved øget udrulning og integration af distribueret vedvarende energi og elektrificering af flere sektorer, hvilket lægger pres på distributionsnet. En omkostnings- og ressourceeffektiv strategi til at aflaste netinfrastrukturen er at udnytte den mulige fleksible drift af distribuerede energiressourcer (DER'er) såsom solcelleanlæg og elektriske køretøjer. Planlægning, realisering og verifikation af en sådan lokal fleksibilitet kræver at fysiske processer og digitale systemer integreres på DER- og distributionsnetniveau til at skabe en samlet koordinerende styring. Den tilknyttede øgede operationelle kompleksitet, nye systembegivenheder og nye årsager til fejl samt fremkosten af behov for data og information stiller nye krav til overblik over driftsituationen på ressource- og netniveau.

Denne afhandling undersøger implementeringen af maskinlæring (ML)-baseret prediktiv analyse til at imødekomme de operationelle krav til overblik over driftsituationen for DER'er og distributionsnet, der opstår som følge af aktivering af lokal fleksibilitet og den tilknyttede digitalisering. I denne sammenhæng behandles tre forskningsemner baseret på syv forskningsartikler.

Det første emne adresserer spørgsmålet *"Hvad er de operationelle udfordringer ved lokal fleksibilitet og den tilknyttede digitalisering, som resulterer i nye krav til kendskab til driftsituations for DER'er og distributionsnet?"*. Realiseringen af fleksibilitet kan være underlagt usikkerhed forårsaget af vejr og forbrugere, beregningsmæssige begrænsninger og data manipulation. Derfor bør prognoser for fleksibilitetsplanlægning være beregningsmæssigt lette og robuste over for manipulation, udover at være præcise. Mens digitalisering muliggør en bred vifte af fleksibilitetsmekanismer, kræver den tilknyttede øgede aktivitet i forbruget og kommende cybertrusler nye løsninger til realtidsovervågning og identifikation af hændelser: Tilstandsestimering på lavspændingsniveau (LV) bør tage højde for mulig fleksibilitetsinduceret stokastisk adfærd, hvilket inkluderer usikkerhedskvantificering og inkorporering af prisindikatorer og andre datakilder. For at øge bevidstheden hos distributionsselskaberne (DSO'er) om fleksibilitetsaktivering fra mekanismer uden deres aktive involvering, bør identifikation af fleksibilitetsaktivering undersøges. Endelig kræver risikoen for koordinerede angreb mod flåder af fleksible ressourcer avancerede koncepter til identifikation af disse angreb mod DER'er, hvilket inkluderer integreret evaluering af cybernetværkstrafik og data om fysiske processer.

Inden for det andet forskningsemne er spørgsmålet *"Hvordan påvirker vejr- og forbrugerinduceret stokastik, beregningsmæssige begrænsninger og cyberangreb realis-*

eringen af fleksibilitet, og hvordan kan data- og modelleringsstrategier baseret på prediktiv analyse muliggøre eller lette en beregningsmæssigt effektiv, omkostningsoptimal og cyber-sikker brug af DER'er som fleksible ressourcer?". En systematisk evaluering af prognosestrategier til optimering af fleksibilitetsplanlægning for et pv-batterisystem for boliger afslører, at uforudsigeligt vejr og forbrugeradfærd reducerer den økonomiske belønning for prosumere med 5-10 procentpoint sammenlignet med et teoretisk optimum. Derudover viser det sig, at ML-baserede prognoser muliggør sådanne næroptimale omkostningsbesparelser, samtidig med at de er beregningsmæssigt lette, kan anvendes med næsten ingen datahistorik og er robuste over for manipulation af vejrinput. Omvendt viser resultaterne, at planlægning af fleksible aktiver baseret på manipulerede prisindikatorer kan omdanne besparelser til ekstra omkostninger og forbrugsspidsreduktion til peak-forstærkende adfærd, hvilket viser nogle af de potentielle konsekvenser af cyber-fysiske angreb. Som midler til cyber-sikker DER-drift undersøges koncepter til identifikation af cyber-fysiske begivenheder. Det vises, at den fælles samtidige evaluering af cybernetværks- og fysiske procesdata ved anvendelse af superviseret ML forbedrer performance af identifikationsalgoritmerne til identifikation af begivenheder og muliggør angrebsdetektion og fejldetektion i én samlet holistisk tilgang. For at udnytte disse fordele udvikles det datadrevne Cyber-Physical Event Reasoning System, CyPhERS, der også samtidig opfylder praktiske krav som behov for få historiske begivenhedsobservationer og menneskelæs- og fortolkningsbare forudsigelser opfyldes,. Det demonstreres, at CyPhERS kan generere informative og menneskelæsbare begivenheds-signaturer, såsom kan evalueres for at udlede begivenhedsinformationer som forekomst, type, påvirkede enheder, angriberes placering og fysisk påvirkning for både kendte og ukendte angreb og fejl.

Det tredje forskningsemne beskæftiger sig med spørgsmålet *"Hvordan påvirker lokal fleksibilitet overvågningen af distributionsnettet, og hvordan kan data- og modelleringsstrategier ved anvendelse af prediktiv analyse give DSO'er overblik over den øjeblikkelige driftssituation på let og effektiv måde når der er stor andel af fleksible ressourcer?"*. På baggrund af en systematisk evaluering af flere fleksibilitetsscenarier vises det, at lokal fleksibilitet kan introducere aleatorisk usikkerhed i datadrevet tilstandsestimering på lavspændingsniveau. Ved anvendelse af et bayesiansk neuralt netværk og tilføjelse af realtidsmålinger af effekt og spænding fra sekundære transformerstationer til input kan fleksibilitetsinduceret usikkerhed pålideligt kvantificeres og delvist modvirkes. For yderligere at støtte DSO'ers operationelle kendskab til driftstilstanden foreslås et datadrevet koncept til identifikation af fleksibilitetsaktivering. Det vises, at identifikation af fleksibilitetsaktivering i aggregerede belastningsdata er muligt, samtidig med at der tages højde for praktiske krav som realtidsdetektion og håndtering af flere og delvist ukendte typer af belastningsændrende begivenheder ved anvendelse af usuperviseret anomalidetektion og open-set klassifikation.

Resultaterne af denne afhandling demonstrerer, hvordan ML-baseret forudsigelig analyse kan udnyttes til at levere løsninger på de operationelle udfordringer for DER'er og distributionsnet i sammenhæng med lokal fleksibilitet og den tilknyttede digitalisering, samtidig med at der tages hensyn til praktiske krav som beregningsmæssig effektivitet, verificerbarhed af forudsigelser og brug af åbent værktøjer.

Contents

Acronyms

AC	alternating current
ANN	artificial neural network
ARIMA	autoregressive integrated moving average
ARP	address resolution protocol
BMS	battery management system
BNN	Bayesian neural network
BRP	balance responsible party
CNN	convolutional neural network
CSIRT	cyber security incident response team
DER	distributed energy resource
DM	data manager
DoS	denial-of-service
DS	data server
DSO	distribution system operator
EMS	energy management system
EV	electric vehicle
EVM	extreme value machine
FCIA	false command injection attack
FDIA	false data injection attack
GBDT	gradient-boosted decision trees
GHI	global horizontal irradiance
HMI	human-machine interface
HPC	high-performance computing
HTM	hierarchical temporal memory
ICT	information and communication technology
IHT	instance hardness threshold
IoT	internet of things
IP	internet protocol
IT	information technology
KNN	k-nearest neighbors
LFM	local flexibility market
LQR	linear quantile regression
LV	low voltage
MAC	media access control
MB	modbus

MITM	man-in-the-middle
ML	machine learning
MV	medium voltage
OT	operational technology
PCA	principle component analysis
PI	prediction interval
PLC	programmable logic controller
PV	photovoltaic
PVMS	photovoltaic management system
RF	random forest
RMSE	root mean squared error
SCADA	supervisory control and data acquisition
SM	smart meter
SMOTE	synthetic minority oversampling technique
SOC	state of charge
SR	spectral residual
SVM	support vector machine
TCP	transmission control protocol
TLS	transport layer security
TPE	tree Parzen estimator
UDP	user datagram protocol

List of Figures

List of Tables

Part I

Thesis Framework and Summary

CHAPTER 1

Introduction

1.1 Context and motivation

Growing concerns about climate change and dependence on finite fossil fuels from authoritarian regimes are driving the transition towards sustainable energy systems. In this context, the proportion of intermittent and distributed renewable generation in the energy mix experiences an ongoing increase. In 2022, solar and wind energy supplied 59.3 % of Denmark's total electricity demand [1]. At the same time, the decarbonization of the mobility, heat, and industrial sectors entails electrification of technologies and systems previously based on fossil fuels. Together, the continuous increase of distributed generation and electrification entails widespread adoption of distributed energy resources (DERs), such as solar plants, battery storages, heat pumps, and electric vehicles (EVs). The presence of DERs raises stochasticity and volatility in both generation and demand due to the influence of weather and consumer behavior, which ultimately results in higher stress for electric distribution grids, for example through load peaks and bidirectional power flows. One measure for enabling large-scale integration of DERs is the extension of grid infrastructure and energy storage capacity. A more cost- and resource-efficient strategy is to exploit the potential for flexible DER operation. By actively shifting a fleet of DERs away from their normal operational patterns in a coordinated manner, usage of existing grid infrastructure can be optimized, for example, through reducing load concurrency and associated peaks. Another aspect is the local matching of generation and demand which has the potential to reduce transmission losses and curtailment of renewable generation. Owners of DERs are incentivized to deploy their resource as a flexible asset by cost savings and active participation and support of a sustainable energy transition. Compared to the process of planning and constructing new grid infrastructure, local flexibility programs have the potential for shorter implementation time horizons, enabling faster integration of substantial amounts of renewable generation [2]. Thus, local flexibility offers both economic and environmental benefits by minimizing or postponing need for further grid infrastructure and storage capacity expansion.

Accessing the potential of local flexibility requires information and communication technology (ICT)-based closed-loop integration of physical processes with digital systems for planning, monitoring, optimization and control at DER and distribution grid level. Together with the associated digitalization, local flexibility amplifies operational complexity for DERs and distribution grids. One factor is emerging data requirements for flexibility planning and realization, which includes ensuring availabil-

ity and integrity of process-relevant data, and need for new information such as load and generation forecasts. Moreover, the complex integration of physical and digital processes introduces new events and failure causes, including cyber-physical attacks and communication breakdowns. Since local flexibility aims at operating distribution grids closer to their capacity limits for efficiency, failure or misuse of flexibility mechanisms can lead to power system reliability issues. Finally, digitalization enables implementation of different flexibility mechanisms with varying actors and control objectives, which increases activeness in distribution grids. The potentially introduced stochasticity and volatility may interfere with operational tools such as fault localization or state estimation. For the listed reasons, integration of local flexibility and the associated digitalization set new awareness requirements for the operation of DERs and distribution grids.

The growing amount of historical and real-time available data, along with recent advances in machine learning (ML)-based predictive analytics, including forecasting, monitoring, and event detection, provide new opportunities for improving operational awareness of DERs and distribution grids. Applying ML in the context of local flexibility comes with particular requirements. Since smaller or residential DERs are typically equipped with simple hardware, computational efficiency must be taken into consideration, which includes careful selection of data sources and models. Another factor is transparency and trustworthiness of predictions. As ML typically comes without any performance guarantees, results must be interpretable by DER and distribution system operators (DSOs) to enable recognition of erroneous or unreliable predictions, which otherwise may entail operational failures in the flexibility planning and realization process.

Exploring the use of ML-based predictive analytics for improved awareness of DER and distribution grid operation in a flexibility context has different facets. First, possible operational challenges of local flexibility and the associated digitalization need to be identified, and resulting requirements for situational awareness derived. Second, related possible negative impacts should be analyzed to understand the scope and significance of the problem. Finally, on this basis, operational awareness strategies and concepts applying ML-based predictive analytics can be developed for efficient, reliable and secure integration of local flexibility, taking technical requirements such as computational efficiency and prediction trustworthiness into consideration.

1.2 Scientific contributions

This Ph.D. thesis focuses on leveraging ML-based predictive analytics to address awareness requirements for the operation of DERs and distribution grids that emerge from integrating local flexibility and the associated digitalization. Consequently, the thesis is situated in the interdisciplinary sphere between electrical engineering, applied data science and cyber security. The overarching objective is to gain insights into what affects the applicability and performance of data-driven monitoring and event

detection techniques in the context of local flexibility management, and to develop data and modeling strategies to leverage their potential under real-world conditions. Thus, a largely empirical and practice-oriented approach is followed, which involves (i) evaluation of realistic data and scenarios, (ii) leveraging publicly available models and techniques, and (iii) taking practical aspects such as limited data and computational resource availability into account. In this context, this thesis addresses three research topics, defined by the following research questions:

RQ1 What are the operational challenges of local flexibility and the associated digitalization that set new requirements on the situational awareness for DERs and distribution grids?

RQ2 How do weather- and consumer-induced stochasticity, computational constraints and cyber attacks impair flexibility realization, and how can data and modeling strategies based on predictive analytics enable or facilitate computationally efficient, cost-optimal and cyber-secure usage of DERs as flexible resources?

RQ3 How does local flexibility affect monitoring of distribution grids, and how can data and modeling strategies applying predictive analytics facilitate effective situational awareness for DSOs under high shares of flexible resources?

The research questions are addressed by the seven independent scientific publications **Paper A**–**Paper G**. A graphical overview of the research topics together with an allocation of the individual papers is provided in Figure 1.1.

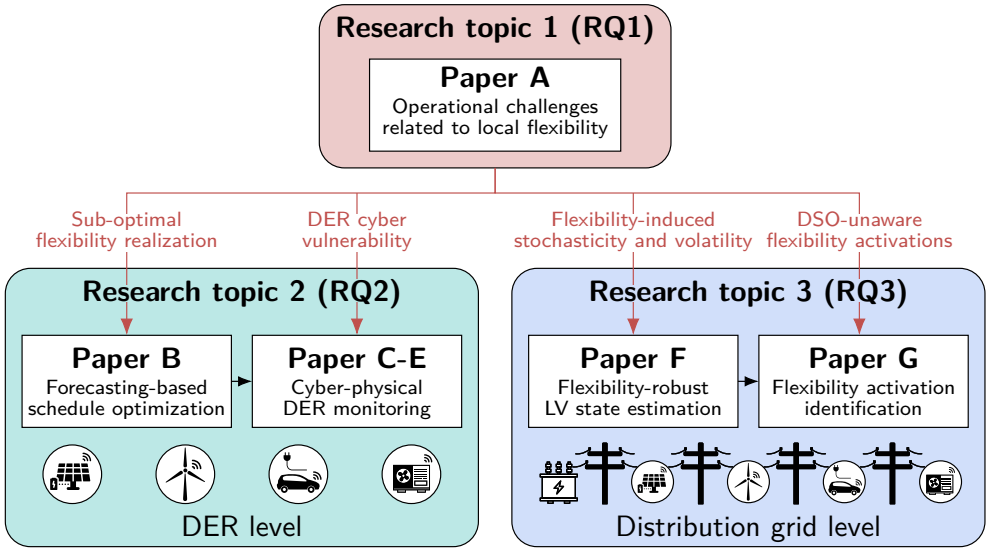


Figure 1.1: Research topics of this thesis and allocation of the individual papers.

The first topic focuses on a qualitative identification of operational challenges of local flexibility integration with particular focus on the associated digitalization, and derivation of resulting requirements for the situational awareness of DERs and distribution grids. **Paper A** systematically formulates threat scenarios for different actors of local flexibility markets (LFMs). A broad spectrum of possible threat origins are taken into account, including device failures, human errors, weather- and consumer-induced stochasticity, and cyber-criminals of varying skill level. On that basis, emerging requirements for operational awareness in distribution grids are derived. The insights provided by **Paper A** serve as motivation for the scientific contributions to the research topics 2 and 3 (see Figure 1.1).

The second research topic is concerned with empirically evaluating previously identified operational challenges of local flexibility integration on DER level, and developing data and modeling strategies based on predictive analytics for addressing those. **Paper B** empirically assesses the impact of weather- and consumer-induced stochasticity and volatility, computational limitations, and malicious data manipulations on the realization of price-based flexibility, considering the case of residential photovoltaic (PV)-battery systems. In that context, several data and modeling strategies for ML-based residential load and generation forecasting are evaluated, and recommendations on best trade-offs between profitability, complexity, and security provided. Motivated by the possible cyber vulnerabilities presented in **Paper A**, and the demonstrated impact of adversarially manipulated flexibility realization (**Paper B**), several works address cyber-physical DER monitoring, which relates to the joint evaluation of physical process and cyber network data. **Paper C** systematically assesses the concept of cyber-physical attack and fault identification applying supervised ML. Inspired by the demonstrated advantages of a cyber-physical approach to event identification, and the practical shortcomings of supervised ML in the context of rare event detection, the new cyber-physical event reasoning system CyPhERS is proposed in **Paper D**. The objective of CyPhERS is the provision of interpretable real-time information about unknown and known types of cyber attacks and physical failures, without need for historical event observations. **Paper E** adapts and demonstrates CyPhERS for monitoring of PV-battery systems.

Research topic 3 involves empirical evaluation of previously identified operational challenges of local flexibility integration at the distribution grid level, as well as development of data and modeling strategies applying predictive analytics for addressing them. **Paper F** assesses the impact of frequent flexibility activations on the accuracy and uncertainty of data-driven low voltage (LV) state estimation. A special focus is on the systematic evaluation of different uncertainty types under several flexibility scenarios. In that context, input data and modeling recommendations for enabling reliable and accurate estimations under high shares of flexible resources applying Bayesian neural networks (BNNs) are provided. **Paper G** proposes and demonstrates a concept for data-driven flexibility activation identification in active power readings as operational awareness support for DSOs. Particular attention is given to the scarcity of historical flexibility activation observations and the differentiation to other known and unknown load-altering events by applying algorithms from the

field of unsupervised detection and open-set classification. The proposed concept represents a use case for the LV state estimations evaluated in **Paper F**.

1.3 Thesis structure

The structure of this thesis is divided into two parts. In Part I, the framework and main contributions of the Ph.D. project are presented. The seven scientific publications constituting this thesis are collected in Part II. Part I is organized into five chapters. Following the introduction (Chapter 1), key contributions to the three defined research topics are successively presented in the Chapters 2–4.

Chapter 2 summarizes central findings on research topic 1 (**RQ1**). Based on content of **Paper A**, several possible operational challenges related to local flexibility integration and the associated digitalization are explained, including a description of their background and possible impacts. Moreover, resulting new awareness requirements for the operation of DERs and distribution grids are derived, and suggestions for addressing those by predictive analytics provided. The gained insights serve as motivation for the contributions on the research topics 2 and 3.

Chapter 3 is concerned with presenting key results on research topic 2 (**RQ2**). On the basis of the findings from **Paper B**, the economic impact of sub-optimal price-based flexibility realization on prosumers is empirically evaluated in Section 3.1. Several influencing factors are taken into account, including consumer- and weather-induced uncertainty, computational constraints, and adversarial data manipulations. On that basis, recommendations on data and modeling strategies for ML-based forecasting enabling robust and near-optimal price-based flexibility realization are presented. Motivated by the demonstrated impact of adversarial manipulations of flexibility realization, concepts for data-driven cyber-physical attack and fault identification for DERs are investigated in Section 3.2. This includes a systematic evaluation of cyber-physical event identification applying supervised ML based on insights from **Paper C**, and presentation of the cyber-physical event reasoning system CyPhERS, which is introduced and demonstrated in **Paper D** and **Paper E**.

Chapter 4 summarizes key findings on research topic 3 (**RQ3**). Based on the insights provided by **Paper F**, the impact of local flexibility on accuracy and uncertainty of data-driven LV state estimation is empirically assessed in Section 4.1, which includes provision of data and modeling recommendations for retaining accurate and reliable estimations under high share of flexible resources. As a potential use case of such LV state estimations, a flexibility activation identification concept for improved DSO awareness is presented in Section 4.2 on the basis of content from **Paper G**.

Chapter 5 concludes the thesis by answering the defined research questions, discussing limitations, and providing opportunities for future research directions.

In order to maintain consistency throughout this thesis, the notation has been partially adjusted compared to the original formulations in the scientific publications.

1.4 List of publications

The seven publications **Paper A**–**Paper G**, which constitute the backbone of this thesis, are listed below. The individual articles are collected in Part II of the thesis under the Chapters A–G. The additional publications [Pub1]–[Pub5] and technical reports [Rep1]–[Rep3] generated during the Ph.D. project but not directly related to the main objective of this thesis are listed beneath the thesis-relevant papers.

Publications included in the thesis

Paper A: **Müller, N.**, K. Heussen, Z. Afzal, M. Ekstedt, and P. Eliasson, "Threat Scenarios and Monitoring Requirements for Cyber-Physical Systems of Flexibility Markets," in *Proceedings of the 2022 IEEE PES Generation, Transmission and Distribution Conference and Exposition – Latin America (IEEE PES GTD Latin America)*, La Paz, 2022, pp. 1-6, doi: 10.1109/IEEEPESGTDLatinAmeri53482.2022.10038290.

Paper B: **Müller, N.**, M. Marinelli, K. Heussen, and C. Ziras, "On the trade-off between profitability, complexity and security of forecasting-based optimization in residential energy management systems," in *Sustainable Energy, Grids and Networks*, vol. 34, 2023, doi: 10.1016/j.segan.2023.101033.

Paper C: **Müller, N.**, C. Ziras, and K. Heussen, "Assessment of Cyber-Physical Intrusion Detection and Classification for Industrial Control Systems," in *Proceedings of the 2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Singapore, 2022, pp. 432-438, doi: 10.1109/SmartGridComm52983.2022.9961010.

Paper D: **Müller, N.**, K. Bao, J. Matthes, and K. Heussen, "CyPhERS: A Cyber-Physical Event Reasoning System providing real-time situational awareness for attack and fault response," in *Computers in Industry*, vol. 151, 2023, doi: 10.1016/j.compind.2023.103982.

Paper E: **Müller, N.**, K. Bao, and K. Heussen, "Cyber-physical event reasoning for distributed energy resources on the case of a PV-battery system," under review at *Sustainable Energy, Grids and Networks*.

Paper F: **Müller, N.**, S. Chevalier, C. Heinrich, K. Heussen, and C. Ziras, "Uncertainty quantification in LV state estimation under high shares of flexible resources," in *Electric Power Systems Research*, vol. 212, 2022, doi: 10.1016/j.epsr.2022.108479.

Paper G: **Müller, N.**, C. Heinrich, K. Heussen, and H. W. Bindner, "Unsupervised detection and open-set classification of fast-ramped flexibility activation events," in *Applied Energy*, vol. 312, 2022, doi: 10.1016/j.apenergy.2022.118647.

Other publications (Pub) not included in the thesis

[Pub1] Herz, G., **N. Müller**, E. Jacobasch, A. Redenius, V. Hille, E. Reichelt, and M. Jahn, "Technical and economic prospects for electrolysis-based direct reduction processes," in *Proceedings of the 5th European Steel Technology and Application Days, ESTAD 2021*, Stockholm, 2021.

[Pub2] Herz, G., C. Rix, E. Jacobasch, **N. Müller**, E. Reichelt, M. Jahn, and A. Michaelis, "Economic assessment of Power-to-Liquid processes – Influence of electrolysis technology and operating conditions," in *Applied Energy*, vol. 292, 2021, doi: 10.1016/j.apenergy.2021.116655.

[Pub3] **Müller, N.**, G. Herz, E. Reichelt, M. Jahn, and A. Michaelis, "Assessment of fossil-free steelmaking based on direct reduction applying high-temperature electrolysis," in *Cleaner Engineering and Technology*, vol. 4, 2021, doi: 10.1016/j.clet.2021.100158.

[Pub4] Jacobasch, E., G. Herz, C. Rix, **N. Müller**, E. Reichelt, M. Jahn, and A. Michaelis, "Economic evaluation of low-carbon steelmaking via coupling of electrolysis and direct reduction," in *Journal of Cleaner Production*, vol. 328, 2021, doi: 10.1016/j.jclepro.2021.129502.

[Pub5] Kraft, O., O. Pohl, U. Jäger, K. Heussen, **N. Müller**, Z. Afzal, M. Ekstedt, H. Farahmand, D. Ivanko, A. Singh, S. Leksawat, and A. Kubis, "Development and Implementation of a Holistic Flexibility Market Architecture," in *Proceedings of the 2022 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, New Orleans, 2022, pp. 1-5, doi: 10.1109/ISGT50606.2022.9817470.

Published reports (Rep) during the Ph.D. studies

[Rep1] **Müller, N.**, K. Heussen, Z. Afzal, M. Ekstedt, and P. Eliasson. "HONOR D6.1 Conceptual model of data streams, detection and verification requirements." Tech. Report, Technical University of Denmark (2021).

[Rep2] Afzal, Z., M. Ekstedt, P. Eliasson, **N. Müller**, and K. Heussen. "HONOR D7.1 Preliminary cyber security assessment of the HONOR flexibility market." Tech. Report, Technical University of Denmark (2021).

[Rep3] Heussen, K., O. Kraft, and **N. Müller**. "HONOR D3.3 Interaction scenarios for flexibility trading, management, control and verification." Tech. Report, Technical University of Denmark (2023).

CHAPTER 2

Operational challenges of local flexibility and the related digitalization

This chapter addresses research topic 1 (**RQ1**) by summarizing main insights of **Paper A** regarding operational challenges for DERs and distribution grids in the context of local flexibility, with a focus on the associated digitalization. Section 2.1 provides a basic description of local flexibility, which covers existing realization mechanisms and technical requirements. Thereafter, Sections 2.2–2.5 address identified challenges, which includes the description of their background and possible impacts, as well as the derivation of resulting operational awareness requirements and suggestion of possible approaches applying predictive analytics. The presented insights motivate the subsequent contributions of **Paper B**–**Paper G** to research topic 2 (Chapter 3) and 3 (Chapter 4). Finally, Section 2.6 concludes the chapter.

2.1 Definition of local flexibility

Flexibility constitutes a somewhat loose term as it covers different aspects of modern power systems. Consequently, several definitions exist with different nuances [3]. A rather general definition is provided by the power sector association Eurelectric, which defines flexibility as "*[…] the modification of generation injection and/or consumption patterns in reaction to an external signal (price signal or activation) in order to provide a service within the energy system.*" [4]. The further specification to *local* flexibility accounts for limitation to service provision by DERs connected to distribution grids. Local flexibility can be largely categorized into price- and market-based mechanisms.

Price-based local flexibility is a mechanism in which electricity prices are dynamically adjusted based on conditions of generation and demand, reflecting the changing costs of generation and grid conditions. It allows DER operators and consumers to respond to price signals and schedule their electricity usage accordingly. When demand for electricity, and therefore the price, is high, consumers have an incentive to reduce or shift their consumption, and vice versa. Various pricing schemes exist, such as day-ahead, time-of-use, and real-time pricing. By matching demand with available generation and reducing load concurrency, price-based flexibility can alleviate stress on the grid, and facilitate integration of intermittent renewable energy sources.

Market-based local flexibility involves creating a marketplace where various stakeholders can buy and sell different flexibility services within a geographically restricted area [5]. A typical setup for such LFMs includes a DSO, a balance responsible party (BRP), multiple aggregators, and a market operator. In this structure, DSOs and BRPs act as buyers of flexibility, while aggregators constitute sellers. Aggregators pool and manage numerous small flexible assets in a portfolio, enabling end users to participate in LFMs. DSOs procure flexibility services to address operational needs within the distribution grid, such as congestion management and voltage control. BRPs, on the other hand, purchase flexibility to optimize their asset portfolios. Aggregators generate profits by scheduling the power demand of their portfolio in a way that flexibility service contracts are fulfilled. Owners of flexible assets earn profits by participating in the aggregator's portfolio.

Figure 2.1 visualizes a DER's flexibility activation by comparing its routine operation with theoretical optimal, scheduled and actual flexibility realization. For price-based mechanisms, optimal flexibility realization refers to flexible DER operation which results in minimum electricity costs (cost-optimal). In the context of market-based flexibility, a flexibility realization is considered optimal if it, as part of the whole aggregator portfolio, results in fulfillment of a DSO-requested service with minimum resources (request-optimal). Both the flexibility scheduling and actual realization can be subject to inaccuracies as further described in Section 2.2.

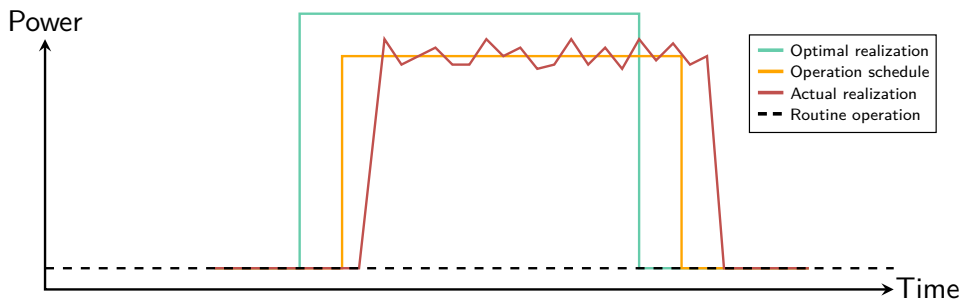


Figure 2.1: Visualization of a DER's flexibility activation together with the associated theoretical cost- or request-optimal flexibility realization, and operation schedule.

For the activation of flexibility, certain technical requirements must be met. The coordination and exchange of information, such as real-time prices or flexibility service schedules, among several actors necessitates a communication infrastructure. This typically includes internet connectivity of DERs and, depending on the underlying mechanism, also remote controllability. To enable appropriate reaction to schedules, control and automation systems are required, which comprise algorithms, software platforms, and computational hardware as typically provided by energy management systems (EMSs). This further includes data analytics, such as load and generation forecasts, for assessing or scheduling flexibility.

2.2 Sub-optimal flexibility realization

This section describes operational challenges associated with sub-optimal flexibility realization, by providing information on background (Section 2.2.1), possible impacts (Section 2.2.2), and operational awareness requirements (Section 2.2.3) based on several threat scenarios and monitoring requirements formulated in **Paper B**.

2.2.1 Background

Sub-optimal flexibility realization of a DER can be the result of inaccurate flexibility schedule determination, or erroneous reaction to a provided schedule (see Figure 2.1). A central impact factor for flexibility scheduling is weather- and consumer-induced uncertainty and volatility which affect involved load and generation forecasts. Depending on the specific flexibility mechanism, those are required for different actors. While for price-based flexibility schedules are typically generated locally based on residential load and generation forecasts, flexibility planning under market-based mechanisms involves forecasts for service requests (e.g., DSO) and offers (aggregator). Another factor are technical limitations of DERs. Response delays may result from a resource's inherent physical restrictions, such as ramping rates, or digital processes including communication delays. Computational constraints may limit the implementation of adequate forecasting, optimization and control algorithms required for both flexibility scheduling and realization. In case of residential DERs, the translation of requested power changes to actual actuator setpoints, such as room temperature for heat pumps, may entail imprecise schedule realization. Another aspect is manipulation of the flexibility scheduling and realization process. DER owners or operators may impair provision of a flexibility service by modifying setpoints either intentionally (gaming) or unintentionally (e.g., changed comfort requirements in case of residential DERs). Manipulations may also have an adversarial background, as further described in Section 2.3.

2.2.2 Possible impacts

From the perspective of individual flexibility asset owners, sub-optimal flexibility realization primarily is of financial concern. In price-based schemes, lower cost savings occur as the potential of following dynamic prices is not fully exploited. For market-based flexibility programs, low accuracy in the provision of flexibility may lowers revenue for participation in an aggregator's portfolio or prohibit admission as flexible resource.

From a grid operation point of view, sub-optimal flexibility realization of a fleet of DERs must be accounted for either through additional means of flexibility or grid extensions. While the former primarily entails economic impact in the form of higher grid operation costs, the latter additionally entails environmental impact

due to higher resource demand and potential slowing down integration of fluctuating renewable generation.

To understand the significance and scope of sub-optimal flexibility realization, economic impact on the level of asset owners and DSOs should be evaluated, taking previously identified influencing factors such as stochasticity, computational limitations and manipulations into consideration.

2.2.3 Operational awareness requirements

Minimizing possible impact of sub-optimal flexibility has at least two angles: improvement of the scheduling process, and better handling of less-reliable flexibility services. Thus, operational awareness requirements include aspects of both prediction of future conditions and real-time monitoring.

For improving scheduling, strategies for the involved forecasts should take previously identified influencing factors into account. This includes accuracy under high volatility, computational efficiency, and robustness against manipulations. Literature has shown that ML can improve forecasting accuracy [6]. An important factor is the simple integration of calendaric and meteorological information, which, for example, facilitates predicting complex load patterns by taking the day of the week into account. Nevertheless, higher computational complexity over traditional statistical methods, and the additional paths for manipulation resulting from dependency on additional and often external data sources can be limiting factors.

While accurate, efficient and robust forecasts can facilitate optimizing flexibility scheduling, a strategy for better handling of sub-optimal flexibility realization is accurate and reliable LV state estimation, facilitating real-time monitoring of flexibility activations. In the case of market-based flexibility, monitoring requested services supports DSOs in responding to possible deviations from the contracted flexibility activation with other operational measures. Being able to handle less reliable services possibly enables exploiting additional flexibility potential in the grid, eventually reducing grid extensions. LV state estimation is complicated by the limited number of measurements, as traditional state estimation techniques cannot be applied to such underdetermined systems [7]. In contrast, data-driven techniques do not dependent on widespread real-time measurements as they can leverage historical data from smart meters and other sources [8].

2.3 DER cyber vulnerability

In this section, operational challenges associated with cyber vulnerabilities of DERs are described, providing information on the background (Section 2.3.1), potential impacts (Section 2.3.2), and requirements for operational awareness (Section 2.3.3) based on several threat scenarios and monitoring requirements formulated in **Paper B**.

2.3.1 Background

Local flexibility relies on ICT infrastructure which allows to schedule, monitor, automate, control, coordinate, and verify flexible operation of DERs. This typically involves connection to public networks for receiving data such as price signals, operation schedules, and meteorological data for load and generation forecasting, as well as sending activation confirmations or other information. Certain programs also rely on remote control capabilities, for example to enable aggregators controlling a portfolio of flexible assets in a coordinated manner. In this environment, cyber criminals can exploit security weaknesses of a DER, including insecure communication protocols, weak encryption, and lack of authentication mechanisms to manipulate, inject, disrupt or steal data, and launch malicious control commands. Multiple DERs may also be attacked in a coordinated manner. One option is the misuse of the remote control permission and capability of aggregators to simultaneously control a portfolio of flexible assets. Another strategy is to exploit common insecure user behavior of DER owners, including the use of default passwords and outdated firmware. A prominent example is the Mirai botnet attack from 2016, which controlled over 600,000 weakly protected internet of things (IoT) devices at its peak [9]. Security experts from Kaspersky ICS CERT estimate that in Europe small- to medium-sized solar and wind parks of approximately 2.8 GW are publicly accessible and remotely controllable due to lack of any security measures, as demonstrated in [10].

2.3.2 Possible impacts

Depending on the objective and skill level of cyber criminals, attacks may target individual DERs and their owners or a fleet of those. On the level of individual assets, attackers may exploit the ICT infrastructure and possible security weaknesses to steal sensitive data for blackmailing or identity theft. Criminals may also target disruption of the physical DER operation, for example, to increase energy consumption or impair participation in price- or market-based flexibility mechanisms, eventually resulting in financial loss.

Attackers of higher skill levels, such as state-sponsored actors, may also launch coordinated cyber-physical attacks against a fleet of DERs to induce load steps or oscillation [11], or to manipulate a scheduled flexibility service with the aim of triggering grid protection mechanisms and causing disconnection of customers.

2.3.3 Operational awareness requirements

The cyber vulnerability of DERs introduces a new dimension of possible event and failure causes, both on the level of individual assets and distribution grid operation. To date, cyber security and process monitoring is typically conducted in isolated silos [12]. In such architectures, effective identification of event root causes and impacts is challenging, which potentially entails delayed and less-informed incident response.

Consequently, an emerging awareness requirement for secure operation of DERs and distribution grids is seen in the integrated monitoring of digital and physical processes and systems. In this context, ML can be beneficial as it has the potential to automate the evaluation of large and heterogeneous cyber-physical data, while avoiding need for manual development of complex cyber-physical models.

2.4 Flexibility-induced stochasticity

This section describes operational challenges of flexibility-induced variability and stochasticity, presenting information on the background (Section 2.4.1), possible impacts (Section 2.4.2), and operational awareness requirements (Section 2.4.3) based on threat scenarios and monitoring requirements formulated in **Paper B**.

2.4.1 Background

The integration of DERs with ICT infrastructure paves the way for the implementation of a broad range of mechanisms for exploiting the flexible operation potential of varying assets such as EVs, heat pumps, and batteries. Asset owners may follow diverse price signals, be part of different aggregator portfolios, pursue other strategies such as maximizing self-consumption targets, or participate in system-wide ancillary services. Given the multitude of possible control objectives, flexible assets deviate from routine operation in versatile manners, which potentially introduces variability and stochasticity to distribution grids.

2.4.2 Possible impacts

The potentially introduced volatility and randomness may impair the performance of tools for monitoring of distribution grids such as LV state estimation and fault localization. While data-driven state estimation approaches can handle low metering device coverage by leveraging historical measurements, the increased load stochasticity and variability may break or complicate their input-output correlation, which ultimately results in inaccurate and less reliable estimations. Incorporating such estimates into distribution system operation potentially entails poor or misinformed control actions, which, in the worst case, can entail local disconnection of customers. Thus, the potentially reduced observability must be accounted for either by widespread installation of metering devices, or extension of grid infrastructure, which both are associated with considerable investments.

To understand the significance of the problem, the impact of flexibility activations on LV state estimation should be evaluated.

2.4.3 Operational awareness requirements

To reduce investments for grid extension or large-scale metering device roll-out, and facilitate reliable grid operation, an emerging requirement is seen in LV state estimation that is robust to local flexibility. Possible strategies include the use of probabilistic data-driven models under incorporation of larger data histories and new information sources. Increased load variability may be compensated by training models on more historical data. On the other hand, stochasticity may partly be compensated by incorporating new data sources such as price-signals and schedules of flexibility portfolios. However, randomness cannot be entirely eliminated, as complete knowledge about flexibility mechanisms and involved processes is hardly feasible. Thus, models should be able to quantify non-avoidable uncertainties in the estimates to enable DSOs to conduct cautious and robust decision-making.

2.5 DSO-unaware flexibility activations

In this section, operational challenges related to flexibility activations without active involvement of DSOs are described, offering information on the background (Section 2.5.1), potential impacts (Section 2.5.2), and requirements for operational awareness (Section 2.5.3) based on threat scenarios and monitoring requirements formulated in **Paper B**.

2.5.1 Background

The integration of DERs with ICT infrastructure enables different flexibility mechanisms and control objectives which do not explicitly involve DSOs. Consequently, flexibility of a fleet of DERs may be activated simultaneously, triggering a significant load modification in the distribution grid without the DSO being aware of it.

2.5.2 Possible impacts

Simultaneous flexibility activation of a fleet of DERs may provoke sudden load steps or local congestion. In case that protection mechanisms are triggered, customers may temporally be disconnected from supply. One example is the concurrent reaction of smart EV chargers to the same low and transregional day-ahead price signal. Such flexibility activations may also interfere with DSO-activated flexibility services. For example, a load reduction service for mitigating an evening peak may overlap with high price signals, making procurement of the service obsolete. Apart from the associated financial loss, the excessive load decrease may provoke over-voltages in extreme cases.

2.5.3 Operational awareness requirements

The large number of nodes in distribution grids render it impractical to manually monitor the network for sudden load changes which potentially lead to local congestion or voltage limit violations. Thus, an automated early-warning system for flexibility activations is seen as an awareness requirement for distribution grid operation in a scenario of high shares of flexible resources. The problem is complicated by the fact that DSOs have no or only limited access to data of individual flexible assets. Moreover, historical observations of flexibility activations and other load-altering events are likely to be scarce. In this context, data-driven anomaly detection and open-set classification are promising techniques. These techniques potentially allow to identify flexibility activations in available load measurements or estimations of specific nodes in the grid, without, or with limited, need for data of individual DERs and historical observations of flexibility activations and other load-altering events. Moreover, these techniques do not depend grid models.

2.6 Summary and reflection

This chapter describes possible operational challenges of local flexibility and the associated digitalization. On that basis, requirements for new or improved operational awareness tools for DERs and distribution grids are derived, including aspects of real-time monitoring and event detection, and predicting future operation.

The effective and robust scheduling of flexibility requires accurate, resource-efficient and cyber-secure forecasts. While ML has the potential to provide forecasts of high accuracy under weather- and consumer-induced variability by exploiting new information sources and growing data histories, the impact of computational restrictions and adversarial manipulations should be evaluated. On this basis, recommendations for optimal forecasting strategies can be derived. For this reason, a systematic comparison of a multitude of forecasting strategies for flexibility schedule optimization is conducted in **Paper B** and summarized in Section 3.1, allowing to assess the significance of different influencing factors, and conclude on data and modeling strategies for optimized price-based flexibility realization.

The digitalization of DER and distribution grid operation is paving the way for a broad spectrum of flexibility mechanisms which promise operational, economic and environmental benefits to several stakeholders. Along with these advancements, the increasing load activity and emerging cyber threats require new or improved real-time monitoring and event detection solutions for DERs and distribution grids. The growing load activity may translates to variability and stochasticity of LV states. While data-driven techniques promise to enable state estimation in underdetermined distribution grids by leveraging offline data, local flexibility may impairs their performance by affecting the input-output correlation. Thus, the impact of frequent flexibility activations on data-driven LV state estimation should be evaluated, and data and modeling strategies for enabling accurate and reliable estimations under

large shares of flexible resources derived. For this purpose, a systematic evaluation of the accuracy and uncertainty of data-driven LV state estimation under several flexibility scenarios is conducted in **Paper F** and summarized in Section 4.1. Further real-time monitoring requirements result from new event and failure causes, including DSO-unaware flexibility activations, and cyber-physical attacks. In both cases, techniques from ML-based predictive analytics offer promising solutions. Data-driven anomaly detection and open-set classification possibly enables to identify flexibility activations based on data available to DSOs and without need for grid models. In this regard, a flexibility activation identification concept is introduced and demonstrated in **Paper G**, which is summarized in Section 4.2. Moreover, ML can automate the processing of large and heterogeneous cyber-physical data for effective real-time attack and fault identification, while avoiding manual development of complex cyber-physical models. In this perspective, a systematic assessment of cyber-physical event identification applying supervised ML is conducted in **Paper C** and summarized in Section 3.2.2. Furthermore, the cyber-physical event reasoning system CyPhERS is proposed and demonstrated in **Paper D** and **Paper E** (see Section 3.2.3), which leverages cyber-physical monitoring for DERs while meeting practical requirements, such as independence from scarce historical event observations and human-verifiable event predictions.

CHAPTER 3

Predictive analytics for efficient, cost-optimal and secure use of DERs as flexible assets

This chapter is concerned with presenting central findings of **Paper B** to **Paper E** on research topic 2 (**RQ2**). In this context, previously identified operational awareness requirements on DER level are addressed, which are related to sub-optimal flexibility realization and DER cyber vulnerability. Section 3.1 evaluates the economic impact of sub-optimal price-based flexibility realization on prosumers based on results from **Paper B**, taking consumer- and weather-induced volatility and stochasticity, computational constraints, and malicious data manipulation into consideration. On that basis, recommendations on accurate, effective and robust forecasting strategies for optimized flexibility realization are provided. Section 3.2 addresses cyber-physical attack and fault identification concepts for DERs, which includes a systematic assessment of the cyber-physical approach to event identification based on **Paper C**, and description of the new cyber-physical event reasoning system CyPhERS introduced and demonstrated in **Paper D** and **Paper E**. Finally, Section 3.3 summarized the chapter and reflects on key outcomes.

3.1 Forecasting strategies for optimized price-based flexibility realization

This section empirically evaluates the economic impact of several influencing factors on price-based flexibility realization, and derives recommendations on data and modeling strategies for forecasting-based flexibility schedule optimization. Section 3.1.1 presents an overview of related works and the contribution of **Paper B**. In Section 3.1.2, background information on the study case and applied methodology are provided. Section 3.1.3 evaluates several forecasting scenarios to gain insights on the influence of weather- and consumer-induced uncertainty, computational constraints and data manipulations on the financial reward of price-based flexibility for prosumers. On that basis, Section 3.1.4 provides recommendations on forecasting strategies for optimized price-based flexibility realization.

3.1.1 Related works and contribution

The literature on concepts for improved price-based flexibility realization is rich [13]. Many works, e.g., [14], demonstrate the financial advantage of applying forecasting-based schedule optimization in comparison to rule-based or heuristic approaches. Some works further evaluate how forecasting accuracy affects cost savings, for example, by artificially adding noise to forecasts [15] or by comparing a limited selection of models [16]. However, a systematic evaluation of forecasting scenarios which allows to 1) understand the impact of different influencing factors, and 2) derive practical recommendations on best practices is missing. Moreover, many existing works are subject to practical limitations such as short evaluation periods and low data resolution, which reduces the significance of the results. Out of this motivation, a systematic comparison of a large set of forecasting cases is conducted in **Paper B** based on more than one year of 5-minutely data of two real prosumers, allowing to derive data and modeling strategies for optimized price-based flexibility realization. **Paper B** also compares forecasting-based schedule optimization with rule-based battery scheduling and demonstrates the advantage of the former. However, since this is in line with findings of existing works, the following summary of **Paper B** focuses on evaluating strategies within forecasting-based optimization.

3.1.2 Background

In Section 3.1.2.1, the study case is introduced. Thereafter, a description of the applied methodology is provided in Section 3.1.2.2.

3.1.2.1 Study case

Prosumer specification The study is based on two residential prosumers, each equipped with rooftop PV, a stationary battery storage, and an EMS (see Figure 3.1). The location of the two prosumers is Roskilde, Denmark. Both prosumers follow the

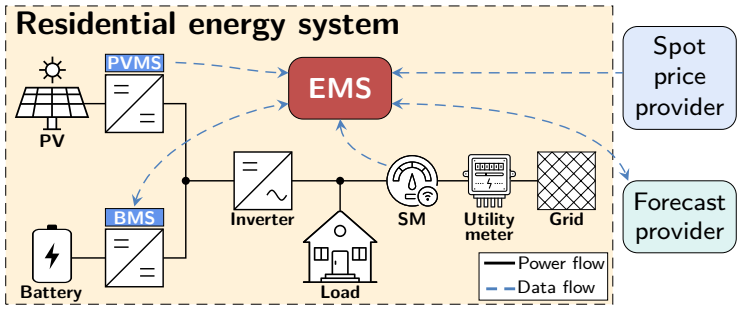


Figure 3.1: Illustration of the two studied prosumers' residential energy system.
Source: Illustration adapted from **Paper B**.

objective of scheduling the battery based on spot prices to minimize energy costs. The PV systems have a capacity of $6\,\mathrm{kW_p}$ (prosumer 1) and $5\,\mathrm{kW_p}$ (prosumer 2). In both cases, the battery storage has an efficiency of $\eta = 0.95$, and an upper and lower state of charge (SOC) limit of $\overline{soc} = 8\,\mathrm{kWh}$ and $\underline{soc} = 0.8\,\mathrm{kWh}$, respectively. The EMS receives power measurements of the grid connection point from the smart meter (SM). The battery management system (BMS) and photovoltaic management system (PVMS) provide PV and battery measurements, respectively. Load consumption is derived from these measurements. All measurements are provided to the EMS as 5-minute average values. As foundation for the optimization of the battery schedule, the EMS receives spot prices and weather forecasts from third parties. The hourly spot prices are provided once per day at 13:00 for the following 24 hours. Weather forecasts comprise hourly averages of the global horizontal irradiance (GHI) and cloud opacity, and are provided each hour of the day for the next 36 hours.

The inclusion of two prosumers is justified by the differences in their production and consumption levels and patterns, supporting generality of the findings. Prosumer 1 has a higher production level, as the greater nominal power of the PV system suggests. Variations in load consumption primarily arise from EV charging in the case of prosumer 1, resulting in higher load levels and less predictable patterns compared to prosumer 2. Furthermore, prosumer 1 acquired a second EV in 2022, leading to a sudden change of the load level and patterns during the data collection period. Lastly, prosumer 1 has a higher self-consumption as a result of actively aligning EV charging with PV production. Considering these characteristics, prosumer 2 can be regarded as a more traditional passive consumer of electricity, while prosumer 1 represents a case of proactive load optimization reflecting future trends.

Data specification Different lengths of historical data are available for the two prosumers. For prosumer 1, approximately three years of data are available, starting from September 1, 2019, and ending on October 30, 2022. The dataset of prosumer 2 comprises 14.5 months of data, spanning from August 15, 2021, to October 30, 2022. In both cases, the last 14 months of data, specifically from September 1, 2021, to October 30, 2022, are reserved for evaluation purposes.

3.1.2.2 Optimization and forecasting methodology

A rolling-horizon optimization of the battery schedule based on spot prices is considered for cost-optimal usage of the residential PV-battery system flexibility. At the center of the schedule optimization are forecasts of the prosumers' PV generation and load consumption, and spot prices.

Schedule optimization The battery schedule optimization problem is formulated according to

$$\min_{\mathcal{W}} \quad \sum_{\tau \in \mathcal{T}} \left[(\lambda_\tau | \hat{\lambda}_\tau + f_\tau) p_\tau^{\mathrm{b}} - \lambda_\tau | \hat{\lambda}_\tau p_\tau^{\mathrm{s}} \right] \Delta T, \tag{3.1a}$$

$$\text{s.t.} \qquad 0 \leq p_\tau^{\mathrm{b}}, \quad 0 \leq p_\tau^{\mathrm{s}} \tag{3.1b}$$

$$0 \leq p_\tau^{\mathrm{c}} \leq \delta_\tau \overline{p}_{\mathrm{inv}} \tag{3.1c}$$

$$0 \leq p_\tau^{\mathrm{d}} \leq (1-\delta_\tau) \overline{p}_{\mathrm{inv}} \tag{3.1d}$$

$$\hat{p}_\tau^{\mathrm{PV}} + p_\tau^{\mathrm{d}} \leq \overline{p}_{\mathrm{inv}} \tag{3.1e}$$

$$p_\tau^{\mathrm{b}} - p_\tau^{\mathrm{s}} = \hat{p}_\tau^{\mathrm{L}} - \hat{p}_\tau^{\mathrm{PV}} + p_\tau^{\mathrm{c}} - p_\tau^{\mathrm{d}} \tag{3.1f}$$

$$soc_{\tau+1} = soc_\tau + \left[p_{\tau+1}^{\mathrm{c}}\ \eta + p_{\tau+1}^{\mathrm{d}}\ /\eta\right] \Delta T \tag{3.1g}$$

$$soc_0 = soc^{\mathrm{s}}, \quad soc_{w_{\mathrm{opt}}} = soc^{\mathrm{e}} \tag{3.1h}$$

$$\underline{soc} \leq soc_\tau \leq \overline{soc}. \tag{3.1i}$$

$\mathcal{W}$ is the set of decision variables. p_τ^{L}, p_τ^{PV}, p_τ^{b}, p_τ^{s}, p_τ^{c}, and p_τ^{d} represent the 5-minute average consumption, PV generation, power bought from the grid, power sold to the grid, battery charging power, and battery discharging power at time step τ, respectively. The inverter capacity limit is denoted as $\overline{p}_{\mathrm{inv}}$. ΔT is the normalized time step duration, which is $\Delta T = 1/12$ for the considered 5-minute time steps. The problem is solved periodically at each time step τ for a look-ahead horizon of size w_{opt}. The set of time steps that comprise the look-ahead horizon is denoted by $\mathcal{T} = \{1, 2, \ldots, w_{\mathrm{opt}}\}$. Spot prices Λ and imposed fees and taxes F have an hourly resolution. Thus, constant prices and fees are considered for each 5-minute time step within the respective hour, represented by λ_τ and f_τ. Values of p_τ^{PV}, p_τ^{L} and λ_τ within the look-ahead horizon that are unknown at the time of calculation are provided as forecasts. These are given as hourly average values denoted by $\hat{P}_h^{\mathrm{PV}}$, $\hat{P}_h^{\mathrm{L}}$ and $\hat{\Lambda}_h$ for a 1-hour time step h. The corresponding constant 5-minute values within h are expressed as $\hat{p}_\tau^{\mathrm{PV}}$, $\hat{p}_\tau^{\mathrm{L}}$ and $\hat{\lambda}_\tau$. The $\lambda_\tau|\hat{\lambda}_\tau$ notation in (3.1a) indicates that, depending on the step τ within $\mathcal{T}$, either true or predicted prices are applied. The notation is neglected for $\hat{p}_\tau^{\mathrm{PV}}$ and $\hat{p}_\tau^{\mathrm{L}}$ for the sake of readability. PV and load forecasts are updated every hour and provided for a horizon of size $w_{\mathrm{pr}} = 36$, which corresponds to an optimization horizon of $w_{\mathrm{opt}} = 432$. The horizon was found to be sufficient to approximate the performance for $w_{\mathrm{pr}} \to \infty$. Price forecasts are performed once per day at 13:00 and extend the true prices by another day. δ_τ represents the battery charging/discharging status and constitutes a binary decision variable. The initial and final SOC values are denoted by soc^{s} and soc^{e}. The associated constraint stated in (3.1h) requires that the battery maintains a SOC of 50 % at the end of the optimization period. All constraints (3.1b)-(3.1i) must hold $\forall\ \tau \in \mathcal{T}$ with the exception of (3.1g) and (3.1h), which are imposed $\forall\ \tau \in \mathcal{T} \backslash w_{\mathrm{opt}}$. Upon implementing the battery schedule obtained at time step τ, a subsequent optimization problem is solved at the next step, applying latest measurements and forecasts. The open-source optimization solver *GLPK* [17] and modeling language *CVXPY* [18] are employed.

PV, load and price forecasts Gradient-boosted decision trees (GBDT) is considered as ML-based forecasting model. GBDT [19] is a frequently used technique known for its efficiency, interpretability and state-of-the-art accuracy, evident from consis-

tent success in data mining and time series forecasting competitions [6]. It combines the predictions of multiple decision trees, forming a set of weak learners. These trees are connected in series, with each learner aiming to minimize the difference between the actual values and the predictions made by the previous tree. The combination of high accuracy and efficiency makes GBDT a good candidate for residential EMSs.

PV and load forecasts are generated at each time step h for a prediction horizon of w_{pr} steps based on lag values $P_h^{\mathrm{PV|L}}, ..., P_{h-w_{\mathrm{hist}}}^{\mathrm{PV|L}}$ and covariates $\boldsymbol{Z}_{h+1}^{\mathrm{PV|L}}, ..., \boldsymbol{Z}_{h+w_{\mathrm{pr}}}^{\mathrm{PV|L}}$ according to

$$\hat{P}_{h+1}^{\mathrm{PV}}, ..., \hat{P}_{h+w_{\mathrm{pr}}}^{\mathrm{PV}} = \Phi\left(P_h^{\mathrm{PV}}, ..., P_{h-w_{\mathrm{hist}}}^{\mathrm{PV}}, \boldsymbol{Z}_{h+1}^{\mathrm{PV}}, ..., \boldsymbol{Z}_{h+w_{\mathrm{pr}}}^{\mathrm{PV}}\right) \tag{3.2}$$

and

$$\hat{P}_{h+1}^{\mathrm{L}}, ..., \hat{P}_{h+w_{\mathrm{pr}}}^{\mathrm{L}} = \Phi\left(P_h^{\mathrm{L}}, ..., P_{h-w_{\mathrm{hist}}}^{\mathrm{L}}, \boldsymbol{Z}_{h+1}^{\mathrm{L}}, ..., \boldsymbol{Z}_{h+w_{\mathrm{pr}}}^{\mathrm{L}}\right), \tag{3.3}$$

where w_{hist} is the length of history window. Spot price forecasts are produced once a day at 13:00 using the lag values $\Lambda_{h+35}, ..., \Lambda_{h-w_{\mathrm{hist}}}$ and covariates $\boldsymbol{Z}_{h+36}^{\Lambda}, ..., \boldsymbol{Z}_{h+60}^{\Lambda}$ according to

$$\hat{\Lambda}_{h+36}, ..., \hat{\Lambda}_{h+60} = \Phi\left(\Lambda_{h+35}, ..., \Lambda_{h-w_{\mathrm{hist}}}, \boldsymbol{Z}_{h+36}^{\Lambda}, ..., \boldsymbol{Z}_{h+60}^{\Lambda}\right). \tag{3.4}$$

The considered covariates are listed in Table 3.1. For PV, load and price forecasting, $\boldsymbol{Z}^{\mathrm{PV}} = \{H, \widehat{GHI}, \widehat{O}\}$, $\boldsymbol{Z}^{\mathrm{L}} = \{H, D, A, \widehat{GHI}, \widehat{O}\}$, and $\boldsymbol{Z}^{\Lambda} = \{H, D\}$, respectively, are applied.

Table 3.1: Covariates considered for GBDT-based forecasting used for battery schedule optimization in the two studied prosumer cases.
Source: Table adapted from **Paper B**.

Covariate	**Sign**	**Value range**
Hour of the day	H	$\{H \in \mathbb{N} \mid H = [0, ..., 23]\}$
Day of the week	D	$\{D \in \mathbb{N} \mid D = [0, ..., 6]\}$
Prosumer absence	A	$\{A \in \mathbb{N} \mid A = [0, 1]\}$
GHI forecasts	$\widehat{GHI}$	$\{\widehat{GHI} \in \mathbb{R} \mid \widehat{GHI} \geq 0\}$
Cloud opacity forecasts	$\widehat{O}$	$\{\widehat{O} \in \mathbb{R} \mid \widehat{O} = [0, ..., 1]\}$

Some of the forecasting scenarios evaluated in Section 3.1.3 consider model selection by tuning hyperparameters listed in Table 3.2 applying the Bayesian optimization algorithm tree Parzen estimator (TPE) [20] on a three-fold time-series cross-validation [21]. TPE finds the optimal set of hyperparameters $\boldsymbol{\omega}_{\mathrm{opt}}$ within a predefined number of samples ($N_{\mathrm{trails}} = 2000$) from the hyperparameter space $\boldsymbol{\Omega}$ by minimizing the average root mean squared error (RMSE) over all N_{folds} folds and w_{pr} prediction steps. On the example of a PV forecaster, this can be expressed as

$$\boldsymbol{\omega}_{\mathrm{opt}}^{\mathrm{PV}} = \arg\min_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \frac{\sum_{\kappa=1}^{N_{\mathrm{folds}}} \sum_{\xi=1}^{w_{\mathrm{pr}}} \sqrt{\sum_{i=1}^{N_{\mathrm{val}}^{(\kappa)}} \frac{\left(\hat{P}_{\xi,i}^{\mathrm{PV}}(\boldsymbol{\omega}) - P_i^{\mathrm{PV}}\right)^2}{N_{\mathrm{val}}^{(\kappa)}}}}{N_{\mathrm{folds}} \cdot w_{\mathrm{pr}}}, \tag{3.5}$$

where $N_{\text{val}}^{(\kappa)}$ is the length of the validation set of the κ-th fold, $\hat{P}_{\xi,i}^{\text{PV}}(\boldsymbol{\omega})$ the ξ-steps ahead forecast for the i-th observation in the validation set of the κ-th fold based on a hyperparameter set $\boldsymbol{\omega}$, and P_i^{PV} the corresponding ground truth.

Table 3.2: Hyperparameters and search spaces considered in the selection of GBDT-based PV, load and spot price forecasting models.
Source: Table adapted from **Paper B**.

No.	Hyperparameter	Search space
1	w_{hist}	[4,...,192]
2	L1 regularization	[0,...,100]
3	Bagging fraction	[0.1,...,1]
4	Max. number of leaves in one tree	[20,...,3000]
5	Feature fraction	[0.1,...,1]
6	Max. depth of a tree	[3,...,21]
7	Number of decision trees	[100,...,10000]
8	Learning rate	[0.001,...,0.3]

In all GBDT-based forecasting scenarios evaluated in Section 3.1.3, weekly model retraining on the full available data history is considered. The GBDT models and the associated selection and retraining procedures are implemented using the open-source libraries *Darts* [22] and *Optuna* [23], respectively.

Apart from the ML-based forecasts applying GBDT, persistence forecasts are considered as those are frequently applied by studies on forecasting-based optimization of PV-battery systems [16, 24]. Moreover, a hypothetical oracle forecast which predicts ground truth values is considered to determine the theoretical optimal performance.

Price and forecast manipulation Spot prices and forecasts are at the center of cost-optimal battery scheduling. Two attack models are considered as basis for evaluating the impact of adversarial manipulations of these central information. The considered price manipulation model (see Figure 3.2) mirrors true values on their

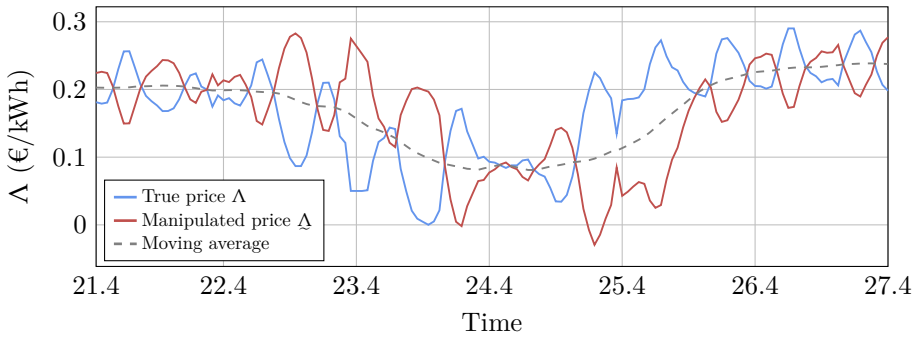


Figure 3.2: Representative excerpt of the spot price manipulation according to (3.6).
Source: Illustration adapted from **Paper B**.

moving average according to

$$\underset{\sim}{\Lambda}_h = \Lambda_h - 2 \cdot \left(\Lambda_h - \sum_{\varsigma=0}^{w_{\text{avg}}} \frac{\Lambda_{h-\varsigma}}{w_{\text{avg}}} \right), \tag{3.6}$$

where $w_{\text{avg}} = 23$. The goal is to achieve a battery behavior that is contrary to cost-optimal operation. Furthermore, the manipulation of GHI and cloud opacity forecasts according to

$$\widehat{\underset{\sim}{GHI}}_{h+\xi} = \widehat{GHI}_{h+\xi} \cdot (1 + \alpha R), \forall \xi \in [1, 2, ..., w_{\text{pr}}] \tag{3.7}$$

and

$$\widehat{\underset{\sim}{O}}_{h+\xi} = \widehat{O}_{h+\xi} \cdot (1 + \alpha R), \forall \xi \in [1, 2, ..., w_{\text{pr}}], \tag{3.8}$$

is considered (see Figure 3.3), where R is a random number drawn from the uniform distribution $R \sim U(-1, 1)$, and α the aggressiveness of the manipulation with $\alpha \in [0.2, 1, 10]$. The objective is to compromise the inputs for PV and load forecasts in order to provoke sub-optimal battery scheduling.

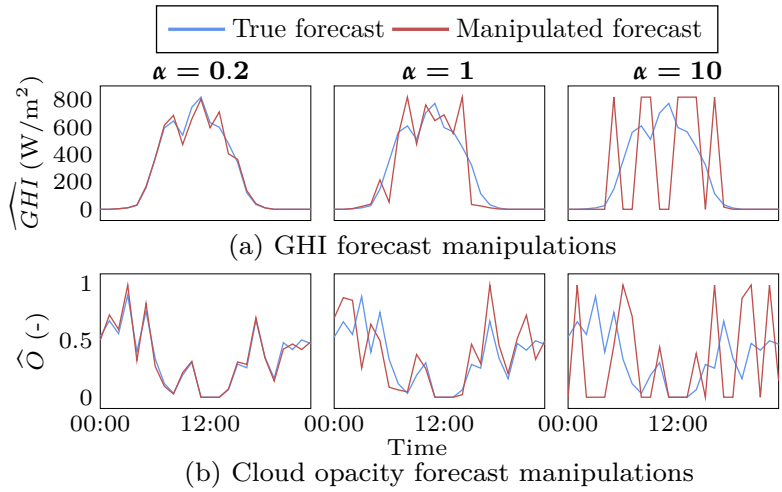


Figure 3.3: Illustration of the (a) GHI and (b) cloud opacity forecast manipulation. Source: Illustration adapted from **Paper B**.

3.1.3 Evaluation of forecasting scenarios and influencing factors

In this section, the impact of several influencing factors on the prosumers economic benefit is evaluated based on a comparison of a selection of forecasting scenarios from **Paper B**. After introducing the selected forecasting scenarios and applied economic

performance indicators in Section 3.1.3.1, different influencing factors are evaluated in Section 3.1.3.2 to Section 3.1.3.4.

3.1.3.1 Forecasting scenarios and performance indicators

Forecasting scenarios An overview of the considered scenarios is provided in Table 3.3. While $\mathrm{GBDT}_{\mathrm{opt}}$ and $\mathrm{GBDT}_{\mathrm{def}}$ consider PV, load and price forecasts based on the same model type, their computational burden significantly diverges due to differences in model selection and data history size. The hyperparameter optimization on up to two years of data in scenario $\mathrm{GBDT}_{\mathrm{opt}}$ is computationally intensive, and entails larger models. For example, the number of decision trees raises from 100 to over 7000 in most cases. While training time of the default GBDT models in scenario $\mathrm{GBDT}_{\mathrm{def}}$ is in the order of seconds to minutes on a standard laptop, the comprehensive model selection process considered in $\mathrm{GBDT}_{\mathrm{opt}}$ takes up to 24 hours on a high-performance computing (HPC) cluster [25].

Table 3.3: Description of forecasting scenarios considered for evaluating forecasting-based prosumer battery schedule optimization in price-based flexibility schemes.
Source: Table based on **Paper B**.

Scenario	Description
Oracle	Perfect knowledge on future PV production, load and prices.
$\mathrm{GBDT}_{\mathrm{opt}}$	PV, load and price forecasts applying GBDT optimized on two years[a] data history based on (3.5). With weekly retraining and all available covariates.
$\mathrm{GBDT}_{\mathrm{def}}$	PV, load and price forecasts based on GBDT with default parameters and two weeks of data history. With weekly retraining and all available covariates.
Persistence	Persistence PV, load and price forecasts.
$\mathrm{GBDT}_{\mathrm{def}}+$ input manip.	$\mathrm{GBDT}_{\mathrm{def}}$ with manipulated weather forecast inputs according to (3.7) and (3.8) and $\alpha = 10$.
$\mathrm{GBDT}_{\mathrm{def}}+$ price manip.	$\mathrm{GBDT}_{\mathrm{def}}$ with manipulated spot prices according to (3.6).

[a]For prosumer 2, re-optimization every 3 months due to shorter data history.

Economic performance indicators The prosumer energy cost under a forecasting scenario C' is determined as

$$K^{C'} = \sum_{i=1}^{N_{\mathrm{eval},\tau}} \left[p_i^{\mathrm{b},C'} (\lambda_i + f_i) - p_i^{\mathrm{s},C'} \lambda_i \right] \Delta T, \tag{3.9}$$

with $N_{\mathrm{eval},\tau}$ being the length of the 14-month evaluation period in 5-minute resolution. The economic benefit under C' is quantified as the disparity from the baseline cost in the absence of a battery (K^{base}), as outlined by

$$B^{C'} = K^{\mathrm{base}} - K^{C'}. \tag{3.10}$$

To facilitate comparison among the scenarios, their benefit is evaluated by contrasting them with the theoretical maximum. Since the maximum benefit is attained when assuming oracle forecasts, the resulting relative benefit under C' is formulated as

$$rB^{C'} = \frac{B^{C'}}{B^{\text{oracle}}}. \tag{3.11}$$

3.1.3.2 Consumer- and weather uncertainty

The relative benefits under the considered forecasting scenarios are depicted in Figure 3.4. The oracle scenario represents a case without any uncertainty about future PV production, load consumption and spot prices. In this case, the storage system enables an additional benefit of $B^{\text{oracle}} = 466\,€$ (prosumer 1) and $555\,€$ (prosumer 2) over the 14-month evaluation period. In comparison, the sophisticated GBDT_{opt} scenario achieves 90 % and 93 % of that theoretical optimum, respectively. To distinguish the influence of sub-optimal PV and load forecasts from price forecasts, a sub-scenario of GBDT_{opt} considering perfect knowledge about future spot prices is included in Figure 3.4. As can be noticed, the improvement over applying realistic price forecasts is marginal. These results indicate that inaccuracies in price forecasting are of minor relevance. Instead, consumer- and weather uncertainty and the resulting sub-optimal PV and load forecasts reduce the economic benefit of using a

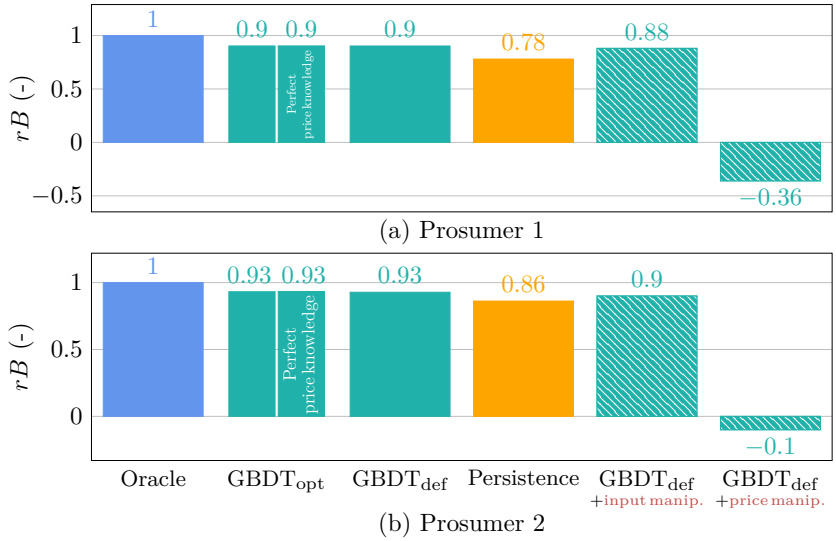


Figure 3.4: Relative economic benefit rB of price-based flexibility for the two considered prosumer cases under selected forecasting scenarios from **Paper B**.
Source: Illustration based on **Paper B**.

residential PV-battery system in a price-based flexibility scheme about 5-10 percentage points compared to the theoretical optimum, which is difficult to avoid even with sophisticated forecasting models. Figure 3.5 illustrates the sub-optimal forecasts on a representative excerpt.

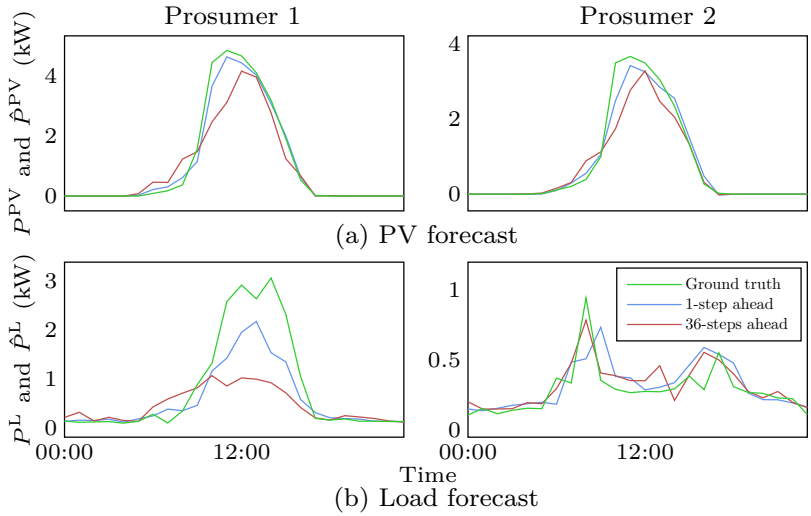


Figure 3.5: Excerpts of the residential 1- and 36-hours ahead (a) PV and (b) load forecasts and associated ground truth for the two studied prosumer cases.
Source: Illustration adapted from **Paper B**.

3.1.3.3 Computational constraints

In contrast to GBDT_{opt}, the scenarios GBDT_{def} and Persistence can both be implemented locally on simple hardware of a residential EMS. Thus, by comparing these three scenarios, insights can be gained regarding the impact of constrained computational resources on the economic benefits for prosumers and the applicability of ML-based forecasting. Figure 3.4 indicates that by applying persistence forecasts, the relative benefit drops to 78 % and 86 %, respectively. On the other hand, applying default GBDT models achieves mostly the same benefit as their significantly more complex counterparts in GBDT_{opt} for both prosumers. These results suggest that computational constraints do not affect prosumers' economic benefit of price-based flexibility, and that ML-based forecasting can be beneficial also under limited computational resources of residential EMSs. Furthermore, since GBDT_{def} in contrast to GBDT_{opt} does not rely on comprehensive data history, another finding is the applicability of ML-based forecasting also for new residential energy systems. Central factor for a good performance based on default parameters and short data history is

the use of weather forecasts as model input, and weekly retraining. The lower relative benefits under use of a default GBDT *without* retraining and/or weather inputs are quantified and included in **Paper B**.

3.1.3.4 Data manipulation

Optimizing the battery schedule depends on external data from third parties (see Figure 3.1). To understand the impact of a loss of integrity in these data streams, two data manipulation scenarios are considered. From Figure 3.4 it can be seen that a manipulation of the weather forecast input of models in GBDT_{def} according to (3.7) and (3.8) with $\alpha = 10$ only reduces rB by two (prosumer 1) and three (prosumer 2) percentage points. Consequently, the economic benefit is still higher than under usage of persistence forecasts. The manipulated inputs do not translate to significantly worse predictions as the regular retraining allows the model to react on the reduced information content in weather features by putting less weight on them. Thus, it approximates the performance of a model that does not use external weather forecasts as input, still outperforming use of persistence forecasts.

In contrast, the considered price manipulation according to (3.6) has severe impact on battery scheduling and turns economic benefits into additional energy costs (see Figure 3.4). Compared to the non-manipulated GBDT_{def} scenario, a loss of 579.32 € (prosumer 1) and 570.34 € (prosumer 2) over the 14-month evaluation period results. These results further indicate that the manipulation is successful in using the prosumers' flexibility to shift consumption to high-price periods, which potentially reinforces an already high demand, putting stress on the grid.

3.1.4 Recommendations on forecasting strategies

Although general recommendations on best principles for price-based flexibility realization require evaluation of large and versatile prosumer portfolios, the results in Section 3.1.3 allow to draw some initial conclusions as they suggest similar findings for two real prosumer cases with substantially different consumption and production behavior. Applying a default GBDT model considering weather forecasts as input and regular retraining is a promising forecasting strategy to achieve near-optimal price-based flexibility realization for a residential PV-battery system, irrespective of possible computational constraints, small data histories, and weather input manipulation. A problem that forecasting strategies cannot solve is price manipulation, which can affect both a prosumer's economics, and grid operation. In case that a fleet of DERs react on the same manipulated price signals, the potential switch from peak shaving to peak reinforcing behavior might induce local congestion or violation of voltage limits. Thus, DERs which are used as flexible resource should be equipped with advanced concepts for real-time identification of such cyber-physical attacks.

3.2 Cyber-physical event identification for DERs

This section addresses cyber-physical attack and fault identification for DERs. In Section 3.2.1, the concept of cyber-physical monitoring is introduced. A systematic assessment of cyber-physical event identification applying supervised ML is provided in Section 3.2.2 based on **Paper C**. Section 3.2.3 introduces and demonstrates the cyber-physical event reasoning system CyPhERS on the basis of **Paper D** and **Paper E**. Finally, Section 3.2.4 concludes this section.

3.2.1 The concept of cyber-physical monitoring

Exploiting DERs as flexible resources drives closed-loop integration of digital and physical processes and equipment. While the interdependency of the cyber and physical domain sets the foundation for flexibility planning, realization and verification, it introduces new and more complex events that potentially can affect DER operation. Both failures of digital networks, devices, and algorithms as well as cyber attacks can entail physical impact. Consequently, the spectrum of possible events and failures increases, and covers two fundamentally distinct domains, rendering identification of root causes more difficult.

Responding to operational incidents of one or multiple DERs is facilitated by real-time event information, such as event type, affected physical and digital devices, attacker location, and physical impact. As simultaneous misoperation of a fleet of DERs or coordinated cyber-physical attacks against those can severely impair distribution grid operation, detecting and identifying operational incidents of DERs in a timely, holistic and reliable manner is of importance both from an asset owner and DSO perspective. Monitoring concepts and systems for DERs are typically exclusive to processing data from either the cyber or physical domain [12]. While evaluating cyber network data potentially allows to distinguish several attack and network failure types, physical impacts cannot be identified. In contrast, observing physical process data provides the basis for detecting and understanding misoperation, but not the underlying attack or failure vectors. Combining information from isolated and domain-specialized monitoring systems in an ex-post scheme can be time-consuming and complex due to incompatibility of the provided information and its representation. Thus, concepts which enable joint real-time monitoring of the heterogeneous data from the cyber and physical domain of a DER should be investigated with the objective to provide valuable information for timely and appropriate attack and fault incident response.

3.2.2 Assessment of cyber-physical event identification

This section systematically evaluates cyber-physical event identification by applying techniques from supervised ML, and is based on content from **Paper C**. Sec-

tion 3.2.2.1 explains the evaluation approach. In Section 3.2.2.2, related works and the contribution are highlighted. The considered study case is introduced in Section 3.2.2.3. Section 3.2.2.4 details the applied supervised event identification pipelines, followed by their systematic evaluation in Section 3.2.2.5.

3.2.2.1 Evaluation approach

The objective of **Paper C** is to understand whether joint processing of cyber network and physical process data enables simultaneous detection and differentiation of several cyber attack and physical failure types, and improves the overall event identification performance. In this context, supervised ML models are considered a valuable tool as the same model type can be trained to explicitly predict several event classes based on different data sources, allowing for an A/B comparison between using purely cyber network and cyber-physical data.

3.2.2.2 Related works and contribution

Several existing works evaluate cyber-physical event identification applying supervised ML, for example [26–28]. While most of them indicate performance advantages of jointly processing cyber network and physical process data, they are subject to methodological shortcomings, weakening the validity of results. These include data leakage as well as limitation to binary cyber attack classification (attack vs. normal). Data leakage is introduced, for example, by shuffling and thus distributing observations of the same attack event over both training and test data. As a result, models are evaluated on the same event they are trained on, entailing an overly optimistic performance assessment. Moreover, evaluating binary attack classification provides no information on whether cyber-physical approaches allow to jointly identify and distinguish several attack and fault types. For these reasons, a systematic and methodologically sound assessment of cyber-physical event identification considering differentiation of multiple attack and fault types is conducted in **Paper C**, and presented in the following.

3.2.2.3 Study case

The availability of datasets describing multiple types of cyber attacks and physical failures on a DER based on both network and process data is scarce. While this circumstance motivated planning and conduction of own experiments (see Section 3.2.3.3), the dataset of a generic cyber-physical laboratory testbed was utilized to proceed with a general evaluation of cyber-physical event identification.

The considered testbed and the associated dataset are introduced in [29]. Figure 3.6 provides a schematic overview of the testbed. The system operates by transferring water between multiple tanks. A complete process cycle involves filling and emptying of all tanks (T1-8). The transfer of water between the tanks is realized by valves (V1-22), pumps (P1-6), and flow sensors (F1-4). A supervisory control and

data acquisition (SCADA) architecture is employed to monitor and control the process, which includes several sensors and actuators, four programmable logic controllers (PLCs), and a SCADA workstation, comprising a human-machine interface (HMI) and a data historian, collectively referred to as the HMI. Communication within the system is based on modbus (MB) with transmission control protocol (TCP)/internet protocol (IP) transport layer. The attacks were launched from an additional Kali Linux machine, which is not depicted in Figure 3.6.

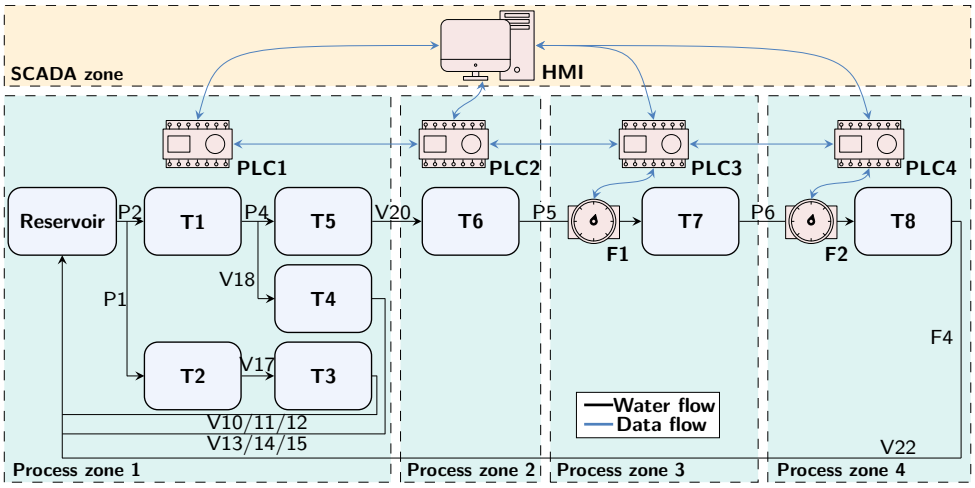


Figure 3.6: Simplified overview of the cyber-physical testbed considered for investigating cyber-physical event identification.
Source: Illustration adapted from **Paper D** and based on [29].

The associated dataset describes both the normal operation of the system as well as the system being affected by several types of cyber attacks and physical faults compromising different components and communication links. The attacks encompass eight man-in-the-middle (MITM), five denial-of-service (DoS), and seven scanning attacks. Three water leaks and six pump or valve breakdowns are included as physical faults. Table 3.4 lists the raw physical and network features of the dataset. The physical data comprises 9210 observations of constant one-second resolution. The network data on average contains 2265 packets per second, and in total comprises $\sim 24.5 \times 10^6$ packets. For a more detailed explanation of the testbed and dataset the reader is referred to [29].

3.2.2.4 Event identification pipelines

Feature extraction and fusion Cyber-physical event identification is challenged by the pronounced heterogeneity of data comprising network traffic and process readings. Moreover, some cyber attack techniques may affect individual packets (e.g., sending a malicious control command), while others are only visible from the context

Table 3.4: Raw cyber network and physical process features of the dataset considered for assessing cyber-physical event identification.
Source: Table adapted from **Paper C**.

No.	Physical features	No.	Network features
1	Timestamp	1	Timestamp
2-9	Fill level of T1-8	2-3	IP address[a]
10-15	On/off state of P1-6	4-5	media access control (MAC) address[a]
16-19	Flow value of F1-4	6-7	Port[a]
20-41	On/off state of V1-22	8	Protocol
		9	TCP flags
		10	Packet size
		11	MB function code
		12	MB response value
		13-14	Number of packets[a]

[a]Both of the source and destination of network packets.

of multiple packets (e.g., replaying valid data transmission). Consequently, a set of features should be extracted from network traffic which includes both features that are sensible to values of single packets, and features that set multiple packets into context, while their format must allow simultaneous processing with attributes from physical data. This is realized by several sample statistics which evaluate network traffic for each second. On the one hand, this enables joint processing with the raw physical features. On the other hand, the selected statistics can indicate individual malicious packets (e.g., MAC/IP mismatch within the current second), and set multiple packets into context (e.g., mean payload size in the current second). A full list of the 161 considered cyber and physical features is provided in **Paper C**.

Event identification pipeline structure Cyber attacks and physical faults encompass a variety of different types which have in common their rare occurrence. Thus, for detecting and identifying those events applying supervised ML, a highly imbalanced multi-class classification problem must be solved. For that purpose, an event identification pipeline structure which comprises a classifier and several preprocessing steps is considered, aiming to improve the classification performance (see Figure 3.7). Scaling is considered to normalize the strongly varying value ranges of the cyber-physical feature set, preventing individual features from dominating the learning process. Dimensionality reduction is applied to compress the information provided by the comparatively large cyber-physical feature set into a smaller number of features, thus lowering the complexity of the classification problem. Decreasing the number of majority class samples (undersampling) and increasing the amount of minority class observations (oversampling) in the training data is considered to prevent the classifier from ignoring low-populated attack and fault classes. Several commonly employed candidate methods are considered for each of the data transformation steps. Scaling techniques include standardization, normalization, and scaling to the maximum absolute value. Principle component analysis (PCA) with Bayesian

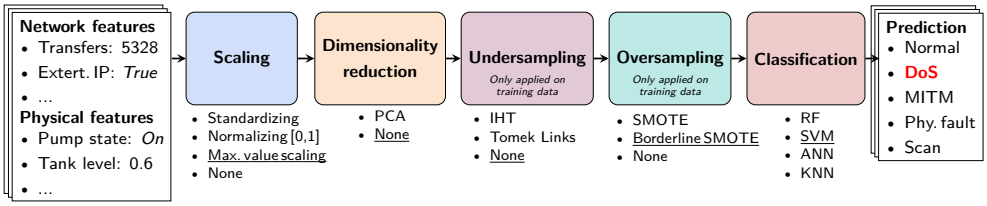


Figure 3.7: Event identification pipeline structure and method selection example.
Source: Illustration based on **Paper C**.

selection of the number of principle components is applied for dimensionality reduction [30]. Undersampling methods include instance hardness threshold (IHT) and Tomek Links removal, while for oversampling the synthetic minority oversampling technique (SMOTE) and borderline SMOTE is considered. The classifiers comprise random forest (RF), k-nearest neighbors (KNN), support vector machine (SVM), and artificial neural network (ANN). The pre-processing steps are optional and can also be bypassed. The pipeline structure and included methods are implemented in Python using the *scikit-learn* library [31], except for the ANN which is implemented using the *Keras* library [32]. For detailed descriptions of the applied techniques, the reader is referred to the documentation of the respective library and [33].

Pipeline selection and training For the evaluation of cyber-physical event identification, different instances of the pipeline structure shown in Figure 3.7 are implemented, which either only process network features or the whole cyber-physical set. Pipelines are developed based on each of the four different classification models (RF, SVM, ANN, and KNN) both for pure network and cyber-physical event identification to substantiate the results. Thus, in total eight pipelines are considered. Training and selection of hyperparameters and pre-processing techniques is conducted on approximately 75 % of the dataset for all pipelines. For that purpose, the entire sequences of the last two instances of each event class[1] are excluded and reserved for performance evaluation. Techniques such as shuffling or cross-validation are intentionally avoided as those would place observations of the same event instance in both training and test data, leading to overly optimistic performances as described in Section 3.2.2.2. The resulting event identification pipelines are summarized in Table 3.5 where they are named according to the underlying classifier. For unspecified hyperparameters, the default values provided by the respective library are employed.

[1]Since the dataset only contains three water leak events, some of which occur simultaneously with valve or pump breakdowns, all fault events are combined into one overarching class.

Table 3.5: Method and hyperparameter selection results for the network and cyber-physical event identification pipelines.
Source: Table adapted from **Paper C**.

	Cyber-physical event identification pipelines			
Step	**RF**	**KNN**	**SVM**	**ANN**
Scaling	Standardiz.	Standardization	Max. value scaling	Max. value scaling
Dim. reduction	None	PCA	None	PCA
Undersampling	None	None	None	None
Oversampling	SMOTE	None	Borderline SMOTE	Borderline SMOTE
Classification	$n_{\text{estimators}}$: 100, $n_{\text{max-features}}$: 17	$n_{\text{neighbors}}$: 5, *Dist. func.*: manhattan, *Weight func.*: distance	*Kernel*: radial-basis function, *Penalty para.*: 10000, *Kernel coeff.*: 0.0175	$n_{\text{hid.-layers}}$: 2, n_{units}: 150, *Act. func.*: ReLu, *Dropout rate*: 0.5, n_{epochs}: 500, *Batch size*: 512
	Network event identification pipelines			
Step	**RF**	**KNN**	**SVM**	**ANN**
Scaling	Max. value sc.	Standardization	Max. value scaling	Max. value scaling
Dim. reduction	None	PCA	None	PCA
Undersampling	IHT	Tomek Links	None	None
Oversampling	None	None	None	Borderline SMOTE
Classification	$n_{\text{estimators}}$: 100, $n_{\text{max-features}}$: 17	$n_{\text{neighbors}}$: 5, *Dist. func.*: manhattan, *Weight func.*: uniform	*Kernel*: radial-basis function, *Penalty para.*: 10000, *Kernel coeff.*: 0.0175	$n_{\text{hid.-layers}}$: 2, n_{units}: 100, *Act. func.*: ReLu, *Dropout rate*: 0.5, n_{epochs}: 500, *Batch size*: 256

3.2.2.5 Evaluation

Performance metrics For assessing the class-wise event identification performance, the F_1 score according to

$$F_{1,l} = \frac{TP_l}{TP_l + \frac{1}{2}(FP_l + FN_l)}, \tag{3.12}$$

is considered, with TP_l, FP_l and FN_l being the number of true positives, false positives and false negatives of the l-th class, respectively. The overall performance is expressed as macro average of the class-wise F_1 scores, following

$$F_1^m = \frac{\sum_{l=1}^{N_{\text{classes}}} F_{1,l}}{N_{\text{classes}}}, \tag{3.13}$$

with N_{classes} being the number of classes.

Comparison of network and cyber-physical event identification The class-wise and average F_1 scores of the eight event identification pipelines are provided in Table 3.6. All pipelines improve in class-specific and overall performance when considering the cyber-physical feature set. Only exception is a slight decrease in normal observation identification by the RF-based pipeline. The identification of scanning attacks improves, even though they do not affect the physical process. This is explained by the fact that physical features reduce confusion among observations

Table 3.6: Class-wise and average F_1 scores for network and cyber-physical event identification, where highest scores are in bold and second best are underlined.
Source: Table adapted from **Paper C**.

	Network features				Cyber-physical features			
Event class	**RF**	**KNN**	**SVM**	**ANN**	**RF**	**KNN**	**SVM**	**ANN**
Normal	0.92	0.93	0.86	0.88	0.91	<u>0.94</u>	**0.95**	0.93
DoS	0.55	0.47	<u>0.96</u>	0.47	0.71	0.49	**1.00**	0.50
MITM	0.87	0.83	0.42	0.68	0.87	<u>0.88</u>	0.81	**0.92**
Physical fault	0.07	0.04	0.00	0.14	0.26	0.06	**0.62**	<u>0.46</u>
Scanning	0.57	0.67	0.80	0.80	**1.00**	0.80	**1.00**	**1.00**
Average (F_1^m)	0.60	0.59	0.61	0.60	0.75	0.63	**0.88**	<u>0.76</u>

of cyber and cyber-physical attacks which have similar impact on the network traffic. The overall performance (F_1^m) averaged over the four classifier types increases by 15.5 percentage points. These results suggest that the joint evaluation of cyber network and physical process data can improve performance of real-time event identification.

The highest overall performance is achieved by the SVM-based cyber-physical event identification pipeline. The good performance under presence of physical faults suggests that cyber-physical event identification has the potential to simultaneously detect and classify cyber attacks and physical faults in real-time. That being said, the F_1 score for physical faults is improvable. While the reason might be complex class boundaries resulting from the merge of different fault types (water leaks and pump/valve breakdowns) in one overarching class, extracting features from physical raw data instead of their direct use could be an option for enhancing the performance.

3.2.3 CyPhERS: A Cyber-Physical Event Reasoning System

This section introduces the **Cy**ber-**Ph**ysical **E**vent **R**easoning **S**ystem CyPhERS. CyPhERS is first introduced in **Paper D**, where it is demonstrated on the previously described cyber-physical laboratory testbed (see Section 3.2.2.3). In **Paper E**, the concept is extended and demonstrated for DER monitoring. This section largely focuses on presenting the refined version of CyPhERS and its demonstration on a PV-battery system based on **Paper E**. Section 3.2.3.1 motivates the development of CyPhERS and describes the fundamental approach. A comparison to related works is provided in Section 3.2.3.2. Thereafter, the conducted experiments and the resulting dataset which serves as the basis to evaluate the application for DER monitoring are described in Section 3.2.3.3. Section 3.2.3.4 and 3.2.3.5 detail the methodology of CyPhERS' Stage 1 and Stage 2, respectively, together with the implementation on the considered PV-battery system case. Finally, the concept is demonstrated in Section 3.2.3.6.

3.2.3.1 Motivation and approach

Motivation The evaluation of cyber-physical event identification in Section 3.2.2 has demonstrated advantages of jointly processing cyber network and physical process data. It enables real-time identification of cyber attacks and physical faults in one integrated approach. Moreover, the identification performance for different attack and fault types is improved compared to exclusive use of network traffic features. These findings suggest that cyber-physical monitoring concepts can facilitate real-time operational awareness for DERs and other cyber-physical systems, and provide valuable and more reliable information to support incident response. However, the demonstrated benefits should be further combined with the fulfillment of additional requirements pertaining to interpretability, practicability, and generality. To create trust in the results of data-driven models and lower the risk for misinformed incident response, the event prediction process should be transparent and understandable. Furthermore, concepts should not depend on historical observations of rarely occurring attacks and faults to enable applicability under realistic conditions. Finally, as in complex cyber-physical architectures unseen events can occur, e.g., new attack strategies, monitoring should not be limited to the identification of a small set of possible attack or fault vectors and instead generalize to providing information on both known and unknown event types. To combine the demonstrated strengths of cyber-physical monitoring with these requirements on interpretability, practicability, and generality, the cyber-physical event reasoning system CyPhERS is developed.

Approach CyPhERS comprises a two-stage process which jointly evaluates network traffic and physical process data to deduce information about events such as their occurrence, type, affected devices, and physical impact in real-time (see Figure 3.8). The first stage generates informative and interpretable event signatures. This is accomplished by combining methods including cyber-physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. Within the second stage, event information are derived from the signatures either through automated matching with a database of predefined signatures or manual interpretation by operators. A detailed description of Stage 1 and 2 follows in the Sections 3.2.3.4 and 3.2.3.5, respectively.

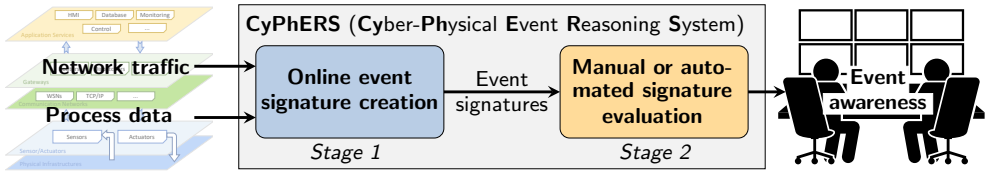


Figure 3.8: Illustration of the cyber-physical event reasoning system CyPhERS.
Source: Illustration adapted from **Paper D**.

3.2.3.2 Related works and contribution

Most of the methods applied in CyPhERS are used in existing works, and associated strengths are demonstrated. While benefits of cyber-physical data fusion are shown in **Paper C**, several works demonstrate the use of unsupervised multivariate anomaly detection to localize affected process components [34, 35]. However, CyPhERS is the first concept which combines these methods and integrates them with anomaly type differentiation to generate informative and human-readable signatures for known and unknown event types that can be evaluated to derive valuable event information, while being independent of historical event observations.

3.2.3.3 Experiments and dataset creation

As mentioned in Section 3.2.2.3, cyber-physical datasets describing DERs that are affected by a variety of attacks or faults are rare. Therefore, own experiments on a real PV-battery system were conducted to collect data which allow to evaluate the applicability of CyPhERS for DER monitoring.

PV-battery system Figure 3.9 depicts the cyber-physical structure of the investigated PV-battery system, which is located at the Karlsruhe Institute of Technology, Germany. It consists of four PV inverters with dedicated solar panel strings (PV1-4) having peak powers ranging from 15.50 kW_p to 16.74 kW_p, four battery inverters and associated battery stacks (BAT1-4) with a capacity of 10.24 kWh and a maximum dis-/charge power of 5 kW, four energy meters (M1-4), a data manager (DM), and a data server (DS). Communication is based on MB, TCP, address resolution protocol (ARP), and user datagram protocol (UDP). BAT1-4 are connected to individual

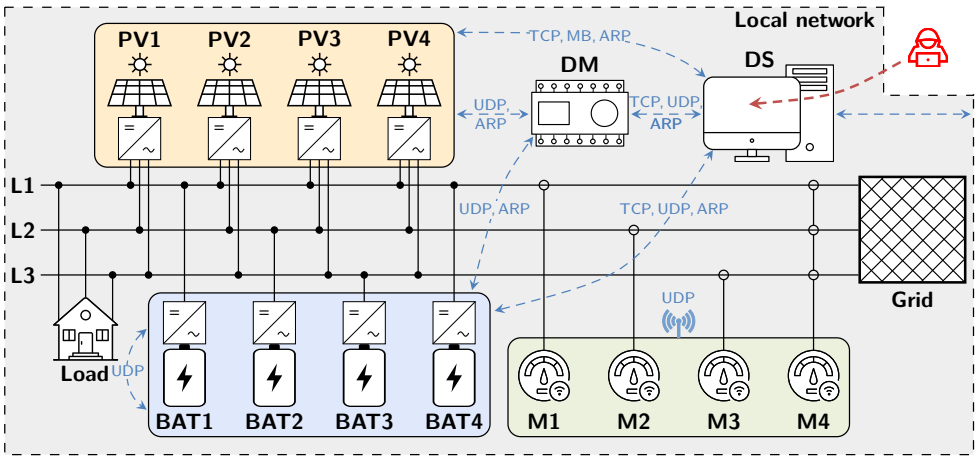


Figure 3.9: Illustration of the PV-battery system used for demonstrating CyPhERS.
Source: Illustration adapted from **Paper E**.

phases (L1-3). PV1-4 are linked to all three lines. While L1-3 are measured individually by M1-M3, M4 measures all three phases, providing measurement redundancy. The load follows typical office building patterns, which includes a higher demand during working hours and a lower one on weekends. In addition, there are periodic spikes in the load caused by the activation of an air compressor. The system aims to minimize power exchange with the grid. Thus, given an appropriate SOC, batteries charge when PV production exceeds the load, and discharge in the opposite case. Power flows P^{L1}-P^{L3} are controlled separately by the linked batteries. The battery controllers get the necessary measurements of P^{L1}-P^{L3} via subscription to the UDP multicast of the respective energy meter (M1, M2, or M3). The DM collects solar panel and battery measurements, including temperatures. The DS hosts data visualization software and is the interface to the external network.

Threat model The attacker obtained virtual access to the local network by hijacking the DS, from where several cyber and cyber-physical attacks are launched, targeting different devices. The attack types are among the most relevant ones for DERs [36–38]. The cyber attacks include SYN scans and HTTPS requests (reconnaissance), as well as ARP spoofing for eavesdropping (data collection). False data injection attacks (FDIAs), false command injection attacks (FCIAs), and replay attacks comprise the cyber-physical attacks. While the FDIAs inject false energy meter power readings which triggers sudden battery dis-/charging, the FCIAs involve shutting down either PV or battery inverters. The replay attacks amplify battery control errors by repeating valid energy meter power readings, leading to battery oscillation. The considered cyber-physical attacks could be part of a static (FDIAs and FCIAs)

Table 3.7: Schedule of attack experiments conducted to collect data for the demonstration of CyPhERS.
Source: Table adapted from **Paper E**.

No.	Attack type	Victim	Start	End
1	FDIA	M3	10:01	10:17
2	ARP spoof	PV3/DM	10:31	10:48
3	HTTPS request	BAT1	11:06	11:06
4	SYN Scan	PV3	11:22	11:34
5	FCIA	PV2	11:47	12:01
6	FDIA	M1	12:14	12:32
7	HTTPS request	DM	12:47	12:47
8	Replay attack	M1	13:05	13:07
9	FCIA	BAT3	13:26	13:39
10	FCIA	PV1	13:56	14:09
11	ARP spoof	PV4/DM	14:30	14:44
12	FCIA	BAT4	15:00	15:19
13	FDIA	M2	15:39	15:48
14	SYN Scan	PV4	16:04	16:08
15	Replay attack	M2	16:20	16:23

or dynamic (replay attack) load-altering attack against power systems in the context of a coordinated manipulation of a DER fleet [11].

Dataset The experiments took place in October 2022. After two weeks of recording the system's normal operation, 15 attacks were launched within one day (see Table 3.7). The experiments were recorded by a passive network packet capture from which a set of physical process and network traffic features is extracted (see Table 3.8). The data resolution ranges from one second to one minute for physical data, depending on the feature. The network data on average consists of 7539 packets per minute.

Table 3.8: Raw cyber network and physical process features of the dataset created for demonstration of CyPhERS.
Source: Table adapted from **Paper E**.

No.	Physical features	No.	Network features
1	Timestamp	1	Timestamp
2	Solar irradiation *GHI*	2-3	IP address (source & destination)
3-14	Active power $P^{\text{PV1-4}}$, $P^{\text{BAT1-4}}$, $P^{\text{M1-4}}$	4-5	MAC address (source & destination)
15-18	Battery state of charge $SOC^{\text{BAT1-4}}$	6	Protocol
19-22	Battery voltage $V^{\text{BAT1-4}}$	7	TCP flags
23-26	Battery temperature $T^{\text{BAT1-4}}$	8	MB function code

3.2.3.4 Online event signature creation (CyPhERS' Stage 1)

CyPhERS' Stage 1, responsible for online event signature creation, is schematically depicted in Figure 3.10. The signatures consist of anomaly flags of several system variables which are derived from both physical sensor readings and cyber network traffic. The flags are generated by a set of data pipelines within the signature extraction system, each consisting of a data-driven time series model of a

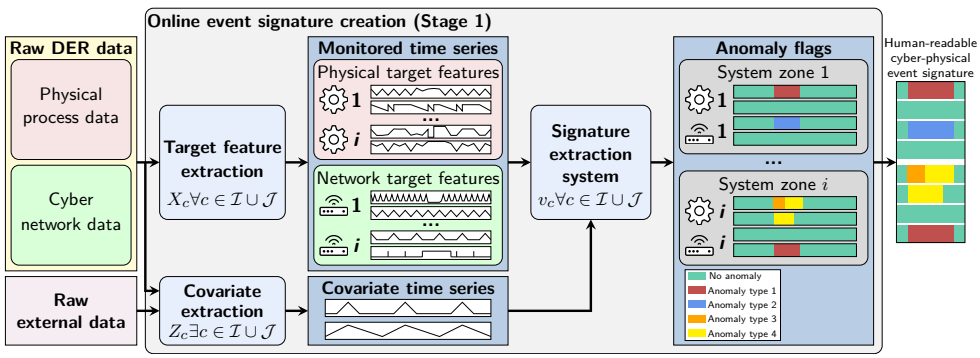


Figure 3.10: Illustration of CyPhERS' online event signature creation (Stage 1).
Source: Illustration adapted from **Paper E**.

specific system variable, and an anomaly detector that flags abnormal deviations between the model and actual observations. Monitored system variables are subsequently denoted *target features*, where $\mathcal{I}$ and $\mathcal{J}$ represent the physical and network feature subset, respectively. The time series of a target feature c is given as $\boldsymbol{X}_c = \{x_1^c, x_2^c, ..., x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$, and the one of a covariate used for modeling c is represented by $\boldsymbol{Z}_c = \{z_1^c, z_2^c, ..., z_N^c \mid z_i^c \in \mathbb{R}\ \forall i\}$. Several anomaly types are considered in the anomaly detector of each target feature, representing distinct abnormal behavior. The series of anomaly flags for a target feature c is denoted $\boldsymbol{v}_c = \{v_1^c, v_2^c, ..., v_N^c \mid v_i^c \in \{-2, -1, 0, 1, 2\}\ \forall i\}$. In the following, target feature and covariate extraction as well as the signature extraction system are explained in more detail.

Target feature and covariate extraction DERs comprise a diverse spectrum of technical systems due to factors such as the variety of physical components (e.g., PV panels, turbines or batteries) and communication protocols (e.g., MB, UDP or ARP). While this precludes definition of a general set of target features, guidelines for extraction of meaningful features can be provided.

Monitoring physical target features serves the purpose of detecting true physical events as well as manipulations of process-relevant data. Accomplishing the former entails employing features that represent operation of all physical components of a DER, enabling localization of affected ones and identification of the associated physical impact. The latter requires monitoring of sensor readings utilized for process control. Attacks can disable a component's functionality (e.g., switch battery off) or exploit their normal functionalities to achieve an abnormal behavior (e.g., control battery to induce load oscillation). Therefore, it is essential to consider a set of physical target features that covers both the technical functionality and behavior of components. Models of functional features should use as input only the variables of the concerned component and its immediate inputs or redundant devices. For behavioral features, selection of model inputs allows to define the context in which the component's behavior is considered abnormal. For DERs exhibiting pronounced volatility or randomness, for example due to weather or user influences, physical components can be described by features that break down a component's operation to simpler abstractions such as the on/off state.

Table 3.9 presents the physical target features and related model inputs for the investigated PV-battery system. The functionality of PV1-4, BAT1-4, and M1-3 is modeled as the average active load applying only variables of the respective component and its immediate inputs or redundant devices, which is denoted as P_{fmean}. $P_{\text{bmean}}^{\text{BAT}i}$, $S^{\text{BAT}i}$ and $P_{\text{osc}}^{\text{BAT}i}$ represent the behavior of the batteries. Anomalies in $P_{\text{bmean}}^{\text{BAT}i}$ and $S^{\text{BAT}i}$ indicate abnormal dis-/charging and in-/activeness, respectively, given the current time and PV feed, while anomalies in $P_{\text{osc}}^{\text{BAT}i}$ can indicate abnormal load oscillation. Behavior of the energy meters is represented by $|P_{\text{sum}}^{\text{Mi}}|$, which can point towards abnormally many or few multicasts, potentially resulting from misuse.

Table 3.9: Overview of physical target features extracted for applying CyPhERS to the investigated PV-battery system case.
Source: Table adapted from **Paper E**.

Target feature	Model input	Type	Description
$P^{\mathrm{PV}i}_{\mathrm{fmean}}$	*GHI* (local)	Functional	For every 60s time step τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{PV}i}$ within τ_{60} according to $P^{\mathrm{PV}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{PV}i}_p / N_{\tau_{60}}$. Anomalies in $P^{\mathrm{PV}i}_{\mathrm{fmean}}$ can indicate disfunction of the i-th solar panel string or its inverter.
$P^{\mathrm{BAT}i}_{\mathrm{fmean}}$	$P^{\mathrm{M}i}_{\mathrm{mean}}$, $V^{\mathrm{BAT}i}_{\mathrm{mean}}$, $SOC^{\mathrm{BAT}i}_{\mathrm{mean}}$, $T^{\mathrm{BAT}i}_{\mathrm{mean}}$	Functional	For every τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{BAT}i}$ within τ_{60} according to $P^{\mathrm{BAT}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{BAT}i}_p / N_{\tau_{60}}$. Anomalies in $P^{\mathrm{BAT}i}_{\mathrm{fmean}}$ can indicate a disfunction of the i-th battery stack or its associated inverter.
$P^{\text{M1-3}}_{\mathrm{fmean}}$	Redundant measurement of M4	Functional	For every 5s time step τ_5 the mean value is determined as average over the N_{τ_5} data packets carrying $P^{\mathrm{M}i}$ within τ_5 according to $P^{\mathrm{M}i}_{\mathrm{mean},\tau_5} = \sum_{p=1}^{N_{\tau_5}} P^{\mathrm{M}i}_p / N_{\tau_5}$. Anomalies can indicate a disfunction of the i-th meter.
$P^{\mathrm{BAT}i}_{\mathrm{osc}}$	$P^{\mathrm{BAT}i}_{\mathrm{osc}}$ lag values	Behavioral	For every 15s time step τ_{15} the absolute sum of power changes $P^{\mathrm{BAT}i}_{\mathrm{osc},\tau_{15}} = \sum_{f=0s}^{15s} \lvert P^{\mathrm{BAT}i}_{\mathrm{mean},\tilde{\tau}+f+1s} - P^{\mathrm{BAT}i}_{\mathrm{mean},\tilde{\tau}+f} \rvert$ is calculated, where $\tilde{\tau}$ is the start time of τ_{15}. Anomalies in $P^{\mathrm{BAT}i}_{\mathrm{osc}}$ can indicate oscillation of the i-th battery.
$S^{\mathrm{BAT}i}$	$P^{\text{PV1-4}}_{\mathrm{mean}}$, time of day	Behavioral	For every τ_{60} the on-off state ($S^{\mathrm{BAT}i}_{\tau_{60}} \in [0,1]$) is determined. Anomalies may indicate that BATi is unexpectedly online/offline given the current time of day and PV feed.
$P^{\mathrm{BAT}i}_{\mathrm{bmean}}$	$P^{\text{PV1-4}}_{\mathrm{mean}}$, time of day	Behavioral	For every τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{BAT}i}$ within τ_{60} according to $P^{\mathrm{BAT}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{BAT}i}_p / N_{\tau_{60}}$. Anomalies in $P^{\mathrm{BAT}i}_{\mathrm{bmean}}$ can indicate abnormal behavior of BATi given the current time of day and PV feed.
$\lvert P^{\text{M1-3}}_{\mathrm{sum}} \rvert$	$P^{\mathrm{M}i}_{\mathrm{mean}}$	Behavioral	For every τ_5 the absolute sum according to $\lvert P^{\mathrm{M}i}_{\mathrm{sum},\tau_5} \rvert = \sum_{p=1}^{N_{\tau_5}} P^{\mathrm{M}i}_p$ is determined. Anomalies can indicate abnormal behavior of the i-th meter given the current $P^{\mathrm{M}i}_{\mathrm{mean}}$ (sending abnormally many or few $P^{\mathrm{M}i}$ packets).

Monitoring network target features follows two main objectives: 1) localization of compromised network devices, and 2) determination of attack types. The former can be achieved by monitoring the traffic of each network device individually. The latter requires extraction of several informative features for each device.

Table 3.10 lists the network target features for the PV-battery system case. Multiple features are extracted for the PV and battery inverters, energy meters, DM, and DS. The features represent occurrence counts of specific protocols, TCP flags and MB function codes within 15-second periods. All network target features are modelled based on their lag values and time of day. The latter is considered a valuable covariate since many processes in operational technology (OT) networks are conducted at specific times, e.g. every hour. While many further informative features exist, only the ones considered most relevant are used for the sake of comprehensibility of the demonstration in Section 3.2.3.6. For the same reason, only packets sent *to* a device are evaluated, except for M1-3, as they send process-relevant data, and the DS, since its connection to external networks renders it a likely target for attackers.

Table 3.10: Overview of network target features extracted for applying CyPhERS to the investigated PV-battery system case.
Source: Table based on **Paper E**.

Target feature[a]	Description[b]
$n^{\mathrm{PV}i}_{\mathrm{UDP_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{UDP_d}}$, $n^{\mathrm{DM}}_{\mathrm{UDP_d}}$, $n^{\mathrm{DS}}_{\mathrm{UDP_d}}$, $n^{\mathrm{M}i}_{\mathrm{UDP_s}}$, $n^{\mathrm{DS}}_{\mathrm{UDP_s}}$	UDP packets send to/from the device.
$n^{\mathrm{PV}i}_{\mathrm{TCP_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{TCP_d}}$, $n^{\mathrm{DM}}_{\mathrm{TCP_d}}$, $n^{\mathrm{DS}}_{\mathrm{TCP_d}}$, $n^{\mathrm{DS}}_{\mathrm{TCP_s}}$	TCP packets send to/from the device.
$n^{\mathrm{PV}i}_{\mathrm{MB_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{MB_d}}$, $n^{\mathrm{DM}}_{\mathrm{MB_d}}$, $n^{\mathrm{DS}}_{\mathrm{MB_d}}$, $n^{\mathrm{DS}}_{\mathrm{MB_s}}$	MB packets send to/from the device.
$n^{\mathrm{PV}i}_{\mathrm{ARP_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{ARP_d}}$, $n^{\mathrm{DM}}_{\mathrm{ARP_d}}$, $n^{\mathrm{DS}}_{\mathrm{ARP_d}}$, $n^{\mathrm{DS}}_{\mathrm{ARP_s}}$	ARP packets send to/from the device.
$n^{\mathrm{PV}i}_{\mathrm{TLS_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{TLS_d}}$, $n^{\mathrm{DM}}_{\mathrm{TLS_d}}$, $n^{\mathrm{DS}}_{\mathrm{TLS_d}}$, $n^{\mathrm{DS}}_{\mathrm{TLS_s}}$	Transport layer security (TLS) packets send to/from the device.
$n^{\mathrm{PV}i}_{\mathrm{SYN_d}}$, $n^{\mathrm{BAT}i}_{\mathrm{SYN_d}}$, $n^{\mathrm{DM}}_{\mathrm{SYN_d}}$, $n^{\mathrm{DS}}_{\mathrm{SYN_d}}$, $n^{\mathrm{DS}}_{\mathrm{SYN_s}}$	Packets with SYN flag send to/from the device.
$n^{\mathrm{PV}i}_{16_{\mathrm{d}}}$, $n^{\mathrm{BAT}i}_{16_{\mathrm{d}}}$	Packets with write register code send to the device.
$n^{\mathrm{PV}i}_{4_{\mathrm{d}}}$, $n^{\mathrm{BAT}i}_{4_{\mathrm{d}}}$	Packets with read register code send to the device.

[a] Indices refer to source (s) and destination (d), respectively.
[b] Packets are counted for 15-second periods.

Signature extraction system The signature extraction system (see Figure 3.10) consists of a set of anomaly detection and classification pipelines, each following the structure depicted in Figure 3.11. The pipelines comprise a target feature model and an associated anomaly detector which flags abnormal deviations between the model and ground truth observations. As DER operation can be subject to weather- and consumer-induced volatility and stochasticity, probabilistic predictions and anomaly decision functions are applied. For each target feature c, the lower quantile q_{L}, median q_{M}, and upper quantile q_{U} are predicted. Given $\boldsymbol{X}_c = \{x_1^c, x_2^c, ..., x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$ and $\boldsymbol{Z}_{c,1}, ..., \boldsymbol{Z}_{c,M} = \{\{z_{1,1}^c, z_{1,2}^c, ..., z_{1,N}^c\}, ..., \{z_{M,1}^c, z_{M,2}^c, ..., z_{M,N}^c\} | z_{j,i}^c \in \mathbb{R} \forall (j,i)\}$, the expected quantile $\hat{x}_t^{q,c}$ at time step t is predicted based on lag values $x_{t-w_{\mathrm{hist}}}^c, ..., x_{t-1}^c$ and covariates $z_{1,t}^c, ..., z_{M,t}^c$ following

$$\hat{x}_t^{q,c} = \Phi\left([x_{t-w_{\mathrm{hist}}}^c, ..., x_{t-1}^c], [z_{1,t}^c, ..., z_{M,t}^c]\right), \forall q \in \{q_{\mathrm{L}}, q_{\mathrm{M}}, q_{\mathrm{U}}\}, \tag{3.14}$$

with M being the number of covariates, and w_{hist} the history window size. Whether and which covariates and lag values are used depends on the target feature. A model

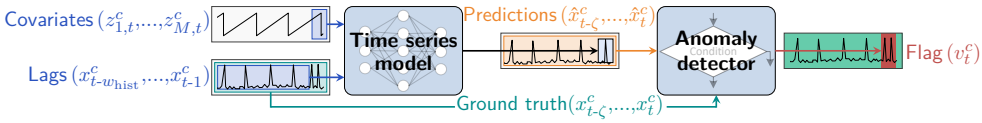


Figure 3.11: Illustration of the anomaly detection and classification pipeline of a target feature c within the signature extraction system of CyPhERS' Stage 1.
Source: Illustration adapted from **Paper E**.

learns to predict $\hat{x}^{q,c}$ by minimizing the quantile loss function [39]

$$\mathcal{L}_q(\hat{x}_i^c, x_i^c) = \max\left[q(x_i^c - \hat{x}_i^c), (q-1)(x_i^c - \hat{x}_i^c)\right] \tag{3.15}$$

for a training set $\boldsymbol{X}_{\text{train}}^c = \{x_1^c, x_2^c, ..., x_{N_{\text{train}}}^c \mid x_i^c \in \mathbb{R}\ \forall i\}$. The quantile predictions of c are forwarded to the anomaly detector, which decides to flag an anomaly based on the distance between the ground truth and the prediction interval (PI) $[\hat{x}^{q_{\text{L}},c}, \hat{x}^{q_{\text{U}},c}]$. Consequently, the distance calculation dynamically adapts to the current model's confidence. The distances are averaged over the last ζ observations according to

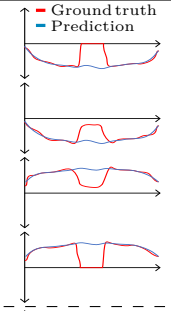
$$\varepsilon_t^c = \frac{\sum_{i=0}^{\zeta-1} \begin{cases} x_{t-i}^c - \hat{x}_{t-i}^{q_{\text{U}},c} & \text{if } x_{t-i}^c > \hat{x}_{t-i}^{q_{\text{U}},c} \\ \hat{x}_{t-i}^{q_{\text{L}},c} - x_{t-i}^c & \text{if } x_{t-i}^c < \hat{x}_{t-i}^{q_{\text{L}},c} \end{cases}}{\zeta}. \tag{3.16}$$

For the studied PV-battery system, $\zeta = 5$, $q_{\text{L}} = 0.01$, and $q_{\text{U}} = 0.99$ are chosen $\forall c \in \mathcal{I}$ and $\mathcal{J}$. On the basis of ε_t^c and additional characteristics of the recent target feature observation, several anomaly types are differentiated, as indicated by the detector's decision function

$$v_t^c = \begin{cases} 2 & \text{if } \overbrace{(\varepsilon_t^c > \gamma_c)}^{\text{Detection}},\ \overbrace{(x_t^c > \hat{x}_t^{q_{\text{M}},c})}^{\text{Direction}} \text{ and } \overbrace{(x_t^c = 0)}^{\text{Is zero}} \\ 1 & \text{if } (\varepsilon_t^c > \gamma_c),\ (x_t^c > \hat{x}_t^{q_{\text{M}},c}) \text{ and } (x_t^c \neq 0) \\ -1 & \text{if } (\varepsilon_t^c > \gamma_c),\ (x_t^c < \hat{x}_t^{q_{\text{M}},c}) \text{ and } (x_t^c \neq 0) \\ -2 & \text{if } (\varepsilon_t^c > \gamma_c),\ (x_t^c < \hat{x}_t^{q_{\text{M}},c}) \text{ and } (x_t^c = 0) \\ 0 & \text{otherwise,} \end{cases} \tag{3.17}$$

with γ_c being a target feature-specific threshold. Details regarding the anomaly types are provided in Table 3.11. Both the information about the direction of an abnormal

Table 3.11: Anomaly types considered for event signature creation in CyPhERS.
Source: Table based on **Paper E**.

Flag v	Anomaly type	Description	Schematic
2	Positive zero	Target feature abnormally high and zero.	
1	Positive non-zero	Target feature abnormally high and non-zero.	
−1	Negative non-zero	Target feature abnormally low and non-zero.	
−2	Negative zero	Target feature abnormally low and zero.	
0	No anomaly	No abnormal behavior.	

deviation and about a target feature being zero facilitates identification of event root causes and physical impacts. For instance, unusually *many* UDP packets from an energy meter may indicates a FDIA, while *zero* PV feed during daytime can point towards a switched off inverter.

Implementing the signature extraction system involves offline model hyperparameter and detector threshold selection for the detection and classification pipelines of all target features (see Figure 3.12).

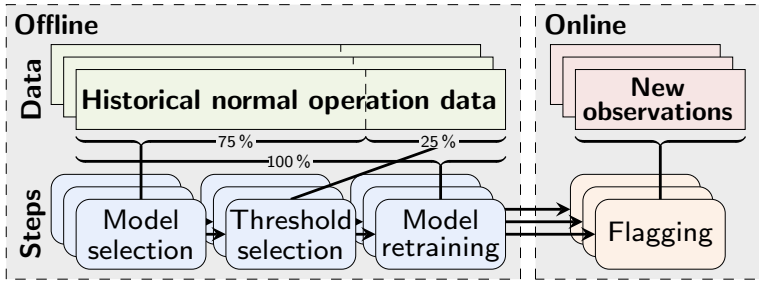


Figure 3.12: Procedure for implementing the anomaly detection and classification pipelines within the signature extraction system of CyPhERS' Stage 1.
Source: Illustration adapted from **Paper E**.

In the investigated study case, target features are modeled using GBDT. The tuned hyperparameters (see Table 3.12) are selected on the first 75 % of the two weeks normal operation data. The detector threshold of a target feature c is determined based on the distances for the remaining 25 % normal operation data $\boldsymbol{E}^c_{\text{test}} = \{\varepsilon^c_1, \varepsilon^c_2, ..., \varepsilon^c_{N_{\text{test}}} \mid \varepsilon^c_i \in \mathbb{R}\ \forall i\}$, and the threshold factor $\beta = 1.1$ according to

$$\gamma_c = \beta \cdot \max\left(\boldsymbol{E}^c_{\text{test}}\right). \tag{3.18}$$

Table 3.12: Tuned hyperparameters and search spaces for the GBDT models of the anomaly detection pipelines in the signature extraction system of CyPhERS' Stage 1.
Source: Table adapted from **Paper E**.

No.	Hyperparameter	Search space
1	$w^{\text{a}}_{\text{hist}}$	[0,...,60]
2	Max. depth of a tree	[3,...,21]
3	Number of decision trees	[100,...,1000]
4	Learning rate	[0.001,...,0.3]

[a]Only for target features considering lag values.

3.2.3.5 Signature evaluation (CyPhERS' Stage 2)

CyPhERS' Stage 2, responsible for evaluating signatures provided by Stage 1, is schematically depicted in Figure 3.13.

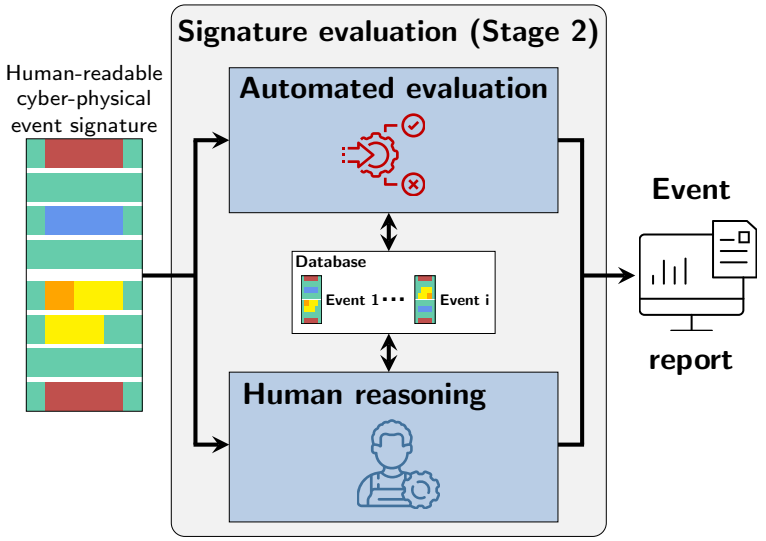


Figure 3.13: Illustration of CyPhERS' signature evaluation (Stage 2).
Source: Illustration adapted from **Paper E**.

Signature database The evaluation involves comparison to a database of predefined signatures. The predefinition can be realized on different level of detail. While simple signatures may be formulated which only indicate affected devices or differentiate between cyber, physical or cyber-physical events, more complex signatures can be defined for specific event types and associated impacts. Consequently, signatures can be formulated both for known and unknown event types, and constantly improved by updating the database. Once a signature provided by Stage 1 matches with one from the database, the signature description (e.g., event type, affected devices, attacker location and physical impact) forms the event hypothesis.

Figure 3.14 illustrates the signatures for attack types of the investigated study case on the example of selected victim devices and physical impacts. For the sake of conciseness, the various protocol count flags are combined in $v_{n_{\text{proto}}}$, and flags in counts of different MB function codes ($v_{n_{16_{\text{d}}}}$ and $v_{n_{4_{\text{d}}}}$) in $v_{n_{\text{fcode}}}$, for each system zone. Detailed descriptions of the signatures are provided in Table 3.13.

Signature matching Contrasting signatures provided by Stage 1 with the database can be done manually through visual comparison or automatically by translating each predefined signature into a set of decision rules of the following form:

flagging of (anomaly type 1 *in* target feature A) *and* (type 2 *in* target feature B) *indicates* (device X *being targeted by* attack type Y *causing* physical impact Z).

As the signatures can be read and interpreted by humans, automated event predictions can be verified by operators, which facilitates identification of mispredictions.

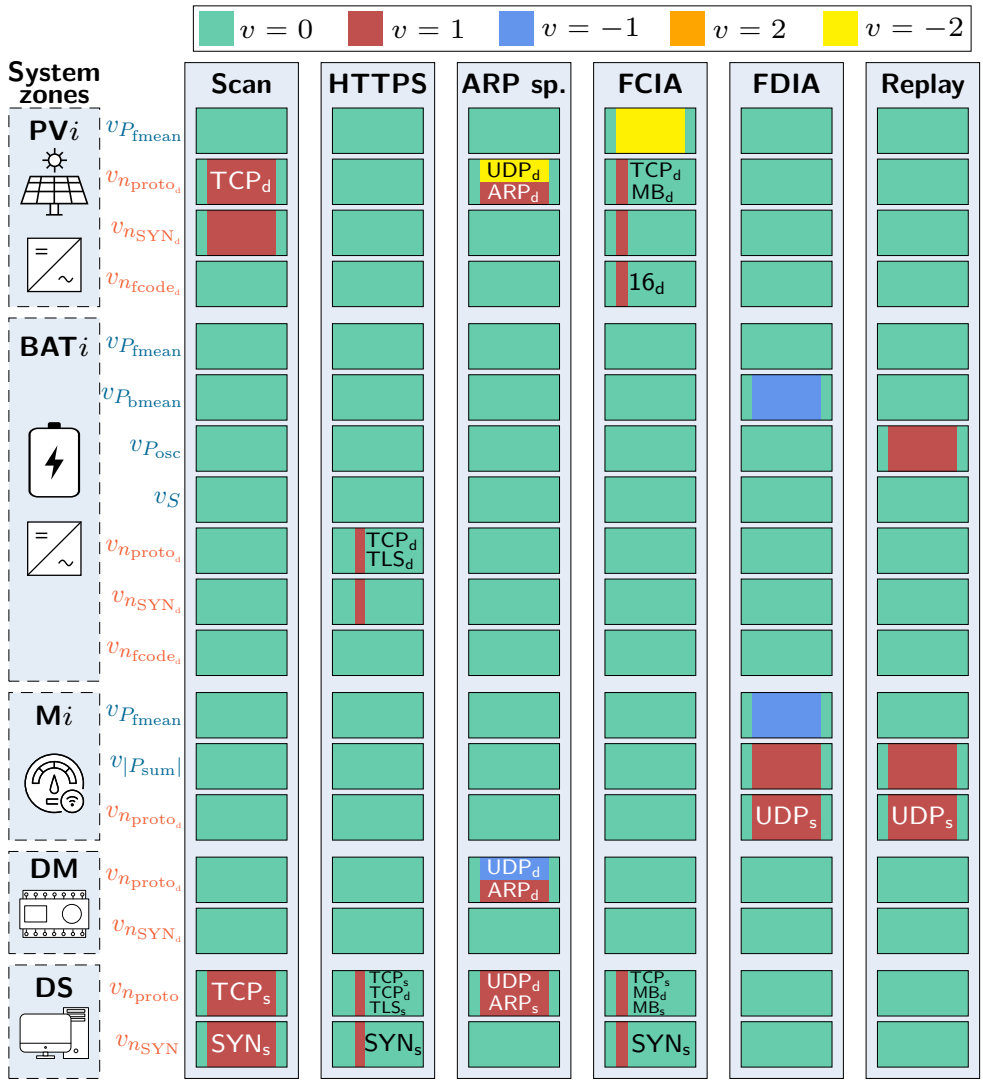


Figure 3.14: Event signatures of the attack types considered for demonstrating CyPhERS, on the example of selected victim devices and physical impacts.
Source: Illustration adapted from **Paper E**.

To proof the feasibility of automating the evaluation of signatures, a simple rule-based system is developed for the study case, which evaluates the latest flags of all target features within a rolling time window T_{eval} of five minutes (see Algorithm 1). In case that $v_c = 0 \ \forall c \in \mathcal{I}$ and $\mathcal{J}$ within T_{eval}, *normal operation* is predicted. If flags within T_{eval} match the rules of a predefined signature, the associated event

Table 3.13: Description of the attack signatures depicted in Figure 3.14.
Source: Table adapted from **Paper E**.

Attack	Event signature description
Scan	A device (e.g., PV inverter) receives an abnormally large number of TCP packets ($v_{n_{\mathrm{TCP_d}}}=1$) with connection request ($v_{n_{\mathrm{SYN_d}}}=1$) over a longer period. Simultaneously, another device (e.g., DS) sends unusually many TCP packets ($v_{n_{\mathrm{TCP_s}}}=1$) with connection request ($v_{n_{\mathrm{SYN_s}}}=1$). Together, this points towards scanning of a victim device (here, the PV inverter), where the attacker is located on a local device (here, the DS). The lack of anomaly flags in physical target features indicates a pure cyber attack without physical impact.
HTTPS request	A device (e.g., battery inverter) receives abnormally many TLS packets ($v_{n_{\mathrm{TLS_d}}}=1$) over short period, pointing towards a web service call (HTTPS request). Simultaneously, another device (e.g., DS) sends more TLS packets than usual ($v_{n_{\mathrm{TLS_s}}}=1$), which suggests the attacker being located on this device. Parallel increase of TCP packets ($v_{n_{\mathrm{TCP_d}}}=1$) and packets with SYN flags ($v_{n_{\mathrm{SYN_d}}}=1$) due to connection establishment between attacker and victim device. The absence of anomaly flags in physical target features indicates a cyber attack without any physical impact.
ARP spoof	Two devices (e.g., PV inverter and DM) receive abnormally many ARP packets ($v_{n_{\mathrm{ARP_d}}}=1$), while another (e.g., DS) sends more than expected ($v_{n_{\mathrm{ARP_s}}}=1$). This points towards ARP spoofing where the attacker is located on a local device (here, the DS). The two victim devices receive less (or no) UDP packets ($v_{n_{\mathrm{UDP_d}}}=-1$ or -2), while the device occupied by the attacker receives more ($v_{n_{\mathrm{UDP_d}}}=1$), which suggests that the communication between the victims is successfully redirected via the occupied device[a]. Lack of flags in physical target features imply eavesdropping instead of manipulation of process-relevant data.
FCIA	A device (e.g., PV inverter) receives an abnormally large number of MB packets ($v_{n_{\mathrm{MB_d}}}=1$) with write register function code 16 ($v_{n_{16_{\mathrm{d}}}}=1$). In parallel, another device (e.g., DS) sends more MB packets than usual ($v_{n_{\mathrm{MB_s}}}=1$). Together, this indicates an attacker sending false control commands to a victim device (here, the PV inverter) from the occupied local device (here, the DS). Parallel increase of TCP packets and packets with SYN flags because of connection establishment between occupied and victim device. Abnormally low and zero PV feed ($v_{P_{\mathrm{fmean}}}=-2$) indicate that the attacker switched off the PV inverter[b].
FDIA	An energy meter Mi sends unusually many UDP packets[c] ($v_{n_{\mathrm{UDP_s}}}=1$) while the absolute sum of its active power readings is too high ($v_{\lvert P_{\mathrm{sum}}\rvert}=1$). Together this points towards unusual frequent broadcasting of active power readings. The parallel abnormally low mean ($v_{P_{\mathrm{fmean}}}=-1$) indicates false $P^{\mathrm{M}i}$ injection imitating grid exports. For the battery which uses Mi readings, an unusually low mean active power given the current time and PV feed ($v_{P_{\mathrm{bmean}}}=-1$) suggests reaction with charging[b]. Absence of anomalies in $P^{\mathrm{BAT}i}_{\mathrm{fmean}}$ underlines that the battery accepts the false data and reacts to them in an expected way.
Replay attack	An energy meter Mi sends abnormally high numbers of UDP packets[c] ($v_{n_{\mathrm{UDP_s}}}=1$), and the absolute sum of its active power measurements is higher than expected ($v_{\lvert P_{\mathrm{sum}}\rvert}=1$). Together this indicates unusually frequent broadcasting of active power readings. As the mean is normal ($v_{P_{\mathrm{fmean}}}=0$), no false data is injected, and instead, a replay of valid $P^{\mathrm{M}i}$ readings is likely. Abnormally high power changes ($v_{P_{\mathrm{osc}}}=1$) of one or more batteries indicates load oscillation due to multiplication of the control error through replaying $P^{\mathrm{M}i}$ values.

[a] Parallel network anomalies for other devices which communicate with the victims possible as victim functionality can be affected by the attack.
[b] Physical impact depends on the injected control command/false data, and the victim device.
[c] Parallel network anomalies for other devices possible due to UDP traffic overloading of those.

description forms the prediction. Otherwise, the system predicts *Unknown abnormal behavior*. Some further defined rules are omitted in Algorithm 1 for brevity. These include variations of attack types (e.g., replay attack *with* or *without* parallel traffic overloading of other devices), and simple rules for unknown event types which predict affected system zones (e.g., *Unknown network anomaly affecting PV2*). Some rules take previous predictions into account. For instance, when predicting a battery being switched off by a FCIA, flag of the false command is considered longer than T_{eval} which prevents transitioning to the prediction of a physical event after five minutes.

Algorithm 1 Simplified representation of the rule-based signature evaluation system. Source: Adapted from **Paper E**.

$T_{\text{eval}} \leftarrow$ Last 5 minutes
if all flags in T_{eval} are zero **then**
  prediction $\leftarrow$ Normal operation
else if $v^{\text{X}}_{n_{\text{TCP}_\text{d}}} = 1$, $v^{\text{X}}_{n_{\text{SYN}_\text{d}}} = 1$, $v^{\text{Y}}_{n_{\text{TCP}_\text{s}}} = 1$, and $v^{\text{Y}}_{n_{\text{SYN}_\text{s}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ Scan of device X from device Y
else if $v^{\text{X}}_{n_{\text{TLS}_\text{d}}} = 1$, $v^{\text{X}}_{n_{\text{TCP}_\text{d}}} = 1$, $v^{\text{X}}_{n_{\text{SYN}_\text{d}}} = 1$, $v^{\text{Y}}_{n_{\text{TLS}_\text{s}}} = 1$, $v^{\text{Y}}_{n_{\text{TCP}_\text{d}}} = 1$, $v^{\text{Y}}_{n_{\text{TCP}_\text{s}}} = 1$, and $v^{\text{Y}}_{n_{\text{SYN}_\text{s}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ HTTPS request of device X from device Y
else if $v^{\text{X}}_{n_{\text{ARP}_\text{d}}} = 1$, $v^{\text{Y}}_{n_{\text{ARP}_\text{d}}} = 1$, $v^{\text{Z}}_{n_{\text{ARP}_\text{s}}} = 1$, $v^{\text{X}}_{n_{\text{UDP}_\text{d}}} = -1$ or -2, $v^{\text{Y}}_{n_{\text{UDP}_\text{d}}} = -1$ or -2, and $v^{\text{Z}}_{n_{\text{UDP}_\text{d}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ ARP spoof against devices X,Y from device Z
else if $v^{\text{X}}_{n_{\text{TCP}_\text{d}}} = 1, v^{\text{X}}_{n_{\text{MB}_\text{d}}} = 1, v^{\text{X}}_{n_{\text{SYN}_\text{d}}} = 1, v^{\text{X}}_{n_{16_\text{d}}} = 1, v^{\text{Y}}_{n_{\text{TCP}_\text{s}}} = 1, v^{\text{Y}}_{n_{\text{MB}_\text{d}}} = 1, v^{\text{Y}}_{n_{\text{MB}_\text{s}}} = 1$, $v^{\text{Y}}_{n_{\text{SYN}_\text{s}}} = 1$, and $v^{\text{X}}_{P_{\text{fmean}}} = -2$ within T_{eval} **then**
  prediction $\leftarrow$ FCIA against device X from device Y with physical impact A (here, switch X off)
else if $v^{\text{M}}_{n_{\text{UDP}_\text{s}}} = 1$, $v^{\text{M}}_{|P_{\text{sum}}|} = 1$, $v^{\text{M}}_{P_{\text{fmean}}} = -1$, and $v^{\text{X}}_{P_{\text{bmean}}} = -1$ within T_{eval} **then**
  prediction $\leftarrow$ FDIA against meter M with physical impact A on device X (here, battery charging)
else if $v^{\text{M}}_{n_{\text{UDP}_\text{s}}} = 1$, $v^{\text{M}}_{|P_{\text{sum}}|} = 1$, $v^{\text{M}}_{P_{\text{fmean}}} = 0$, and $v^{\text{X}}_{P_{\text{osc}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ Replay attack ag. meter M with physical impact A on device X (here, battery oscillation)
else
  prediction $\leftarrow$ Unknown abnormal behavior

3.2.3.6 Demonstration

The application of CyPhERS to the PV-battery system study case is demonstrated on the example of ARP spoofing and FCIAs. A complete evaluation on all attack types of the study case can be found in **Paper E**. Moreover, a demonstration of joint cyber attack and physical fault identification as well as a benchmarking with existing detection concepts is provided in **Paper D**.

ARP spoofing attacks Figure 3.15 depicts the event signatures provided by CyPhERS' Stage 1 during the two ARP spoofs together with the predictions of the rule-based signature evaluation system (Stage 2). System zones without flagged anomalies are not shown. In both cases, signatures match the predefined signature of ARP spoofing attacks (see Figure 3.14). Consequently, the rule-based system predicts ARP spoofing against PV3/DM and PV4/DM, respectively, from an attacker located

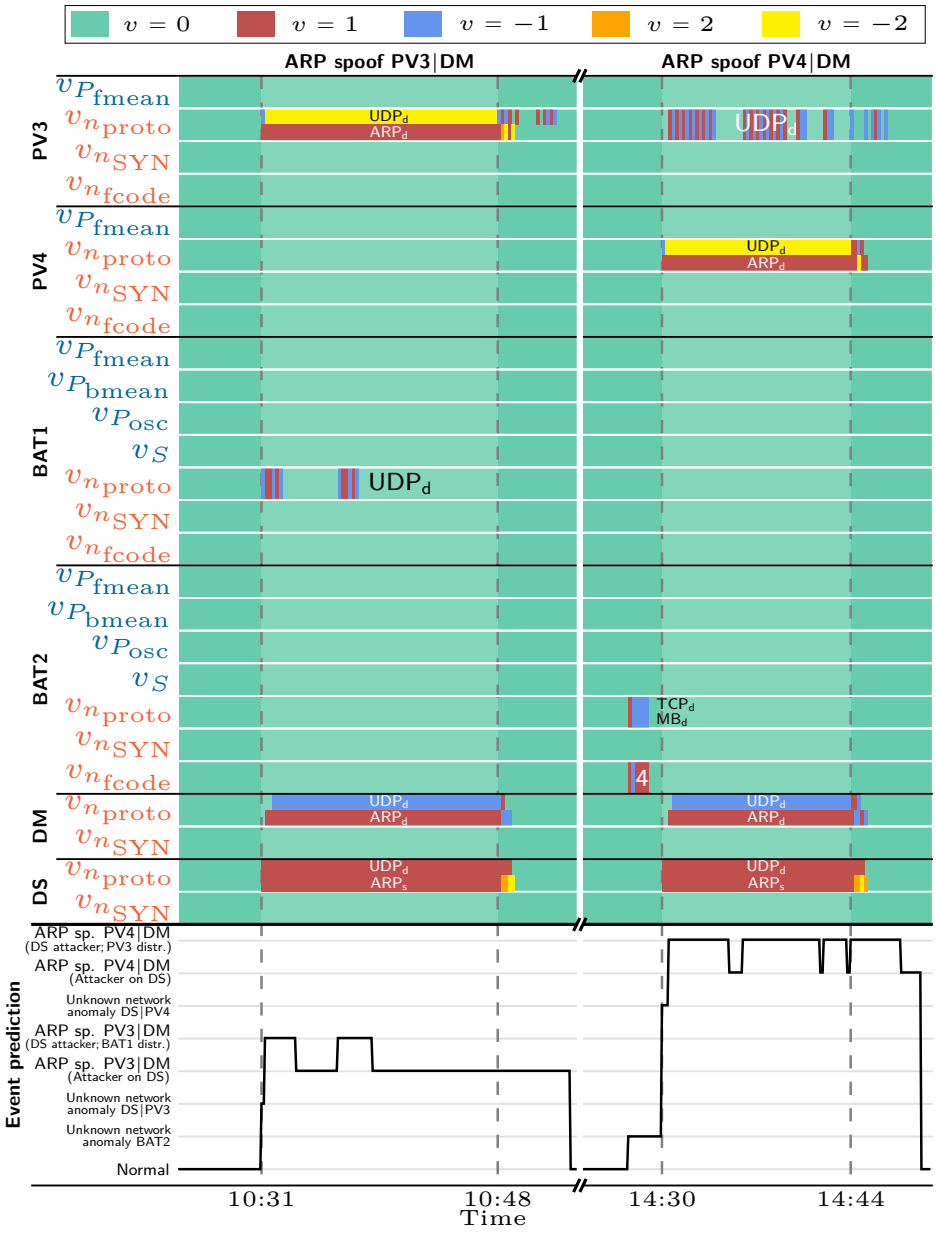


Figure 3.15: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the ARP spoofs.
Source: Illustration based on **Paper E**.

on the DS. The parallel network anomaly flags for BAT1 and PV3, respectively, indicate that the attacks distract the DM which affects its UDP communication pattern with non-victim devices. As a result, the predictions switch between ARP spoofing with and without parallel traffic distraction of other devices (see Figure 3.15). Shortly before the second ARP spoof, an additional unknown network anomaly for BAT2 is identified. This example demonstrates that CyPhERS can provide information also for unknown event types in an automated fashion, including their occurrence, affected devices, and differentiation between cyber, physical and cyber-physical events.

Figure 3.16 illustrates the prediction and flagging of the underlying anomaly detection and classification pipelines for (a) $n^{\text{PV4}}_{\text{ARP}_\text{d}}$ and (b) $n^{\text{DM}}_{\text{UDP}_\text{d}}$. From Figure 3.16 (a) it becomes evident that the ARP spoofs lead to pronounced global anomalies in the number of ARP packets which are instantly detected. Due to the non-deterministic occurrence of ARP packets during regular operation, the GBDT model is unable to learn the small peaks. Instead, it captures them with a constant PI, which demonstrates that the model approximates a simple static but precise threshold for target features which lack learnable patterns. Figure 3.16 (b) shows that the DM's UDP packet pattern maintains during ARP spoofing attacks due to communication with non-victim devices, however, on a lower level. The level decrease is successfully detected by the underlying pipeline.

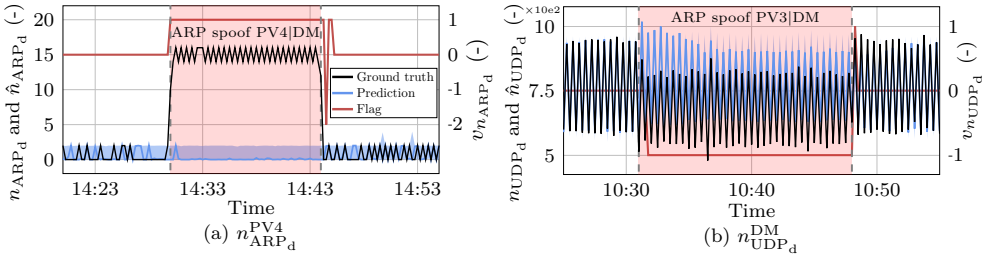


Figure 3.16: Ground truth, prediction (98 % PI and median), and anomaly flag for (a) $n^{\text{PV4}}_{\text{ARP}_\text{d}}$ during second ARP spoof, and (b) $n^{\text{DM}}_{\text{UDP}_\text{d}}$ during first ARP spoof.
Source: Illustration based on **Paper E**.

False command injection attacks The event signatures of Stage 1 and the rule-based predictions of Stage 2 during the four FCIAs are illustrated in Figure 3.17. The anomaly flags correspond to the FCIA signature in every instance (see Figure 3.14), enabling identification of the attack type, victims, attacker location, and physical impact, as indicated by the rule-based predictions. The detection of false command-induced anomalies is depicted in Figure 3.18 on the case of the FCIA against PV1 and $n^{\text{DS}}_{\text{MB}_\text{s}}$. The associated GBDT model learned that large peaks occur at full hours, and that small positive peaks are typically followed by negative ones. Since the peaks resulting from injection of false commands are not followed by such a negative peak,

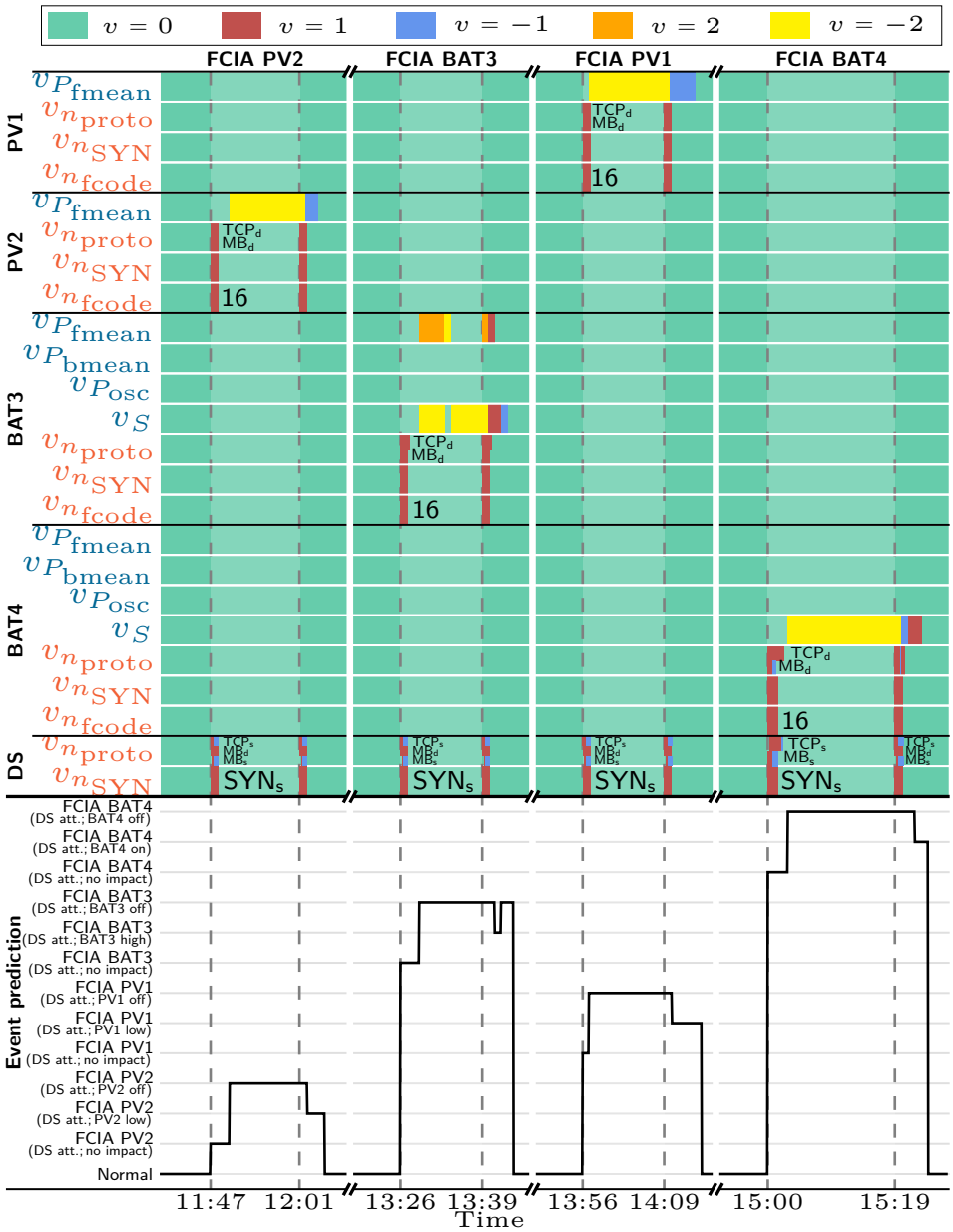


Figure 3.17: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the FCIAs.
Source: Illustration based on **Paper E**.

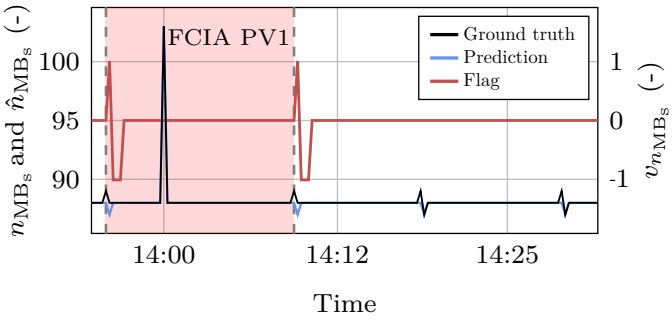


Figure 3.18: Ground truth, prediction (98 % PI and median) and anomaly flag for $n^{\mathrm{DS}}_{\mathrm{MB_s}}$ during the FCIA against PV1.
Source: Illustration adapted from **Paper E**.

they are flagged as anomalies. This demonstrates that the use of data-driven time series models enables detection of complex local anomalies.

The presence of yellow or orange flags ($v = -2$ or 2) in physical target features of the victim devices (see Figure 3.17) indicate that the attacker switched off the respective inverter. Figure 3.19 depicts the detection[2] of the FCIA-induced physical impacts on the example of (a) $P^{\mathrm{PV1}}_{\mathrm{fmean}}$ and (b) $P^{\mathrm{BAT3}}_{\mathrm{fmean}}$. It can be seen that modeling physical target features exhibits greater uncertainties in contrast to network traffic modeling. Consequently, less pronounced impacts may be missed as the case for $P^{\mathrm{BAT4}}_{\mathrm{fmean}}$ and $P^{\mathrm{BAT4}}_{\mathrm{bmean}}$ during the FCIA against BAT4. In contrast, the switch-off is indicated by anomalies in the battery state S^{BAT4}, which underlines the significance of such abstracting features for applying CyPhERS on DERs with pronounced uncertainty.

[2]Note that the predicted abrupt transition from charging to discharging in Figure 3.19 (b) is a consequence of the battery usually compensating for the compressor load peak if not switched off.

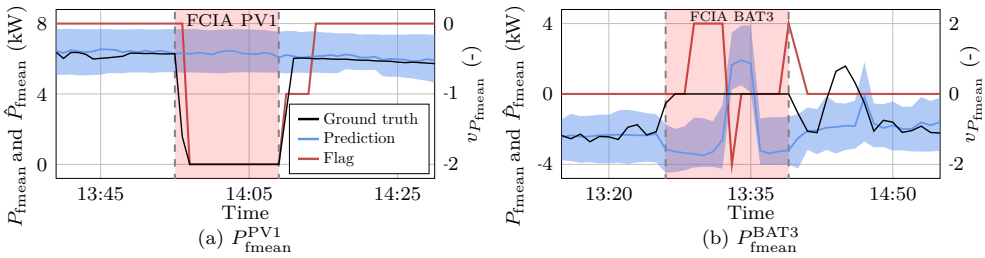


Figure 3.19: Ground truth, prediction (98 % PI and median) and anomaly flag for (a) $P^{\mathrm{PV1}}_{\mathrm{fmean}}$ at FCIA against PV1, and (b) $P^{\mathrm{BAT3}}_{\mathrm{fmean}}$ at FCIA against BAT3.
Source: Illustration based on **Paper E**.

3.2.4 Remaining barriers for cyber-physical DER monitoring

The results of Section 3.2.2 demonstrate that cyber-physical monitoring enables integrated real-time identification and differentiation of cyber attacks, cyber-physical attacks and physical faults, and improves the event identification performance compared to exclusive monitoring of cyber network traffic, which both constitute important factors for facilitating timely and adequate incident response. With the introduction of CyPhERS, these advantages are leveraged for DER monitoring while central practical requirements, including the independence of scarce attack and fault samples and verifiability of the event predictions, are taken into account. Although CyPhERS eliminates fundamental barriers for an application of cyber-physical DER monitoring in real environments, some remaining practical challenges must be addressed. While the implementation of CyPhERS is transferable among systems comprising the same digital and physical architectures and processes (e.g., residential PV-battery systems of the same vendor), it is not generalizable across different DER types. Necessary adjustments include the selection of relevant target features and predefinition of event signatures. These customizations require expert knowledge of both the system being monitored and possible attack and failure vectors, which is typically not provided by the same stakeholder. Thus, it is important to develop concepts which automate the selection of target features and definition of event signatures by mapping common attack and failure vectors defined by information technology (IT)/OT specialists onto system-specifications provided by DER operators or vendors.

3.3 Summary and reflection

This chapter empirically evaluates several influencing factors on price-based flexibility realization, and proposes data and modeling strategies on the basis of ML-based predictive analytics to address operational awareness requirements for DERs related to sub-optimal flexibility realization and cyber vulnerability. The overarching objective is to facilitate efficient, cost-optimal and secure use of DERs as flexible resources.

Based on content from **Paper B**, the impact of consumer- and weather stochasticity, computational constraints, and adversarial data manipulations on the financial reward of using a residential PV-battery system in a price-based flexibility mechanism is evaluated. For that purpose, the economic benefit under several scenarios of forecasting-based battery schedule optimization is evaluated on two real prosumer cases with substantially different production and consumption behavior. A theoretical maximum benefit of $B^{\text{oracle}} = 466\,€$ (prosumer 1) and $555\,€$ (prosumer 2) over a 14-month period is determined. It is shown that consumer and weather stochasticity affect the involved PV and load forecasts and consequently reduce the economic benefit to prosumers by about 5-10 percentage points, which is difficult to avoid even with sophisticated forecasting models. The results further suggest that virtually the same financial reward can be achieved with lightweight default GBDT forecasting models and almost no data history. Key factors are weekly model retraining and

use of weather forecasts as model input. In contrast, applying often used persistence forecasts lowers the economic benefit by further 7-12 percentage points. It is also demonstrated that manipulation of the weather inputs for GBDT-based forecasting reduces the economic benefit only by 2-3 percentage points, as the models learn to reduce weight on these compromised features via regular retraining. These findings suggest that computational constraints, short data histories (e.g., new PV plant), and weather input manipulation have low impact on a prosumer's financial reward from price-based flexibility, since they hardly affect the advantageous performance of the applied ML-based forecasts. Based on these findings, the use of a default GBDT model considering weather forecasts as input and weekly retraining is recommended to achieve near-optimal price-based flexibility realization for a residential PV-battery system, irrespective of possible computational limitations, short data histories, and weather input manipulation. It is further demonstrated that adversarial price manipulation can have significant impact on a prosumer's economics, and put stress on grids by reinforcing load peaks. Thus, it is suggested to equip DERs which are used as flexible resource with advanced concepts for real-time identification of such cyber-physical attacks.

On the basis of content from **Paper C**, cyber-physical event identification is systematically assessed as a potential strategy to improve real-time monitoring of DERs. Several supervised event identification pipelines are applied to a highly imbalanced multi-class classification problem comprising different types of cyber(-physical) attacks and physical faults. The pipelines are either trained exclusively on features extracted from cyber network traffic or on a cyber-physical feature set, which adds attributes from physical process data. By switching to the cyber-physical set, the F_1^m score averaged over all pipelines improves by 15.5 percentage points. Moreover, it is demonstrated that cyber-physical event identification allows to detect and classify cyber attacks, cyber-physical attacks, and physical faults in one integrated approach. Both the improved performance and the progression towards holistic event identification constitute important factors for facilitating timely and adequate DER incident response. Motivated by the findings of **Paper C**, the **Cy**ber-**Ph**ysical **E**vent **R**easoning **S**ystem CyPhERS is introduced based on content from **Paper D** and **Paper E**. CyPhERS leverages the advantages of cyber-physical monitoring while addressing practical requirements such as the independence of scarce attack and fault samples, verifiability of event predictions, and information provision for both known and unknown event types. CyPhERS is a two-stage process which evaluates network traffic and physical process data to infer information on events such as their occurrence, type, affected devices, and physical impact in real-time. Stage 1 produces informative and interpretable event signatures by leveraging methods including cyber-physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. In Stage 2, event information are derived from the signatures either through interpretation by the operator or automated matching with a database of predefined signatures. CyPhERS is tested on data collected from own experiments on a real PV-battery system, covering several cyber and cyber-physical attacks. The results demonstrate that CyPhERS automatically identifies the evalu-

ated attack types in real-time, and provides information on included unknown events such as occurrence, affected devices, and classification as cyber, physical or cyber-physical event. At the same time, CyPhERS is independent of historical event observations, and allows verification of automatically generated event predictions due to provision of human-readable and -interpretable event signatures. A remaining practical challenge is the selection of monitored system variables (target features) and the predefinition of event signatures as they require to combine expert knowledge of both the monitored system and possible attack and failure vectors.

While this chapter is focused on data and modeling strategies for operational challenges on DER level, the findings are also of relevance for distribution grid operation. The simple realization of near-optimal price-based flexibility can be a motivation for residential DER owners to actively manage and optimize their consumption, and thus has the potential to exploit new flexibility capacities in distribution grids. Such increased and near-optimal price-based flexibility realization can foster efficient distribution grid operation through more effective load balancing and peak shaving in case of local pricing schemes. Another aspect is the integration of CyPhERS systems of individual DERs to form a bottom-up security architecture for distribution grids. Attack reports from several distributed CyPhERS systems could be gathered and collectively analyzed by a cyber security incident response team (CSIRT). The CSIRT could then inform DSOs about possible coordinated attacks against DERs and other cyber incidents within their networks.

CHAPTER 4

Predictive analytics for distribution grid monitoring under high shares of flexible assets

This chapter presents main findings of **Paper F** and **Paper G** on research topic 3 (**RQ3**). In this context, previously identified operational awareness requirements on distribution grid level are addressed, which are related to flexibility-induced stochasticity and DSO-unaware flexibility activations. Section 4.1 evaluates the impact of local flexibility on the performance of data-driven LV state estimation based on **Paper F**, and derives recommendations for providing accurate and reliable estimations in scenarios of high shares of flexible resources. Section 4.2 introduces a flexibility activation identification concept for DSOs based on **Paper G**, and thereby presents a use case for LV state estimations. Finally, Section 4.3 summarized the chapter and reflects on key outcomes.

4.1 Flexibility-tolerant LV state estimation

This section systematically evaluates the impact of frequent flexibility activations on the performance of data-driven LV state estimation on the case of a real suburban medium voltage (MV)-LV network and load profiles based on **Paper F**. Section 4.1.1 provides a brief overview of related works and the contribution. In Section 4.1.2, the evaluation approach is detailed. The considered study case is introduced in Section 4.1.3. Thereafter, the performance of data-driven LV state estimation is evaluated under several local flexibility scenarios (Section 4.1.4) and model input information scenarios (Section 4.1.5). On that basis, Section 4.1.6 provides recommendations on data and modeling strategies for flexibility-tolerant LV state estimation.

4.1.1 Related works and contribution

Data-driven LV state estimation is frequently addressed in the literature, applying different methods including ANNs [40] and analog-search techniques [41]. Some works apply probabilistic methods to quantify uncertainty potentially introduced by the lower metering coverage and fluctuating renewable generation [41, 42]. The influence of different PV penetration levels on the estimation performance is evaluated in [43]

and [41]. However, the impact of local flexibility mechanisms on data-driven LV state estimation has not been addressed yet. In **Paper F**, a systematic evaluation of the accuracy and uncertainty of data-driven LV state estimation under several flexibility usage scenarios is conducted, and data and modeling strategies for flexibility-robust estimations derived.

4.1.2 Evaluation approach

Data-driven LV state estimation is a promising concept for underdetermined distribution systems due to the capability to cope with low meter coverage by leveraging historical data from smart meters and other data sources. In Section 2.4.2 it is discussed that local flexibility mechanisms may increase load volatility and stochasticity, and thus entail less reliable data-driven LV state estimation by breaking or complicating the model's input-output correlation. In this section, data-driven LV state estimation is systematically evaluated under several flexibility and model input information scenarios. More precisely, a BNN is applied to quantify both epistemic and aleatoric uncertainty in the estimation of LV states under all considered scenarios. Epistemic (model) uncertainty arises from a lack of knowledge about the modeled process. It is associated with insufficient data or model assumptions and can be reduced through use of more data or better suited models. Flexibility activations may complicate the state estimation problem due to increased load variability and therefore require larger data quantities to sufficiently describe the more complex relations. Thus, they potentially increase epistemic uncertainty. Aleatoric (data) uncertainty refers to the inherent randomness in the modeled process and cannot be reduced with larger data histories or more complex models. Flexibility activations may entail load ambiguity so that the same model input (e.g., substation measurements) can be mapped to several LV states. Thus, they potentially introduce aleatoric uncertainty. By considering different forms of estimation uncertainty that potentially are introduced by flexibility activations, the impact of local flexibility on the reliability of data-driven LV state estimation can be comprehensively quantified.

BNN description BNNs combine Bayesian uncertainty quantification with the predictive power of ANNs [44]. In contrast to traditional ANNs, the model parameters (weights and biases) are given as conditional probability distributions, representing their uncertainty. The BNN can be represented by

$$[\hat{y}, \hat{\sigma}^2] = f_{\mathrm{BNN}}^{\hat{W}}(\boldsymbol{u}), \tag{4.1}$$

with $\hat{\boldsymbol{W}}$ being the set of model parameters, and $\boldsymbol{u}$ the input feature vector. The model outputs estimates of both the predictive mean $\hat{y}$ and variance $\hat{\sigma}^2$, allowing to consider aleatoric uncertainty. Let $\boldsymbol{U}_{\mathrm{train}} = \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{N_{\mathrm{train}}} \mid \boldsymbol{u}_i \in \mathbb{R} \ \forall i\}$ and $\boldsymbol{Y}_{\mathrm{train}} = \{y_1, \ldots, y_{N_{\mathrm{train}}} \mid y_i \in \mathbb{R} \ \forall i\}$ be the training input and output data, respectively, of size N_{train}. The posterior distribution $p(\boldsymbol{W}|\boldsymbol{U}_{\mathrm{train}}, \boldsymbol{Y}_{\mathrm{train}})$ over the model parameters, given the training data $\{\boldsymbol{U}_{\mathrm{train}}, \boldsymbol{Y}_{\mathrm{train}}\}$ and Gaussian priors $\mathcal{N}(0, I)$ is

calculated by Bayes' rule. The predictive distribution for an observation $\boldsymbol{u}$ is derived from marginalizing over the posterior distribution [45], following

$$p(y|\boldsymbol{u}, \boldsymbol{U}_{\text{train}}, \boldsymbol{Y}_{\text{train}}) = \int p(y|\boldsymbol{u}, \boldsymbol{W}) p(\boldsymbol{W}|\boldsymbol{U}_{\text{train}}, \boldsymbol{Y}_{\text{train}}) \mathrm{d}\boldsymbol{W}. \tag{4.2}$$

The true posterior is typically intractable due to non-linearity and -conjugacy. Instead, the posterior is approximated by minimizing the Kullback-Leibler divergence between $p(\boldsymbol{W}|\boldsymbol{U}_{\text{train}}, \boldsymbol{Y}_{\text{train}})$ and a surrogate distribution. To enable simultaneous output of $\hat{y}$ and $\hat{\sigma}^2$ as shown in (4.1), the loss function according to

$$\mathcal{L}_{\text{BNN}}(\hat{y}, y, \hat{\sigma}) = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \frac{1}{2\hat{\sigma}_i^2} ||y_i - \hat{y}_i||^2 + \frac{1}{2} \log \hat{\sigma}_i^2, \tag{4.3}$$

is applied [46].

Taking epistemic uncertainty into account requires multiple predictions of $\hat{y}$ and $\hat{\sigma}^2$ for the same input $\boldsymbol{u}$, each based on a new set of sampled model parameters $\hat{\boldsymbol{W}}_d$. As a result, the predictive mean of the BNN follows

$$\widetilde{\mathbb{E}}(y) \approx \frac{1}{N_{\text{sample}}} \sum_{d=1}^{N_{\text{sample}}} f_{\text{BNN}}^{\hat{W}_d}(\boldsymbol{u}), \tag{4.4}$$

where N_{sample} denotes the number of sampled predictions. The predictive uncertainty is approximated by

$$\begin{aligned} \widetilde{\text{Var}}(y) \approx & \underbrace{\left[\frac{1}{N_{\text{sample}}} \sum_{d=1}^{N_{\text{sample}}} \hat{y}_d^2 - \left(\frac{1}{N_{\text{sample}}} \sum_{d=1}^{N_{\text{sample}}} \hat{y}_d \right)^2 \right]}_{\text{epistemic}} \\ & + \underbrace{\frac{1}{N_{\text{sample}}} \sum_{d=1}^{N_{\text{sample}}} \hat{\sigma}_d^2}_{\text{aleatoric}}. \end{aligned} \tag{4.5}$$

For an in-depth description of BNNs, readers are referred to [46]. As a benchmark for the BNN, linear regression is applied and trained to estimate quantiles by minimizing the quantile loss according to (3.15), which in the following is referred to as linear quantile regression (LQR).

Performance metric A comprehensive performance quantification of probabilistic models requires to take several attributes such as reliability and sharpness into account [47]. Reliability refers to the degree of proximity between the predicted distribution and the actual distribution. For example, if a 90 % PI covers 90 % of the

observations, the estimates are considered reliable. On the other hand, sharpness addresses the tightness of PIs, and thus expresses how informative the estimates are. The averaged Pinball loss constitutes a comprehensive probabilistic performance metric which takes both reliability and sharpness of probabilistic estimations into account, where a lower value indicates better performance [47]. Thus, for evaluating and comparing the performance of data-driven probabilistic LV state estimation under several flexibility and information scenarios, the Pinball loss, which is calculated as

$$Pinball_t = \begin{cases} (y_t - \hat{y}_t^q)q, & y_t \geq \hat{y}_t^q \\ (\hat{y}_t^q - y_t)(1-q), & y_t < \hat{y}_t^q \end{cases} \tag{4.6}$$

for a time step t and a quantile q, is considered and averaged over the quantiles $q = 0.01, \ldots, 0.99$.

4.1.3 Study case

This section presents the study case, which involves description of the network and customer profiles (Section 4.1.3.1), estimation problem (Section 4.1.3.2), evaluated scenarios (Section 4.1.3.3), and model implementation (Section 4.1.3.4).

4.1.3.1 Network and customer profiles

The study is based on a real suburban MV-LV network which serves a total of 564 residential customers located in Bornholm, Denmark (see Figure 4.1). It comprises six 10/0.4 kV secondary substations connected to a 60/10 kV primary substation, which is referred to as SubP. The high voltage side of SubP serves as the reference voltage, set at 1 pu. While the networks below most secondary substations are represented as load buses constituting aggregated (re-)active power of the associated customers, the network below substation SuBS is considered in more detail.

The applied residential customer profiles consist of real five-minute average active and reactive power measurements for the year 2018, collected during the EcoGrid 2.0 project [48]. Most customers use a heat pump or electric heater with an average yearly consumption of ∼8 MWh, while around 10 % have PV systems of ∼6 kW_p installed. Some profiles include limited flexible operation of heating loads, as a result of experimental demonstration of an LFM.

4.1.3.2 Estimation problem

DSOs typically have real-time observability up to the primary substation, and in some cases to the secondary level. The considered problem is to provide a probabilistic estimation of voltages at nodes 1 to 6 (see Figure 4.1) under different flexibility usage and input information scenarios. It is assumed that DSOs have access to SM energy readings with a daily delay. Based on these data and an accurate network model,

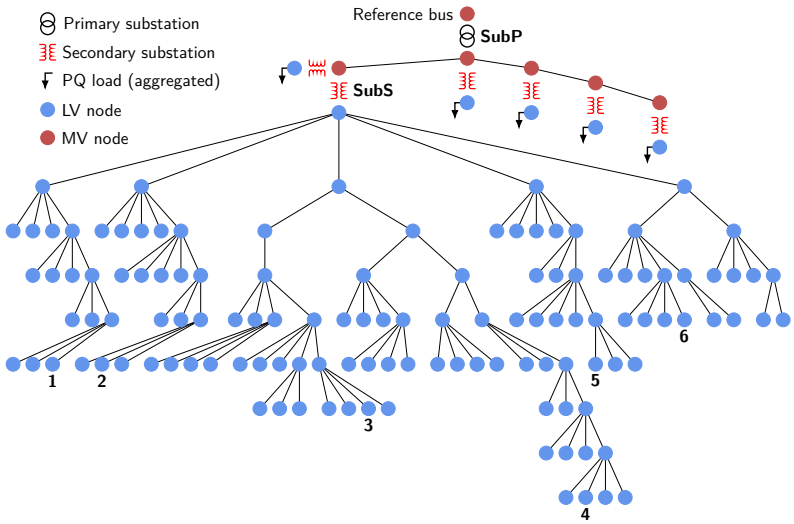


Figure 4.1: Depiction of the real MV-LV network used for evaluating data-driven LV state estimation under several flexibility scenarios, and predicted node voltages 1-6.
Source: Illustration adapted from **Paper F**.

the DSO can create a historical dataset containing the relevant voltages by running power flows.

4.1.3.3 Evaluated scenarios

The flexibility from heating loads in the original customer profiles is marginal. To study the impact of larger shares of flexible resources, several scenarios are simulated in which smart EV charging profiles are added in different parts of the network. The charging periods and associated energy needs are sourced from [49]. It is assumed that users optimize the charging pattern based on day-ahead spot prices of the Danish bidding zone DK2 to minimize costs. For creating datasets of the scenarios, the customer profiles are first randomly assigned to the leaf nodes below SubS and the aggregated load profiles of the remaining secondary substations. Next, charging profiles are added on varying nodes depending on the scenario. The network voltages are obtained by running alternating current (AC) power flow applying a network model which has been validated with real network measurements. Apart from the flexibility scenarios, varying levels of DSO information availability are considered to account for different possible observability levels and to investigate how model inputs affect the estimation performance under a high share of flexible demand. The considered flexibility scenarios FS1-3 and information scenarios IS1-3 are defined in the following.

FS1: original customer profiles FS1 exclusively considers the original residential profiles, thus representing a case of mild flexibility utilization.

FS2: original customer profiles and EVs below adjacent secondary substations The customer placement of FS1 is kept, but EVs applying smart charging are added to each customer below the five adjacent secondary substations of SubS. FS2 showcases significant load flexibility, however, exclusively below adjacent substations.

FS3: original customer profiles and EVs below all secondary substations An EV performing smart charging is additionally added to each customer under SubS. FS3 presents a case of pronounced load flexibility within the entire distribution grid.

IS1: low information availability The DSO has access to SM load readings with a delay of 24 hours, real-time weather data (temperature and solar irradiation), calendric features (weekday vs. weekend indicator and time of day), and spot prices. IS1 assumes zero DSO real-time grid observability.

IS2: typical information availability Real-time active and reactive power measurements from the primary substation SubP are added to the features from IS1.

IS3: high information availability Real-time (re-)active power and voltage readings from the secondary substation SubS are added on top of the features from IS2.

4.1.3.4 Model implementation

For training, selection and evaluation of the state estimation models, the scenario datasets are split into a training, validation and test set with a partition of 80/10/10. Model selection is realized by minimizing the validation loss, resulting in using Adam optimizer, tanh activation function, two hidden layers, and a batch size of 64 in all scenarios. The epochs range from 2000 to 10000, while the units in the hidden layers vary between 5 and 12. Before being applied on the test set, the selected models are retrained on the whole training and validation data of the respective scenario. The BNN is implemented in Python applying the *Tensorflow Probability* library [50].

4.1.4 Flexibility scenario evaluation

The three flexibility scenarios are evaluated for IS2, which constitutes the most typical information availability scenario for DSOs.

Qualitative evaluation Figure 4.2 depicts estimates of the BNN and the associated actual observations for the same period under FS1-3. In all cases, epistemic uncertainty is marginal. The high certainty about model parameters indicates that the approximately eleven months of training data are sufficient to learn the existing correlation between primary substation measurements and LV states. This also holds for FS2 and FS3, where load variability from flexible resources potentially results in

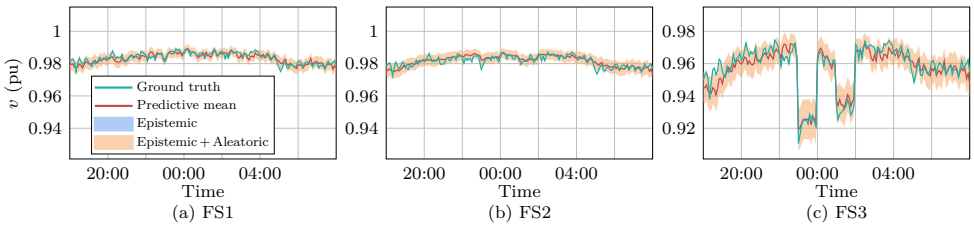


Figure 4.2: Predictive mean and 90 % PI, split into epistemic and aleatoric uncertainty, provided by the BNN for an excerpt of FS1-3 based on IS2 and node voltage 4.
Source: Illustration based on **Paper F**.

more complex input-output relations. It can be concluded that possible load volatility associated with local flexibility has minor impact on data-driven LV state estimation.

Instead, aleatoric uncertainty is dominating in all scenarios, which cannot be reduced by larger data quantities or more complex models as it results from process-inherent randomness. The quality of estimates under FS1 and FS2 is similar, which suggests that flexible resources below adjacent secondary substations have limited impact on data-driven LV state estimation. In contrast, aleatoric uncertainty increases under FS3, which indicates that flexible resources within the same LV network lower the correlation between primary substation measurements and LV states by introducing randomness.

Quantitative evaluation Figure 4.3 illustrates the quantitative impact of the different flexibility scenarios on the performance of probabilistic data-driven LV state estimation in form of the Pinball loss averaged over all estimated node voltages. The values are scaled to the weakest performance under all considered flexibility and information scenarios. The results support the initial findings from the qualitative evaluation. Flexible resources below adjacent secondary substations (FS2) only marginally

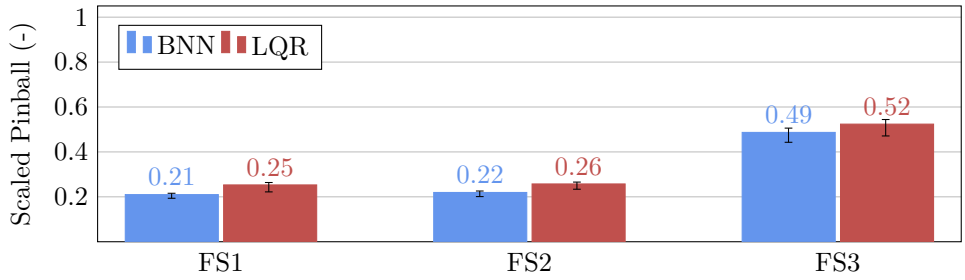


Figure 4.3: Pinball score of the BNN and LQR for FS1-3 based on IS2, averaged over the estimated node voltages 1-6 and scaled to the largest score across all scenarios. Black bars indicate min. and max. values among individual node voltage scores.
Illustration based on **Paper F**.

decrease estimation performance. On the other hand, the Pinball score significantly increases under FS3 due to the introduced randomness, which forces the models to widen the PIs and thereby reduces the sharpness of the probabilistic estimates. It can be concluded that flexible resources below the same secondary substation entail less informative data-driven LV state estimations. From Figure 4.3 it is further noticeable that the BNN performs better compared to LQR in all flexibility scenarios, which can be an argument for using such more advanced but computational intensive models.

4.1.5 Information scenario evaluation

The results in Section 4.1.4 indicate that aleatoric uncertainty is dominant in the data-driven LV state estimation problem, and further increases through flexible resources below the same secondary substation (FS3). While larger data histories and more complex models cannot reduce aleatoric uncertainty, additional data sources potentially can by altering the considered regression problem. Therefore, the impact of the three DSO information availability scenarios on the estimation performance is evaluated in the following on FS3.

Qualitative evaluation Figure 4.4 depicts estimates of the BNN and the associated actual observations for a representative period of FS3 under the information scenarios IS1-3. It can be seen that, irrespective of the model input, aleatoric uncertainty is dominating while epistemic uncertainty is marginal, which supports the findings from Section 4.1.4. Under IS1, the BNN entirely misses pronounced voltage drops resulting from simultaneous activation of multiple EV charging processes. In contrast, the drops are captured under IS2 and IS3. It can be concluded that real-time distribution grid measurements from primary substation level or below are crucial for data-driven LV state estimation under high shares of flexible resources. Moreover, the additional incorporation of real-time measurements from the secondary substation SubS (IS3) seems to further reduce aleatoric uncertainty, and thus enable more informative estimates under local flexibility. Another finding from Figure 4.4 is that the PIs increase during the voltage drops. This indicates that the BNN provides PIs

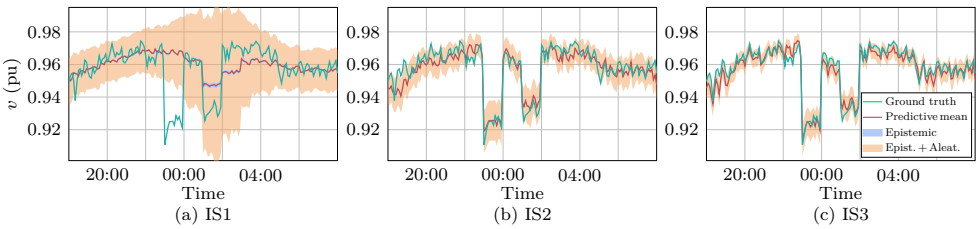


Figure 4.4: Predictive mean and 90 % PI, split into epistemic and aleatoric uncertainty, provided by the BNN under IS1-3 for an excerpt of FS3 and node voltage 4.
Source: Illustration based on **Paper F**.

of high resolution which successfully capture the increased randomness during times of pronounced flexibility activation events.

Quantitative evaluation Figure 4.5 illustrates the quantitative impact of the different DSO information availability scenarios on the performance of probabilistic data-driven LV state estimation under high shares of flexible resources in form of the scaled Pinball loss averaged over all estimated node voltages. The results support the initial findings of the qualitative evaluation. Incorporating real-time measurements from secondary substation SubS (IS3) improves the prediction performance by reducing aleatoric uncertainty induced by flexible resources and thus facilitates informative data-driven LV state estimation in a scenario of pronounced local flexibility. By using information from SubS, the BNN achieves a scaled Pinball score of 0.27 for FS3, which is in a similar performance range as in scenarios with marginal load flexibility (see FS1 in Figure 4.3).

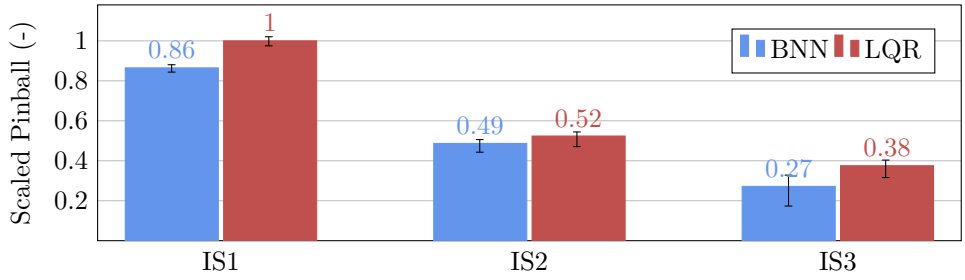


Figure 4.5: Pinball score of the BNN and LQR for IS1-3 and FS3, averaged over the estimated node voltages 1-6 and scaled to the largest score of all scenarios. Black bars indicate minimum and maximum values among individual node voltage scores.
Source: Illustration based on **Paper F**.

4.1.6 Recommendations for flexibility-tolerant LV state estimation

The findings in Section 4.1.4 and 4.1.5 allow some initial recommendations on data and modeling strategies for reliable and informative LV state estimation under high shares of flexible resources. Large-scale integration of flexible resources can introduce uncertainty to data-driven LV state estimation. Thus, probabilistic models should be considered to provide reliable estimates and thus reduce risk for misinformed grid operation. Primarily, models need to be capable of quantifying aleatoric uncertainty, which is the predominant type introduced through flexible resources. Nevertheless, quantifying epistemic uncertainty can be advantageous. On the one hand, it becomes more relevant in situations of changing process conditions, for example, new residential load patterns. On the other hand, it can indicate need for model retraining. The marginal epistemic (model) uncertainty further indicates that data histories of about one year seem to be sufficient for training probabilistic LV state estimation models.

Another finding from Section 4.1.4 and 4.1.5 is that the BNN improves the Pinball loss averaged over all considered flexibility and information scenarios by 16 % compared to LQR. Thus, the higher computational burden of such more complex models as the BNN is considered justifiable as it facilitates accurate and reliable LV state estimation. Finally, providing a BNN with real-time power and voltage readings from secondary substations enables an estimation performance close to a scenario without flexible resources. Thus, equipping secondary substations with real-time metering can be a strategy to counteract the impact of flexible resources on LV state estimation, and thus, a cost-effective way to increase observability in distribution grids without widespread installation of real-time meters below secondary substation level. Such reliable and informative LV state estimates can furthermore constitute the foundation for other distribution grid monitoring functionalities such as the flexibility activation identification concept introduced in the following section.

4.2 Flexibility activation identification for DSOs

This section presents a data-driven flexibility activation identification concept for DSOs based on **Paper G**. First, flexibility activation identification is motivated in Section 4.2.1. In Section 4.2.2, a short overview of related works and the contribution is provided. Practical requirements for flexibility activation identification are highlighted in Section 4.2.3. Thereafter, Section 4.2.4 details the developed identification pipeline. The considered study case is introduced in Section 4.2.5, followed by a demonstration of the pipeline in Section 4.2.6. Finally, Section 4.2.7 discusses remaining barriers and limitations.

4.2.1 Motivation

Section 2.5 previously described that simultaneous activation of a fleet of flexible assets can occur without active involvement of DSOs, for example, due to concurrent reaction of smart EV chargers to low-price signals. While online identification of such flexibility activations would generally improve a DSO's situational awareness, it in particular supports timely countermeasures in case of critical activations that are threatening to cause congestions or voltage violations. Another use case for flexibility activation identification is automated online verification of DSO-requested flexibility services. Although flexibility activations may also be identifiable through visual inspection of LV states and load flows in control rooms, the large number of grid nodes that potentially need to be monitored, and the risk of human failure can render exclusive manual supervision insufficient. Thus, providing mechanisms to automatically identify flexibility activations and verify successful activation of requested services has the potential to increase trust of system operators in local flexibility mechanisms.

4.2.2 Related works and contribution

Literature on event identification in distribution grids is rich. For example, unsupervised anomaly detection in load time series is investigated applying different models such as autoregressive integrated moving average (ARIMA) [51], hierarchical temporal memory (HTM) [52] and variational autoencoders [53]. Another largely addressed field is the classification of power quality events [54]. In this context, only a limited number of works examines open-set classification [55, 56], which comes with the advantage of being able to identify observations of unknown event types where traditional closed-set classifiers fail. While many works examine event detection and classification in distribution grids, the identification of flexibility activations has not been addressed yet. Therefore, **Paper G** introduces a data-driven flexibility activation identification pipeline which is based on unsupervised detection and open-set classification, and demonstrates the feasibility of the main components.

4.2.3 Requirements

Within **Paper G**, requirements that should be considered in the development of flexibility activation identification concepts are provided, which are listed in the following.

Independence of flexibility asset data DSOs typically have no or limited access to real-time data of flexible assets, in particular on residential level. Thus, concepts should be based on processing real-time power measurements or estimates of different nodes in the grid.

Real-time detection Flexibility activations should be detected at an early stage. In this way, operators can be supported in responding to possible critical activation events or failure of requested services.

Computational efficiency With a view to growing data amounts, edge computing [57] and data compression [58] may become more relevant in future distribution grid operation for avoiding data latency, integrity and availability issues. Thus, being computationally lightweight to foster distributed implementation on edge devices such as PLCs and intelligent electronic devices, and capable of processing compressed data is considered desirable for flexibility activation identification concepts.

Limited need for historical flexibility activation observations Simultaneous activation of a fleet of flexible assets in future distribution grids may only happens occasionally, depending on the implemented flexibility mechanisms. Moreover, manual extraction of flexibility activation samples from historical data is tedious and time-consuming. Consequently, the dependency of identification concepts on historical observations of flexibility events should be kept at a minimum.

Handling multiple and unknown load-altering event types Sudden load variations in distribution grids can have several backgrounds. These include physical damage or failure of grid equipment, planned switching operations and reconfiguration, and large social events. Moreover, the integrity of active power readings may be compromised by cyber attacks or communication failures, potentially entailing artificial events of similar appearance to real load modifications. Thus, identification concepts should be capable of differentiating flexibility activations from other event types in active power readings. As some of these may have never occurred in the past, distinction from both previously observed and unobserved event types is desirable.

4.2.4 Flexibility activation identification pipeline

This section describes the proposed flexibility activation identification pipeline (see Figure 4.6). The fundamental approach is introduced in Section 4.2.4.1. Thereafter, details on the main building blocks of the pipeline are provided, which comprises unsupervised event detection (Section 4.2.4.2) and open-set classification (Section 4.2.4.3).

4.2.4.1 Approach

The proposed concept aims to identify flexibility activations in active power readings or estimates of distribution grid nodes, while meeting the requirements defined in Section 4.2.3. For that purpose, it is proposed to split the event identification problem into unsupervised event detection and open-set classification. The detector is based on point-wise unsupervised anomaly detection. In case that a load-altering event is detected, the event sampler extracts the related time series sequence and forwards it to an extreme value machine (EVM) open-set classifier [59]. In contrast to an one-step identification approach which classifies the time series sequence of the most recent load observations in a rolling fashion, the proposed pipeline has several practical advantages. To classify a load-altering event, a model typically requires the full time series sequence of the event, which introduces pronounced identification delays. By decoupling event detection from identification through adding a point-wise anomaly detector upstream, operators can be informed about occurrence of a load-altering event in near real-time instead of waiting for the delayed classifier prediction. The proposed scheme further reduces computational burden as the classifier is only applied on sequences of previously detected events. The computational advantage of such event-triggered classification is complemented by using a lightweight persistence

Figure 4.6: Illustration of the proposed flexibility activation identification pipeline.
Source: Illustration adapted from **Paper G**.

forecast as foundation for the event detection step. Moreover, by applying unsupervised anomaly detection, the detection of flexibility activations is independent of historical event observations. Thus, events can be detected even in a scenario without any samples of previous activations, where a one-step classification approach would not applicable. Finally, the applied EVM open-set classifier is capable of differentiating flexibility activations from normal operation and other known and unknown load-altering event types.

4.2.4.2 Unsupervised event detection

The flexibility activation detection problem is considered as unsupervised univariate anomaly detection in 5-minutely averaged active power readings of the form $\boldsymbol{p} = \{p_1, ..., p_N \mid p_i \in \mathbb{R}\, \forall i\}$. A load observation p_t at time step t is declared abnormal if the associated anomaly score s_t exceeds a predefined threshold γ, which is expressed by the decision function

$$v_t = \begin{cases} 1 \text{ (anomaly)} & \text{if } s_t > \gamma \\ 0 \text{ (normal)} & \text{otherwise.} \end{cases} \tag{4.7}$$

The use of a trivial persistence forecast is suggested to determine s_t according to

$$s_t = |p_t - \hat{p}_t| = |p_t - p_{t-1}|, \tag{4.8}$$

where the anomaly score is defined as the distance between the ground truth p_t and the expected value $\hat{p}_t$ provided by the persistence forecast $\hat{p}_t = p_{t-1}$. In case of processing differenced load data $\Delta\boldsymbol{p} = \{\Delta p_1, ..., \Delta p_N \mid \Delta p_i = p_i - p_{i-1},\ p_i \in \mathbb{R}\, \forall i\}$ potentially resulting from data compression [60], persistence-based detection reduces to comparing the differenced values to the threshold γ. It is assumed that more complex and computationally intensive models cannot significantly improve the prediction accuracy due to the small and mainly random changes in load data of 5-minute resolution. To evaluate this hypothesis, the persistence-based detection is benchmarked against a variety of detectors applying different advanced statistical and ML forecasting models. These include HTM [61], ARIMA [62], convolutional neural network (CNN) [63], and spectral residual (SR) [64]. The models determine the anomaly score s_t in (4.7) either directly by learning a mapping function from lag values of a history window of size w_{hist} according to

$$s_t = \Phi([p_{t-1}, ..., p_{t-w_{\text{hist}}}]), \tag{4.9}$$

or indirectly via predicting the expected value used in $s_t = |p_t - \hat{p}_t|$, following

$$\hat{p}_t = \Phi([p_{t-1}, ..., p_{t-w_{\text{hist}}}]). \tag{4.10}$$

4.2.4.3 Open-set classification

The identification of previously detected abnormal load deviation events is considered as open-set classification problem. In Figure 4.7, closed- and open-set classification

is compared. The more commonly considered closed-set problem assumes that a training dataset describes all possible event classes, which is a strong assumption for distribution grids given the variety of load-altering events and rare occurrence of some. Observations of event types not considered during training are wrongly assigned to one of the known classes by a closed-set classifier, which weakens its performance.

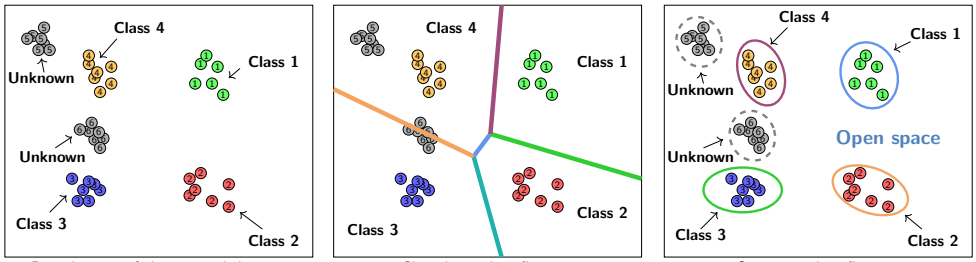


Figure 4.7: Illustrative comparison of closed- and open-set classification.
Source: Illustration adapted from **Paper G** and based on [65].

In contrast, open-set classifiers promise better performance in scenarios of limited knowledge about possible event types, because observations of unknown classes are identified as such. Let $\Delta \boldsymbol{p}_{\text{seq}} = \{\Delta p_1, ..., \Delta p_{N_{\text{seq}}} | \, \Delta p_i \in \mathbb{R} \, \forall i\}$ be the extracted sequence of a previously detected load-altering event, and $\boldsymbol{u}$ the associated feature vector consisting of the attributes listed in Table 4.1. The considered EVM classifier models event classes which are included in the training data by a set of radial inclusion functions, as schematically illustrated in Figure 4.7. Based on the radial inclusion function of an event class C_l, the EVM determines the probability $\widehat{Pr}(C_l|\boldsymbol{u})$ that a newly detected event represented by $\boldsymbol{u}$ belongs to C_l. The decision function of the EVM is given by

$$\hat{y}^* = \begin{cases} \arg\max_{l \in \{1,...,N_{\text{classes,train}}\}} \widehat{Pr}(C_l|\boldsymbol{u}) & \text{if } \widehat{Pr}(C_l|\boldsymbol{u}) \geq \rho \\ unknown & \text{otherwise,} \end{cases} \tag{4.11}$$

Table 4.1: Extracted features of the time series sequence $\Delta \boldsymbol{p}_{\text{seq}}$ of previously detected load-altering events, which are used as input in the classification step.
Source: Table adapted from **Paper G**.

Feature	Definition
Mean $\mu_{\Delta \boldsymbol{p}_{\text{seq}}}$	$\frac{1}{N_{\text{seq}}} \left(\sum_{i=1}^{N_{\text{seq}}} \Delta p_i \right)$
Standard deviation $\sigma_{\Delta \boldsymbol{p}_{\text{seq}}}$	$\sqrt{\frac{1}{N_{\text{seq}}-1} \sum_{i=1}^{N_{\text{seq}}} (\Delta p_i - \mu_{\Delta \boldsymbol{p}_{\text{seq}}})^2}$
Minimum value $\Delta p_{\min}$	$\min(\Delta \boldsymbol{p}_{\text{seq}})$
Maximum value $\Delta p_{\max}$	$\max(\Delta \boldsymbol{p}_{\text{seq}})$
Number of zeros n_0	$\text{count}(\Delta p_i \overset{!}{=} 0 \in \Delta \boldsymbol{p}_{\text{seq}})$
Steps between min. and max. value n_{minmax}	$\text{abs}(\text{index}(\Delta p_{\min}) - \text{index}(\Delta p_{\max}))$

where ρ is a threshold defining the boundary between the set of known classes and the unknown open space, $N_{\text{classes,train}}$ the number of known classes, and $\hat{y}^*$ the predicted class. A second variant of the EVM with $\rho = 0$ is considered as closed-set classification benchmark model.

4.2.5 Study case

This section describes the considered study case, which involves presenting the datasets used for evaluating unsupervised flexibility activation detection (Section 4.2.5.1) and open-set classification (Section 4.2.5.2), as well as the associated model implementation (Section 4.2.5.3).

4.2.5.1 Dataset for evaluating flexibility activation detection

The active power dataset is constructed by aggregating the load of 450 households and extends over a period of 6.5 month, starting from September 15, 2017. The 5-minute average residential load profiles were collected during the EcoGrid 2.0 project [48], where heat pumps and electric heaters were controlled by adjusting temperature set-points or throttle signals in the context of an LFM demonstration. The resulting load reduction and increase events are illustrated in Figure 4.8. Activation periods are in the range of 30-120 minutes. Each observation of the dataset is labeled either as normal operation or as being part of a flexibility activation. In total, 205 flexibility activation events are considered.

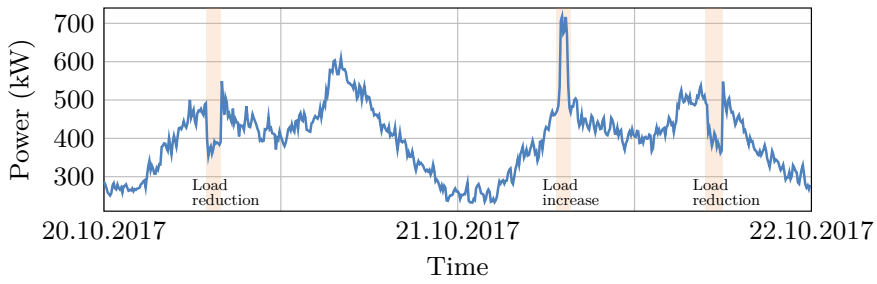


Figure 4.8: Excerpt of the active power dataset and included flexibility activation events used to evaluate flexibility activation detection.
Source: Illustration adapted from **Paper G**.

4.2.5.2 Dataset for evaluating open-set classification

For the evaluation of open-set flexibility activation classification, five event classes are considered (see Figure 4.9). It is assumed that historical observations of the classes flexibility activation (FA) and normal operation (NO) are available. Thus, 205 associated time series sequences are extracted from the load dataset for each

of them, where 90 % of the sequences are reserved for model selection and training, and the remaining share for testing. The Monday peak (MP), frozen value (FV), and data unavailability (DU) classes are assumed to be unknown and thus only considered during model evaluation. The MP class comprises real load peaks occurring every Monday at 8 am due to collective heat up of electric water boilers to avoid bacteria growth. The FV class simulates a data measurement or communication failure where power readings stagnate over a certain period. In the DU class, it is assumed that a subset of individual measurements forming an aggregated active power data stream (e.g., neighborhood consumption based on individual SMs) is unavailable. Seven instances of each unknown class are added to the test dataset. For the time series sequences of all event instances, the features summarized in Table 4.1 are extracted, together forming the final dataset.

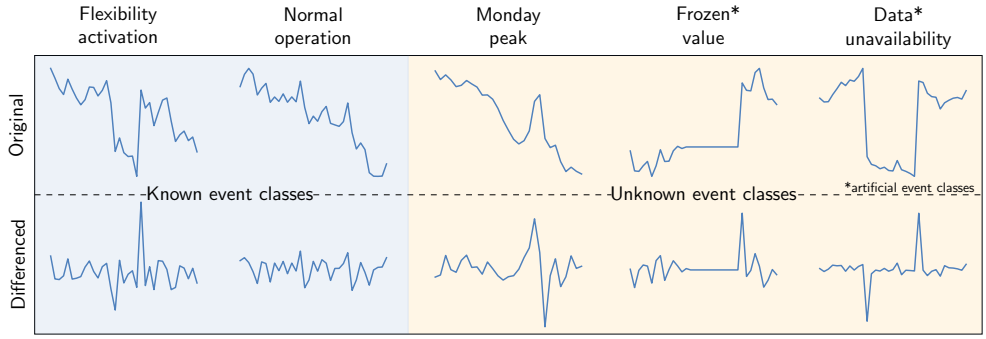


Figure 4.9: Illustration of the load-altering events considered for evaluating open-set flexibility activation classification.
Source: Illustration based on **Paper G**.

4.2.5.3 Model implementation

The triviality of the proposed persistence forecast-based detector renders model selection and (re-)training unnecessary. Details on the implementation of the detectors applying statistical and ML forecasting models, which are considered for benchmarking, can be found in **Paper G**. The hyperparameter selection for the EVM classifier is realized by applying a 5-fold cross-validation on the training set. The features are standardized within every individual split, avoiding data leakage. For selecting the threshold ρ, the model exhibiting the smallest ρ while still fulfilling the classification performance requirement $F_1^m \geq 0.8$ (see (3.13)) during cross-validation is selected. The resulting hyperparameters are listed in Table 4.2.

Table 4.2: Selected hyperparameters and associated search spaces of the open-set EVM classifier applied for flexibility activation classification.
Source: Table adapted from **Paper G**.

Hyperparameter	Search space	Selected value
Tailsize	[1,100]	7
Distance multiplier	[0.1,1.1]	0.9
Distance metric	Canberra, Cosine, Euclidean	Canberra
Threshold ρ	[0.1, 0.99999]	0.9

4.2.6 Demonstration

This section demonstrates and evaluates the two main components of the proposed flexibility activation identification pipeline. In Section 4.2.6.1, applied performance metrics are introduced. Thereafter, unsupervised detection and open-set classification of flexibility activation events is evaluated in Section 4.2.6.2 and 4.2.6.3, respectively.

4.2.6.1 Performance metrics

Detection metrics For evaluating the binary flexibility activation detection problem, the class-specific F_1 score according to (3.12) is considered, as the importance is on accurately indicating the rarely occurring flexibility activations instead of normal operation. Detecting one or multiple observations of a single flexibility activation event is considered as one true positive. Entirely missing an event counts as single false negative. In contrast, the calculation of false positives and true negatives is conducted point-wise.

In order to evaluate the timeliness of detection, the average detection delay is considered, which is calculated according to

$$\bar{\delta}_{\text{det}} = \frac{1}{N_{\text{det}}} \sum_{i=1}^{N_{\text{det}}} (t_{\text{det},i} - t_{\text{FA,start},i}), \tag{4.12}$$

where N_{det} is the number of detected flexibility activation events, and $t_{\text{FA,start},i}$ and $t_{\text{det},i}$ the start and first-detection time of the i-th detected event, respectively. The execution time of the detectors is in the range of milliseconds and therefore neglected.

Classification metrics Event classification is evaluated on the macro-averaged F_1 score (3.13), giving equal importance to the identification of the FA and NO class. Importantly, the calculation is only based on the F_1 scores of the known classes, as simply treating all unknown events as one additional class would bias the performance results [65]. The impact of unknown classes on the performance reflects in potentially higher false positive and negative rates of the known classes.

Section 4.2.6.3 evaluates the influence of varying numbers of unknown classes, which typically is expressed as the openness of a test set, following

$$Op = 1 - \sqrt{\frac{2N_{\text{classes,train}}}{N_{\text{classes,test}} + N_{\text{classes,target}}}}, \tag{4.13}$$

where $Op \in [0, 1]$, and $N_{\text{classes,target}} = N_{\text{classes,train}}$ in the considered case. Higher Op values indicate more unknown classes, while for the closed-set problem $Op = 0$.

4.2.6.2 Unsupervised detection of flexibility activation events

Figure 4.10 depicts the F_1 score over varying thresholds, the maximum F_1 score, and the average detection delay at an optimal threshold[1] γ_{opt} for the proposed persistence forecast-based detection and all considered benchmark detectors. From Figure 4.10 (b) it can be seen that the highest $F_{1,\text{max}}$ score is achieved by the persistence-, ARIMA-, and CNN-based detectors, which indicates that using the proposed simple persistence detector can achieve the same flexibility activation detection performance as application of sophisticated statistical or ML forecasting models. Moreover, Figure 4.10 (a) indicates that these three detectors behave almost identical over varying thresholds. This is explained by two factors: On the one hand, changes between consecutive load observations are rather small on the considered 5-minute resolution, which makes a persistence forecast a reasonable approach. On the other hand, existing changes largely result from random fluctuations with few learnable patterns. For these reasons, the ARIMA and CNN model approximate a persistence forecast, resulting in similar detection behavior and performance. The distinct behavior of the other two detectors is possibly explained by differences in the modeling procedure, referring to additional Fourier transformation of the load time series in the case of SR, and the model-internal anomaly score calculation of the HTM. In terms of early detection, the persistence detector exhibits a marginal advantage, as Figure 4.10 (c) shows. For

[1] γ_{opt} is determined based on the flexibility activation detection (*FAD*) score as detailed in **Paper G**.

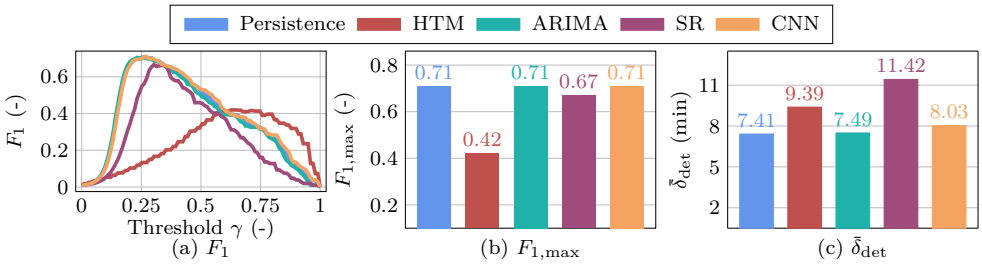


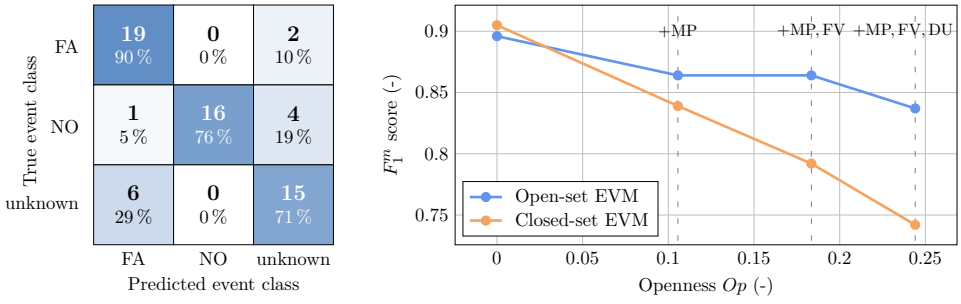
Figure 4.10: F_1 score (a), maximum F_1 score $F_{1,\text{max}}$ (b), and average detection delay $\bar{\delta}_{\text{det}}$ at γ_{opt} (c) of the proposed persistence-based detector and all benchmarks.
Source: Illustration based on **Paper G**.

γ_{opt}, 191 of the 205 flexibility activations (93 %) are detected by the persistence detector, while 498 out of 39827 normal operation observations (1.25 %) are false positives. The relatively high detection rate mainly is a consequence of the fast-ramp behavior of the flexibility activations within the investigated dataset. The associated sudden change of slope directly translates to a high anomaly score, as (4.8) indicates. From these findings, it can be concluded that simple persistence forecast-based anomaly detection can be an effective approach for detecting fast-ramped flexibility activations in an automated manner.

4.2.6.3 Open-set classification of flexibility activation events

Figure 4.11 (a) depicts the confusion matrix for applying the open-set EVM on the test dataset considering all event classes. The MP, FV and DU class are summarized as *unknown*. The EVM correctly classifies 90 % of all FA and 76 % of the NO class samples. 71 % of the observations from unknown classes are identified as such. Within the known classes, the confusion is small. In contrast, the presence of unknown classes entails several false positives and negatives for the known classes. The corresponding macro F_1 score is $F_1^m = 0.837$, which indicates feasibility of classifying flexibility activations in the more realistic open-set scenario.

Figure 4.11 (b) compares the open- and closed-set EVM classifier under increasing numbers of unknown event classes. For $Op = 0$, the closed-set classifier performs better as it avoids flagging observations of known event classes as *unknown*. As soon as an unknown class is added, the open-set classifier exhibits a better performance, as it identifies observations from unknown classes as such, whereas the closed-set classifier wrongly assigns all of them to one of the known classes. The F_1^m score of the open-set classifier decreases as well since the identification of unknown observations is not free of mistakes. However, it falls slower so that for a rising openness the performance



(a) Confusion matrix for $Op = 24.7\,\%$

(b) Comparison of open- and closed set classification

Figure 4.11: Confusion matrix of the open-set EVM under consideration of all unknown load-altering event classes ($Op = 24.7\,\%$) (a), and (b) comparison of the open- and closed-set EVM for a varying openness Op.
Source: Illustration based on **Paper G**.

advantage compared to closed-set classification increases. These results demonstrate that applying open-set classifiers can be beneficial for flexibility activation classification in more realistic distribution grid scenarios which involve occurrence of unknown load-altering event types.

4.2.7 Remaining barriers and limitations

The results in Section 4.2.6 demonstrate the fundamental feasibility of flexibility activation detection and classification with lightweight models and under consideration of the more realistic scenario of multiple and partly unknown load-altering event types. Nevertheless, the proposed flexibility activation identification pipeline and the conducted study are subject to limitations. The detection is limited to fast-ramped flexibility activations. For detecting flexibility events of different behavior, an anomaly detection approach based on univariate time-series forecasts, in particular applying a trivial persistence model, is not appropriate. A potential strategy is the incorporation of covariates as done in the studies described in Section 3.1 and 3.2.3. Moreover, a detection delay of $\sim$7.5 minutes should be improvable. As the delay is partly explained by considering 5-minutely load averages, the concept should be investigated on data of higher resolution. The classification is based on full sequences of flexibility activations and thus can be considered an ex-post analysis. A potential strategy is the use of early classification algorithms [66]. Moreover, the classification performance is evaluated on a small number of event types and observations. To substantiate the findings, studies on larger and more versatile datasets should be conducted. Finally, the proposed identification pipeline has to be demonstrated as a whole, since the event sampling procedure proposed in **Paper G** is not yet tested.

4.3 Summary and reflection

This chapter empirically evaluates the impact of local flexibility on distribution grid monitoring, and proposes data and modeling strategies on the basis of ML-based predictive analytics to address operational awareness requirements of distribution grids related to flexibility-induced stochasticity and DSO-unaware flexibility activations. The overarching objective is to facilitate effective situational awareness of DSOs under high shares of flexible resources.

Based on content from **Paper F**, the impact of high shares of flexible resources on the reliability and accuracy of data-driven LV state estimation is evaluated. For that purpose, a BNN capable of quantifying aleatoric and epistemic estimation uncertainty is evaluated under several flexibility usage and DSO information availability scenarios. The BNN estimates node voltages at the end of LV feeders based on real-time primary substation power readings as well as weather and price features. Existing input-output relations are learned offline from a dataset which is based on historical SM data. The results indicate that local flexibility below adjacent secondary substations

has little impact. In contrast, high shares of flexible resources beneath the same secondary substation as where voltages are estimated introduce prediction uncertainty. As aleatoric uncertainty is the predominant type, it can be concluded that integration of flexible resources introduces stochasticity which cannot be counteracted by more advanced models and larger data histories. Instead, the estimation problem must be modified through additional information sources. The results demonstrate that the incorporation of real-time power and voltage readings from secondary substations enables the BNN to provide LV state estimates similarly reliable and informative as in scenarios of marginal flexibility usage. Thus, using advanced probabilistic regression models, capable of quantifying aleatoric uncertainty, in combination with secondary substation readings enables leveraging the advantage of data-driven state estimation of being applicable to underdetermined distribution grids also under high shares of flexible resources.

On the basis of content from **Paper G**, a data-driven concept for flexibility activation identification in active power readings is presented, constituting a potential use case for accurate LV state estimations. The concept is intended for online identification of DSO-unaware flexibility activations, for example from simultaneous reaction of smart EV chargers to low-price signals, and online verification of DSO-requested flexibility services. The proposed data pipeline separates the identification problem into unsupervised detection and open-set classification to account for several practical requirements. These include real-time detection, computational efficiency, limited need for historical event observations, and handling of multiple and partly unknown load-altering event types. The unsupervised detection of flexibility activations is treated as point-wise anomaly detection problem, where a persistence forecast provides the normal behavior reference. Activations are flagged given a sufficiently large deviation between the reference and actual observations. The application of trivial persistence forecasting is argued by the typically small and random changes in load data of high resolution, such as the considered 5-minute averages. The subsequent classification step is based on an EVM open-set classifier which distinguishes flexibility activation events from normal operation, while identifying observations of other unknown load-altering event types as such. The detection step is evaluated based on an active power time series which comprises 450 households and includes 205 real flexibility activations. The results show that at an optimal detection threshold 93 % of the activations are detected with an average delay of 7.41 minutes, while 1.25 % of the normal operation observations are falsely declared as being part of a flexibility activation. Moreover, it is demonstrated that applying more advanced statistical and ML models (e.g., ARIMA and CNN) results in a similar performance as they approximate persistence forecasts due to small and random changes between consecutive load observations. For evaluating the classification step, a test set comprising observations of flexibility activations and normal operation as well as samples of three unknown load-altering event classes is considered. The EVM correctly classifies 90 % of all flexibility activation events, while 71 % of the observations from unknown classes are identified as such, resulting in a macro F_1 score of $F_1^m = 0.837$. Moreover, it is demonstrated that applying open-set classifiers allows to reduce the performance degradation under

increasing numbers of unknown load-altering event classes in comparison to typically applied closed-set classification models. While the general feasibility of detecting and classifying flexibility activations, taking the formulated practical requirements into account, is demonstrated, remaining limitations must be addressed. Those include restriction to detection of fast-ramped flexibility activations, and delayed classification due to processing of complete event time series sequences.

The data and modeling strategies proposed in this chapter constitute efficient approaches to facilitate situational awareness of DSOs in scenarios of significant usage of local flexibility, as they exploit existing online and offline data sources, and minimize need for additional metering devices. As such they have the potential to create trust and willingness for integrating higher shares and varying mechanisms of local flexibility, which ultimately can support more efficient and sustainable grid operation.

CHAPTER 5
Conclusion and research outlook

This thesis investigates the use of ML-based predictive analytics for meeting awareness requirements for DER and distribution grid operation that arise from the integration of local flexibility and the related digitalization. The work is placed within the interdisciplinary area between electrical engineering, applied data science and cyber security. The primary objective is to gain insights into what influences applicability and performance of data-driven monitoring and event detection techniques in the context of local flexibility management, and to develop data and modeling strategies for leveraging their potential under real-world conditions. Thus, a predominantly empirical and practice-oriented approach is applied, which involves (i) evaluation of realistic data and scenarios, (ii) leveraging publicly accessible models and techniques, and (iii) accounting for practical aspects such as limited data and computational resource availability. In this context, three research topics are defined and addressed by seven independent scientific publications. The associated research questions are concluded in the following.

RQ1 *What are the operational challenges of local flexibility and the associated digitalization that set new requirements on the situational awareness for DERs and distribution grids?*

Flexibility realization can be compromised by different factors including weather- and consumer-induced uncertainty, technical limitations and unintentional or malicious manipulation. The impact of sub-optimal flexibility realization is mainly of economic nature. While flexibility asset owners face lower electricity cost savings or revenues from service provision, DSOs may require investments in grid extension, for example, to handle more pronounced load peaks. Tackling sub-optimal flexibility realization requires, among others, forecasts for optimized flexibility scheduling which are accurate and robust under possible load and generation variability, computational limitations, and data manipulation.

Local flexibility relies on ICT for planning, realizing and verifying flexible operation of DERs, which drives their connection to public networks and remote control functionality. In this environment, cyber criminals can exploit poor authentication mechanisms and other security weaknesses to steal and manipulate data or launch malicious control commands. As part of a coordinated cyber-physical attack targeting a fleet of DERs, flexible assets can be misused to launch load-altering attacks against distribution grids, which may trigger protection mechanisms and interrupt customer supply. Thus, a need for advanced attack identification concepts for DERs is seen.

Local flexibility can be based on several mechanisms with different control objectives, which potentially introduces load variability and stochasticity in distribution grids. As a consequence, distribution grid monitoring can be subject to larger uncertainties, which ultimately can entail poor or misinformed control actions. Thus, monitoring tools such as LV state estimation should be robust to high shares of flexible resources, which potentially involves uncertainty quantification and incorporation of further data sources such as price signals.

Some flexibility mechanisms may not actively involve DSOs. Consequently, events such as the simultaneous reaction of multiple smart EV chargers to a low-price signal may introduce pronounced load modifications without system operators being aware of it. In severe cases, triggered protection mechanisms may entail disconnection of consumers from supply. Thus, an automated online identification system for flexibility activations is considered useful for supporting distribution grid monitoring and operation in a scenario of high shares of flexible resources.

RQ2 *How do weather- and consumer-induced stochasticity, computational constraints and cyber attacks impair flexibility realization, and how can data and modeling strategies based on predictive analytics enable or facilitate computationally efficient, cost-optimal and cyber-secure usage of DERs as flexible resources?*

Unpredictable weather and consumer behavior, computational limitations and malicious data manipulation can impair price-based flexibility realization by lowering the accuracy of load and generation forecasts needed for optimizing the operation schedule of a DER. One consequence can be lower financial rewards for owners of flexible assets. On the example of two real prosumers equipped with PV-battery systems, it is shown that unpredictable consumer and weather behavior reduces cost savings by about 5-10 percentage points compared to the theoretical case of assuming perfect forecasts, which cannot be avoided even under use of sophisticated forecasting models. The impact of computational limitations is found to be marginal as applying lightweight default GBDT models with almost no data history practically achieves the same financial rewards. Primary factors are regular model retraining and use of weather forecasts as model input. In comparison, using often considered persistence forecasts reduces economic benefits by further 7-12 percentage points. These findings indicate that ML-based forecasting enables near-optimal price-based flexibility realization also under practical challenges of limited computational resources and short data histories. The performance of ML-based forecasting can be affected by manipulation of the model inputs. It is shown that adding noise to weather inputs results in up to 3 percentage points lower cost savings. More severe degradation is avoided as the GBDT models learn via retraining to put less weight on affected inputs, thus approximating a model that does not use weather forecasts as input. Other attacks against flexible assets can cause more significant impacts on prosumers and grid operation. It is demonstrated that mirroring price data on their moving average turns cost-optimal battery scheduling into an energy cost driver for prosumers. In case that a fleet of flexible assets reacts to the same manipulated prices, grid operation can be af-

fected by transforming the assets' peak-shaving into peak-reinforcing behavior. Thus, DERs should be equipped with advanced concepts for real-time identification of such cyber-physical attacks to facilitate incident response.

One strategy for improving real-time monitoring of DERs is the joint evaluation of physical process and cyber network data applying data-driven models. The systematic evaluation of such cyber-physical event identification on the case of classifying several attack and fault types demonstrates a macro-averaged F_1 score improvement of 15.5 percentage points compared to exclusively processing cyber network data. It is further shown that cyber-physical event identification allows to detect and classify cyber attacks, cyber-physical attacks and physical faults in one integrated approach. Both the improved classification performance and the progression towards holistic event identification can facilitate timely and adequate incident response, supporting a cyber-secure usage of DERs as flexible assets. To leverage cyber-physical monitoring in non-academic environments, several practical requirements must be met, including independence of scarce attack and fault samples, verifiability of event predictions, and information provision for both known and unknown event types. For that purpose, the cyber-physical event reasoning system CyPhERS is proposed. CyPhERS first generates informative and human-interpretable event signatures based on cyber-physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. Event information is then derived from the signatures through manual interpretation or automated matching against a database of predefined signatures. The evaluation of CyPhERS on several cyber and cyber-physical attack types targeting a real PV-battery system demonstrates feasibility of providing human-verifiable real-time event information such as occurrence, type, affected devices, attacker location, and physical impact both for known and unknown attack and other event types without need for historical event observations.

RQ3 *How does local flexibility affect monitoring of distribution grids, and how can data and modeling strategies applying predictive analytics facilitate effective situational awareness for DSOs under high shares of flexible resources?*

Observability in distribution grids with low meter coverage can be supported by data-driven LV state estimation, as it is applicable to underdetermined systems by leveraging offline measurements. Local flexibility can affect the performance of data-driven LV state estimation. On the example of price-based smart EV charging, it is shown that high shares of flexible resources below the same secondary substation as where states are estimated can lower the estimation performance in terms of the Pinball loss by around 28 percentage points. Key factor is the increase of aleatoric uncertainty which cannot be counteracted by more advanced models and larger data histories. In contrast, adding real-time power and voltage readings from secondary substations to the inputs of a sophisticated BNN-based state estimator reduces the performance degradation to about 6 percentage points. Moreover, compared to LQR-based estimation, the BNN on average achieves a 16 % lower Pinball loss. It can be concluded that informative and reliable LV state estimates can be provided also

under high shares of flexible resources by using advanced probabilistic regression models, capable of quantifying aleatoric uncertainty, in combination with secondary substation readings. By avoiding widespread installation of metering devices below secondary substation level, this approach facilitates situational awareness of DSOs in a cost-effective manner.

The situational awareness of grid operators can be further supported by online identification of flexibility activations in load measurements or estimates of specific nodes in the grid, allowing operators to i) detect load modifications from flexibility mechanisms which do not actively involve DSOs, and ii) verify activation of DSO-requested services. The proposed data-driven concept separates the identification problem into unsupervised detection and subsequent open-set classification to account for several practical requirements including real-time detection, computational efficiency, limited need for historical event observations, and handling of multiple and partly unknown load-altering events. A simple persistence forecast is applied to provide expected values in the detection step, while an EVM is used for open-set classification. In the evaluated case comprising of 205 real flexibility activation events from a LFM demonstration, 93 % of the activations are detected with an average delay of 7.41 minutes, while 1.25 % of the normal operation observations are falsely declared as being part of a flexibility activation. It is further shown that detection based on advanced statistical and ML models (e.g., ARIMA and CNN) cannot improve the performance compared to using simple persistence forecasts due to small and largely random changes between consecutive observations in the considered load data of 5-minute resolution. The classification of flexibility activation events is evaluated on a dataset comprising flexibility activation and normal operation observations as well as samples of three unknown load-altering event classes. The applied EVM open-set classifier correctly identifies 90 % of all flexibility activation events, while 71 % of the observations from unknown classes are classified as such, resulting in a macro-averaged F_1 score of $F_1^m = 0.837$. In contrast, using a traditional closed-set classifier lowers the performance to $F_1^m = 0.742$ as observations of unknown load-altering events are falsely assigned to one of the known classes. These results indicate the general feasibility of flexibility activation detection and classification as operational awareness support for DSOs, under fulfillment of the formulated practical requirements.

Common conclusion and reflection The results of this thesis underline the importance of accompanying the adoption of local flexibility with concepts for improved operational awareness both on DER and distribution grid level in order to facilitate effective, reliable and secure use of flexibility potentials and the related economic, operational and environmental benefits. In this regard, it is shown how ML-based predictive analytics can be leveraged to provide solutions to operational challenges for DERs and distribution grids in the context of local flexibility and the associated digitalization, while respecting practical requirements such as computational efficiency, prediction verifiability and use of publicly available tools. Nevertheless, careful preliminary assessment of the suitability of ML tools for a specific problem

and benchmarking with simpler concepts is crucial to avoid unnecessary run-time and computational overhead as the case of flexibility activation detection exemplifies.

In the course of this Ph.D. project, two further needs for facilitating the development of operational awareness concepts based on predictive analytics became apparent. The first aspect is the demand for a universal and customizable data-processing and prediction pipeline, applicable to regression, forecasting and detection problems, since most of the concepts evaluated and developed in this thesis and comparable studies apply similar models, data processing steps and implementation procedures. Thus, a pipeline for regression, forecasting and anomaly detection problems was developed along this Ph.D. project aiming to mitigate duplicating efforts and frequently seen flaws such as data leakage. The pipeline is wrapped around the *Darts* and *Optuna* libraries and covers all typical steps from feature extraction to model evaluation. With few clicks, users can choose between regression, forecasting and detection, different models ranging from persistence forecast to most recent deep learning algorithms, sequential or parallel processing, and other options. A beta version of the pipeline is publicly available[1] and will soon be transferred into a Python library.

The second aspect is the need for more publicly available datasets. Developing data-driven concepts depends on datasets of sufficient size and quality. Those are often either hard to find, confidential, or overused, which complicates concept development and comparison, and lowers generality of results. For these reasons, further comprehensive attack experiments were conducted on the PV-battery system described in Section 3.2.3.3 in the course of this Ph.D. project. The resulting cyber-physical dataset will soon be made publicly available.

Limitations Focusing on the evaluation of data from real systems as, for example, in **Paper B** (real prosumer generation and demand), **Paper E** (real DER process readings and network traffic), and **Paper G** (real flexibility activations) avoids simulation model assumptions and constraints, and allows to take practical issues such as measuring errors or network packet re-transmissions into account. Together this facilitates investigation of ML-based monitoring and event detection techniques under real-world conditions. However, as the availability of such data is often limited and their generation time-consuming, most of the findings provided by this thesis are based on specific study cases. While this allows to demonstrate basic trends and concept feasibility, results may vary for certain other cases. One example is the limitation to Danish prosumers in **Paper B**. Although the findings are similar despite substantially different consumption patterns, results may differ for prosumers located in other countries, for example, due to other electricity tariffs and meteorological conditions. Another example is the exclusive demonstration of CyPhERS on a PV-battery system in **Paper E**. The considered case is insightful due to the relatively high complexity resulting from involving energy generation, storage, and consumption, weather- and consumer influences, and multiple digital and physical components. Nevertheless, other DERs may introduce further monitoring challenges,

[1]Link: https://gitlab.com/Nils_Mueller/flexml

for example, by coupling several energy sectors as in the case of electrolyzers or combined heat and power units. These examples highlight that it would be beneficial to substantiate the findings of this thesis by evaluating further systems and scenarios.

Future research directions The results of this thesis set the foundation for further research directions and studies. While several possible paths are provided throughout the previous chapters, two main directions are highlighted in the following.

A promising direction is the extension of the proposed cyber-physical event reasoning system CyPhERS. One opportunity is the integration of additional data sources such as host logs or human interactions with the system (e.g., maintenance activity schedules), which would allow to extend CyPhERS to detection of initial steps of cyber kill chains and other events such as human errors. Moreover, it should be investigated whether ML models can be leveraged for automated creation of a database of predefined event signatures by providing models with information on the architecture of a DER and typical attack and fault vectors. Finally, the development of a bottom-up security architecture for distribution grids based on the aggregation and joint evaluation of event reports from a network of CyPhERS systems is considered a promising study path. While coordinated attacks against a fleet of DERs pose a risk for grid operation, DSOs cannot monitor individual resources as they typically neither have access to their data nor the capacity to process such large data quantities. Instead, a collective evaluation of attack reports from distributed CyPhERS systems by a CSIRT has the potential to provide DSOs with information including location and aggregated load capacity of affected DERs, allowing system operators to assess possible risks and counteract with measures such as grid reconfiguration.

Another research direction is seen in the extension of data-driven LV state estimation to forecasting. State forecasting can be a valuable tool for distribution grid planning and operation, for example, in the context of congestion management mechanisms. However, by systematically avoiding congestions and other problematic grid conditions, for example by means of local flexibility services from LFMs, historical data lack observations of those. Consequently, ML models cannot be explicitly trained on predicting critical grid states. Thus, it should be investigated if data-driven models are able to predict future problematic grid conditions by extrapolating from the learned relations between non-critical grid states and associated flexibility services. In this context, it would be of interest to examine whether learning the relation between grid states and different flexibility services further allows to use LV state forecasting for comparing and selecting among several flexibility service offers.

Bibliography

[1] The Copenhagen Post. *New record as wind and solar power account for close to 60 percent of Denmark's annual electricity consumption*. 2022. URL: https://cphpost.dk/2022-12-30/news/new-record-as-wind-and-solar-power-account-for-close-to-60-percent-of-denmarks-annual-energy-consumption (visited on May 23, 2023).

[2] C. Corinaldesi et al. "European Case Studies for Impact of Market-driven Flexibility Management in Distribution Systems." In: *Proceedings of the 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 2019, pages 1–6. DOI: 10.1109/SmartGridComm.2019.8909689.

[3] M. Z. Degefa, I. B. Sperstad, and H. Sæle. "Comprehensive classifications and characterizations of power system flexibility resources." In: *Electric Power Systems Research* 194 (2021). DOI: 10.1016/j.epsr.2021.107022.

[4] Eurelectric. *Flexibility and Aggregation: Requirements for their interaction in the market*. 2014. URL: https://www.usef.energy/app/uploads/2016/12/EURELECTRIC-Flexibility-and-Aggregation-jan-2014.pdf (visited on May 23, 2023).

[5] P. Olivella-Rosell et al. "Optimization problem for meeting distribution system operator requests in local flexibility markets with distributed energy resources." In: *Applied Energy* 210 (2018), pages 881–895. DOI: 10.1016/j.apenergy.2017.08.136.

[6] C. S. Bojer and J. P. Meldgaard. "Kaggle forecasting competitions: An overlooked learning opportunity." In: *International Journal of Forecasting* 37.2 (2021), pages 587–603. DOI: 10.1016/j.ijforecast.2020.07.007.

[7] F. Ahmad et al. "Distribution system state estimation-A step towards smart grid." In: *Renewable and Sustainable Energy Reviews* 81 (2018), pages 2659–2671. DOI: 10.1016/j.rser.2017.06.071.

[8] M. Pertl et al. "Validation of a robust neural real-time voltage estimator for active distribution grids on field data." In: *Electric Power Systems Research* 154 (2018), pages 182–192. DOI: 10.1016/j.epsr.2017.08.016.

[9] M. Antonakakis et al. "Understanding the Mirai Botnet." In: *Proceedings of the 26th USENIX Security Symposium (USENIX Security 17)*. 2017, pages 1093–1110.

[10] Plusminus. *Leichtes Spiel für Hacker (Easy game for hackers)*. 2023. URL: https://www.tagesschau.de/investigativ/swr/plusminus-solar-windkraft-anlagen-sicherheit-hacker-101.html (visited on May 26, 2023).

[11] S. Lakshminarayana, S. Adhikari, and C. Maple. "Analysis of IoT-Based Load Altering Attacks Against Power Grids Using the Theory of Second-Order Dynamical Systems." In: *IEEE Transactions on Smart Grid* 12.5 (2021), pages 4415–4425. DOI: 10.1109/TSG.2021.3070313.

[12] Y. Li and J. Yan. "Cybersecurity of Smart Inverters in the Smart Grid: A Survey." In: *IEEE Transactions on Power Electronics* 38.2 (2023), pages 2364–2383. DOI: 10.1109/TPEL.2022.3206239.

[13] M. Beaudin and H. Zareipour. "Home energy management systems: A review of modelling and complexity." In: *Renewable and Sustainable Energy Reviews* 45 (2015), pages 318–335. DOI: 10.1016/j.rser.2015.01.046.

[14] J. Salpakari and P. Lund. "Optimal and rule-based control strategies for energy flexibility in buildings with PV." In: *Applied Energy* 161 (2016), pages 425–436. DOI: 10.1016/j.apenergy.2015.10.036.

[15] Y. Zhang et al. "Model predictive control-based operation management for a residential microgrid with considering forecast uncertainties and demand response strategies." In: *IET Generation, Transmission & Distribution* 10.10 (2016), pages 2367–2378. DOI: 10.1049/iet-gtd.2015.1127.

[16] M. Elkazaz et al. "Performance Assessment of an Energy Management System for a Home Microgrid with PV Generation." In: *Energies* 13.13 (2020). DOI: 10.3390/en13133436.

[17] GNU project. *GLPK (GNU Linear Programming Kit)*. 2023. URL: https://www.gnu.org/software/glpk/ (visited on May 28, 2023).

[18] S. Diamond and S. Boyd. "CVXPY: A Python-Embedded Modeling Language for Convex Optimization." In: *Journal of Machine Learning Research* 17.1 (2016), pages 2909–2913.

[19] J. H. Friedman. "Greedy function approximation: A gradient boosting machine." In: *The Annals of Statistics* 29.5 (2001), pages 1189–1232. DOI: 10.1214/aos/1013203451.

[20] J. Bergstra et al. "Algorithms for Hyper-Parameter Optimization." In: *In Proceedings of Advances in Neural Information Processing Systems*. Volume 24. 2011.

[21] R. Hyndman and G. Athanasopoulos. *Forecasting: Principles and Practice*. 3rd. OTexts, 2021.

[22] J. Herzen et al. "Darts: User-Friendly Modern Machine Learning for Time Series." In: *Journal of Machine Learning Research* 23.124 (2022), pages 1–6.

[23] T. Akiba et al. "Optuna: A Next-Generation Hyperparameter Optimization Framework." In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pages 2623–2631. DOI: 10.1145/3292500.3330701.

[24] D. Azuatalam et al. "Energy management of small-scale PV-battery systems: A systematic review considering practical implementation, computational requirements, quality of input data and battery degradation." In: *Renewable and Sustainable Energy Reviews* 112 (2019), pages 555–570. DOI: 10.1016/j.rser.2019.06.007.

[25] DTU Computing Center. *DTU Computing Center resources*. 2022. DOI: 10.48714/DTU.HPC.0001.

[26] A. Sahu et al. "Multi-Source Multi-Domain Data Fusion for Cyberattack Detection in Power Systems." In: *IEEE Access* 9 (2021), pages 119118–119138. DOI: 10.1109/ACCESS.2021.3106873.

[27] G. Loukas et al. "Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning." In: *IEEE Access* 6 (2018), pages 3491–3508. DOI: 10.1109/ACCESS.2017.2782159.

[28] T. P. Vuong, G. Loukas, and D. Gan. "Performance Evaluation of Cyber-Physical Intrusion Detection on a Robotic Vehicle." In: *Proceedings of the 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*. 2015, pages 2106–2113. DOI: 10.1109/CIT/IUCC/DASC/PICOM.2015.313.

[29] L. Faramondi et al. "A Hardware-in-the-Loop Water Distribution Testbed Dataset for Cyber-Physical Security Testing." In: *IEEE Access* 9 (2021), pages 122385–122396. DOI: 10.1109/ACCESS.2021.3109465.

[30] T. Minka. "Automatic Choice of Dimensionality for PCA." In: *Proceedings of Advances in Neural Information Processing Systems*. Volume 13. 2000.

[31] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python." In: *Journal of Machine Learning Research* 12 (2011), pages 2825–2830.

[32] F. Chollet. *Keras*. 2015. URL: https://keras.io (visited on June 22, 2023).

[33] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., 2009.

[34] C. Zhang et al. "A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data." In: *Proceedings of the 33th AAAI Conference on Artificial Intelligence*. 2019, pages 1409–1416. DOI: 10.1609/aaai.v33i01.33011409.

[35] K. Hundman et al. "Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding." In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pages 387–395. DOI: 10.1145/3219819.3219845.

[36] N. D. Tuyen et al. "A Comprehensive Review of Cybersecurity in Inverter-Based Smart Power System Amid the Boom of Renewable Energy." In: *IEEE Access* 10 (2022), pages 35846–35875. DOI: 10.1109/ACCESS.2022.3163551.

[37] M. K. Hasan et al. "Review on cyber-physical and cyber-security system in smart grid: Standards, protocols, constraints, and recommendations." In: *Journal of Network and Computer Applications* 209 (2023). DOI: 10.1016/j.jnca.2022.103540.

[38] F. Li et al. "A Review of Cyber-Attack Methods in Cyber-Physical Power System." In: *Proceedings of the 2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP)*. 2019, pages 1335–1339. DOI: 10.1109/APAP47170.2019.9225126.

[39] Q. Wang et al. "A Comprehensive Survey of Loss Functions in Machine Learning." In: *Annals of Data Science* 9 (2022), pages 187–212. DOI: 10.1007/s40745-020-00253-5.

[40] E. Manitsas et al. "Distribution System State Estimation Using an Artificial Neural Network Approach for Pseudo Measurement Modeling." In: *IEEE Transactions on Power Systems* 27.4 (2012), pages 1888–1896. DOI: 10.1109/TPWRS.2012.2187804.

[41] R. Bessa et al. "Probabilistic Low-Voltage State Estimation Using Analog-Search Techniques." In: *Proceedings of the 2018 Power Systems Computation Conference (PSCC)*. 2018, pages 1–7. DOI: 10.23919/PSCC.2018.8443074.

[42] Y. Huang et al. "Probabilistic State Estimation Approach for AC/MTDC Distribution System Using Deep Belief Network With Non-Gaussian Uncertainties." In: *IEEE Sensors Journal* 19.20 (2019), pages 9422–9430. DOI: 10.1109/JSEN.2019.2926089.

[43] M. Pertl et al. "Voltage estimation in active distribution grids using neural networks." In: *Proceedings of the 2016 IEEE Power and Energy Society General Meeting (PESGM)*. 2016, pages 1–5. DOI: 10.1109/PESGM.2016.7741758.

[44] L. V. Jospin et al. "Hands-On Bayesian Neural Networks—A Tutorial for Deep Learning Users." In: *IEEE Computational Intelligence Magazine* 17.2 (2022), pages 29–48. DOI: 10.1109/MCI.2022.3155327.

[45] A. G. Wilson and P. Izmailov. "Bayesian Deep Learning and a Probabilistic Perspective of Generalization." In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. 2020, pages 4697–4708.

[46] A. Kendall and Y. Gal. "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pages 5580–5590.

[47] T. Hong and S. Fan. "Probabilistic electric load forecasting: A tutorial review." In: *International Journal of Forecasting* 32.3 (2016), pages 914–938. DOI: 10.1016/j.ijforecast.2015.11.011.

[48] C. Heinrich et al. "EcoGrid 2.0: A large-scale field trial of a local flexibility market." In: *Applied Energy* 261 (2020). DOI: 10.1016/j.apenergy.2019.114399.

[49] UK Department for Transport. *Electric Chargepoint Analysis 2017: Domestics*. 2017. URL: https://www.gov.uk/government/statistics/electric-chargepoint-analysis-2017-domestics (visited on June 22, 2023).

[50] J. V. Dillon et al. "Tensorflow distributions." In: *arXiv preprint arXiv:1711.10604* (2017).

[51] J.-S. Chou and A. S. Telaga. "Real-time detection of anomalous power consumption." In: *Renewable and Sustainable Energy Reviews* 33 (2014), pages 400–411. DOI: 10.1016/j.rser.2014.01.088.

[52] A. Barua et al. "Hierarchical Temporal Memory-Based One-Pass Learning for Real-Time Anomaly Detection and Simultaneous Data Prediction in Smart Grids." In: *IEEE Transactions on Dependable and Secure Computing* 19.3 (2022), pages 1770–1782. DOI: 10.1109/TDSC.2020.3037054.

[53] J. Pereira and M. Silveira. "Unsupervised Anomaly Detection in Energy Time Series Data Using Variational Recurrent Autoencoders with Attention." In: *Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2018, pages 1275–1282. DOI: 10.1109/ICMLA.2018.00207.

[54] R. K. Beniwal et al. "A Critical Analysis of Methodologies for Detection and Classification of Power Quality Events in Smart Grid." In: *IEEE Access* 9 (2021), pages 83507–83534. DOI: 10.1109/ACCESS.2021.3087016.

[55] A. E. Lazzaretti et al. "A new approach for event classification and novelty detection in power distribution networks." In: *Proceedings of the 2013 IEEE Power & Energy Society General Meeting*. 2013, pages 1–5. DOI: 10.1109/PESMG.2013.6672703.

[56] N. Huang et al. "Mechanical Fault Diagnosis of High Voltage Circuit Breakers with Unknown Fault Type Using Hybrid Classifier Based on LMD and Time Segmentation Energy Entropy." In: *Entropy* 18.9 (2016). DOI: 10.3390/e18090322.

[57] C. Feng et al. "Smart grid encounters edge computing: opportunities and applications." In: *Advances in Applied Energy* 1 (2021). DOI: https://doi.org/10.1016/j.adapen.2020.100006.

[58] J. C. S. de Souza, T. M. L. Assis, and B. C. Pal. "Data Compression in Smart Distribution Systems via Singular Value Decomposition." In: *IEEE Transactions on Smart Grid* 8.1 (2017), pages 275–284. DOI: 10.1109/TSG.2015.2456979.

[59] E. M. Rudd et al. "The Extreme Value Machine." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.3 (2018), pages 762–768. DOI: 10.1109/TPAMI.2017.2707495.

[60] S. W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing, 1997.

[61] J. Wu, W. Zeng, and F. Yan. "Hierarchical Temporal Memory method for time-series-based anomaly detection." In: *Neurocomputing* 273 (2018), pages 535–546. DOI: 10.1016/j.neucom.2017.08.026.

[62] R. H. Shumway and D. S. Stoffer. "Time Series Regression and ARIMA Models." In: *Time Series Analysis and Its Applications*. Springer New York Inc., 2000, pages 89–212. DOI: 10.1007/978-1-4757-3261-0_2.

[63] S. Albawi, T. A. Mohammed, and S. Al-Zawi. "Understanding of a convolutional neural network." In: *Proceedings of the 2017 International Conference on Engineering and Technology (ICET)*. 2017, pages 1–6. DOI: 10.1109/ICEngTechnol.2017.8308186.

[64] H. Ren et al. "Time-Series Anomaly Detection Service at Microsoft." In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pages 3009–3017. DOI: 10.1145/3292500.3330680.

[65] C. Geng, S.-J. Huang, and S. Chen. "Recent Advances in Open Set Recognition: A Survey." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.10 (2021), pages 3614–3631. DOI: 10.1109/TPAMI.2020.2981604.

[66] A. Gupta et al. "Approaches and Applications of Early Classification of Time Series: A Review." In: *IEEE Transactions on Artificial Intelligence* 1.1 (2020), pages 47–61. DOI: 10.1109/TAI.2020.3027279.

Part II

Collection of relevant publications

PAPER A

Nils Müller, Kai Heussen, Zeeshan Afzal, Mathias Ekstedt, Per Eliasson

Threat Scenarios and Monitoring Requirements for Cyber-Physical Systems of Flexibility Markets

Müller, N., K. Heussen, Z. Afzal, M. Ekstedt, and P. Eliasson, "Threat Scenarios and Monitoring Requirements for Cyber-Physical Systems of Flexibility Markets," in *Proceedings of the 2022 IEEE PES Generation, Transmission and Distribution Conference and Exposition–Latin America (IEEE PES GTD Latin America)*, La Paz, 2022, pp. 1-6, doi: 10.1109/IEEEPESGTDLatinAmeri53482.2022.10038290.

Threat Scenarios and Monitoring Requirements for Cyber-Physical Systems of Flexibility Markets

Nils Müller, Kai Heussen
Wind and Energy Systems Department
Technical University of Denmark
Lyngby, Denmark
{nilmu; kheu}@dtu.dk

Zeeshan Afzal, Mathias Ekstedt
Department of Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden
{zafzal; mekstedt}@kth.se

Per Eliasson
Foreseeti AB
Stockholm, Sweden
per.eliasson@foreseeti.com

***Abstract*—The ongoing integration of renewable generation and distributed energy resources introduces new challenges to distribution network operation. Due to the increasing volatility and uncertainty, distribution system operators (DSOs) are seeking concepts to enable more active management and control. Flexibility markets (FMs) offer a platform for economically efficient trading of electricity flexibility between DSOs and other participants. The integration of cyber, physical and market domains of multiple participants makes FMs a system of cyber-physical systems (CPSs). While cross-domain integration sets the foundation for efficient deployment of flexibility, it introduces new physical and cyber vulnerabilities to participants. This work systematically formulates threat scenarios for the CPSs of FMs, revealing several remaining security challenges across all domains. Based on the threat scenarios, unresolved monitoring requirements for secure participation of DSOs in FMs are identified, providing the basis for future works that address these gaps with new technical concepts.**

***Index Terms*—distribution grids, flexibility markets, threat scenarios, monitoring requirements, cyber-physical power systems**

I. Introduction

To reach the European goal of carbon neutrality in 2050, electricity generation and consumption must undergo radical changes. While the share of renewable generation needs to increase, electrification through devices such as electric vehicles (EVs) and heat pumpss (HPs) will drive up and reshape electricity demand. This extensive installation of distributed energy resources (DERs) will introduce more uncertainty and volatility, which radically changes usage of distribution networks (DNs), potentially requiring expensive grid reinforcements. A widely discussed alternative is the use of end user flexibility, referred to as demand response [1]. By reducing equipment loading at peak hours, distribution system operators (DSOs) can use local flexibility to delay or avoid investments for reinforcement of transformers and power lines.

As a framework for the integration of local flexibility, a widely promoted approach are flexibility markets (FMs) [2]. FMs constitute a competitive trading platform for electricity flexibility in a geographically restricted area such as towns [3]. A typical setup of market participants consists of a DSO, a balance responsible party (BRP), several aggregators and a market operator. Aggregators pool and manage multiple small residential flexibility assets. In this way, they enable end users to participate in FMs. DSOs and BRPs typically are flexibility buyers, while aggregators constitute sellers. DSOs procure flexibility for operational purposes, such as congestion management or voltage control. BRPs buy flexibility for portfolio optimization. By adjusting power demand of aggregated flexibility assets, aggregators make profits according to flexibility contracts. Owners of flexibility assets earn profits by providing DERs, such as HPs or EVs, to aggregators.

The foundation of a FM is a strong integration of cyber, physical and market domains of multiple actors, making it a system of cyber-physical systems (CPSs). While this cross-domain integration sets the foundation for efficient deployment of flexibility assets, it introduces new vulnerabilities to involved participants and their systems. By applying end user flexibility to avoid critical grid states, DSO grid operation becomes partly dependent on third parties. Moreover, the required use of information and communication technology (ICT), including less secure public networks, and the strong coupling with the physical and market domain opens doors for cyber criminals, aiming at social or financial damage. In addition, incorporating home devices of end users as flexibility assets also requires transmission, storage and processing of sensitive data.

This paper contributes to the identification and analysis of possible risks and security requirements in the CPSs of FMs. The work first provides an overview of possible threat scenarios, which result from a comprehensive and original system analysis. Thereafter, unresolved monitoring requirements for secure participation of DSOs in FMs are derived from the threat scenarios. Objective of this work is to provide a foundation and motivation for future works addressing the identified gaps with new technical concepts and case studies.

A. Related work

As highlighted by [4] and [5], the influence of flexibility on power system security constitutes a research gap, as most existing works focus solely on benefits of flexibility usage. Some works shed light on specific physical threats, such as uncertain customer behavior [6] or financial threats, e.g. financial risk due to the intermittent nature of flexibility assets [7]. Other

This work is funded by the Innovation Fund Denmark under File No. 91363 and the Swedish Energy Agency as part of the ERA-Net project HONOR.

works such as [8] and [9] investigate cyber threats introduced by the application of new smart grid technologies, including smart meters (SMs) and advanced metering infrastructure. In [10], a number of cyber threats are identified and mapped to grid assets and threat agents. The work also addresses possible security controls to reduce exposure to threats. However, these works only focus on particular threats or threat categories and do not specifically address FMs.

In [5] possible positive and negative impacts of flexibility on the security of supply are discussed from a physical and a cyber perspective. A major physical threat is seen in the rebound effect of flexibility activations which may shift load peaks, and results in even more severe situations. A flexibility-induced cyber threat is seen in load-altering attacks that may impact the bulk power system without compromising better protected assets on transmission level. To the best of the authors knowledge, [5] is the only work that provides cyber and physical threat scenarios in the context of flexibility. However, as threat scenarios are no major concern of the work, it does not provide a comprehensive and systematic overview. Moreover, it neither takes characteristics of FMs into account nor derives unresolved security requirements to motivate new research directions for future studies.

B. Contribution and paper structure

The contribution of this paper is twofold:

- Systematic formulation of threat scenarios for the CPSs of FMs. Scenarios result from an original system analysis and consider origins in various domains, emphasizing the interaction among the cyber, physical and market domain.
- Identification of unresolved monitoring requirements for DSOs participating in FMs as foundation for new technical concepts and case studies addressing these gaps.

The remainder of this paper is structured as follows. Section II provides a systematic overview of threat scenarios for the CPSs of FMs. Section III identifies monitoring requirements for participation of DSOs in FMs. Finally, Section IV concludes the paper.

II. Threat scenarios

This section is concerned with the systematic formulation of threat scenarios for the CPSs of FMs. Subsection II-A presents the scenario formulation approach, followed by scenario descriptions in Subsections II-B to II-J.

A. Threat scenario formulation

To describe and compare scenarios with various backgrounds, a domain-neutral formulation is required, which still captures key information. Fig. 1 represents the applied formulation concept. Threat origin, affected component and threat impact are selected as domain-independent key information. The threat origin comprises two groups, namely external and internal. In Table I the considered origins are listed and allocated to one of the two groups, supplemented by information on their background. Table I indicates the broad spectrum of origins, enabling a holistic threat scenario investigation.

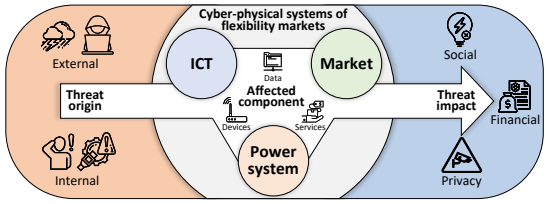


Fig. 1. Schematic representation of the threat scenario formulation.

Typically, a critical situation develops around a specific component or component type in a system. A cyber attacker will most likely try to manipulate a specific data stream or device to launch the attack. A famous example is the Industroyer malware attack on the Ukrainian power system, which targeted the control of circuit breakers in substations [11]. An aggregator who falsely determines the potential of its portfolio will affect a flexibility service traded in the market.

At the end of every scenario there is a potential negative impact, typically of social or financial nature. However, other impact, such as a loss of private information, are taken into account. To allow for a better overview and to demonstrate how fundamentally different threat origins can result in similar critical situations, scenarios are grouped by the affected component. Table II summarizes the scenarios, including threat origins, given as numbers referring to Table I, and impacts.

B. SM-based scenarios

SMs may provide data to aggregators for flexibility planning and verification. Meters typically use a programmable logic controller (PLC) interface to communicate with the utility and have capability to remotely switch power on or off.

1) Unauthorized access and modification of SM data: Cyber-criminals could gain access to sensitive meter data such as consumption, credentials, and firmware information by exploiting known software vulnerabilities or by decrypting PLC communication that uses weak encryption. Additionally, if cyber-criminals take over the meter communication with the aggregator (e.g., by using the encryption key), they can send wrong consumption data. The asset owner may be fined for breaking contracts and removed from the portfolio. Having hold of private data and options for financial damage, cyber-criminals may aim at blackmailing flexibility asset owners.

TABLE I
Overview and description of threat origins.

Nr.	Threat origin	Group	Background
1	Device failure	Internal	accidentally
2	Human error	Internal	accidentally
3	Market actors	Internal	financial gain
4	Insiders	Internal	dissatisfaction
5	Consumer behavior	Internal	randomness
6	Weather	External	volatility, randomness
7	Price signals	External	volatility, randomness
8	Organized cyber-criminals	External	financial gain
9	State-sponsored actors	External	political

2) Accessing and controlling multiple SMs: As extension of Scenario II-B1, state-sponsored actors may aim at accessing and controlling multiple SMs. Attackers may use weaknesses of SMs, such as static encryption keys. Typically, SMs deployed by a DSO share the same encryption key. Thus, if attackers gain access to the encryption key of one SM, they are able to extract useful information, such as the energy consumption behavior, of entire neighborhoods. At this stage, attackers can also take over remote on/off switching. In [12], it is demonstrated how attackers can cause line trips through load oscillations by exploiting the switching capability of multiple SMs. The result may be power outages, resulting in social and financial cost. Another attack path to target multiple SMs is to launch a remote or physical attack on meter data concentrators.

C. Local controller-based scenarios

Various actors of FMs rely on local controllers. By interfacing the cyber and physical domain, they constitute critical system components, introducing potential threats.

1) Modification of substation controller: Primary and some secondary substations are equipped with controllers, such as PLCs, remote terminal units (RTUs) and intelligent electronic devices (IEDs). State-sponsored actors could place infected rootkits onto one or multiple local controllers. By sending malicious control signals to circuit breakers and protection relays attackers could damage grid facility and disconnect customers. To hide the attack, normal operation values could be returned to the central control room. In [13], it is demonstrated that attackers can create such false data that will not raise an alarm by existing algorithms for bad data detection.

2) Modification of flexibility asset controller: Flexibility assets such as EVs or HPs are often controlled by home energy management systems (HEMSs). These internet-connected systems typically have remote control capability and are based on off-the-shelf soft- and hardware, making them vulnerable to cyber attacks and asset owner modification. Flexibility asset owners may aim at financial gain by modifying setpoint boundaries or increasing setpoints before a service is activated, which manipulates the flexibility service of the aggregator. Organized cyber-criminals could make use of weak password security and encryption to gain access to individual HEMSs. To blackmail customers, attackers may collect sensitive data, change setpoints to impair customer comfort and degrade flexibility assets, or increase costs by raising consumption or mitigating contracted flexibility activations. State-sponsored actors could infiltrate local controllers of multiple small or individual large flexibility assets. By changing the setpoints or switching assets on or off, they could introduce load peaks or oscillations to trigger transformer protection, resulting in customer disconnection. A coordinated attack on flexibility assets and grid protection mechanisms may result in severe physical damage of grid facilities and blackouts.

3) Failure of large flexibility asset controller: The activation of large flexibility assets may fails due to soft- or hardware failures. Compared to defects of small assets, the impact may be severe. An industrial plant could provide flexibility by reducing production capacity during times of high EVs charging. Under these conditions, an activation failure could lead to congestion at the transformer.

TABLE II
SUMMARY OF THE IDENTIFIED THREAT SCENARIOS.

Nr.	Threat scenario	Impact	Origin
	SM-based scenarios		
1	Unauthorized access & modification of SM data	privacy	3,8
2	Accessing and controlling multiple SMs	social, privacy	8,9
	Local controller-based scenarios		
3	Modification of substation controller	social, financial	9
4	Modification of flexibility asset controller	social, privacy	3,8,9
5	Failure of large flexibility asset controller	social, financial	1
	Flexibility activation signal-based scenarios		
6	Tamper or disrupt flexibility activation signals	social, financial	4,8,9
7	Unintentional wrong activation of flexible assets	social, financial	2
8	Parallel flexibility activations with opposing or reinforcing effects	financial	3,7
	Historical data-based scenarios		
9	Compromised data on DSO or aggregator data historian	financial	1,2,4,9
	Flexibility request-based scenarios		
10	High uncertainty in the determination of flexibility needs	financial	5,6,7
11	Uncertainty about power system states due to frequent flexibility activations	social, financial	3
12	Parallel events resulting in sudden change of flexibility needs	social, financial	1,2,7,9
	Flexibility offer-based scenarios		
13	Place wrong flexibility offers on the FM	social, financial	4,9
14	High uncertainty in the determination of flexibility offers	financial	5,6
	Flexibility measurement or schedule-based scenarios		
15	Disrupt or manipulate flexibility measurements and schedules	financial, privacy	3,8
	Flexibility asset-based scenarios		
16	Unavailability of flexibility assets	financial	1,5,8,9
	Vendor soft- and hardware-based scenarios		
17	Compromise vendor software and systems	social, financial	9

D. Flexibility activation signal-based scenarios

Flexibility activation signals comprise activation requests from flexibility buyers to sellers, and activation signals from aggregators to small flexibility assets. The transmission is typically conducted via public networks.

1) Tamper or disrupt flexibility activation signals: Aggregator employees may launch insider attacks, such as sending activation signals at wrong times or preventing required flexibility activations. Insiders of DSOs may send wrong activation requests. Flexibility activation signals could also be manipulated by cyber-criminals or state-sponsored actors through false data injection attacks, exploiting insecure authentication or weak encryption. Attackers could also flood flexibility assets with activation signals to disrupt the activation process. In all these scenarios, attackers could prevent or temper required flexibility activations to leave congestions or voltage violations unresolved or even intensify them. Moreover, attackers could initiate critical grid states by activating flexibility assets. In both cases high social and financial costs are likely.

2) Unintentional wrong activation of flexibility assets: Human errors of various actors, such as DSOs or aggregators, and

in different process steps, from determining flexibility needs or potential to preparing and sending activation requests, could initiate wrong flexibility activations. Equivalent to intentional attacks, damage could be of social and financial nature.

3) Parallel flexibility activations with opposing or reinforcing effects: Flexibility services can be requested by different actors with distinct purposes. While a DSO may intend to prevent congestion, a BRP aims at portfolio optimization. Thus, flexibility services with opposing effects could be activated simultaneously, resulting in financial damage as services may be procured without achieving the desired outcome. At the same time, price-based demand response introduces additional flexibility activations in DNs. DSOs might be unaware of future behavior of price-driven loads during flexibility planning. Thus, a risk for network violations exists if the DSO service is reinforced by price-driven flexibility.

E. Historical data-based scenarios

Historical data is of high importance for several actors in FMs. Threats emerge from potential data loss or manipulation.

1) Compromised data on DSO or aggregator data historian: Historical data provide necessary information for flexibility planning, activation and verification. Typically, they are not checked for integrity, after being stored. However, integrity could be affected by human and transfer errors or attacks. Model development based on compromised data will weaken performance or might render models useless. Financial damage may result due to imprecise flexibility planning and verification. In severe cases, power system monitoring techniques may fail, leaving critical grid conditions unresolved.

F. Flexibility request-based scenarios

To procure flexibility, DSOs and BRPs submit flexibility requests to the FM. Depending on the market concept, requests can be formulated from intraday to months ahead.

1) High uncertainty in the determination of flexibility needs: DNs face increasing volatility due to the dependency of distributed energy resources on weather, consumer behavior and price signals. At the same time, low-voltage (LV) grid states are highly underdetermined due to low real-time meter device coverage (low observability). The resulting uncertainty complicates forecasting of flexibility needs and requires DSOs to request larger flexibility capacities, which increases costs.

2) Uncertainty about power system states due to frequent flexibility activations: DSOs request and activate flexibility to avoid or postpone expensive grid extensions. However, frequent activations may break correlation between the few available measurements (e.g. primary substation and weather data) and system states at the end of LV feeders [14]. Thus, FM operation might deteriorate the accuracy of LV state estimation, making critical states potentially unobservable to DSOs. Based on inaccurate state estimations a DSO might activate unnecessary or even counteracting flexibility, resulting in financial costs. In severe cases, the triggering of protection mechanisms might cause disconnection of customers.

3) Parallel events resulting in sudden change of flexibility needs: Different events, including line failures or shut down of large industrial loads, can lead to sudden change of the DNs condition. Additionally, load peaks from simultaneous EV charging and other new events will be introduced to DNs in the upcoming years. If they occur during flexibility activation periods, such events may change grid condition in a way that activation is not required or even critical. Moreover, state-sponsored actors could launch attacks on other systems, e.g. large battery energy storage systems or industrial plants, during activation periods to modify the grid condition. Due to the low observability of DNs, the detection of such events may be challenging.

G. Flexibility offer-based scenarios

To sell flexibility, aggregators submit flexibility offers to FMs. Depending on the market scheme, flexibility can be offered from intraday to months ahead.

1) Place wrong flexibility offers on the FM: If offers on the market do not reflect the actual potential, flexibility activations will likely not match the problem to solve. State-sponsored actors or insiders could tamper offers or place wrong offers on the market in the name of verified market participants. In less serious cases aggregators will have to pay a refund. In severe cases critical grid conditions might not be solved by wrong flexibility offers.

2) High uncertainty in the determination of flexibility offers: Determination of flexibility potential is subject to uncertainties. The capacity of an aggregator portfolio is dependent on the comfort requirements of customers, weather, customer behavior and other portfolio changes. In particular, weather and customer behavior uncertainties directly translate into uncertainty of flexibility offers. Moreover, in most cases the demand of small flexibility assets is controlled indirectly, e.g by adjusting temperature setpoints. As the translation of temperature setpoints to power consumption is dependent on external factors, additional uncertainties are introduced during activation. Unreliable offers mainly reduce financial profit for aggregators. However, severe uncertainties might make the use of flexibility for DSOs unreliable, and lead to more expensive but reliable alternatives, such as grid extensions. In case a DSO relies on a flexibility offer to solve a critical condition, high uncertainty might result in disconnection of end users.

H. Flexibility measurement or schedule-based scenarios

Reliable measurements of flexibility assets are required for service planning, activation and verification. Besides SM readings, additional data may come from devices such as photovoltaic meters. To define the activation process, aggregators and flexibility asset owners agree on flexibility schedules.

1) Disrupt or manipulate flexibility measurements and schedules: Several actors might have an interest in manipulating flexibility measurements and schedules either by gaming or data tampering. Aggregators or flexibility asset owners could manipulate flexibility activation recordings for financial gain. Exemplary, for baseline services an asset owner could increase

consumption before an activation period, to imitate a service by just returning to normal consumption level. Cyber-criminals that can sniff and modify data in networks of aggregators could compromise measurements, e.g. for blackmailing. One way is the modification of flexibility portfolio recordings to disrupt the service verification process. As a result, aggregators might receive fines for not fulfilling contractual agreements. Attackers could also modify the schedules which aggregators send to the assets, resulting in wrong activations. In mild cases, aggregators will be fined. In severe cases, wrong activations might trigger grid protection, resulting in disconnection of customers and thus high social costs.

I. Flexibility asset-based scenarios

Flexibility assets comprise a variety of DERs, owned by end users or companies. They reach from small loads such as refrigerators to large loads, including industrial processes.

1) Unavailability of flexibility assets: During activation periods assets may not be available due to software failures, manual setpoint altering by asset owners or unforeseeable changes in the physical process of industrial flexibility assets. Moreover, cyber-criminals or state-sponsored actors could disturb communication by denial-of-service attacks. Since asset owners break the contract in cases of a failed activation, such scenarios would result in a financial penalty. Especially in case of large flexibility assets, unavailability might lead to unresolved congestions and voltage violations.

J. Vendor soft- and hardware-based scenarios

All actors of FMs are dependent on services of third-parties, such as vendors. The required trust introduces potential risks.

1) Compromise vendor software and systems: State-sponsored actors could install malicious code in vendor soft- or hardware. Attackers may install a backdoor in a PLC. This backdoor can later be used to manipulate DSO operation in many ways. The impact of such events may go beyond single end users, as EV or HP vendors provide soft- or hardware to multiple asset owners. The recent SolarWinds hack demonstrates the severity of such attacks [15].

III. Unresolved Monitoring Requirements for Secure DSO Participation in FMs

This section identifies unresolved monitoring requirements for DSOs participating in FMs For that purpose, threat scenarios from Section II are mapped onto a generic cyber-physical monitoring architecture of DSOs, shown in Fig. 2.

1) Quantifying flexibility-induced uncertainty: Threat scenario II-F2 discusses that frequent flexibility activations could introduce uncertainty to LV state estimation. At the same time, flexibility is used to operate power systems closer to capacity limits. Under these conditions, deterministic point estimations may fail silently, potentially impacting critical decisions. On the contrary, probabilistic approaches provide information about reliability of estimates. Incorporating such uncertainty quantification into the decision making process allows situational adjustment of control actions and thus to lower the risk for wrong actions while retaining efficiency. Thus, quantifying flexibility-induced uncertainty is seen as important requirement for power system monitoring (Fig. 2).

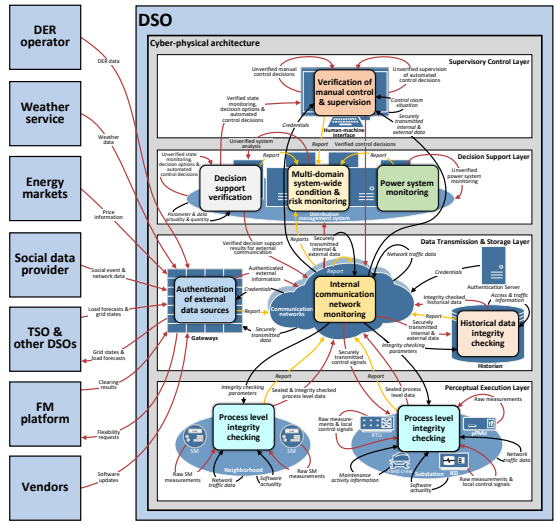


Fig. 2. Cyber-physical monitoring architecture of a DSO. Colored fields: generic monitoring requirements, which are considered a prerequisite to observe the DSO's CPS. Red links: information for power system operation. Black links: information for fulfilling monitoring requirements. Yellow links: reports of monitoring solutions satisfying the requirements.

Uncertainty quantification also improves verification of data-driven decision support tools (Fig. 2). Models that increase uncertainty under appearance of unseen flexibility activation events or system states provide operators with requirement indicators for retraining or additional input features [14].

Finally, uncertainty quantification facilitates analyzing the system-wide CPS condition (Fig. 2). If critical power system states are reported to a system-wide multi-domain condition monitor, probability of occurrence could be included. The improved interpretability could reduce false alarms, enabling more reliable system-wide monitoring of a DSO's CPS.

2) Flexibility activation detection: Several threat scenarios (II-C2, II-D2, II-D3) demonstrate that flexibility can be activated without the DSO being aware of it. Such activations might occur intentionally through other market participants and cyber attackers or unintentionally due to human errors. To enable immediate counteractions in case of critical activations, early detection is required. Moreover, early detection would allow online verification of successful activation of DSO-requested flexibility by the operator [16]. Thus, for power system monitoring (Fig. 2) automated real-time detection of flexibility activations is seen as an important requirement.

3) Flexibility scenario monitoring: In threat scenario II-F1 and II-F2, respectively, the difficulty of determining flexibility needs and flexibility activation demand is described. Especially under the aforementioned uncertainty and low observability

of DNs, flexibility planning becomes a challenging task for DSOs. Thus, tools providing probabilistic power system state scenarios under various flexibility services are considered an important requirement for power system monitoring (Fig. 2). Depending on the market concept, tool requirements may look different. For day or week-ahead procurement, tools will be required to provide state forecasts under various flexibility services. Market concepts that include real-time procurement of flexibility require tools for mapping available flexibility offers onto the current grid state.

4) Integration of multi-domain information: Many threat scenarios demonstrate a strong FM-induced interaction and dependency among cyber, physical and market domains. Cyber attackers may intend to cause physical damage (II-C2, II-D1, II-E1) or disturb market actions (II-B1, II-G1, II-H1), while insufficient coordination (II-D3) or wrong flexibility offers (II-G1) on the FM platform may result in physical impact. This interdependency has two consequences: on the one hand, underlying events are likely to leave traces in multiple domains. On the other hand, the root cause of a specific event can lay in different domains. As an example, a denial-of-service attack against activation of a large flexibility asset leaves traces in physical measurements and cyber network data. Moreover, the activation failure could also be caused by a hardware failure or human error. Thus, a monitoring requirement is seen in the integration of information from multiple domains to i) incorporate all available traces and ii) take possible threat origins in various domains into account. Among others, this somewhat general requirement could facilitate process-level or historical data integrity checking (Fig. 2). One example is the integrated detection and classification of cyber attacks and physical faults by fusion of cyber network and physical process data [17]. A central challenge for integration of multi-domain information is seen in the fusion of heterogeneous data.

5) Interpretable unsupervised intrusion and anomaly detection for flexibility assets: In threat scenario II-B2 and II-C2, respectively, it is demonstrated that edge devices, such as SMs and HEMSs, have security weaknesses (e.g., static encryption keys) which can be exploited by cyber attackers. FMs will make power system operation partly dependent on such less protected devices. Thus, from the perspective of the DSO, advanced intrusion and anomaly detection systems for flexibility assets are considered as an important requirement for process level integrity checking (Fig. 2). Challenges include computational constraints, lack of data describing the various attacks and anomalies, and the multitude of anomalies (e.g. cyber attacks, soft- and hardware faults and human errors) complicating root cause analysis. A potential approach is seen in machine learning-based unsupervised anomaly detection on information stream level. Unsupervised models do not require observations of anomalies. Moreover, detecting anomalies on information stream level (e.g. destination IP addresses, customer setpoints and power demand) instead of system-wide, retains interpretability for root cause analysis also in an unsupervised scheme.

IV. Conclusion

In this work, threat scenarios for the CPSs of FMs are systematically formulated and presented. 17 scenarios across all system domains are introduced, revealing several remaining security challenges. Among others, scenarios include simultaneous control of multiple flexibility assets by cyber attackers exploiting weak encryption, and uncertainty in the determination of flexibility needs and offers due to low meter coverage and high load variability in DNs. Based on the threat scenarios, unresolved monitoring requirements for secure participation of DSOs in FMs are identified. Requirements include interpretable unsupervised anomaly detection for flexibility assets on information stream level and quantification of flexibility-induced uncertainty. By identifying such unresolved monitoring requirements, a foundation for new technical concepts and case studies addressing these gaps is provided.

References

[1] K. Spiliotis, A. I. Ramos Gutierrez, and R. Belmans, "Demand flexibility versus physical network expansions in distribution grids," *Applied Energy*, vol. 182, pp. 613–624, 2016.

[2] X. Jin and Q. Wu, "Local flexibility markets: Literature review on concepts, models and clearing methods," *Applied Energy*, vol. 261, 2020.

[3] P. Olivella-Rosell et al., "Optimization problem for meeting distribution system operator requests in local flexibility markets with distributed energy resources," *Applied energy*, vol. 210, pp. 881–895, 2018.

[4] M. Alizadeh et al., "Flexibility in future power systems with high renewable penetration: A review," *Renewable and Sustainable Energy Reviews*, vol. 57, pp. 1186–1193, 2016.

[5] I. B. Sperstad, M. Z. Degefa, and G. Kjølle, "The impact of flexible resources in distribution systems on the security of electricity supply: A literature review," *Electric Power Systems Research*, vol. 188, 2020.

[6] B. Zeng, G. Wu, J. Wang, J. Zhang, and M. Zeng, "Impact of behavior-driven demand response on supply adequacy in smart distribution systems," *Applied Energy*, vol. 202, pp. 125–137, 2017.

[7] T. Ghose, H. W. Pandey, and K. R. Gadham, "Risk assessment of microgrid aggregators considering demand response and uncertain renewable energy sources," *Journal of Modern Power Systems and Clean Energy*, vol. 7, no. 6, pp. 1619–1631, 2019.

[8] A. Chehri, I. Fofana, and X. Yang, "Security risk modeling in smart grid critical infrastructures in the era of big data and artificial intelligence," *Sustainability*, vol. 13, no. 6, 2021.

[9] M. Costache and V. Tudor, "Security aspects in the advanced metering infrastructure," M.Sc. Thesis, Chalmers University of Technology, Department of Civil and Environment, Gothenburg, Sweden, 2011.

[10] L. Marinos, "Smart grid threat landscape and good practice guide," *White Paper, European Network and Information Security Agency (ENISA); ENISA: Attiki, Greece*, 2013.

[11] N. Kshetri and J. Voas, "Hacking power grids: A current problem," *Computer*, vol. 50, no. 12, pp. 91–95, 2017.

[12] F. Alanazi, J. Kim, and E. Cotilla-Sanchez, "Load oscillating attacks of smart grids: Demand strategies and vulnerability analysis," *arXiv preprint arXiv:2105.00350*, 2021.

[13] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, p. 21–32, 2009.

[14] N. Müller, S. Chevalier, C. Heinrich, K. Heussen, and C. Ziras, "Uncertainty quantification in LV state estimation under high shares of flexible resources," *Electric Power Systems Research*, vol. 212, 2022.

[15] S. Peisert et al., "Perspectives on the solarwinds incident," *IEEE Security Privacy*, vol. 19, no. 2, pp. 7–13, 2021.

[16] N. Müller, C. Heinrich, K. Heussen, and H. W. Bindner, "Unsupervised detection and open-set classification of fast-ramped flexibility activation events," *Applied Energy*, vol. 312, 2022.

[17] N. Müller, C. Ziras, and K. Heussen, "Assessment of cyber-physical intrusion detection and classification for industrial control systems," in *2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2022.

PAPER B

Nils Müller, Mattia Marinelli, Kai Heussen, Charalampos Ziras

On the trade-off between profitability, complexity and security of forecasting-based optimization in residential energy management systems

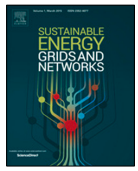
Müller, N., M. Marinelli, K. Heussen, and C. Ziras, "On the trade-off between profitability, complexity and security of forecasting-based optimization in residential energy management systems," in *Sustainable Energy, Grids and Networks*, vol. 34, 2023, doi: 10.1016/j.segan.2023.101033.

Sustainable Energy, Grids and Networks 34 (2023) 101033

Contents lists available at ScienceDirect

Sustainable Energy, Grids and Networks

journal homepage: www.elsevier.com/locate/segan



On the trade-off between profitability, complexity and security of forecasting-based optimization in residential energy management systems

Nils Müller, Mattia Marinelli, Kai Heussen, Charalampos Ziras *

Wind and Energy Systems Department, Technical University of Denmark, Building 330, Risø campus, 4000 Roskilde, Denmark

ARTICLE INFO

Article history:
Received 12 January 2023
Received in revised form 14 March 2023
Accepted 17 March 2023
Available online 21 March 2023

Keywords:
Energy management system
Prosumer
Flexibility
Forecasting
Machine learning
Security

ABSTRACT

With the emergence of affordable access to data sources, machine learning models and computational resources, sophisticated control concepts for residential energy management systems (EMSs) are on the rise. At the heart of those are production and consumption forecasts. Given the wide spectrum of implementation opportunities, selection of appropriate forecasting strategies is challenging. This work systematically evaluates forecasting-based optimization for residential EMSs in terms of trade-offs between economic profitability, computational complexity and security. The foundation of the study is two real prosumer cases equipped with a photovoltaic-battery system. Results demonstrate that, within the considered scenarios, best trade-offs are achieved based on forecasts of a default gradient-boosted decision trees model, using a short initial training set, weather forecast inputs and regular retraining. Over 90% of the theoretical maximum economic benefit is achieved in this scenario, at significantly lower computational complexity than others with similar savings, while being applicable to new systems without large data history. In terms of security, this scenario exhibits tolerance against weather input manipulation. However, sensitivity to price tampering may require data integrity checking in residential EMSs.

1. Introduction

Electricity grids are facing increasing shares of volatile renewable generation and variable consumption due to the electrification of the mobility and heating sectors [1,2]. The resulting larger temporal changes in supply and demand are entailing a need for electricity flexibility, with a great potential found within the residential sector [3]. At the same time, the increase and fluctuations in electricity prices motivate consumers to optimize their consumption. In that context, residential energy management systems (EMSs) constitute a promising solution, as controlling flexible energy resources allows to simultaneously provide (1) flexibility to the power system, and (2) financial benefits to consumers.

Given their steady cost decrease, photovoltaic (PV)-battery systems have become prominent examples of residential flexibility assets [4]. Existing EMSs typically apply simple myopic heuristics or rule-based controls for battery scheduling, without consideration of future electricity prices, production and consumption [5]. More advanced approaches combine optimization techniques with PV production and load forecasts. Given the affordable or even free access to weather and electricity prices data, machine learning (ML) models and computational resources, such more advanced concepts slowly find their way into application.

The use of forecasts is at the center of these optimization-based approaches. However, the wide range of implementation options and limitations makes selection of an appropriate forecasting strategy a compelling task. Overly complex models may provide minimal improvements at the cost of computational overhead. On the contrary, the lack of historical data for newly installed PV-battery systems may render the implementation of advanced models infeasible. Finally, strategies relying on data integration via the Internet (e.g., weather data or cloud-based forecasts) may open new opportunities for adversaries aiming at financial damage, for example, through data manipulation. These observations illustrate the need for a systematic and holistic assessment of different forecasting strategies for optimization in residential EMSs, considering trade-offs of profitability, complexity and security (see Fig. 1). In a nutshell, this can be expressed by the following research question: *"Under which conditions of data availability, computing resources and model complexity can forecasting-based battery scheduling in residential EMSs provide best trade-offs regarding economic profitability, computational complexity and security?"* To address this question, this work evaluates

* Corresponding author.
E-mail address: chazi@dtu.dk (C. Ziras).

https://doi.org/10.1016/j.segan.2023.101033

Nomenclature

Abbreviations

ANN	Artificial neural network
BMS	Battery management system
EMS	Energy management system
GBDT	Gradient-boosted decision trees
GHI	Global horizontal irradiance
HPC	High-performance computing
IoT	Internet of things
ML	Machine learning
PV	Photovoltaic
PVMS	Photovoltaic management system
RMSE	Root mean squared error
SM	Smart meter
SOC	State of charge
TPE	Tree Parzen Estimator

Parameters

α	Aggressiveness of the attack [–]
ΔT	Normalized duration of a time step [–]
η	Battery efficiency [–]
$\overline{p}_{\text{inv}}$	Inverter power capacity limit [kW]
$\overline{s}$	Battery upper SOC limit [kWh]
$\underline{s}$	Battery lower SOC limit [kWh]
d	Feature dimension [–]
k	Depth of decision trees [–]
L	Dataset length [–]
m	Number of decision trees [–]
N	Number [–]
v	Number of nodes in decision trees [–]
w	Time series window length [–]

Sets

$\boldsymbol{\Omega}$	Hyperparameter space
$\boldsymbol{\omega}$	Set of hyperparameters
$\mathcal{T}$	Set of steps in the optimization horizon
$\mathcal{W}$	Set of optimization variables

Indices

σ	Time step in moving average window
τ	5-min time step
C	Forecasting case
C'	EMS scenario
h	1-h time step
i	Dataset observation
j	1-h time steps ahead index
q	Time-series cross-validation fold

Variables

$\boldsymbol{X}$	Vector of covariate values [–]
δ	Battery charging/discharging status [–]
$\hat{\Lambda}$	Forecast of spot price [€/kWh]
$\hat{P}^{\text{L}}$	1-h avg. load forecast [kW]
$\hat{P}^{\text{PV}}$	1-h avg. PV production forecast [kW]
Λ	Spot price of a 1-h time step [€/kWh]
λ	Spot price of a 5-min time step [€/kWh]
$nRMSE$	Normalized RMSE [–]
$rRMSE$	Relative RMSE [–]
$\tilde{\Lambda}$	Manipulated spot price [€/kWh]
$\widehat{GHI}$	1-h average GHI forecast [W/m^2]
$\widehat{O}$	1-h avg. cloud opacity forecast [–]
A	Indicator of prosumer absence [–]
B	Economic benefit [€]
D	Day of the week [–]
F	Fees and taxes [€/kWh]
f	Fees and taxes of a 5-min time step [€/kWh]
H	Hour of the day [–]
K	Energy cost [€]
M	Memory need [–]
O	Approximated number of operations [–]
p	5-min avg. net demand ($p^{\text{L}} - p^{\text{PV}}$) [kW]
p^{b}	5-min avg. power from the grid [kW]
p^{c}	5-min avg. battery charging power [kW]
p^{d}	5-min avg. battery discharging power [kW]
P^{L}	1-h avg. load consumption [kW]
p^{L}	5-min avg. load consumption [kW]
P^{PV}	1-h average PV production [kW]
p^{PV}	5-min avg. PV production [kW]
p^{s}	5-min avg. power sold to the grid [kW]
R	Random number drawn from uniform distribution [–]
rB	Relative economic benefit [–]
rM	Relative memory need [–]
rO	Relative approximated number of operations [–]
s	SOC at end of a 5-min time step [kWh]

optimization-based control in residential EMS under several forecasting cases defined by a variety of model types, data availability scenarios and modeling strategies on two real prosumer cases.

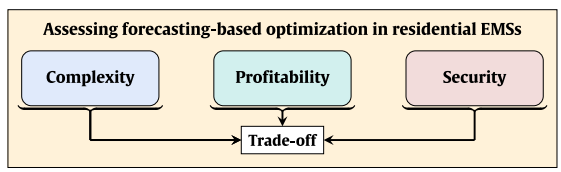


Fig. 1. Aspects for evaluating forecasting-based optimization in EMSs.

1.1. Related work

Most works on optimization in residential EMSs assess forecasting only by means of profitability. Typical approaches include the comparison of state-of-the-art rule-based control with forecasting-based optimization techniques [6,7], and the evaluation of different levels of forecast accuracy [8–12]. The authors of [7] compare a rule- and optimization-based strategy over the period of one year. Results demonstrate an up to 25% cost reduction by applying the latter. However, perfect weather and load forecasts (i.e., actual measured values) are assumed. The authors of [9] evaluate the impact of forecast uncertainty. Instead of evaluating real forecasts, random errors are artificially added to measurements to model forecasting uncertainty. Moreover, the evaluation is only based on simulation data for one week on a 30-min resolution. As shown in [11,13], time resolution, and thus the frequency at which the system is re-optimized (optimization frequency), has significant impact on economic performance assessment. In [8,10], the authors compare the impact

of perfect and realistic forecasts on economic performance. Both consider data of hourly resolution. Moreover, [8] is based on an artificially created dataset. Separate consumption and production data sources are combined, which breaks any existing correlation among those. The authors of [11] compare various forecasts for PV generation and load consumption. Several types of persistence models as well as perfect forecasts are considered for both PV and load forecasts. Additionally, artificial neural network (ANN)-based forecasts are used for load consumption, and an irradiation forecast-based PV model for PV forecasts. Compared to the previously described works, optimization frequency is two minutes. Moreover, several sensitivity analyses are conducted, including varying forecast errors, optimization frequencies and battery capacities. Results demonstrate that advanced forecast models allow for further price savings compared to persistence. However, prices are assumed to be always known for the next 24 h, which is not the case for spot prices. Moreover, PV generation and load consumption profiles of different buildings from different regions are combined, and PV production is artificially scaled. Finally, the impact of data availability (e.g., different amounts of training data) and modeling strategies (e.g., retraining) is not explored.

In contrast to the above-mentioned works, some extend economic evaluation with considerations of computational complexity. In [5,6], the authors compare multiple EMS strategies, covering both rule-based heuristics and forecasting-based optimization. The authors claim that the former achieve near-optimal solutions with lower computing resources compared to the latter. However, the optimization only runs on a 30-min frequency. Moreover, only persistence forecasts for PV generation and load consumption are considered as realistic forecasting approach. The authors of [14] propose a multi-objective predictive energy management strategy. The proposed prediction model is compared to several ML-based PV and load forecast models regarding profitability and computational complexity. However, only hourly data and one-step ahead predictions are considered.

In summary, the review of related literature demonstrates that most works only evaluate forecasting-based optimization in terms of profitability. Further, a large fraction exhibits methodologically shortcomings as they use short evaluation sets with low data resolution (30–60 min), assume to know prices for the entire optimization period or rely on artificially constructed prosumer datasets. To the best of the authors knowledge, no work systematically assesses several forecasting strategies for optimization in residential EMSs regarding economic profitability, computational complexity and security.

1.2. Contribution and paper structure

The main contributions of this work are as follows:

- Systematic and holistic evaluation of multiple scenarios of forecasting-based battery schedule optimization in residential EMSs.
- Consideration of various forecast cases defined by different model types, data availability and modeling strategies.
- Investigation of two real prosumer cases on an evaluation period of more than one year, considering an optimization frequency of 5 min and realistic price availability.
- Recommendations on optimal strategies for forecasting-based optimization in residential EMSs regarding trade-offs between economic profitability, computational complexity and security.

The remainder of this paper is structured as follows: In Section 2, the investigated prosumer scenarios are described. Section 3 introduces the applied methodology with regards to control strategies and underlying forecasting cases. In Section 4, details on the experimental setup and metrics are provided. Results are presented and evaluated in Section 5. Finally, a discussion of result implications is provided in Section 6, followed by a conclusion and view on future work in Section 7.

2. Prosumer concept and case description

In this study, two different residential prosumers are considered, which are each equipped with rooftop PV, a stationary storage system and an EMS (see Fig. 2). A common PV-battery inverter is assumed. The EMS comes with a dedicated smart meter smart meter (SM) that measures power at the grid connection point. PV and battery measurements are provided by the battery management system (battery management system (BMS)) and PV management system PV management system (PVMS), respectively, while load consumption is deducted from these measurements. The BMS further provides the current state of charge state of charge (SOC). While the measurements are sampled at high rates, they are usually available to users in extracted reports at, for example, 5-min resolution. The present work is based on such 5-min average values. Load and PV variations within the averaging period are not taken into account. Note that another meter is installed by the utility company for billing, but typically these meters provide only accumulated energy import and export values at 15- to 60-min rate. The prosumers are subject to instantaneous summation netting, that is imports and exports are summed up separately on the net result of all three phases [13].

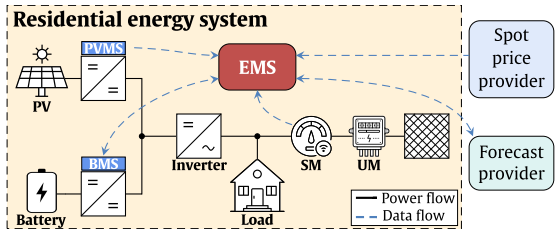


Fig. 2. Schematic representation of the residential energy system of both considered prosumers.

Apart from residential energy system-specific data, such as power measurements or the battery's SOC, the EMS has access to spot prices, which are published every day at 13:00 for the following day. Additionally, prosumers may acquire forecasts from third parties. One scenario is receiving weather forecasts, which can be used to generate PV and load forecasts locally. Another is direct procurement of the latter, for example, from providers of cloud-based forecasts. In this case, prosumers may need to provide historical and/or real-time measurements.

The consideration of two prosumers is justified by different production and consumption levels and patterns, allowing a broader evaluation of forecasting-based optimization. Production levels differ due to a higher nominal power of the PV plant of prosumer 1. Differences in load consumption mainly result from electric vehicle (EV) charging in case of prosumer 1, which entails a higher load level and less predictable patterns compared to the second one. Moreover, prosumer 1 purchased another EV in 2022, resulting in a change of load level and patterns during the recordings. Finally, prosumer 1 exhibits a higher self-consumption due to alignment of EV charging to PV production. Based on these characteristics, prosumer 2 can be considered a more traditional passive consumer, while prosumer 1 represents an already active future consumer. More details on the prosumers' setup follow in Section 4.1.

3. Methodology

This section introduces the underlying methodology of the different scenarios of forecasting-based optimization and a rule-based control benchmark, which together are referred to as *EMS scenarios* in the following. Section 3.1 describes the considered battery control strategies. In Section 3.2, different cases of PV and load forecasts are presented, which constitute the central foundation of the EMS scenarios. Finally, the realization of spot price forecasts is addressed in Section 3.3, followed by the introduction of two data manipulation scenarios in Section 3.4.

3.1. Battery control strategies

This subsection describes the two considered battery control strategies, namely an offline rule-based control benchmark (Section 3.1.1) and forecasting-based rolling-horizon optimization (Section 3.1.2).

3.1.1. Offline rule-based control benchmark

Rule-based controllers not depending on any prosumer-external information are common in real-life applications because they are robust, easy to implement and have minimal computational requirements. Thus, this offline approach will serve as a benchmark in the present study. The control mode that is used in this work, and many commercial PV-battery systems, minimizes the exchange of energy between the prosumer and the grid [5]. Let p^{L}_τ and p^{PV}_τ denote the 5-min average consumption and PV generation at time step τ, respectively. The difference of the two is the net demand $p_\tau = p^{\mathrm{L}}_\tau - p^{\mathrm{PV}}_\tau$. Each time step τ has a normalized duration of ΔT, where $\Delta T = 1/12$ for a 5-min step duration. When there is power surplus from the prosumer side, energy is stored in the battery. Once the battery is fully charged, excess energy is fed to the grid. If there is power deficit, energy from the battery supplies the load. If this is not possible because there is no sufficient energy stored, power is drawn from the grid. In all cases, battery inverter constraints are taken into account. A common inverter for the PV and the battery is assumed (see Fig. 2), so that the total power produced by the PV and flowing out of the battery cannot exceed the inverter's power capacity. The described control logic is summarized in the form of an algorithmic description in Algorithm 1, where η, p^{c}_τ and p^{d}_τ denote the battery's efficiency, charging and discharging power, respectively. s_τ is the SOC at the end of a 5-min period τ. The average power bought from/sold to the grid at τ is represented by p^{b}_τ and p^{s}_τ, while $\bar{s}$, $\underline{s}$ and $\overline{p}_{\mathrm{inv}}$ represent the upper/lower SOC limit and inverter power capacity, respectively.

Algorithm 1 Rule-based strategy for minimizing energy exchanges with the network.

$p_\tau \leftarrow p^{\mathrm{L}}_\tau - p^{\mathrm{PV}}_\tau$
if $p_\tau \geq 0$ **then**
  $p^{\mathrm{d}}_\tau \leftarrow \min(\overline{p}_{\mathrm{inv}} - p^{\mathrm{PV}}_\tau, p_\tau, (s_{\tau-1} - \underline{s})\eta/\Delta T)$
  $p^{\mathrm{b}}_\tau \leftarrow p_\tau - p^{\mathrm{d}}_\tau$
  $p^{\mathrm{s}}_\tau \leftarrow 0$
  $p^{\mathrm{c}}_\tau \leftarrow 0$
else
  $p^{\mathrm{c}}_\tau \leftarrow \min(\overline{p}_{\mathrm{inv}}, -p_\tau, (\bar{s} - s_{\tau-1})/(\eta\Delta T))$
  $p^{\mathrm{s}}_\tau \leftarrow -p_\tau - p^{\mathrm{c}}_\tau$
  $p^{\mathrm{b}}_\tau \leftarrow 0$
  $p^{\mathrm{d}}_\tau \leftarrow 0$

3.1.2. Forecasting-based rolling-horizon optimization

More advanced battery control schemes consider schedule optimization, which typically relies on prosumer-external information such as spot prices (see Fig. 2). In this work, an optimization problem according to

$$\min_{\mathcal{W}} \quad \sum_{\tau\in\mathcal{T}} \left[(\lambda_\tau|\hat{\lambda}_\tau + f_\tau)p^{\mathrm{b}}_\tau - \lambda_\tau|\hat{\lambda}_\tau p^{\mathrm{s}}_\tau\right]\Delta T, \tag{1a}$$

$$\text{s.t.} \quad 0 \leq p^{\mathrm{b}}_\tau, \quad 0 \leq p^{\mathrm{s}}_\tau \tag{1b}$$

$$0 \leq p^{\mathrm{c}}_\tau \leq \delta_\tau \overline{p}_{\mathrm{inv}} \tag{1c}$$

$$0 \leq p^{\mathrm{d}}_\tau \leq (1-\delta_\tau)\overline{p}_{\mathrm{inv}} \tag{1d}$$

$$\hat{p}^{\mathrm{PV}}_\tau + p^{\mathrm{d}}_\tau \leq \overline{p}_{\mathrm{inv}} \tag{1e}$$

$$p^{\mathrm{b}}_\tau - p^{\mathrm{s}}_\tau = \hat{p}^{\mathrm{L}}_\tau - \hat{p}^{\mathrm{PV}}_\tau + p^{\mathrm{c}}_\tau - p^{\mathrm{d}}_\tau \tag{1f}$$

$$s_{\tau+1} = s_\tau + \left[p^{\mathrm{c}}_{\tau+1}\,\eta + p^{\mathrm{d}}_{\tau+1}\,/\eta\right]\Delta T \tag{1g}$$

$$s_0 = s^{\mathrm{s}}, \quad s_{w_{\mathrm{opt}}} = s^{\mathrm{e}} \tag{1h}$$

$$\underline{s} \leq s_\tau \leq \bar{s} \tag{1i}$$

is considered. The set of optimization variables is denoted by $\mathcal{W}$. At every 5-min time step τ, the problem is solved over a look-ahead horizon of size w_{opt}, corresponding to a set of steps $\mathcal{T} = \{1, 2, \ldots, w_{\mathrm{opt}}\}$. Spot prices Λ and imposed fees and taxes F are hourly. Thus, constant values are used for each 5-min step τ within the respective hour, represented by λ_τ and f_τ. Unknown future values of p^{PV}_τ, p^{L}_τ and λ_τ within the look-ahead horizon are supplemented with forecasts. These are provided as hourly values and denoted by $\hat{P}^{\mathrm{PV}}_h$, $\hat{P}^{\mathrm{L}}_h$ and $\hat{\Lambda}_h$ for a 1-h time step h. For all 5-min time steps τ within the corresponding hour h, constant forecasts are considered, which are referred to as $\hat{p}^{\mathrm{PV}}_\tau$, $\hat{p}^{\mathrm{L}}_\tau$ and $\hat{\lambda}_\tau$. Whether true or forecasted prices are used in (1a) depends on the step τ within $\mathcal{T}$, which is indicated through a $\lambda_\tau|\hat{\lambda}_\tau$ notation. PV and load forecasts for a prediction horizon of size w_{pr} are performed every hour in a rolling fashion. Consequently, new forecasts are available every hour. As a default, this work assumes $w_{\mathrm{pr}} = 36$, which corresponds to an optimization horizon of $w_{\mathrm{opt}} = 432$. Initial studies suggest that $w_{\mathrm{pr}} = 36$ is sufficiently large to approximate the performance for $w_{\mathrm{pr}} \to \infty$. The impact of varying prediction horizons is evaluated in Section 5. Prices are published every day at 13:00 for the upcoming day. Price forecasts are performed at the same time and extend the published prices by another day. Further details on the PV, load and price forecasts follow in Section 3.2.

The battery charging/discharging status is represented by δ_τ and constitutes a binary decision variable. Starting and ending SOC values are denoted by s^{s} and s^{e}. The related constraint in (1h) requires the battery to be half charged at the end of the optimization period. All constraints (1b)–(1i) are imposed $\forall\ \tau \in \mathcal{T}$ except for (1g) and (1h), which hold $\forall\ \tau \in \mathcal{T}\backslash w_{\mathrm{opt}}$. After implementing the resulting optimal battery schedule at τ, a new optimization problem is solved at the following step based on the latest measurements and forecasts. *GLPK* [15] and *CVXPY* [16] are used as open-source optimization solver and modeling language, respectively.

3.2. Forecasting cases

This subsection describes the considered cases of PV and load forecasts required for battery schedule optimization (see Section 3.1.2). The set of forecasting cases is created by varying model type, data availability and modeling strategies. While Sections 3.2.1 and 3.2.2 introduce the considered scenarios of data availability and modeling strategies, respectively, applied forecast model types are introduced in Section 3.2.3–3.2.5. An overview of the 18 resulting cases (C1–C18) is provided in Table 1.

3.2.1. Data availability

In this work, different availability and usage scenarios are considered with respect to historical training data and external weather forecasts (see Table 1). As for historical data, a small and large set are considered. The former represents a scenario of minimal available data history, which might result from a newly installed PV plant, metering device or EMS. For the latter, a long operation history of the PV-battery system is assumed. Varying the training set size aims to assess whether (1) data-driven forecasting can be applied to new systems "out-of-the-box", and (2) extensive data histories allow for significant forecast improvements and additional economic benefits.

As for external weather data, GHI and cloud opacity forecasts are considered. These can be obtained against payment or partly free of charge [17,18]. The comparison of cases with and without use of weather forecasts enables to assess if additional effort and potential costs of incorporating external weather data are justified by savings through higher forecast accuracy. Moreover, it allows contrasting profitability gains with security concerns arising from the dependency on potentially manipulated external data.

Both larger training sets and additional features increase the amount of data to be processed, which ultimately impacts computational complexity. If the computational burden exceeds the capabilities of typical EMS hardware, cloud-computing may be necessary. In this case, sensitive consumption data may need to be provided to third-parties, which should be taken into account in the assessment of different forecasting strategies.

3.2.2. Modeling strategies

The considered modeling strategies comprise model (hyperparameter) selection and regular retraining (see Table 1). Both are typical procedures which usually improve accuracy, however, at the cost of increased computational complexity. Contrasting cases with and without applying these strategies provides insight regarding optimal trade-offs between profitability and computational burden. Similar to the processing of extensive data (see Section 3.2.1), complex modeling strategies may necessitate cloud computing. Resulting privacy concerns should be considered in the evaluation of forecasting strategies.

3.2.3. Naïve forecast

The first model type is a naïve persistence forecast, which is a popular benchmark and frequently applied by studies on forecasting-based optimization of PV-battery systems [6,11]. The term *persistence* stems from the fact that values within the prediction horizon of size w_{pr} are assumed to be the same as in a previous period. Daily persistence is considered for PV forecasting according to

$$\hat{P}^{\mathrm{PV}}_{h+1}, \ldots, \hat{P}^{\mathrm{PV}}_{h+w_{\mathrm{pr}}} = P^{\mathrm{PV}}_{h+1-24}, \ldots, P^{\mathrm{PV}}_{h+w_{\mathrm{pr}}-2\cdot 24}. \tag{2}$$

To account for different consumption patterns between weekdays and weekends, weekly persistence is applied for load forecasting as given by

$$\hat{P}^{\mathrm{L}}_{h+1}, \ldots, \hat{P}^{\mathrm{L}}_{h+w_{\mathrm{pr}}} = P^{\mathrm{L}}_{h+1-7\cdot 24}, \ldots, P^{\mathrm{L}}_{h+w_{\mathrm{pr}}-7\cdot 24}. \tag{3}$$

Eqs. (2) and (3) are implemented as rolling forecasts, generating predictions at each time step h for the following w_{pr} hours. The model is implemented in Python using the open-source forecasting library *Darts* [19]. Persistence forecasts are independent of training data, external weather data as well as model selection or training processes (see Table 1). This simplicity renders it also an attractive strategy for residential EMSs, as shown by the frequent consideration in many related works.

Table 1
Overview of forecasting cases with regards to model type, data availability and modeling strategies.

Case	Model	Data availability		Modeling strategies	
		Train set size	Weather data	Selection	Retraining
C1	Naïve	None	No	No	No
C2	GBDT	Small	No	No	No
C3	GBDT	Small	No	No	Yes
C4	GBDT	Small	No	Yes	No
C5	GBDT	Small	No	Yes	Yes
C6	GBDT	Small	Yes	No	No
C7	GBDT	Small	Yes	No	Yes
C8	GBDT	Small	Yes	Yes	No
C9	GBDT	Small	Yes	Yes	Yes
C10	GBDT	Large	No	No	No
C11	GBDT	Large	No	No	Yes
C12	GBDT	Large	No	Yes	No
C13	GBDT	Large	No	Yes	Yes
C14	GBDT	Large	Yes	No	No
C15	GBDT	Large	Yes	No	Yes
C16	GBDT	Large	Yes	Yes	No
C17	GBDT	Large	Yes	Yes	Yes
C18	Oracle	–	–	–	–

3.2.4. GBDT forecasts

To enable a fair comparison of different data availability and modeling strategy scenarios for PV and load forecasting, the same model type (gradient-boosted decision trees (GBDT)) is considered for the cases C2-C17 (see Table 1). Although comparing different ML models would provide additional insights, it is out of the scope of this work. GBDT [20] is a widely applied ML technique. Its popularity arises from its efficiency, interpretability and state-of-the-art accuracy, as, for example, demonstrated by regularly winning data mining and time series forecasting competitions [21]. Moreover, they are actively researched and improved as many recent versions, such as XGBoost [22], LightGBM [23] and CatBoost [24], demonstrate. GBDT combines the predictions of many individual decision trees, which constitute a set of weak learners. The trees are connected in series, so that each learner tries to minimize the residual between ground truth and prediction of the previous tree. The simultaneous high accuracy and efficiency renders GBDT a promising candidate for residential EMS applications.

GBDT is applied to predict the expected values for a prediction horizon of w_{pr} steps at time step h based on lag values $P^{\mathrm{PV|L}}_{h}, \ldots, P^{\mathrm{PV|L}}_{h-w_{\mathrm{hist}}}$ and covariates $\boldsymbol{X}^{\mathrm{PV|L}}_{h+1}, \ldots, \boldsymbol{X}^{\mathrm{PV|L}}_{h+w_{\mathrm{pr}}}$ according to

$$\hat{P}^{\mathrm{PV}}_{h+1}, \ldots, \hat{P}^{\mathrm{PV}}_{h+w_{\mathrm{pr}}} = \Phi\left(P^{\mathrm{PV}}_{h}, \ldots, P^{\mathrm{PV}}_{h-w_{\mathrm{hist}}}, \boldsymbol{X}^{\mathrm{PV}}_{h+1}, \ldots, \boldsymbol{X}^{\mathrm{PV}}_{h+w_{\mathrm{pr}}}\right) \tag{4}$$

and

$$\hat{P}^{\mathrm{L}}_{h+1}, \ldots, \hat{P}^{\mathrm{L}}_{h+w_{\mathrm{pr}}} = \Phi\left(P^{\mathrm{L}}_{h}, \ldots, P^{\mathrm{L}}_{h-w_{\mathrm{hist}}}, \boldsymbol{X}^{\mathrm{L}}_{h+1}, \ldots, \boldsymbol{X}^{\mathrm{L}}_{h+w_{\mathrm{pr}}}\right), \tag{5}$$

where a history window of w_{hist} steps is considered. The forecasts are generated every hour in a rolling fashion to provide up-to-date predictions. Covariates comprise calendric features, prosumer absence and external weather forecasts (see Table 2). Absence feature A assumes that prosumers can enter holidays in their EMS to allow load forecast models for better predictions in these periods. Since no correlation between PV generation and day of the week D and prosumer absence A exists, they are excluded for PV forecasting. As a result, $\boldsymbol{X}^{\mathrm{PV}} = \{H, \widehat{GHI}, \widehat{O}\}$ and $\boldsymbol{X}^{\mathrm{L}} = \{H, D, A, \widehat{GHI}, \widehat{O}\}$ in cases considering use of weather forecasts, and $\boldsymbol{X}^{\mathrm{PV}} = \{H\}$ and $\boldsymbol{X}^{\mathrm{L}} = \{H, D, A\}$ in cases without.

For scenarios applying model selection, the automatic hyperparameter optimization software *optuna* [25] is used in combination with three-fold time-series cross-validation [26]. The tuned hyperparameters and respective search spaces are listed in

Table 2
Covariates considered for GBDT-based forecasting.

Covariate	Sign	Value range
Hour of the day	H	$\{H \in \mathbb{N} \mid H = [0, \ldots, 23]\}$
Day of the week	D	$\{D \in \mathbb{N} \mid D = [0, \ldots, 6]\}$
Prosumer absence	A	$\{A \in \mathbb{N} \mid A = [0, 1]\}$
GHI forecasts	$\widehat{GHI}$	$\{\widehat{GHI} \in \mathbb{R} \mid \widehat{GHI} \geq 0\}$
Cloud opacity forecasts	$\widehat{O}$	$\{\widehat{O} \in \mathbb{R} \mid \widehat{O} = [0, \ldots, 1]\}$

Table 3. For all other hyperparameters, default values according to [27] are used. To find optimal sets of hyperparameters for PV ($\boldsymbol{\omega}^{\mathrm{PV}}_{\mathrm{opt}}$) and load forecasting ($\boldsymbol{\omega}^{\mathrm{L}}_{\mathrm{opt}}$) within the hyperparameter space $\boldsymbol{\Omega}$, the average root mean squared error (RMSE) over all N_{folds} folds and w_{pr} prediction steps is minimized by

$$\boldsymbol{\omega}^{\mathrm{PV|L}}_{\mathrm{opt}} = \arg\min_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \frac{\sum_{q=1}^{N_{\mathrm{folds}}} \sum_{j=1}^{w_{\mathrm{pr}}} \sqrt{\sum_{i=1}^{L^{(q)}_{\mathrm{val}}} \frac{\left(\hat{P}^{\mathrm{PV|L}}_{j,i}(\boldsymbol{\omega}) - P^{\mathrm{PV|L}}_{i}\right)^2}{L^{(q)}_{\mathrm{val}}}}}{N_{\mathrm{folds}} \cdot w_{\mathrm{pr}}}, \tag{6}$$

where $L^{(q)}_{\mathrm{val}}$ is the length of the validation set of the qth fold, $\hat{P}^{\mathrm{PV|L}}_{j,i}(\boldsymbol{\omega})$ the j-steps ahead forecast for the ith observation in the validation set of the qth fold based on a hyperparameter set $\boldsymbol{\omega}$ and $P^{\mathrm{PV|L}}_{i}$ the corresponding ground truth. The Bayesian optimization algorithm Tree Parzen Estimator (TPE) [28] is applied to find $\boldsymbol{\omega}_{\mathrm{opt}}$ according to (6) within a predefined number of hyperparameter set samples of $N_{\mathrm{trails}} = 2000$. While a large data history is considered sufficient for identification of optimal hyperparameters, small training datasets are likely to require regular model reselection, for example, due to lack of samples of all seasons of a year. Therefore, in cases considering a large data history (see Table 1), model selection is only conducted once based on the initial training set. For the ones with little training data, model selection is conducted repeatedly every three months. Scenarios without model selection use default parameters [27] and $w_{\mathrm{hist}} = 48$.

Table 3
Tuned hyperparameters and associated search spaces for GBDT-based forecasting.

No.	Hyperparameter	Search space
1	w_{hist}	[4, …, 192]
2	L1 regularization	[0, …, 100]
3	Bagging fraction	[0.1, …, 1]
4	Max. number of leaves in one tree	[20, …, 3000]
5	Feature fraction	[0.1, …, 1]
6	Max. depth of a tree	[3, …, 21]
7	Number of decision trees	[100, …, 10000]
8	Learning rate	[0.001, …, 0.3]

In cases which consider retraining (see Table 1), the GBDT model is repeatedly trained every week based on the entire previous data history. If no retraining is considered, only initial training is conducted. All GBDT-based cases (C2-C17) are implemented in Python using the open-source forecasting library *Darts* [19].

3.2.5. Oracle forecast

In addition to the naïve lower-end forecast benchmark, an oracle forecast is considered to quantify the theoretical optimum. The oracle forecast is characterized by perfect knowledge of the future, which includes that time resolution and w_{pr} are converging to infinity. This behavior is approximated with assuming perfect forecasts for every 5-min time step τ within a prediction horizon of $w_{\mathrm{pr}} = 2016$ steps (seven days) according to

$$\hat{p}^{\mathrm{PV}}_{\tau+1}, \ldots, \hat{p}^{\mathrm{PV}}_{\tau+w_{\mathrm{pr}}} = p^{\mathrm{PV}}_{\tau+1}, \ldots, p^{\mathrm{PV}}_{\tau+w_{\mathrm{pr}}} \tag{7}$$

and

$$\hat{p}^{\mathrm{L}}_{\tau+1}, \ldots, \hat{p}^{\mathrm{L}}_{\tau+w_{\mathrm{pr}}} = p^{\mathrm{L}}_{\tau+1}, \ldots, p^{\mathrm{L}}_{\tau+w_{\mathrm{pr}}}. \tag{8}$$

As the oracle forecast only constitutes a theoretical benchmark, no reasonable definition of data availability scenarios and modeling strategies can be made (see Table 1).

3.3. Spot price forecast

Hourly spot prices are typically published every day at 13:00 for the following day [29]. This leads to varying spot price knowledge horizons between 12 and 35 h, depending on the time of the day. Thus, if optimization horizons of more than 12 h are considered, price forecasts are required. Spot price forecasting is a complex task which depends on inputs such as wind production, consumption, calendric features and many more [30]. Recently, first providers offer access to advanced forecasts [31]. However, for the evaluation period considered in this work, historical forecasts could not be acquired. To avoid the assumption of knowing true prices for the entire optimization period, price forecasts are generated based on a GBDT model. Together with the publishing of spot prices for the next day, prices for the day after tomorrow are predicted at 13:00 according to

$$\hat{\Lambda}_{h+36}, \ldots, \hat{\Lambda}_{h+60} = \Phi\left(\Lambda_{h+35}, \ldots, \Lambda_{h-w_{\mathrm{hist}}}, \boldsymbol{X}^{\Lambda}_{h+36}, \ldots, \boldsymbol{X}^{\Lambda}_{h+60}\right), \tag{9}$$

with $\boldsymbol{X}^{\Lambda} = \{H, D\}$. Since advanced spot price forecasting is not the focus of this work, only lag values and calendric covariates are considered. Optimal hyperparameters are selected on a two-year history following a similar approach to (6). During the prediction of the evaluation set, the model is retrained on a daily basis. Note that varying spot price forecasts is not explicitly part of the case study. However, to validate this comparatively simple approach and assess if more complex price forecasts can be justified by significant economic benefits, a comparison to assuming true prices is included in Section 5.2.2.

3.4. Data manipulation

Cost-optimal scheduling of batteries requires external data such as spot prices and weather forecasts (see Fig. 2). While the required connection to the internet is the foundation for such smart applications, it also introduces new cyber vulnerabilities. Events such as the Mirai botnet in 2016 have shown that attacks on distributed internet of things (IoT)-devices are a reality [32]. Thus, also potential damage should be taken into consideration whenever assessing the advancements of IoT-based applications. Among the most famous and critical attacks in power systems are false data injections [33]. Based on an impact quantification of such attacks, the different EMS scenarios can be better assessed in terms of trade-offs between profitability and security. For that purpose, this subsection introduces two data manipulation scenarios. While Section 3.4.1 describes manipulation of spot price data, Section 3.4.2 is concerned with tampering of external weather forecasts.

3.4.1. Spot price manipulation

The objective of the considered price manipulation is to approximate an opposite behavior of cost-optimal operation. For that purpose, the attack model mirrors prices on their moving average according to

$$\underset{\sim}{\Lambda}_h = \Lambda_h - 2 \cdot \left(\Lambda_h - \sum_{\sigma=0}^{w_{\mathrm{avg}}} \frac{\Lambda_{h-\sigma}}{w_{\mathrm{avg}}}\right), \tag{10}$$

where $w_{\mathrm{avg}} = 23$. One attacker's motivation could be financial damage of prosumers. However, more critical is the potential switch from peak shaving to peak reinforcing behavior of flexible residential loads. If able to manipulate price input of multiple EMSs, an attacker could target overloading situations entailing disconnection of customers. The attack model in (10) is exemplarily depicted in Fig. 3.

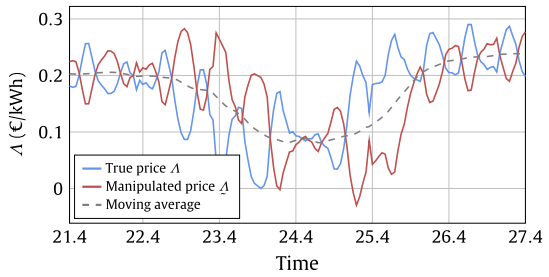


Fig. 3. Exemplary depiction of the spot price manipulation according to (10) on an excerpt from April 2022.

3.4.2. GHI and cloud opacity forecast manipulation

Reducing the accuracy of inputs for PV and load forecasts is likely to translate to lower economic benefits due to sub-optimal battery scheduling. Therefore, an attacker's motivation for manipulating weather forecast inputs could be financial damage. As damage increases over time, attackers may try to keep modifications discreet to avoid detection. This behavior is imitated by adding noise of different intensity levels to GHI and cloud opacity forecasts according to

$$\underline{\widehat{GHI}}_{h+j} = \widehat{GHI}_{h+j} \cdot (1 + \alpha R)\,, \forall j \in [1, 2, ..., w_{\text{pr}}] \tag{11}$$

and

$$\underline{\widehat{O}}_{h+j} = \widehat{O}_{h+j} \cdot (1 + \alpha R)\,, \forall j \in [1, 2, ..., w_{\text{pr}}], \tag{12}$$

with R being a random number drawn from the uniform distribution $R \sim U(-1, 1)$ and α the aggressiveness of the attack with $\alpha \in [0.2, 1, 10]$. To further hide the attack, physically implausible values are avoided by containing manipulated GHI values between zero and the maximum value in the respective region, and cloud opacity between zero and one. The attack models in (11) and (12) are exemplarily depicted in Fig. 4.

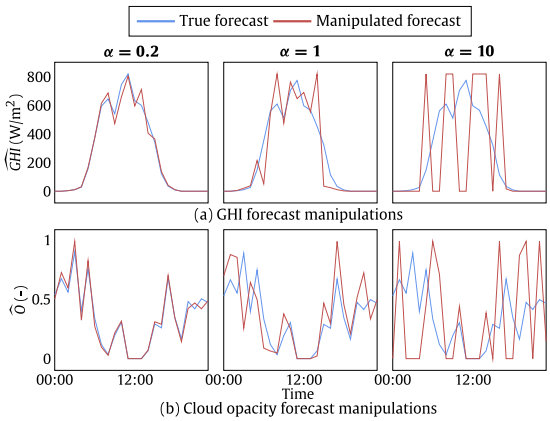


Fig. 4. Exemplary depiction of (a) GHI and (b) cloud opacity forecast manipulations on an excerpt from June 2022.

4. Experimental setup and metrics

This section is concerned with the experimental setup and applied metrics of the present study. Section 4.1 provides prosumer and data specifications. Thereafter, the applied forecasting performance metrics (Section 4.2) and economic performance indicators (Section 4.3) are introduced.

4.1. Prosumer and data specification

The two prosumers are located in Roskilde, Denmark, and are subject to the DK2 day-ahead price zone. Both follow the schematic representation in Fig. 2. They are equipped with a PV system of 6 kW_p (prosumer 1) and 5 kW_p (prosumer 2), respectively. For the battery system, $\bar{s} = 8\,\text{kWh}$, $\underline{s} = 0.8\,\text{kWh}$ and $\eta = 0.95$ is considered. Moreover, the inverter comes with $\bar{p}_{\text{inv}} = 6\,\text{kW}$ for prosumer 1 and $\bar{p}_{\text{inv}} = 5\,\text{kW}$ for prosumer 2. All components of the residential energy systems follow a dimensioning typical for the Danish case. The self-consumption of the two prosumers without battery is 67% and 44%, respectively. In both cases the EMS receives PV and load measurements as 5-min averages. External weather forecasts (GHI and cloud opacity) are provided with an hourly resolution. For the two prosumers, historical data of different length is available. The dataset of prosumer 1 comprises approximately three years, beginning on the 1st of September 2019 and ending on the 30th of October 2022. For prosumer 2, 14.5 months between the 15th of August 2021 and the 30th of October 2022 are available. In both cases, the last 14 months (1st of September 2021 to 30th of October 2022) are reserved for evaluation in Section 5. As detailed in Section 3.2.1, historical training data of different size are considered within the forecasting cases. While the small set comprises the last two weeks of August 2021, the large one spans over two years from 1st of September 2019 to 31st of August 2021. Since historical data of prosumer 2 are not available before the 15th of August 2021, forecasting cases considering two years of training data are only evaluated on prosumer 1.

4.2. Forecasting performance metrics

This subsection introduces the performance metrics applied to evaluate accuracy (Section 4.2.1) and computational complexity (Section 4.2.2) of the forecasting cases.

4.2.1. Accuracy

The accuracy for a j-steps ahead PV or load prediction under forecasting case C is quantified based on the normalized RMSE according to

$$\begin{aligned} nRMSE_j^{\text{PV|L},C} &= \frac{\sqrt{\frac{\sum_{i=1}^{L_\text{eval}} \left(\hat{P}_{j,i}^{\text{PV|L},C} - P_i^{\text{PV|L}}\right)^2}{L_\text{eval}}}}{\frac{\sum_{i=1}^{L_\text{eval}} P_i^{\text{PV|L}}}{L_\text{eval}}} \\ &= \frac{\sqrt{L_\text{eval} \sum_{i=1}^{L_\text{eval}} \left(\hat{P}_{j,i}^{\text{PV|L},C} - P_i^{\text{PV|L}}\right)^2}}{\sum_{i=1}^{L_\text{eval}} P_i^{\text{PV|L}}}, \end{aligned} \tag{13}$$

where L_eval is the length of the 14 month evaluation set. Normalization removes the impact of the scale of PV production and load consumption and thus facilitates comparison among *different* prosumers. The overall performance of the considered multi-step forecasts is quantified as the average over the entire forecasting horizon of size w_{pr} according to

$$nRMSE_{\text{avg}}^{\text{PV|L},C} = \frac{\sum_{j=1}^{w_{\text{pr}}} nRMSE_j^{\text{PV|L},C}}{w_{\text{pr}}}. \tag{14}$$

For comparison of different forecasting cases on the *same* prosumer, the relative averaged normalized RMSE $rRMSE_{\text{avg}}^{\text{PV|L},C}$ is

considered, which follows from dividing $nRMSE_{\text{avg}}^{\text{PV},C}$ and $nRMSE_{\text{avg}}^{\text{L},C}$ by the respective highest value among all cases[1] as given by

$$rRMSE_{\text{avg}}^{\text{PV|L},C} = \frac{nRMSE_{\text{avg}}^{\text{PV|L},C}}{\max\left\{nRMSE_{\text{avg}}^{\text{PV|L,C1}}, \ldots, nRMSE_{\text{avg}}^{\text{PV|L,C17}}\right\}}. \tag{15}$$

4.2.2. Computational complexity

The computational complexity of an ML model in terms of training time, prediction time and space can be expressed with the respective big O notation [34]. For the considered GBDT model, these are given as $\mathcal{O}_{\text{train}}\,(L_{\text{train}}\log(L_{\text{train}})dm)$ (train time complexity), $\mathcal{O}_{\text{pred}}\,(km)$ (prediction time complexity) and $\mathcal{O}_{\text{space}}\,(vm)$ (space complexity), where L_{train} is the length of the training set, d the feature dimension, m the number of trees, k the depth of the trees and v the number of nodes in the trees [35,36]. Based on these notations, the number of operations for training and prediction as well as memory needs of a PV and load forecasting model of case C are approximated by

$$O_{\text{train}}^{\text{PV|L},C} \approx N_{\text{trails}}^{C} L_{\text{train}}^{C}\log(L_{\text{train}}^{C})d^{\text{PV|L},C}m^{\text{PV|L},C}, \tag{16}$$

$$O_{\text{pred}}^{\text{PV|L},C} \approx k^{\text{PV|L},C}m^{\text{PV|L},C} \tag{17}$$

and

$$M^{\text{PV|L},C} \approx v_{\text{max}}^{C} m_{\text{max}}^{C}, \tag{18}$$

where m_{max}^{C} and v_{max}^{C} constitute the maximum implemented number of trees and nodes in trees, respectively.[2] For cases considering retraining, $L_{\text{train}}^{\text{PV|L},C}$ is defined by the size of the largest retraining set during evaluation. To facilitate comparison among the different forecasting cases, $O_{\text{train}}^{\text{PV|L},C}$, $O_{\text{pred}}^{\text{PV|L},C}$ and $M^{\text{PV|L},C}$ are divided by the respective maximum value across all cases according to

$$rO_{\text{train}}^{\text{PV|L},C} = \frac{O_{\text{train}}^{\text{PV|L},C}}{\max\left\{O_{\text{train}}^{\text{PV|L,C1}}, \ldots, O_{\text{train}}^{\text{PV|L,C17}}\right\}}, \tag{19}$$

$$rO_{\text{pred}}^{\text{PV|L},C} = \frac{O_{\text{pred}}^{\text{PV|L},C}}{\max\left\{O_{\text{pred}}^{\text{PV|L,C1}}, \ldots, O_{\text{pred}}^{\text{PV|L,C17}}\right\}} \tag{20}$$

and

$$rM^{\text{PV|L},C} = \frac{M^{\text{PV|L},C}}{\max\left\{M^{\text{PV|L,C1}}, \ldots, M^{\text{PV|L,C17}}\right\}}. \tag{21}$$

4.3. Economic performance indicators

The set of evaluated EMS scenarios comprises rolling-horizon optimization based on the forecasting cases C1–C18 and the offline rule-based benchmark. For simplicity, optimization-based EMS scenarios are referred to as their underlying forecasting case. The energy cost under an EMS scenario C' is calculated by

$$K^{C'} = \sum_{i=1}^{L_{\text{eval},\tau}} \left[p_i^{\text{b},C'}(\lambda_i + f_i) - p_i^{\text{s},C'}\lambda_i\right]\Delta T, \tag{22}$$

where $L_{\text{eval},\tau}$ is the length of the evaluation period in 5-min resolution. Based on the costs, the economic benefit under a scenario C' is expressed as the difference to the baseline cost without a battery (K^{base}) according to

$$B^{C'} = K^{\text{base}} - K^{C'}. \tag{23}$$

To simplify comparison among the scenarios, their benefit is assessed by comparing with the theoretical maximum. As the maximum benefit is achieved in case of assuming oracle forecasts (C18), the resulting relative benefit under an EMS scenario C' can be expressed as

$$rB^{C'} = \frac{B^{C'}}{B^{\text{C18}}}. \tag{24}$$

Note that in case of a monthly cost analysis of the evaluation period, the relative benefit $rB^{C'}$ will be denoted as $rB_m^{C'}$.

5. Results

This section evaluates the considered EMS scenarios regarding profitability, complexity and security. In preparation of that, Section 5.1 examines the underlying forecasting cases in terms of accuracy and computational complexity. Thereafter, the evaluation of EMS scenarios follows in Section 5.2. Unless otherwise stated, results are based on the default forecasting horizon $w_{\text{pr}} = 36$ (see Section 3.1.2).

5.1. Performance evaluation of forecasting cases

In this subsection, the forecasting cases are first evaluated regarding accuracy (Section 5.1.1) and computational complexity (Section 5.1.2). In Section 5.1.3, a conclusion on the trade-off between these factors is provided. Finally, Section 5.1.4 analyzes the error behavior of PV and load forecasts over the forecasting horizon.

5.1.1. Accuracy

A performance overview for all cases is provided in Table 4. Note that $rRMSE_{\text{avg}}$ is depicted, which shows the relative performance of each case against the worst forecast for a given variable and prosumer. Highest accuracy is written in bold, second best is underlined and third best dotted underlined. The *absolute* averaged RMSEs of the best performing cases are $RMSE_{\text{avg}}^{\text{PV,C17}} = 0.3196\,\text{kW}$ and $RMSE_{\text{avg}}^{\text{L,C9}} = 0.5065\,\text{kW}$ (prosumer 1) as well as $RMSE_{\text{avg}}^{\text{PV,C9}} = 0.2569\,\text{kW}$ and $RMSE_{\text{avg}}^{\text{L,C9}} = 0.2555\,\text{kW}$ (prosumer 2).

Model type. The persistence model (C1) forms the lower performance end for load forecasting. For PV forecasting, the GBDT model without large training data, external weather forecasts, model selection and retraining (C2) performs worst. Apart from this exception, GBDT-based forecasts outperform the persistence benchmark by at least 12.3 percentage points, which proves the existence of learnable patterns in P^{PV} and P^{L}, justifying the application of ML models.

Historical data size. The availability of comprehensive training data improves accuracy significantly in cases without use of external weather input or retraining (e.g., C2 vs. C10 and C6 vs. C14). Other cases only exhibit minor improvements, which is caused by two factors. On the one hand, using highly correlated weather forecasts as input simplifies the problem and makes comprehensive training data obsolete. On the other hand, retraining exploits newly incoming data and thus minimizes the need for large data histories. These findings suggest that data-driven forecasting can also be applied in scenarios of small data histories, such as newly installed EMSs. For load forecasts, large training data can even worsen results (e.g., C9 vs. C17). The reason is a changing consumption behavior of prosumer 1 due to purchase of a second EV in 2022, which renders older load data less useful and hinders model selection and training.

[1] C18 (oracle) is excluded as it constitutes no realistic forecasting case.

[2] In cases considering model selection, combinations of v and m may occur which entail higher memory needs than the finally selected model.

Table 4
Accuracy and computational complexity for PV and load forecasts. Highest accuracy in bold, second best underlined and third best dotted underlined.

Case	Model	Train set size	Weather data	Selection	Retraining	Accuracy				Computational complexity									
						$rRMSE_{avg}$ (–)				rO_{train} (–)				rO_{pred} (–)				rM (–)	
						Prosumer 1		Prosumer 2		Prosumer 1		Prosumer 2		Prosumer 1		Prosumer 2		Prosumer 1		Prosumer 2	
						PV	Load	PV	Load	PV	Load	PV	Load	PV	Load	PV	Load	PV	Load	PV	Load
C1	Naïve	–	No	No	No	0.916	1	0.954	1	0	0	0	0	0	0	0	0	0	0	0	0
C2	GBDT	Small	No	No	No	1	0.999	1	0.78	0	0	0	0	0	0	0	0	0	0	0	0
C3	GBDT	Small	No	No	Yes	0.754	0.782	0.778	0.715	0	0	0	0	0	0	0	0	0	0	0	0
C4	GBDT	Small	No	Yes	No	0.795	0.838	0.835	0.722	0.9	0.3	0.9	0.5	0.6	0.9	0.3	0.8	1	1	1	1
C5	GBDT	Small	No	Yes	Yes	0.719	0.759	0.753	0.702	1	0.3	1	0.5	0.6	0.9	0.3	0.8	1	1	1	1
C6	GBDT	Small	Yes	No	No	0.541	0.877	0.567	0.769	0	0	0	0	0	0	0	0	0	0	0	0
C7	GBDT	Small	Yes	No	Yes	0.401	0.678	0.429	0.709	0	0	0	0	0	0	0	0	0	0	0	0
C8	GBDT	Small	Yes	Yes	No	0.412	0.714	0.446	0.725	0.9	0.3	0.9	0.9	1	1	1	1	1	1	1	1
C9	GBDT	Small	Yes	Yes	Yes	0.381	**0.664**	**0.411**	**0.701**	1	0.4	1	1	1	1	1	1	1	1	1	1
C10	GBDT	Large	No	No	No	0.705	0.828	–	–	0	0	–	–	0	0	–	–	0	0	–	–
C11	GBDT	Large	No	No	Yes	0.701	0.761	–	–	0	0	–	–	0	0	–	–	0	0	–	–
C12	GBDT	Large	No	Yes	No	0.694	0.791	–	–	0.2	0.2	–	–	0	0.1	–	–	1	1	–	–
C13	GBDT	Large	No	Yes	Yes	0.693	0.752	–	–	0.4	0.3	–	–	0	0.1	–	–	1	1	–	–
C14	GBDT	Large	Yes	No	No	0.377	0.77	–	–	0	0	–	–	0	0	–	–	0	0	–	–
C15	GBDT	Large	Yes	No	Yes	0.373	0.693	–	–	0	0	–	–	0	0	–	–	0	0	–	–
C16	GBDT	Large	Yes	Yes	No	0.363	0.753	–	–	0.5	0.6	–	–	0.1	0.7	–	–	1	1	–	–
C17	GBDT	Large	Yes	Yes	Yes	**0.36**	0.681	–	–	0.8	1	–	–	0.1	0.7	–	–	1	1	–	–
C18	Oracle	–	–	–	–	0	0	0	0	–	–	–	–	–	–	–	–	–	–	–	–

Weather forecasts. Incorporating GHI and cloud opacity forecasts improves both PV and load forecasting in all cases (e.g., C3 vs. C7). While the impact is significant for PV forecasts as anticipated, the potential for load forecast improvement depends on the prosumer. As prosumer 1 is aligning consumption (EV charging) with PV production, the impact is stronger compared to prosumer 2.

Model selection. Model selection improves accuracy in all cases. However, the improvement compared to using default hyperparameters usually is only in the range of 1 to 3 percentage points (e.g., C15 vs. C17).

Retraining. Cases considering retraining exhibit lower *rRMSE* values compared to their non-retrained counterparts (e.g., C6 vs. C7). As expected, the impact is larger for models with small initial training sets. Nevertheless, even for cases assuming two years of training data, retraining can significantly improve accuracy, given that new changes in data distributions appear. As mentioned above, this is the case for load consumption of prosumer 1, which explains the comparatively large improvement for P^{L} forecasts, for example, between C16 and C17. In contrast, PV models trained on two years of historical data have learned most patterns, rendering improvement through frequent retraining marginal (C16 vs. C17). Compared to model selection, the improvement potential of retraining is at least in the same range and in many cases larger.

5.1.2. Computational complexity

An overview of the computational complexities for all cases is provided in Table 4. As for $rRMSE_{avg}$, relative complexities of each case against the highest value for a given variable and prosumer are shown.

Time complexity. From Table 4 it can be seen that model selection is the dominating factor for both training and prediction time complexity (rO_{train} and rO_{pred}). Cases without selection appear as 0 because their complexities are orders of magnitude smaller. The time required for the considered model selection process (see Section 3.2.4) and training of one GBDT model is in the range of 2 to 24 h on a high-performance computing (HPC) cluster [37]. This process could not be realized locally using the hardware of a residential EMS and would require cloud computing. In contrast, training a default model is in the order of seconds to minutes on a standard laptop, allowing local realization. The strong impact of model selection results from the entailed increase of model size, which is a key driver of rO_{train} and rO_{pred}. For example, in most cases the number of decision trees raises from 100 (default value) to over 7000. In addition, selection requires training of N_{trails} models, which further increases rO_{train}.

The use of weather forecasts as model input also impacts rO_{train} and rO_{pred}, as can be seen from comparison of C13 and C17. This can be mainly explained by a higher feature dimension. Retraining does not affect rO_{pred} as it does not increase model size (e.g., C12 vs. C13). In contrast, since retraining is conducted on larger data than initial training, rO_{train} increases as the comparison of C16 and C17 shows. Nevertheless, the impact of additional weather input and retraining is negligible in cases without model selection.

Space complexity. Space complexity is also dominated by model selection (see Table 4). In these cases, rM is orders of magnitude larger since the defined hyperparameter space (see Table 3) requires evaluation of models of significantly larger size compared to the default model.

5.1.3. Trade-off between accuracy and complexity

The highest accuracy (apart from oracle case C18) is achieved by GBDT-based forecasting using external weather forecasts, model selection and retraining (C9 and C17). As can be seen from Table 4, this comes at the cost of high time and space complexity. However, second or third best results are achieved by models with significantly smaller complexity (C7 and C15). In these cases, a default model is combined with the use of external weather data and retraining. The loss in accuracy in terms of $rRMSE_{avg}$ is in the range of 1 to 2 percentage points. It can be concluded that C7 and C15 provide best trade-offs with regards to accuracy and complexity. An example of the forecasts under C7 is depicted in Fig. 5.

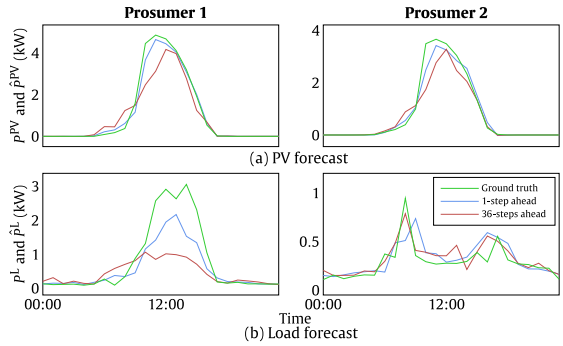


Fig. 5. Representative excerpts of 1- and 36-steps (hours) ahead forecasts of (a) PV production and (b) load consumption.

5.1.4. Error development over the prediction horizon

Understanding the error behavior over the forecasting horizon adds further insight to the comparison of average performance values in Section 5.1.1. The $nRMSE^{PV,C9}$ and $nRMSE^{L,C9}$ are depicted in Fig. 6 over the prediction horizon.

In all cases the error increases with the horizon. This constitutes a typical behavior as predictions further into the future are usually more difficult. PV forecast errors of both prosumers follow a similar trend, which can be explained by their local proximity. The error increases rapidly up to a four-steps ahead horizon. Thereafter, the gradient drops. The accuracy of load forecasts is lower than for PV forecasts, which can be attributed to the randomness of consumer behavior. Moreover, the $nRMSE^{L,C9}$ evolves differently among the prosumers. While errors begin in a similar range at one-step ahead, they follow different gradients over the consecutive horizon. The error of prosumer 2 almost stays constant. This stems from comparatively similar load patterns among different days. The resulting correlation with time of the day renders load forecasting rather a regression problem, explaining the stable $nRMSE^{L,C9}$ over the horizon. For prosumer 1 the error increases rapidly until five-steps ahead, followed by a saturation phase. In this case, load patterns exhibit more variation between different days due to less predictable EV charging. The error can be kept small within the first hours, due to similarity with the most recent lag values of P^{L}. Once this effect cancels out, the stronger variation results in higher $nRMSE^{L,C9}$ values compared to prosumer 2.

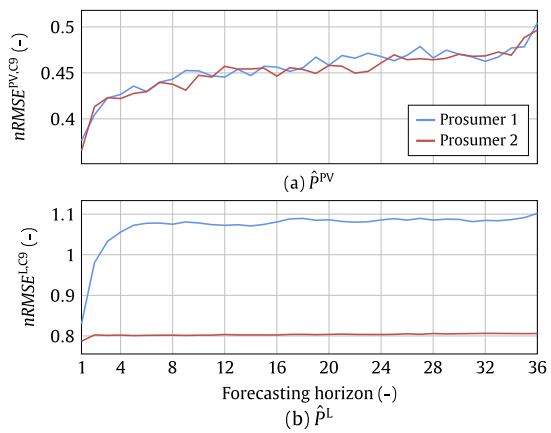


Fig. 6. *nRMSE* of (a) $\hat{P}^{PV}$ and (b) $\hat{P}^{L}$ for both prosumers over the forecasting horizon exemplary on C9.

5.2. Performance evaluation of EMS scenarios

This subsection evaluates the economic benefit of the EMS scenarios, considering trade-offs with computational complexity and security. Section 5.2.1 provides a summary of the prosumers energy and cost quantities. In Section 5.2.2, scenarios are compared and recommendations on best trade-offs provided. Based on the suggested scenario, the impact of time of the year (Section 5.2.3), optimization horizon (Section 5.2.4) and data manipulation (Section 5.2.5) on the economic value is assessed.

5.2.1. Prosumers overview

Table 5 provides an overview of the prosumers energy and cost characteristics. Prosumer 1 exhibits higher production and consumption values over the 14-month evaluation period. Both exhibit negative energy costs[3] without using a battery, equal to $K^{base} = -282$ € and -43 €, respectively. The addition of a storage system enables a maximum additional benefit of $B^{C18} = 466$ € and 555 € for the theoretical EMS scenario considering optimization based on perfect forecasts (C18). Consequently, the maximum total revenue over the evaluation period amounts to 748 € and 598 €, respectively. The offline rule-based benchmark, which minimizes energy exchanges, achieves 45% and 56% of B^{C18} for prosumer 1 and 2, respectively. The following evaluation examines what fraction of the maximum theoretical benefit B^{C18} the various EMS scenarios achieve based on their underlying forecasting case.

Table 5
Overview of energy and cost quantities of prosumer 1 and 2 based on the full 14-month evaluation period.

Energy and cost quantities	Prosumer 1	Prosumer 2
PV production (kWh)	7505	5748
Consumption (kWh)	4766	3210
Base cost K^{base} (€)	−282	−43
Max benefit B^{C18} (€)	466	555
Rule-based benefit B^{rb} (€)	209	310

5.2.2. Scenario comparison and recommendation

In Fig. 7, the relative benefit *rB* over the entire 14-month evaluation period is depicted for all EMS scenarios.

Impact of price forecast. The impact of spot price forecast accuracy is assessed on C17 (prosumer 1) and C9 (prosumer 2). Perfect price forecasts increase *rB* by 0.0004 (prosumer 1) and 0.0019 percentage points (prosumer 2). This translates to an additional benefit of 0.2051 € and 1.072 €, respectively, over a period of 14 months. It can be concluded that sophisticated price forecasts as extension of available prices are not required for cost-optimal control of residential PV-battery systems.

Impact of PV and load forecasts. From C1–C9, it can be seen that differences of *rB* between scenarios exhibit similar trends for the two prosumers. In both cases, even naïve persistence-based optimization (C1) achieves significant improvements compared to offline rule-based control (78% and 86% of the theoretical optimum), without need for model training and selection or external weather forecasts. Consequently, even in the simplest case, rolling-horizon optimization enables additional gains of 152.11 € and 168.2 € to the prosumers compared to the offline rule-based scheme. Nevertheless, with the exception of C2, all GBDT-based scenarios outperform persistence-based optimization, motivating the use of ML. This is in line with the $rRMSE_{avg}$-based findings presented in Section 5.1.1.

As can be seen from Fig. 7, highest relative benefits are achieved by C7, C9, C15 and C17 ($rB = 0.9$ for prosumer 1 and $rB = 0.93$ for prosumer 2). This translates to additional income of 56.01 € and 36.53 €, respectively, through use of ML models compared to simple persistence forecasting. Among these scenarios, C7 exhibits advantages from a computational complexity perspective, since it avoids model selection (opposed to C9 and C17) and dependency on large historical data (in contrast to C15 and C17). On the one hand, this indicates that ML-based forecasts are also economically beneficial for new systems without extensive data history. On the other hand, it suggests that the computational burden of extensive model selection (see Section 5.1.2) is economically not justified.

[3] Negative energy costs result from the fact that revenues for PV production exceed electricity costs in the evaluation period.

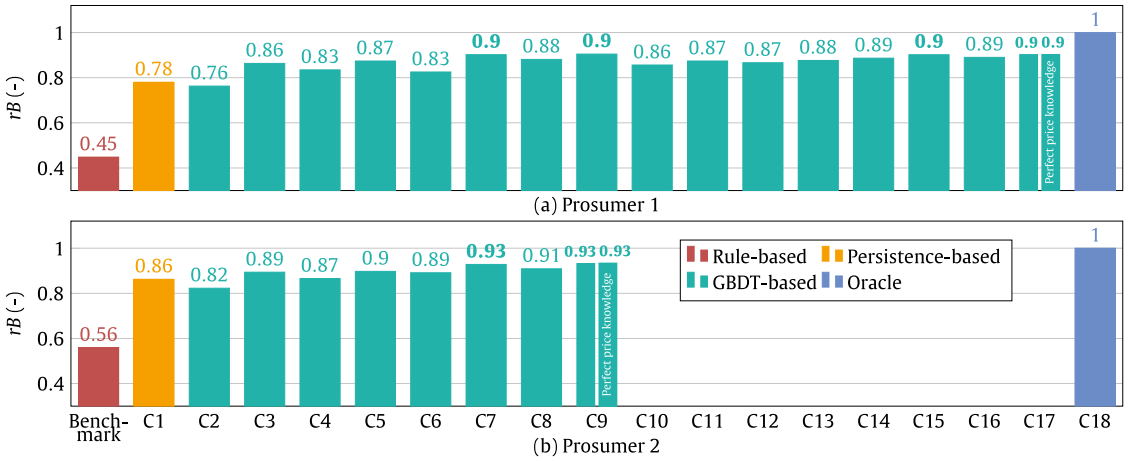


Fig. 7. Relative benefit rB of all EMS scenarios over the full evaluation period for (a) prosumer 1 and (b) prosumer 2. Best values are written in bold (excluding oracle). Impact of assuming perfect future spot price knowledge is exemplary shown on C17 and C9.

The computational simplicity of C7 also translates to benefits in terms of security. The use of default parameters facilitates local implementation, avoiding any need for externalization of PV and load forecasts to cloud-based solutions (see Section 5.1.2). Thus, no (potentially sensitive) data need to be provided to third-parties. Nevertheless, C7 is dependent on external weather forecasts, which might introduces opportunities for adversaries, as further evaluated in Section 5.2.5. In contrast, C3 avoids using external weather data, while exhibiting the same advantages in terms of computational complexity. However, the potential security advantage comes at the cost of a benefit reduction of 3-4 percentage points, which translates to 19.09€ for prosumer 1 and 17.76€ for prosumer 2. Note that all optimization-based scenarios (C1–C18) require external price data. Thus, to run the residential energy system (Fig. 2) isolated from public networks, offline rule-based control is the only opportunity among the considered EMS scenarios. The impact of price manipulations is evaluated in Section 5.2.5.

To conclude, the EMS scenario providing best trade-offs in terms of economic profitability and computational complexity is seen in rolling-horizon optimization based on forecasts of a default GBDT model, using a short two weeks initial training set, external weather forecast inputs and weekly retraining (C7). It achieves the same financial gain as models resulting from extensive selection processes at significantly lower computational costs. Moreover, it can be applied to new systems with short data history. The small computational burden also eases local implementation, providing data security advantages. Nevertheless, for a holistic assessment of C7, the sensitivity to attacks on required external data streams (price and weather forecasts) must be quantified, which follows in Section 5.2.5. In the subsequent sections, C7 is considered as representative case for GBDT-based forecasts.

5.2.3. Impact of time of the year

Fig. 8 depicts the monthly relative benefit rB_m for offline rule-based control as well as persistence-, GBDT- and oracle-based optimization. Relative benefits are volatile under the rule-based scheme in both prosumer cases, ranging from $rB_m = 0.102$ to $rB_m = 0.887$. The pronounced under-performance of rule-based control around December 2021 and July 2022 is driven by two factors. On the one hand, the margin for battery utilization is low, either due to small PV production (December) or low consumption because of holidays (July). While optimization exploits the remaining benefits, the battery is barely used under rule-based control. On the other hand, the respective months exhibit particularly high and volatile prices. As the considered rule-based scheme only discharges to cover load demand, high prices cannot be actively exploited by grid exports.

Optimization-based battery scheduling provides more stable values over the year, even in case of persistence forecasts. Therefore, the difference between persistence- and GBDT-based optimization is largely stable. While persistence-based optimization is outperformed by rule-based control in some months, GBDT-based optimization achieves the best results across the entire evaluation period.

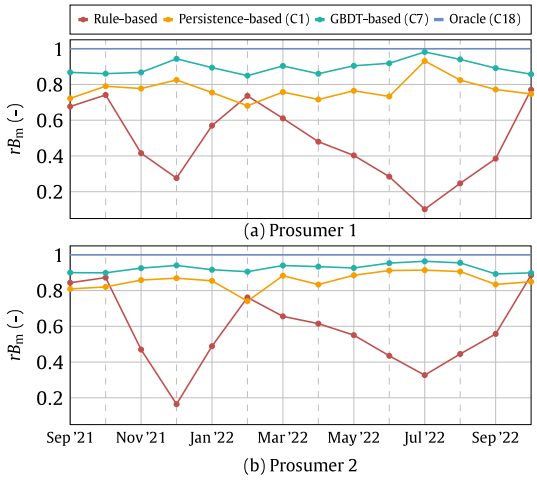


Fig. 8. Monthly relative benefit rB_m for (a) prosumer 1 and (b) prosumer 2.

5.2.4. Impact of prediction horizon

In Fig. 9, the relative benefit rB is depicted for varying forecasting horizons.[4] The oracle-based scenario (C18) and offline rule-based benchmark are included for the sake of easier comparison, although they do not depend on w_{pr} and therefore remain constant. On horizons $w_{\text{pr}} < 4$, rule-based control outperforms persistence- and GBDT-based optimization. This can be explained by under-utilization of the battery due to the 50% SOC condition at the end of the optimization horizon (see Section 3.1.2). To satisfy this constraint for short horizons, the battery remains at a SOC close to 50% even in periods of abundant PV production, instead of charging. Removing this constraint further deteriorates performance, as the myopic optimal decision is to fully discharge the battery. For $w_{\text{pr}} \geq 4$, optimization outperforms rule-based control. In case of GBDT-based optimization, the relative benefit saturates around $w_{\text{pr}} = 16$ at $rB = 0.9$ (prosumer 1) and $rB = 0.93$ (prosumer 2), respectively. Thus, horizons between $w_{\text{pr}} = 16$ and $w_{\text{pr}} = 20$ should be favored because the additional economic benefit of a longer horizon is negligible and unnecessarily increases complexity.

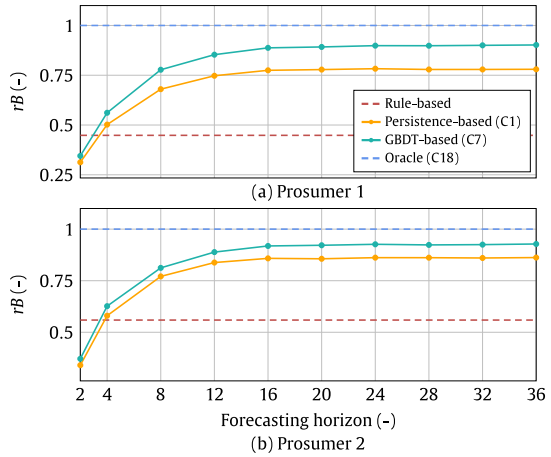


Fig. 9. Relative benefit rB over the forecasting horizon for (a) prosumer 1 and (b) prosumer 2.

5.2.5. Impact of data manipulation

The impact of weather forecast and price manipulation on the economic benefits is evaluated on C7 and shown in Fig. 10.

Weather forecast manipulation. The manipulation of weather forecast model input exhibits minor economic impact on both prosumers. Even for $\alpha = 10$, the relative benefit over the entire evaluation period only drops from $rB^{\text{C7}} = 0.902$ to $rB^{\text{C7}} = 0.877$ (prosumer 1) and $rB^{\text{C7}} = 0.928$ to $rB^{\text{C7}} = 0.904$ (prosumer 2), which translates to loss of 11.62 € and 13.53 €, respectively. This behavior can be explained by two factors. (1) Regular retraining allows the model to recognize and react on reduced information content of weather forecasts by putting less weight on these inputs. If the weather forecasts would contain no information, the model would approximate C3, which neglects external weather data. Therefore, the maximum reduction which can result from weather forecast manipulation can be quantified for C7 as $rB^{\text{C7}} = rB^{\text{C3}} = 0.863$ (prosumer 1) and $rB^{\text{C7}} = rB^{\text{C3}} = 0.894$ (prosumer 2), respectively. (2) As the introduced errors are randomly distributed around the true values, simultaneously processing a sequence of $w_{\text{pr}} = 36$ steps allows the model to derive a rolling average of the weather inputs. Therefore, they provide information even under high noise levels as the case for $\alpha = 10$ (see Fig. 4). It can be concluded that manipulation of external weather forecasts constitutes only a small economic risk for prosumers applying forecasting-based optimization on their PV-battery system. Therefore, avoiding use of external weather forecasts for security reasons is not well justified. This supports the highlighting of C7 in Section 5.2.2 as best trade-off in terms of profitability, complexity and security within the considered scenarios.

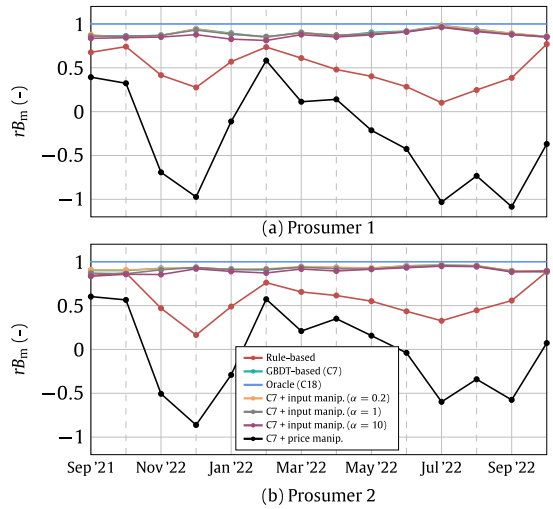


Fig. 10. Impact of data manipulation on the monthly relative benefit rB_m for (a) prosumer 1 and (b) prosumer 2.

Spot price manipulation. As can be seen from Fig. 10, price manipulation severely decreases the relative benefits in both prosumer cases. Benefits are lower than under offline rule-based control for each month of the evaluation period. Over the entire evaluation period, values drop from $rB^{\text{C7}} = 0.902$ to $rB^{\text{C7}} = -0.362$ (prosumer 1) and $rB^{\text{C7}} = 0.928$ to $rB^{\text{C7}} = -0.099$ (prosumer 2). This translates to a loss of 579.32 € and 570.34 € respectively, compared to the non-manipulated cases. It can be concluded that a manipulation of spot prices would reduce the economic benefit of optimization-based battery scheduling drastically and even generate additional cost compared to a scenario without a battery. Since all optimization-based scenarios (C1–C18) depend on price data, only offline rule-based control mitigates such risk within the considered EMS scenarios.

Although the potential impact of price manipulation is high, it would require an attack to last for months. To avoid price manipulations remaining undetected over long periods, residential EMSs should be equipped with concepts for spot price data integrity checking. In this case, forecasting-based rolling-horizon optimization according to C7 still provides the best trade-off in terms of economic profitability, computational complexity and security within the evaluated scenarios.

6. Discussion

In this section, implications of the results from Section 5 are discussed in a broader context. Considered aspects include the

[4] Note that $w_{\text{pr}} = 1$ is not included, since it may often lead to infeasible optimization problems due to the ending SOC constraint.

use of ML in residential EMSs (Section 6.1), result transferability (Section 6.2) and market readiness (Section 6.3).

6.1. ML for residential EMSs

Scenario C7 demonstrates that highest economic benefits from PV-battery systems can even be obtained by using default ML forecasting models with almost no initial training data. Instead of model selection and extensive training data, the incorporation of weather forecasts and frequent retraining is of key relevance. This importance results from characteristics of residential production and consumption patterns. On the one hand, the strong correlation of PV production and weather features enables accurate forecasts even with simple models. On the other hand, residential consumption exhibits regular changes (e.g., through new electronic devices), which reduces the value of extensive data histories and explains the importance of frequent retraining. These findings suggest that ML-based forecasting is beneficial for residential EMSs. However, data- and computation-intensive approaches, including deep learning, are not suitable and justifiable.

An alternative to combining ML-based forecasting with optimization may be seen in more advanced rule-based concepts, which include price and weather forecast information. An argument often used in favor of the latter is low computational burden. However, the simplicity of C7 can hardly be undercut, and it avoids manual development and tweaking of control rules. Both concepts can be locally implemented and thus avoid provision of sensitive consumption data to third parties. However, ML-based forecasting additionally exhibits robustness against weather input manipulations, since it automatically puts less weight on affected features through retraining. Last but not least, rule-based approaches can only approximate the economic benefits achieved by forecasting-based optimization.

6.2. Transferability

Section 5 shows similar impact of forecasting strategies on both prosumers. For example, relative benefits exhibit the same trends over varying forecasting cases (see Fig. 7) and horizons (see Fig. 9). Since prosumers with substantially different production and consumption levels, patterns and uncertainties are considered, this similarity points towards transferability of recommendations on forecasting strategies to other prosumers. Nevertheless, case studies with large and versatile prosumer portfolios covering different locations, weather conditions and component dimensions are required to substantiate the findings.

The main difference between the two is a lower benefit level across all scenarios for prosumer 1. This is explained by the prosumers' contrast with respect to predictability and optimization potential. While prosumer 2 exhibits strong repetitiveness in load patterns, EV charging introduces more randomness in the other case. Moreover, the active alignment of EV charging to PV production reduces the margin for further load optimization in contrast to the passive behavior of prosumer 2. Given that the two considered users represent rather extreme cases, it can be expected that the relative benefits of other prosumers in many cases will lie between those two.

Although results are based on prosumers located in Denmark, findings transfer to other regions with similar instantaneous netting schemes, for example, Belgium and parts of the United States (Nevada, Arizona and New York) [13]. Further states and confederations move towards instantaneous metering (e.g., Netherlands) or are promoting the roll-out of SMs (European Union), which provide the technical means for employing time-varying prices and short netting intervals.

6.3. Market readiness

The data [17], models [19] and optimization algorithms [15] used in C7 can already be acquired and used free of charge for private use. Moreover, simple hardware in the range of existing EMSs or small single-board computers is capable of hosting such applications. Therefore, forecasting-based optimization as in C7 can be considered market ready.

7. Conclusion and future work

In this work, trade-offs between economic profitability, computational complexity and security of forecasting-based optimization in residential EMSs are evaluated. Two PV-battery systems of real prosumers exhibiting different production and consumption characteristics serve as the foundation of the study. Several forecasting cases are considered, which result from variations of model type, data availability and modeling strategies. The resulting EMS scenarios and underlying forecasts are systematically quantified and assessed regarding forecasting accuracy, computational complexity and economic benefits, including sensitivity analyses on time of the year and length of the forecasting horizon. Moreover, two data manipulation scenarios are included to quantify possible attack impact and assess the EMS scenarios in terms of security. Results show that the theoretical maximum benefit over a 14-month period in the two prosumer cases is 466 € and 555 €, respectively, compared to a scenario without battery. Optimization based on naïve persistence forecasts achieves 78% (prosumer 1) and 86% (prosumer 2) of this upper limit. The relative benefits further raise to 90% and 93%, respectively, in scenarios considering GBDT-based forecasts. This performance increase already is achieved in a scenario which (1) can be applied to new systems with short data history and (2) can be implemented locally without need for extensive computing resources (e.g., cloud computing). The highlighted scenario does not depend on sophisticated price forecasts and is tolerant against manipulations of weather model inputs. However, due to sensitivity to price manipulations, incorporation of concepts for price data integrity checking into residential EMSs should be considered.

Future studies should evaluate the profitability, complexity and security trade-offs for other residential energy systems, such as electric vehicle- or heat-pump-based setups. Another aspect of interest is to understand if intra-hourly forecasts can provide further benefits despite the high randomness of load consumption. Finally, new error metrics for forecast model selection should be developed, which improve translation of forecasting accuracy to financial gains and thus might increase efficiency and benefits of hyperparameter selection.

CRediT authorship contribution statement

Nils Müller: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing – original draft, Visualization. **Mattia Marinelli:** Resources, Writing – review & editing, Supervision. **Kai Heussen:** Conceptualization, Writing – review & editing, Supervision, Funding acquisition. **Charalampos Ziras:** Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgment

This work is funded by the Innovation Fund Denmark (IFD) under File No. 91363, and the EU Horizon project EV4EU under grant agreement 101056765.

References

[1] H. Holttinen, J. Kiviluoma, D. Flynn, J.C. Smith, A. Orths, P.B. Eriksen, N. Cutululis, L. Söder, M. Korpås, A. Estanqueiro, J. MacDowell, A. Tuohy, T.K. Vrana, M. O'Malley, System impact studies for near 100% renewable energy systems dominated by inverter based variable generation, IEEE Trans. Power Syst. 37 (4) (2022) 3249–3258, http://dx.doi.org/10.1109/TPWRS.2020.3034924.

[2] J. Ostergaard, C. Ziras, H.W. Bindner, J. Kazempour, M. Marinelli, P. Markussen, S.H. Rosted, J.S. Christensen, Energy security through demand-side flexibility: The case of Denmark, IEEE Power Energy Mag. 19 (2) (2021) 46–55, http://dx.doi.org/10.1109/MPE.2020.3043615.

[3] L. Söder, P.D. Lund, H. Koduvere, T.F. Bolkesjø, G.H. Rossebø, E. Rosenlund-Soysal, K. Skytte, J. Katz, D. Blumberga, A review of demand side flexibility potential in northern europe, Renew. Sustain. Energy Rev. 91 (2018) 654–664, http://dx.doi.org/10.1016/j.rser.2018.03.104.

[4] X. Han, J. Garrison, G. Hug, Techno-economic analysis of PV-battery systems in Switzerland, Renew. Sustain. Energy Rev. 158 (2022) http://dx.doi.org/10.1016/j.rser.2021.112028.

[5] D. Azuatalam, K. Paridari, Y. Ma, M. Förstl, A.C. Chapman, G. Verbič, Energy management of small-scale PV-battery systems: A systematic review considering practical implementation, computational requirements, quality of input data and battery degradation, Renew. Sustain. Energy Rev. 112 (2019) 555–570, http://dx.doi.org/10.1016/j.rser.2019.06.007.

[6] D. Azuatalam, M. Förstl, K. Paridari, Y. Ma, A.C. Chapman, G. Verbič, Techno-economic analysis of residential PV-battery self-consumption, in: 2018 Asia-Pacific Solar Research Conference (APSRC), Vol. 186, 2018, pp. 171–178, https://scholar.google.com/scholar?hl=de&as_sdt=0%2C5&q=Techno-economic+Analysis+of+Residential+PV-battery+Self-consumption&btnG=.

[7] J. Salpakari, P. Lund, Optimal and rule-based control strategies for energy flexibility in buildings with PV, Appl. Energy 161 (2016) 425–436, http://dx.doi.org/10.1016/j.apenergy.2015.10.036.

[8] C. Sun, F. Sun, S.J. Moura, Nonlinear predictive energy management of residential buildings with photovoltaics & batteries, J. Power Sources 325 (2016) 723–731, http://dx.doi.org/10.1016/j.jpowsour.2016.06.076.

[9] Y. Zhang, R. Wang, T. Zhang, Y. Liu, B. Guo, Model predictive control-based operation management for a residential microgrid with considering forecast uncertainties and demand response strategies, IET Gener., Transm. Distribution 10 (10) (2016) 2367–2378, http://dx.doi.org/10.1049/iet-gtd.2015.1127.

[10] Y. Iwafune, T. Ikegami, J.G. da Silva Fonseca, T. Oozeki, K. Ogimoto, Cooperative home energy management using batteries for a photovoltaic system considering the diversity of households, Energy Convers. Manage. 96 (2015) 322–329, http://dx.doi.org/10.1016/j.enconman.2015.02.083.

[11] M. Elkazaz, M. Sumner, S. Pholboon, R. Davies, D. Thomas, Performance assessment of an energy management system for a home microgrid with PV generation, Energies 13 (13) (2020) http://dx.doi.org/10.3390/en13133436.

[12] D. van der Meer, G.C. Wang, J. Munkhammar, An alternative optimal strategy for stochastic model predictive control of a residential battery energy management system with solar photovoltaic, Appl. Energy 283 (2021) http://dx.doi.org/10.1016/j.apenergy.2020.116289.

[13] C. Ziras, L. Calearo, M. Marinelli, The effect of net metering methods on prosumer energy settlements, Sustain. Energy Grids Netw. 27 (2021) http://dx.doi.org/10.1016/j.segan.2021.100519.

[14] K. Shivam, J.-C. Tzou, S.-C. Wu, A multi-objective predictive energy management strategy for residential grid-connected PV-battery hybrid systems based on machine learning technique, Energy Convers. Manage. 237 (2021) http://dx.doi.org/10.1016/j.enconman.2021.114103.

[15] GNU project, 2023, https://www.gnu.org/software/glpk/, (Accessed: 2023-03-07).

[16] S. Diamond, S. Boyd, CVXPY: A python-embedded modeling language for convex optimization, 2022, https://www.cvxpy.org, (Accessed: 2022-12-15).

[17] Free rooftop solar forecasting, 2022, https://solcast.com/free-rooftop-solar-forecasting, (Accessed: 2022-11-24).

[18] Current weather and forecast - OpenWeatherMap, 2022, https://openweathermap.org/, (Accessed: 2022-11-24).

[19] J. Herzen, F. Lässig, S.G. Piazzetta, T. Neuer, L. Tafti, G. Raille, T. Van Pottelbergh, M. Pasieka, A. Skrodzki, N. Huguenin, M. Dumonal, J. Kościsz, D. Bader, F. Gusset, M. Benheddi, C. Williamson, M. Kosinski, M. Petrik, G. Grosch, Darts: User-friendly modern machine learning for time series, J. Mach. Learn. Res. 23 (124) (2022) 1–6.

[20] J.H. Friedman, Greedy function approximation: A gradient boosting machine, Ann. Statist. 29 (5) (2001) 1189–1232.

[21] C.S. Bojer, J.P. Meldgaard, Kaggle forecasting competitions: An overlooked learning opportunity, Int. J. Forecast. 37 (2) (2021) 587–603, http://dx.doi.org/10.1016/j.ijforecast.2020.07.007.

[22] T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, 2016, pp. 785–794, http://dx.doi.org/10.1145/2939672.2939785.

[23] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A highly efficient gradient boosting decision tree, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS '17, 2017, pp. 3149–3157, https://scholar.google.de/scholar?hl=de&as_sdt=0%2C5&q=LightGBM%3A+A+highly+efficient+gradient+boosting+decision+tree%2C+proceedings.neurips&btnG=.

[24] L. Prokhorenkova, G. Gusev, A. Vorobev, A.V. Dorogush, A. Gulin, CatBoost: unbiased boosting with categorical features, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 31, 2018.

[25] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, 2019, pp. 2623–2631, http://dx.doi.org/10.1145/3292500.3330701.

[26] R. Hyndman, G. Athanasopoulos, Forecasting: Principles and Practice, second ed., OTexts, Australia, 2018, https://scholar.google.com/scholar?hl=de&as_sdt=0%2C5&q=+Forecasting%3A+principles+and+practice+hyndman+2018&btnG=.

[27] LightGBM documentation, 2022, https://lightgbm.readthedocs.io/en/latest/Parameters.html, (Accessed: 2022-11-24).

[28] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, in: Advances in Neural Information Processing Systems, Vol. 24, 2011.

[29] Day-ahead prices, 2022, https://www.nordpoolgroup.com/en/Market-data1/Dayahead/, (Accessed: 2022-11-24).

[30] A. Jedrzejewski, J. Lago, G. Marcjasz, R. Weron, Electricity price forecasting: The dawn of machine learning, IEEE Power Energy Mag. 20 (3) (2022) 24–31, http://dx.doi.org/10.1109/MPE.2022.3150809.

[31] Carnot spot price forecast, 2022, https://www.carnot.dk/, (Accessed: 2022-11-24).

[32] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, DDoS in the IoT: Mirai and other botnets, Computer 50 (7) (2017) 80–84, http://dx.doi.org/10.1109/MC.2017.201.

[33] B. Bostami, M. Ahmed, S. Choudhury, False data injection attacks in internet of things, in: Performability in Internet of Things, Springer International Publishing, 2019, pp. 47–58.

[34] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, third ed., The MIT Press, 2009, https://scholar.google.com/scholar?hl=de&as_sdt=0%2C5&q=Introduction+to+algorithms+2022&btnG=.

[35] H.M. Sani, C. Lei, D. Neagu, Computational complexity analysis of decision tree algorithms, in: International Conference on Innovative Techniques and Applications of Artificial Intelligence, Springer, 2018, pp. 191–197.

[36] Time complexities of machine learning algorithms, 2022, https://7-hiddenlayers.com/time-complexities-of-ml-algorithms, (Accessed: 2022-11-24).

[37] DTU Computing Center, DTU Computing Center resources, 2022, http://dx.doi.org/10.48714/DTU.HPC.0001.

PAPER C

Nils Müller, Charalampos Ziras, Kai Heussen

Assessment of Cyber-Physical Intrusion Detection and Classification for Industrial Control Systems

Müller, N., C. Ziras, and K. Heussen, "Assessment of Cyber-Physical Intrusion Detection and Classification for Industrial Control Systems," in *Proceedings of the 2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Singapore, 2022, pp. 432-438, doi: 10.1109/SmartGridComm52983.2022.9961010.

Assessment of Cyber-Physical Intrusion Detection and Classification for Industrial Control Systems

Nils Müller, Charalampos Ziras, Kai Heussen
Wind and Energy Systems Department
Technical University of Denmark
Lyngby, Denmark
{nilmu; chazi; kh}@dtu.dk

***Abstract*—The increasing interaction of industrial control systems (ICSs) with public networks and digital devices introduces new cyber threats to power systems and other critical infrastructure. Recent cyber-physical attacks such as Stuxnet and Irongate revealed unexpected ICS vulnerabilities and a need for improved security measures. Intrusion detection systems constitute a key security technology, which typically monitors cyber network data for detecting malicious activities. However, a central characteristic of modern ICSs is the increasing interdependency of physical and cyber network processes. Thus, the integration of network and physical process data is seen as a promising approach to improve predictability in real-time intrusion detection for ICSs by accounting for physical constraints and underlying process patterns. This work systematically assesses machine learning-based cyber-physical intrusion detection and multi-class classification through a comparison to its purely network data-based counterpart and evaluation of misclassifications and detection delay. Multiple supervised detection and classification pipelines are applied on a recent cyber-physical dataset, which describes various cyber attacks and physical faults on a generic ICS. A key finding is that the integration of physical process data improves detection and classification of all considered attack types. In addition, it enables simultaneous processing of attacks and faults, paving the way for holistic cross-domain root cause identification.**

***Index Terms*—cyber-physical, intrusion detection, industrial control systems, machine learning, power systems**

I. Introduction

In recent years, industrial control systems (ICSs) face an ongoing opening to the internet [1]. Previously isolated systems relying on private networks and specifically designed protocols increasingly use public networks and digital devices to achieve several business benefits. However, along with advantages such as cost-efficiency and process flexibility, new cyber vulnerabilities emerge, as evidenced by a growing number of cyber attacks on ICSs [2]. Besides the introduction of new cyber threats, the ongoing integration of cyber and physical components transforms modern ICSs to complex cyber-physical systems. In such cyber-physical ICSs, failures can originate from a variety of hard- or software faults, human errors or malicious activities. Distinguishing attacks from physical faults or human errors is a particularly challenging task in such highly integrated and complex systems, which complicates the identification of root causes.

This work is funded by Innovation Fund Denmark (File No. 91363) and Horizon 2020 project INTERPRETER under grant agreement No. 864360.

Intrusion detection systems (IDSs) are responsible for detecting malicious activities by monitoring and analyzing either ICSs end-device (host-based IDS) or network data (network-based IDS). In recent years, the use of machine learning (ML) for IDSs has attracted increasing interest for reasons such as the ability to capture complex properties of ICS operation and attacks, lower central processing unit (CPU) load compared to conventional IDSs, higher detection speed, reduced need for expert knowledge due to generalizability, and the exploitation of steadily increasing amounts of data in ICSs [3]. In this context, supervised multi-class classification comes with the advantage of enabling automated distinction between different types of attacks and other anomalies such as physical faults, facilitating the identification of root causes.

A promising approach to improve supervised intrusion detection and classification is the integration of cyber network with physical process data [4]. In this way, the underlying physical constraints and patterns of an ICS are included into the IDS, potentially improving predictability. A few studies investigate such supervised cyber-physical detection and classification of cyber attacks. In [5] the authors propose a multi-layer cyber attack detection system, which combines a supervised and exclusively network data-based classification step with an empirical model for detecting abnormal operation in physical process data. However, the authors consider a binary classification problem (normal vs. attack) which weakens the determination of causational factors. Moreover, as the dataset was shuffled and randomly divided into training and test data, samples of a specific attack event can be found in both sets, which entails data leakage, weakening the validity of the results. In [6]–[8] ML-based intrusion detection and classification in ICSs is investigated considering multiple attack types as well as cyber network and physical process features. However, none of these works compares cyber-physical with network intrusion detection, or evaluates misclassifications and detection delay. In [9]–[11] the benefit of integrating physical process data into a supervised classification-based IDS is demonstrated for robotic vehicles. These works neither consider a multi-class classification problem nor a scenario including both cyber attacks and physical faults. The thematically and methodologically closest works are [12] and [13]. In [12] the authors introduce the dataset which also forms foundation for the present

work. To demonstrate a use case of the dataset, they compare several supervised classifiers for intrusion detection. Although the work compares the use of cyber network and physical process data, the integration of both information sources is not considered. Moreover, only a binary classification problem is examined. Finally, models are evaluated using K-fold cross-validation (CV), which leads to data leakage as samples from the same attack or fault event are placed in the training and test datasets. In [13] the authors compare several unsupervised and supervised models for cyber-physical intrusion detection in power systems. The study explicitly compares network to cyber-physical intrusion detection. However, in total only four man-in-the-middle (MITM) attack events are considered and investigated individually, again leading to data leakage.

The review reveals that existing works on supervised cyber-physical intrusion detection and classification either i) lack systematic assessment through comparison to a purely network data-based approach and evaluation of misclassifications and detection delay, or ii) do not consider multi-class classification, weakening root cause identification, or iii) suffer from methodological issues, limiting the validity of results.

This work systematically assesses real-time cyber-physical intrusion detection and multi-class classification based on a comparison to its exclusively network data-based counterpart and evaluation of misclassifications and detection delay. Various supervised detection and classification pipelines are implemented and evaluated on a recent dataset of a generic ICS [12], describing several cyber attack and physical fault types based on physical process and cyber network data.

A. Contribution and paper structure

The main contributions of this work are as follows:

- Systematic comparison of ML-based network and cyber-physical intrusion detection and multi-class classification for ICSs, evaluating multiple classification pipelines.
- Attack and fault class-wise analysis of misclassifications and detection delay for cyber-physical intrusion detection.
- Proposal and assessment of prediction filtering for reduction of misclassifications.
- Transferability evaluation of the investigated generic ICS to control systems in the power sector.

The remainder of the paper is structured as follows. In Section II the investigated dataset is introduced, and the transferability of the underlying ICS to power sector control systems evaluated. Section III provides a description of the data preparation as well as applied models and techniques. In Section IV results are presented and discussed, followed by a conclusion and a view on future work in Section V.

II. Dataset description and transferability

This section first describes the dataset under investigation (Subsection II-A). Thereafter, Subsection II-B sheds light on the transferability of this work's results by comparing the investigated generic ICS to control systems in the power sector.

A. Dataset

The dataset used in this study was acquired from a hardware-in-the-loop water distribution testbed and is introduced in [12]. The system distributes water across eight tanks, where one process cycle is defined by a full filling/emptying process of each tank. This procedure is steadily repeated, rendering it a cyclical process. Water flow between the tanks is realized by valves, pumps, pressure sensors and flow sensors. The process is controlled by a typical supervisory control and data acquisition (SCADA) architecture consisting of multiple sensors and actuators (field instrumentation control layer), four programmable logic controllers (PLCs) (process control layer), and a SCADA workstation, including an human-machine interface (HMI) and data historian (supervisory control layer). Communication is conducted via the MODBUS TCP/IP protocol. An additional Kali Linux machine is included for launching cyber attacks. The process consists of four stages, each of which is controlled by one of the four PLCs.

The dataset describes the normal operation of the system as well as several types of cyber attacks and physical faults against different components and communication links. Among the cyber attacks are eight different MITM attacks, five denial-of-service (DoS), and seven scanning attacks. Moreover, three different water leaks and six sensor and pump breakdowns are included as physical events, which partly appear simultaneously and are interchangeably called attacks or faults by the authors. In the present work, they will be referred to as physical faults. The dataset consists of two sub-datasets, namely a cyber network and physical process dataset. While the physical dataset has a constant one-second resolution, the network dataset on average has 2633 observations per second. The raw features are listed in Table I. For a more detailed explanation of the dataset the reader is referred to [12].

TABLE I
RAW CYBER NETWORK AND PHYSICAL PROCESS FEATURES.

No.	Physical features	No.	Network features
1	Timestamp	1	Timestamp
2-9	Pressure sensor value of tank 1-8	2-3	IP address (src. & dst.)[a]
		4-5	MAC address (src. & dst.)
10-15	State of pump 1-6	6-7	Port (src. & dst.)
16-19	Flow sensor value of flow sensor 1-4	8	Protocol
		9	TCP flags
20-41	State of valve 1-22	10	Payload size
		11	MODBUS function code
		12	MODBUS response value
		13-14	No. of packets[b] (src. & dst.)

[a] Src. and dst. refer to source and destination, respectively.
[b] Refers to packets of the same device during the last two seconds.

B. Transferability to control systems in the power sector

The considered ICS constitutes a generic test bed with characteristics potentially differing from its real-world counterparts. To shed light on the transferability of the results of this study, a contextualization of the investigated ICS is required. Prominent representatives of real-world ICSs are control systems in the power sector, such as distribution system

operator's SCADA or substation automation systems (SASs), upon which the following comparison is based. An overview is given in Table II.

An important commonality is the SCADA architecture and its components. Thus, the investigated attack scenarios, targeting various SCADA components, provide realistic scenarios for both system types. Another similarity is the well-defined and steady configuration of the physical process and cyber network. Moreover, both systems show continuous and repetitive patterns of the physical process either due to process cycles (generic ICS) or seasonality (power systems). Such deterministic configurations and patterns allow to model both systems with a set of physical and network features.

Differences are mainly related to the physical process. In contrast to the investigated ICS, power systems are subject to external influences such as weather and customer behavior, increasing process volatility. Moreover, compared to a distribution system operator's SCADA system, the number of physical and network components is relatively small. Thus, a central difference is system complexity due to higher volatility and number of components in power systems.

To conclude, the present generic ICS can be considered a valuable test case for smaller control systems in the power sector such as SASs. However, due to a lack of system complexity, investigation of intrusion detection for larger SCADA systems will require more extensive test beds.

TABLE II
COMPARISON OF THE INVESTIGATED ICS TO CONTROL SYSTEMS IN THE POWER SECTOR.

No.	Similarities	No.	Differences
1	SCADA architecture & components	1	Volatility & trend pattern
2	Attack types & target components	2	Physical fault types
3	Steady network & process configuration (e.g., IP addresses, protocols & number of physical devices)	3	External impacts (e.g., weather & customer behavior)
4	Continuous, repetitive and thus deterministic network & process patterns	4	Type of communication protocol
5	Continuous, discrete & categorical features	5	Number of physical & network components

III. METHODOLOGY

This section first describes the preparation of the dataset. Thereafter, the data pipelines applied to the intrusion detection and classification problem are introduced.

A. Data preparation

1) Data partitioning: A good practice to test performance and generalization of a fully specified ML model is the application to a holdout test dataset which stems from the same target distribution as the training set but was not previously seen [14]. The present dataset describes attack or fault events in time series format, where a specific event corresponds to a sequence of observations. As a specific event can only occur once, its entire sequence must either be placed in the training or test dataset. Placing observations from a single event in both the training and test set will assume information from future events during model training. Thus, data shuffling or CV-based performance evaluation will result in an overly optimistic model performance assessment.

In this work, the time series format of the investigated dataset is considered to perform a fair evaluation of the model performance. The entire sequences of the last two events of each attack or fault[1] class are reserved for the test dataset and not considered during model training. As a result, the training set consists of the first $80\,\%$ of all normal operation observations, $73.07\,\%$ of the DoS observations, $80.91\,\%$ of the MITM observations, $77.75\,\%$ of the physical fault observations, and $71.42\,\%$ of the scanning observations. In this way, the risk for an overly optimistic performance assessment through data leakage is minimized, while a typical $75/25$ ratio between the training and test set can be maintained. Thus, future works examining the present dataset are encouraged to use the same partitioning in order to improve validity and comparison of results.

2) Feature extraction and fusion: Cyber-physical intrusion detection and classification simultaneously processes physical process and network traffic data. As these originate from different domains, they usually exhibit unequal characteristics such as observation rates and noise levels. Thus, feature extraction from raw data typically requires different approaches for these two data sources.

In this work, the extraction of features from raw network traffic is realized by several sample statistics. An overview of the considered statistics and resulting set of network features is given in Table III. The selected statistics evaluate the traffic for each second. The objective is to retain existing and extract additional information compared to consideration of individual data packets, while network and physical process features are aligned and the number of model executions reduced. As discussed in Subsection II-B, ICSs usually exhibit well-defined and static network configurations, which include, for example, fixed sets of IP addresses or ports. This characteristic can be exploited by statistics which indicate occurrence of instances or instance combinations not present during normal operation. Related features include MAC/IP address mismatch occurrence (Feature no. 2-3) and abnormal instance occurrence (Feature no. 4-14). The set of normal instances of a raw network feature is extracted from the normal operation observations of the training dataset (see Subsection III-A1). To retain the information detail of individual data packets, an abnormal instance (combination) occurrence is already indicated if a single packet is affected during the considered second. Additional information is extracted by contextualizing packets within each second based on several counts and mean values (see Table III). If, for example, only normal packets are received during a DoS attack, but at an unusual rate, it can only be detected based on the additional information provided by the context of multiple packets.

[1]Since there are only three water leak events in the dataset, some of which also occur simultaneously with sensor or pump breakdowns, all events of physical faults are combined into a *physical fault* event class.

TABLE III
EXTRACTED NETWORK TRAFFIC AND PHYSICAL PROCESS FEATURES.

Feature no.	Extracted network features	Description	Underlying raw features[a]
1	Number of data transfers	Count of data packets transferred during the last second.	Raw network traffic data index
2-3	MAC/IP mismatch occurrence	Mismatch indication between the IP and MAC address of at least one network device within the last second.	IP and MAC address
4-14	Abnormal instance occurrence	Indication of an abnormal instance occurrence within the respective raw network feature during the last second.	All raw network features except timestamp and number of packets
15-25	Number of abnormal instance occurrences	Count of occurrences of abnormal instances within a specific raw network feature during the last second.	All raw network features except timestamp and number of packets
26-36	Number of normal instance occurrences	Count of occurrences of normal instances within a specific raw network feature during the last second.	All raw network features except timestamp and number of packets
37-106	Number of occurrences for each instance	Individual occurrences count for all instances of the respective raw network feature during the last second.	All raw network features except timestamp, port and number of packets
107-117	Number of different instances	Count of distinct instances of a specific raw network feature within the last second.	All raw network features except timestamp and number of packets
118-128	Number of NaN occurrences	Count of NaN occurrences within a raw network feature during the last second.	All raw network features except timestamp and number of packets
129-131	Mean value	Mean value of the respective raw network feature during the last second.	Payload size and number of packets
132-177	Number of different class-specific instances[b]	Count of distinct event class-specific instances of the respective raw network feature during the last second.	All raw network features except timestamp, IP address and MAC address
Feature no.	**Extracted physical features**	**Description**	**Underlying raw features**
178-185	Pressure value of tank 1-8	Raw pressure value of the respective tank.	Pressure sensor value of tank 1-8
186-191	State of pump 1-6	Raw state of the respective pump.	State of pump 1-6
192-195	Value of flow sensor 1-4	Raw value of the respective flow sensor.	Value of flow sensor 1-4
196-217	State of valve 1-22	Raw state of the respective valve.	State of valve 1-22
218	Normal progress of a process cycle	Normal state of the current process cycle, defined on the range between zero and one.	Pressure value of tank 1
219	Sine transformed normal progress of a process cycle	Sine transformation of the normal progress of a process cycle.	Pressure value of tank 1
220	Cosine transformed normal progress of a process cycle	Cosine transformation of the normal progress of a process cycle.	Pressure value of tank 1

[a]Source and destination considered in case of IP address, MAC address, port and number of packets.
[b]Event class-specific instances are defined only based on the training dataset.

As discussed in Subsection II-B, the volatility, and hence noise level, of raw physical process features is comparatively small in the present case. For that reason, no processing, such as data smoothing, is required and raw features can directly be used (see Table III). In addition, the normal progress of a process cycle is extracted with the associated feature vector $P = \{p_1, p_2, ..., p_N \mid p_i \in \mathbb{R} \forall i\}$ of length N. Values of P are defined on the range $p_i \in [0, d]$, where d corresponds to the usual duration of a process cycle, which is derived from pressure sensor values of tank 1 within the training dataset. While P can represent an expected progress of, for example, 10 percentage points between $p = 0.85d$ and $p = 0.95d$, the same progress from $p = 0.95d$ to $p = 0.05d$ in the next process cycle is not described properly due to the jump discontinuity. To eliminate the discontinuity, and hence account for the cyclical nature of the process, the additional feature vectors P_{sin} and P_{cos} are extracted by applying sine and cosine transformation [15] on P according to

$$p_{\text{sin},i} = \sin\left(\frac{2\pi p_i}{d}\right), \text{ and } p_{\text{cos},i} = \cos\left(\frac{2\pi p_i}{d}\right), \quad (1)$$

$\forall i \in [1, N]$. Note that cosine transformation is required as the sine function alone is not bijective, which would lead to ambiguity in the process cycle progress.

In total, 220 features are extracted (see Table III), some of which exhibit constant values and thus are non-informative. After removing the non-informative features, the final dataset comprises 161 features and 9185 observations.

B. Intrusion detection and classification data pipelines

Supervised detection and classification of attacks and faults constitutes a highly imbalanced multi-class classification problem. To improve the classification performance, several up- and downstream data transformation steps are considered. Typical steps comprise scaling, dimensionality reduction, undersampling and oversampling [16]. Together with classification and prediction filtering, these define the intrusion detection and classification data pipeline considered in this work (see Fig. 1). Note that prediction filtering stems from result evaluation in Section IV and is not considered during model selection. As depicted in Fig. 1, the data pipeline maps observations of a cyber-physical dataset to the considered event classes. For each of the transformation steps, several candidate methods are considered, which represent the most widely used techniques. Scaling candidates include feature standardization by removing the mean and scaling to unit variance, normalization to values between zero and one, and scaling to the maximum absolute value. Principal component analysis (PCA) with Bayesian selection of the number of principle components is considered for dimensionality reduction [17]. Methods for undersampling include instance hardness threshold (IHT) and removal of Tomek Links, while synthetic

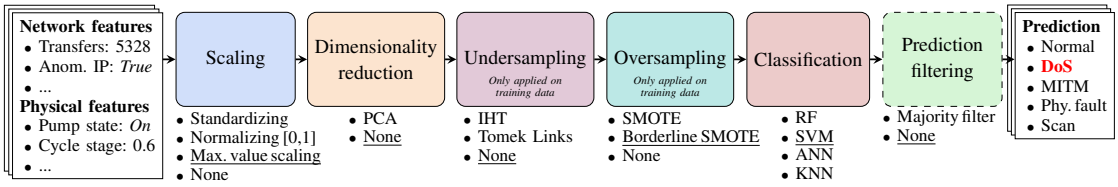


Fig. 1. Data pipeline for cyber-physical intrusion detection and multi-class classification, which maps observations of a dataset (left side) to event classes (right side). Underscores indicate an exemplary selection of methods for all data transformation steps.

minority oversampling technique (SMOTE) and Borderline SMOTE constitute the oversampling methods. Note that undersampling and oversampling is only applied on training data. The classification models include a random forest (RF), k-nearest neighbors (KNN), a support vector machine (SVM) and an artificial neural network (ANN). Prediction filtering is realized by a moving majority filter which outputs the most frequent label of the past six predictions. As scanning attacks do not necessarily appear in sequences, they are excluded from the filtering process. All data transformation steps, except for classification, can also be bypassed. For some classifiers specific transformation steps, such as bypassing scaling for SVM, are excluded due to numerical issues.

The pipeline and most of the embedded data transformation methods and classification models are implemented in Python using the *scikit-learn* library [18]. An exception is the ANN which is implemented using the deep learning library *Keras* [19]. Due to the multitude of models and techniques considered in this work, theoretical descriptions are omitted for brevity. Thus, for detailed backgrounds, the reader is referred to the respective library documentation as well as [14] and [16]. The selection of transformation methods and hyperparameters is conducted based on the training set (see Section III-A1) applying a shuffled and stratified 5-fold CV grid search. Although shuffling introduces data leakage during model selection, it is required to ensure observations of each attack type in all folds. While this may result in selection of non-optimal hyperparameters, the evaluation of the selected models is unaffected. The method and hyperparameter selection is summarized in Table IV, where selected pipelines are referred to as the respective classifier. For hyperparameters not defined in Table IV, the library's default values are used. Before being applied to test data, the selected detection and classification pipelines are retrained on the full training set.

TABLE IV
METHOD AND HYPERPARAMETER SELECTION RESULTS.

	Cyber-physical intrusion detection & classification pipelines			
Step	**RF**	**KNN**	**SVM**	**ANN**
Scaling	Standard.	Standard.	Max. val. sc.	Max. val. sc.
Dim. red.	None	PCA	None	PCA
Unders.	None	None	None	None
Overs.	SMOTE	None	Bor. SMOTE	Bor. SMOTE
Classif.	$n_{\text{estimators}}$: 100, $n_{\text{max-features}}$: 17	$n_{\text{neighbors}}$: 5, *Dist. func.*: manhattan, *Weight func.*: distance	*Kernel*: radial-basis function, *Penalty para.*: 10000, *Kernel coeff.*: 0.0175	$n_{\text{hid.-layers}}$: 2, n_{units}: 150, *Act. func.*: Relu, *Dropout rate*: 0.5, n_{epochs}: 500, *Batch size*: 512
	Network intrusion detection & classification pipelines			
Step	**RF**	**KNN**	**SVM**	**ANN**
Scaling	Max. val. sc.	Standard.	Max. val. sc.	Max. val. sc.
Dim. red.	None	PCA	None	PCA
Unders.	IHT	Tomek Links	None	None
Overs.	None	None	None	Bor. SMOTE
Classif.	$n_{\text{estimators}}$: 100, $n_{\text{max-features}}$: 17	$n_{\text{neighbors}}$: 5, *Dist. func.*: manhattan, *Weight func.*: uniform	*Kernel*: radial-basis function, *Penalty para.*: 10000, *Kernel coeff.*: 0.0175	$n_{\text{hid.-layers}}$: 2, n_{units}: 100, *Act. func.*: Relu, *Dropout rate*: 0.5, n_{epochs}: 500, *Batch size*: 256

IV. PERFORMANCE EVALUATION

This section assesses the performance of cyber-physical intrusion detection and multi-class classification. Subsection IV-A introduces the applied performance metrics. In Subsection IV-B a comparison to network intrusion detection and classification is conducted, while Subsection IV-C evaluates detection delay and misclassifications.

A. Metrics

To evaluate the class-wise detection and classification performance, this study considers the F_1 score according to

$$F_{1,i} = \frac{TP_i}{TP_i + \frac{1}{2}(FP_i + FN_i)}, \tag{2}$$

where TP_i, FP_i and FN_i are the number of true positives, false positives and false negatives of the i-th class, respectively. The overall performance is assessed based on a macro average of the class-wise F_1 scores, given as

$$F_1^m = \frac{\sum_{i=1}^{N_{\text{classes}}} F_{1,i}}{N_{\text{classes}}}, \tag{3}$$

with N_{classes} being the number of classes. As seen from (3), the macro average F_1^m treats all classes evenly, which is important given the high cost of missing observations of the less-populated attack classes. Average detection delay of class i and average detection delay over all classes are given by

$$\tau_i = \frac{\sum_{j=1}^{N_{\text{events},i}} (t_{\text{det},j,i} - t_{\text{start},j,i})}{N_{\text{events},i}}, \text{ and } \tau = \frac{\sum_{i=1}^{N_{\text{classes}}} \tau_i}{N_{\text{classes}}}, \tag{4}$$

where $N_{\text{events},i}$ is the number of events, $t_{\text{start},j,i}$ the start time of the j-th event and $t_{\text{det},j,i}$ the first-detection time of the j-th event of the i-th class, respectively.

B. Comparison of network and cyber-physical intrusion detection and multi-class classification

In Table V the F_1 scores of all detection and classification pipelines are listed. The highest scores are in bold, while the second best scores are underlined. For network intrusion detection, all models show a similar overall performance, despite the differences on a class level. As expected, physical faults are barely detected with pure network data. However, most models detect some physical fault observations, especially the ANN. It can be concluded that network data provide some information about the physical process, which for instance could result from altered payload sizes or higher NaN occurrences.

The use of the cyber-physical feature set improves class-wise and overall performance for all models, with detection of normal observations by the RF being the only exception. This allows the conclusion that incorporating physical process data has the potential to improve supervised intrusion detection and classification in ICSs. Interestingly, also the classification of scanning attacks improves, although these do not affect the physical process. In fact, the absence of physical impact can be informative in the case where observations of *different* attack types exhibit *similar* impact on network traffic. If only one of the attack types also impacts the physical process, classification of both will be improved by incorporating physical data, due to less confusion between these attacks. As a result, physical process features also improve detection and classification of purely network traffic-affecting attacks.

The number of scanning attack observations is small compared to the other attack types. Nevertheless, most of the studied cyber-physical pipelines perfectly detect and classify scanning attacks, despite the very few training examples. It can be inferred that the extracted features in Table III very well capture the distinctive characteristics of scanning attacks.

Although ANNs can capture highly non-linear and complex relationships, they achieve only second best performance. An explanation may be the insufficient number of observations. However, as training data of cyber attacks usually is scarce, ANNs might not be appropriate for the given problem. The highest overall performance is achieved by the SVM, due to the superior exploitation of physical process information. Moreover, SVMs are known to be accurate also in high dimensional spaces, which may be an advantage given the relatively large feature-to-observation ratio. The good performance of the SVM under existence of physical faults demonstrates that integration of physical process data also allows simultaneous detection and classification of events of fundamentally different nature, paving the way for holistic cross-domain root cause analysis. Nevertheless, the comparatively low F_1 score for physical faults requires further improvements such as comprehensive physical feature extraction.

While the overall performance (F_1^m) on average improves by 15.5 percentage points, the models show very different class-specific improvements. Although detection of physical faults clearly improves for most models, KNN sets an exception. Moreover, only the SVM and ANN show strong improvements for MITM detection, while the RF shows a comparatively good improvement for DoS attacks. This complementarity suggests further investigation of ensemble modeling, e.g., combination of a SVM and ANN.

TABLE V
CLASS-WISE AND AVERAGE F_1 SCORES FOR NETWORK AND CYBER-PHYSICAL INTRUSION DETECTION AND CLASSIFICATION.

	Network features				Cyber-physical features			
Event class	**RF**	**KNN**	**SVM**	**ANN**	**RF**	**KNN**	**SVM**	**ANN**
Normal	0.92	0.93	0.86	0.88	0.91	0.94	**0.95**	0.93
DoS	0.55	0.47	0.96	0.47	0.71	0.49	**1.00**	0.50
MITM	0.87	0.83	0.42	0.68	0.87	0.88	0.81	**0.92**
Phy. fault	0.07	0.04	0.00	0.14	0.26	0.06	**0.62**	0.46
Scanning	0.57	0.67	0.80	0.80	**1.00**	0.80	**1.00**	**1.00**
Avg. (F_1^m)	0.60	0.59	0.61	0.60	0.75	0.63	**0.88**	0.76

C. Misclassification and detection delay evaluation of cyber-physical intrusion detection and multi-class classification

Misclassifications and detection delay are investigated on the SVM considering cyber-physical features due to superior performance. The confusion matrix in Fig. 2 reveals that the SVM mainly confuses physical faults and MITM attacks with normal operation and vice versa. Moreover, it shows almost no confusion among the attack and fault classes.

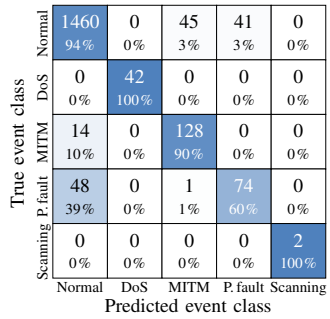


Fig. 2. Confusion matrix of the SVM for the cyber-physical feature set.

To further evaluate the location of misclassifications, the true and predicted labels of an excerpt of the test dataset are depicted in Fig. 3(a) in time series format. It can be noticed that misclassifications do not primarily appear during transition between event classes and are rather distributed. However, some increased emergence can be noticed at the transition from physical fault to normal operation (15:55:56) and at the beginning and end of the MITM attack between 16:08:26 and 16:10:01. High misclassification densities exist around 16:00:55 and 16:06:20 during normal operation, which might result from unlabeled irregular process behavior or noise. Finally, most misclassifications occur individually and not in sequences. This finding suggests filtering of the classification output (prediction filtering), as indicated in Fig. 1. Fig. 3(b) depicts the predictions after applying the majority filter described in Subsection III-B. The implemented filter is

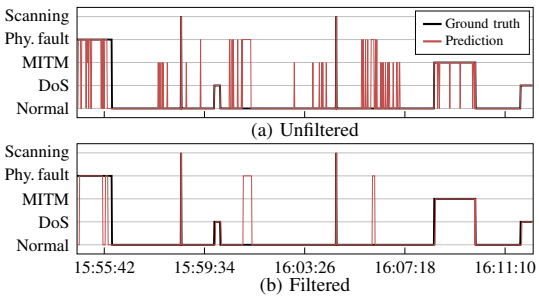


Fig. 3. Unfiltered (a) and filtered (b) predicted and true test dataset labels.

simple and not the result of a purely training data-based model selection process. Thus, results are of preliminary nature and further investigation is required. From a comparison of Fig. 3(a) and (b), it is noticeable that individual misclassifications are filtered out and only sequences remain, which greatly reduces the number of false positives. A quantitative assessment of the additional prediction filtering step in terms of F_1 score and detection delay for the test set excerpt of Fig. 3 is given in Table VI. Prediction filtering improves overall detection performance (F_1^m) by 3 percentage points. While detection of physical faults and MITM attacks greatly improves, the performance of DoS detection decreases. This can be explained by the prediction filtering-induced false negatives and positives at the beginning and end of the initially perfectly classified DoS events. From Table VI it can also be seen that the unfiltered SVM immediately detects all event classes except for physical faults. Prediction filtering increases detection delay, since several seconds of an event need to elapse to reach majority within the filter sequence.

TABLE VI
COMPARISON OF THE UNFILTERED AND FILTERED SVM BASED ON F1 SCORE AND DETECTION DELAY FOR THE TIME SERIES DEPICTED IN FIG. 3.

Model	Metric	Normal	DoS	MITM	P. fault	Scan	Average
SVM unfilt.	F_1 [-]	0.94	1.00	0.85	0.62	1.00	0.88
	τ_i [s]	–	0.00	0.00	3.00	0.00	0.75
SVM filtered	F_1 [-]	0.96	0.90	0.98	0.72	1.00	0.91
	τ_i [s]	–	3.00	3.00	6.00	0.00	3.00

V. CONCLUSION AND FUTURE WORK

This work assesses ML-based cyber-physical intrusion detection and multi-class classification for ICSs. For that purpose, a systematic comparison to a purely network data-based approach is conducted, followed by an evaluation of misclassifications and detection delay. An average F_1^m improvement of 15 percentage points across several supervised classification pipelines demonstrates the benefit of incorporating physical process data into intrusion detection and classification. Moreover, simultaneous processing of cyber attacks and physical faults is demonstrated, which paves the way to holistic cross-domain root cause analysis. Based on the evaluation of misclassifications, filtering of the classifier output (prediction filtering) is proposed to reduce false positives, which, however, comes at the cost of higher detection delays.

A remaining problem of cyber-physical intrusion detection and classification is the dependency on usually scarce attack samples. A potential solution is seen in applying attack sample-independent unsupervised methods. The often weaker performance of such methods in distinguishing attack types may be counteracted by considering cyber-physical input data.

REFERENCES

[1] M. R. Asghar, Q. Hu, and S. Zeadally, "Cybersecurity in industrial control systems: Issues, technologies, and challenges," *Computer Networks*, vol. 165, 2019.

[2] ThoughtLab, "Cybersecurity solutions for a riskier world." https://thoughtlabgroup.com/wp-content/uploads/2022/05/Cybersecurity-Solutions-for-a-Riskier-World-eBook_FINAL-2-1.pdf, 2022. [Accessed: 01/09/2022].

[3] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, "A detailed investigation and analysis of using machine learning techniques for intrusion detection," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 686–728, 2019.

[4] A. Ayodeji, Y.-k. Liu, N. Chao, and L.-q. Yang, "A new perspective towards the development of robust data-driven intrusion detection for industrial control systems," *Nuclear Engineering and Technology*, vol. 52, no. 12, pp. 2687–2698, 2020.

[5] F. Zhang, H. A. D. E. Kodituwakku, J. W. Hines, and J. Coble, "Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4362–4369, 2019.

[6] J. Yeckle and S. Abdelwahed, "An evaluation of selection method in the classification of scada datasets based on the characteristics of the data and priority of performance," in *Proceedings of the International Conference on Compute and Data Analysis*, pp. 98–103, 2017.

[7] M. Keshk, N. Moustafa, E. Sitnikova, and G. Creech, "Privacy preservation intrusion detection technique for scada systems," in *Military Communications and Information Systems Conference*, IEEE, 2017.

[8] R. C. B. Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, and S. Pan, "Machine learning for power system disturbance and cyber-attack discrimination," in *2014 7th International symposium on resilient control systems (ISRCS)*, pp. 1–8, IEEE, 2014.

[9] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon, and D. Gan, "Cloud-based cyber-physical intrusion detection for vehicles using deep learning," *Ieee Access*, vol. 6, pp. 3491–3508, 2017.

[10] T. P. Vuong, G. Loukas, and D. Gan, "Performance evaluation of cyber-physical intrusion detection on a robotic vehicle," in *2015 IEEE International Conference on Computer and Information Technology*, pp. 2106–2113, IEEE, 2015.

[11] T. P. Vuong, G. Loukas, D. Gan, and A. Bezemskij, "Decision tree-based detection of denial of service and command injection attacks on robotic vehicles," in *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, IEEE, 2015.

[12] L. Faramondi, F. Flammini, S. Guarino, and R. Setola, "A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing," *IEEE Access*, vol. 9, 2021.

[13] A. Sahu, Z. Mao, P. Wlazlo, H. Huang, K. Davis, A. Goulart, and S. Zonouz, "Multi-source multi-domain data fusion for cyberattack detection in power systems," *IEEE Access*, vol. 9, 2021.

[14] J. Friedman, T. Hastie, R. Tibshirani, *et al.*, *The elements of statistical learning*, vol. 1. Springer series in statistics New York, 2009.

[15] D. Chakraborty and H. Elzarka, "Advanced machine learning techniques for building performance simulation: a comparative analysis," *Journal of Building Performance Simulation*, vol. 12, no. 2, pp. 193–207, 2019.

[16] A. Fernndez, S. Garca, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from Imbalanced Data Sets*. Springer Publishing Company, Incorporated, 1st ed., 2018.

[17] T. Minka, "Automatic choice of dimensionality for pca," *Advances in neural information processing systems*, vol. 13, 2000.

[18] F. Pedregosa et al., "Scikit-learn." https://scikit-learn.org/, 2011. [Accessed: 25/01/2022].

[19] F. Chollet, "Keras." https://keras.io, 2015. [Accessed: 25/01/2022].

PAPER D

Nils Müller, Kaibin Bao, Jörg Matthes, Kai Heussen

CyPhERS: A Cyber-Physical Event Reasoning System providing real-time situational awareness for attack and fault response

Müller, N., K. Bao, J. Matthes, and K. Heussen, "CyPhERS: A Cyber-Physical Event Reasoning System providing real-time situational awareness for attack and fault response," in *Computers in Industry*, vol. 151, 2023, doi: 10.1016/j.compind.2023.103982.

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.sciencedirect.com/journal/computers-in-industry

CyPhERS: A cyber-physical event reasoning system providing real-time situational awareness for attack and fault response

Nils Müller [a,*], Kaibin Bao [b], Jörg Matthes [b], Kai Heussen [a]

[a] Wind and Energy Systems Department, Technical University of Denmark, Building 330, Risø campus, 4000 Roskilde, Denmark
[b] Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, Building 445, Campus North, 76344, Eggenstein-Leopoldshafen, Germany

ARTICLE INFO

Dataset link: https://ieee-dataport.org/open-access/hardware-loop-water-distribution-testbed-wdt-dataset-cyber-physical-security-testing

Keywords:
Cyber-physical systems
Situational awareness
Event identification
Machine learning
Cyber security

ABSTRACT

Cyber–physical systems (CPSs) constitute the backbone of critical infrastructures such as power grids or water distribution networks. Operating failures in these systems can cause serious risks for society. To avoid or minimize downtime, operators require real-time awareness about critical incidents. However, online event identification in CPSs is challenged by the complex interdependency of numerous physical and digital components, requiring to take cyber attacks and physical failures equally into account. The online event identification problem is further complicated through the lack of historical observations of critical but rare events, and the continuous evolution of cyber attack strategies. This work introduces and demonstrates CyPhERS, a **Cy**ber-**Ph**ysical **E**vent **R**easoning **S**ystem. CyPhERS provides real-time information pertaining the occurrence, location, physical impact, and root cause of potentially critical events in CPSs, without the need for historical event observations. Key novelty of CyPhERS is the capability to generate informative and interpretable event signatures of known and unknown types of both cyber attacks and physical failures. The concept is evaluated and benchmarked on a demonstration case that comprises a multitude of attack and fault events targeting various components of a CPS. The results demonstrate that the event signatures provide relevant and inferable information on both known and unknown event types.

1. Introduction

The recent development of critical infrastructure such as power grids and water distribution networks is driven by digitalization and automation. The closed-loop integration of physical processes with computer systems and communication technologies renders them cyber–physical systems (CPSs). In such systems, critical incidents can arise from failures of a variety of interconnected physical and digital devices (Alguliyev et al., 2018; Colabianchi et al., 2021). The increasing trend of connecting critical infrastructure to the internet adds cyber attacks as another dimension of possible incident causes (Maglaras et al., 2018). Cyber attacks against CPSs constitute a particular risk, as they can entail damage to physical equipment or even humans. Appropriate countermeasures for critical incidents are facilitated by real-time information about affected devices, root causes and physical impact. As incidents can be caused by failure or attack against a variety of physical and digital devices, integrated monitoring of the cyber and physical domain is advantageous (Frattini et al., 2019). The problem is further complicated by new attack types or unseen physical failures, where no prior knowledge is available for event identification. Consequently, a monitoring system for CPSs is required which provides real-time information about unknown and known types of both cyber attacks and physical failures.

Nowadays, monitoring of the cyber and physical domain is largely conducted in isolated silos, for example through intrusion or fault detection systems. Some recent works propose integration of both through supervised machine learning (ML) (Müller et al., 2022; Ayodeji et al., 2020). While these approaches excel as they automate event identification, they come with two inherent drawbacks: (1) Substantial amounts of naturally scarce historical attack and fault samples are required. (2) Inability to provide information on unknown event types. These shortcomings motivate the following question: *How can operators of CPSs be provided with relevant information for real-time incident response, given the lack of historical critical event observations and the variety of known and unknown attack and fault types potentially affecting different physical or digital components of a CPS?* To address this question, this work proposes CyPhERS, a new **Cy**ber-**Ph**ysical **E**vent **R**easoning **S**ystem (see Fig. 1). CyPhERS comprises a two-stage process which infers event information such as occurrence, location, root cause, and physical impact from joint evaluation of network traffic and physical process data in real time. Stage 1 creates informative event signatures

* Corresponding author.
E-mail address: nilmu@dtu.dk (N. Müller).

https://doi.org/10.1016/j.compind.2023.103982
Received 15 February 2023; Received in revised form 16 May 2023; Accepted 21 June 2023

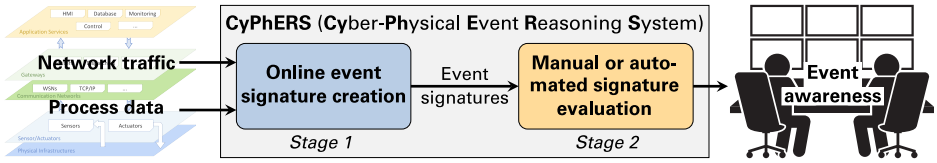


Fig. 1. Schematic representation of CyPhERS.

of unknown and known cyber attacks and faults by combining methods including cyber–physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. In Stage 2, the event signatures are evaluated either by automated matching with a signature database of known events, or through manual interpretation by the operator.

1.1. Related works

Literature on attack or fault identification for CPSs is rich (Zhang et al., 2022; Luo et al., 2021; Giraldo et al., 2018; Cai et al., 2019; Alguliyev et al., 2018; Lindemann et al., 2021; Dalzochio et al., 2020). Many works propose methods which are independent of historical event observations, and able to detect both known and unknown events. Table 1 summarizes these approaches and classifies them by the groups A-C. Conceptual differences exist in the considered data sources (process data, network traffic or both), and the information they provide (e.g., event occurrence, location or impact). A description of methods falling under group A-C is provided in Section 1.1.1. Thereafter, a comparison with CyPhERS follows in Section 1.1.2.

1.1.1. Existing event identification concepts for CPSs

Group A. Works summarized in group A are conceptually characterized by the joint evaluation of multiple CPS variables (e.g., multiple sensor readings), where considered variables are derived either exclusively from physical process (Xi et al., 2022; Li et al., 2019a; Feng and Tian, 2021) or network traffic data (Huong et al., 2021). Unsupervised multivariate time series anomaly detection is applied to detect deviations from normal behavior induced by attacks or faults. Their output is a binary description of the system state (*normal* vs. *abnormal*). Differences among the works mainly exist in the applied ML models and detection rules. The proposed methods can detect unknown and known attack or fault types in real time, given that these entail anomalies in the monitored data. However, as they provide system-wide anomaly flags they only inform about occurrence of an event, while further information, e.g., affected devices or event type, is neglected. Moreover, their restriction to either network or process data limits the events they can detect. While exclusively monitoring network data is blind to physical faults and physical impacts of cyber attacks, limiting to process data misses pure cyber events and only detects cyber–physical attacks when their impact on the system already happened.

Group B. Methods falling under group B apply unsupervised multivariate time series anomaly detection either on process (Zhang et al., 2019; Tuli et al., 2022; Khoshnevisan and Fan, 2019; Hallac et al., 2017; Song et al., 2018; Hundman et al., 2018) or network data (Su et al., 2019; Navarro and Rossi, 2020), similar to group A. Central difference is the provision of feature-level rather than system-wide anomaly flags. By identifying the system variables with the highest individual anomaly scores, affected components are localized in the CPSs. Thus, these concepts provide additional information about detected events to operators. However, the scores only describe the intensity of the deviation from normal behavior, while further characteristics of an anomaly (i.e., polytypic anomaly flags), for example the direction of the deviation or occurrence of missing data, are not considered. Consequently, information on the physical impact or root causes is limited. Moreover, as for group A, monitoring is limited to either network or process data. Consequently, they cannot detect and distinguish anomalies induced by both cyber attacks and physical failures.

Group C. Methods in group C apply unsupervised multivariate time series anomaly detection to features of both process and network data (Niu et al., 2019; Bezemskij et al., 2016; Heartfield et al., 2021). Consequently, they are equally capable of detecting anomalies caused by cyber attacks and physical failures. Moreover, compared to the subset of methods from group A and B which only monitors process data, cyber–physical attacks can be detected earlier and potentially before impacting the process. However, only system-wide monotypic anomaly flags are provided. Therefore, these methods only inform operators about the occurrence of an event without further context information. In contrast to the concepts represented by group A and B, literature on cyber–physical unsupervised anomaly detection for CPSs is rare.

1.1.2. Comparison of existing concepts to CyPhERS

CyPhERS combines strategies of the described concepts (group A-C), such as fusion of process and network data, unsupervised multivariate time series anomaly detection, and provision of feature-level anomaly flags (see Table 1). It further leverages their associated advantages by considering polytypic anomaly flags. Together, this allows CyPhERS to generate highly informative and recognizable event signatures in Stage 1 (see Fig. 1). Moreover, CyPhERS comprises strategies for manual and automated evaluation of the event signatures (Stage 2).

Table 1
Comparison of CyPhERS to existing event sample-independent attack or fault identification concepts.

Group	Concept description	Event coverage	Early detection	Localization	Cause and impact identification
A	Multivariate physical *or* network features, system-wide monotypic anomaly flags	o	o	–	–
B	Multivariate physical *or* network features, feature-level monotypic anomaly flags	o	o	o	–
C	Multivariate physical *and* network features, system-wide monotypic anomaly flags	+	+	–	–
CyPhERS	Multivariate physical *and* network features, feature-level polytypic anomaly flags	+	+	+	+

1.2. Contribution and paper structure

The main contributions of this work are as follows:

- Introduction of CyPhERS, a cyber–physical event reasoning system which provides real-time information about unknown and known attack and fault types in CPSs, while being independent of historical event observations.
- Concept demonstration, evaluation and benchmarking on a CPS study case, considering a variety of attack and fault types affecting several system components.
- Discussion of possible modifications to further improve and extend CyPhERS.

The remainder of the paper is structured as follows: In Section 2, CyPhERS is conceptually introduced. The considered demonstration case is presented in Section 3. Section 4 explains methodological details of CyPhERS, and demonstrates its implementation on the given case. In Section 5, results of applying CyPhERS on the study case are presented. Finally, demonstration results are discussed in Section 6, followed by a conclusion and view on future work in Section 7.

2. Conceptual introduction of CyPhERS

This section introduces the two stages of CyPhERS at a conceptual level. Section 2.1 provides details on the online event signature creation (Stage 1). Thereafter, signature evaluation (Stage 2) is explained in Section 2.2. Finally, Section 2.3 provides a taxonomy of CPSs that CyPhERS can be applied to.

2.1. Online event signature creation (Stage 1)

Stage 1 of CyPhERS is schematically outlined in Fig. 2. To provide event signatures that contain information about occurrence, location, root cause and physical impact of unknown and known types of both attacks and faults in real time, CyPhERS combines several strategies, which are introduced in the following.

2.1.1. Multi-domain information

First element is the fusion and joint evaluation of physical process and cyber network data of a CPS (see Fig. 2), which is required for real-time detection and differentiation of attacks and failures. Moreover, cyber–physical monitoring allows to determine whether a cyber attack already entails physical impact or detect it early enough in network traffic to mitigate damage through countermeasures such as isolation of affected devices. Other data such as maintenance activities and schedules (human domain) can potentially be included to take human errors into consideration.

2.1.2. Feature-level monitoring

Second element is the individual monitoring of multiple system variables (see Fig. 2), aiming for a more detailed picture of critical events in contrast to monitoring the system state. Within CyPhERS, the set of monitored variables is multivariate in two dimensions:

- Monitoring variables of multiple physical and network components (*cross-device multivariate monitoring*).
- Monitoring several variables of a single component (*in-device multivariate monitoring*).

While cross-device multivariate monitoring aims at localizing affected devices, in-device multivariate monitoring is supposed to provide further details about a component's abnormal behavior. The monitored system variables are derived, for example, from sensor measurements or traffic of network devices. In the following, the resulting set of monitored system variables is referred to as *target features*, with $\mathcal{I}$ being the physical target feature and $\mathcal{J}$ the network target feature subset. The time series of an arbitrary target feature c is defined as $X_c = \{x_1^c, x_2^c, \dots, x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$. Details on the extraction of target features follow in Section 4.1.

2.1.3. Unsupervised time series anomaly detection considering covariates

CyPhERS applies unsupervised time series anomaly detection[1] to identify the occurrence of critical events. Time series models are applied to provide normal behavior references of individual target features, which are compared to actual observations for detecting abnormal system behavior. Central argument is the independence of historical event observations, and ability to detect both known and unknown event types, given that they entail anomalies. Monitoring target features in time series format allows to detect deviations from normal behavior which are only abnormal in a specific temporal context (local anomalies) (Cook et al., 2020). Furthermore, covariates are considered for time series modeling. Covariates allow to provide models with further system internal or external information (see Fig. 2). As a result, situational anomalies can be detected which are only abnormal in the context of the provided covariates. A covariate time series of a target feature c is defined as $Z_c = \{z_1^c, z_2^c, \dots, z_N^c \mid z_i^c \in \mathbb{R}\ \forall i\}$. Covariate extraction is detailed in Section 4.2.

2.1.4. Differentiation of anomaly types

The fourth element is the semantic differentiation of multiple anomaly types (see Fig. 2). In case that an anomaly is flagged for a target feature c, it is further classified based on information such

[1] Sometimes referred to as self-supervised anomaly detection.

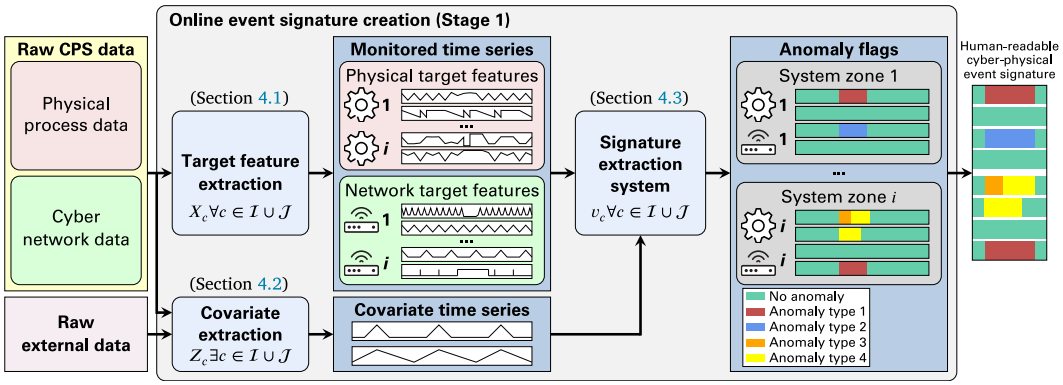


Fig. 2. Schematic overview of CyPhERS' online event signature creation (Stage 1). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

as the direction of the deviation (e.g., abnormally *many* data packets received by network device X). Considering various anomaly types provides additional information for identification of event root causes and impact. The series of flags provided by the signature extraction system for a target feature c is given as $v_c = \{v_1^c, v_2^c, \dots, v_N^c \mid v_i^c \in \{-2, -1, 0, 1, 2\}\ \forall i\}$. A detailed explanation of the signature extraction system, including anomaly types, follows in Section 4.3.

2.1.5. Event signature visualization

By covering multiple domains, system variables and anomaly types, Stage 1 of CyPhERS provides dense information about critical events in form of anomaly flag series of a set of target features. To ease extraction of these information, the flag series are re-organized by grouping them for each system zone of a CPS (see Fig. 2). A system zone comprises a group of physical components and network devices which are directly related, such as a set of physically connected process units and a programmable logic controller (PLC) monitoring and controlling them. Due to the logical relation within a system zone, anomaly flags of different target features can be quickly related. As a result, Stage 1 of CyPhERS provides information-rich and human-readable event signatures.

2.2. Signature evaluation (Stage 2)

The concept of CyPhERS' Stage 2 is schematically depicted in Fig. 3. In Stage 2, the event signatures of Stage 1 are evaluated, which can be realized through interpretation by human operators as well as by automated reasoning systems. The provided signatures are event specific and distinguishable. Thus, for known attacks or faults they can be predefined and stored in a database. Once Stage 1 indicates occurrence of an event, the associated signature can be compared to the database. In case of a signature match, the stored information about event type, affected component, root cause, and/or physical impact provide the event hypothesis. Signature matching can either be conducted by the operator through visual comparison or an automated evaluation system. One automation approach would be the transformation of a signature into a set of rules, e.g.,

> *flagging of* (anomaly type 1 *in* target feature X) *and* (type 2 *in* target feature Y) *indicates* (device A *being targeted by* attack type B *causing* physical impact C).

Signatures can also be defined for unknown event types based on partial knowledge. In this instance, they carry reduced information, e.g.,

> *flagging of* (anomaly type 1 *in* target feature X) *indicates* (device A *failure* [type unknown] *causing* physical impact B).

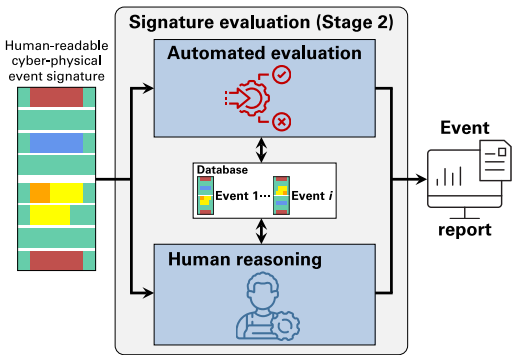


Fig. 3. Schematic overview of CyPhERS' signature evaluation (Stage 2).

In case of unknown or undefined signatures, automated evaluation cannot infer event information. In these situations, operators may deduce information such as affected system components or physical impact based on process expertise. The minimum information CyPhERS provides in any event case is the occurrence of abnormal system behavior.

2.3. Application scope of CyPhERS

The CyPhERS concept is applicable to CPSs which fulfill the following requirements: (1) Real-time data availability of and (2) learnable normal behavior patterns in both process and network traffic data. CPS traffic typically exhibits periodical patterns as it arises from automated processes (e.g., polling) in static network architectures with a consistent number of devices (Barbosa et al., 2012). Moreover, many technical systems exhibit learnable process patterns, including manufacturing processes (Hsieh et al., 2019), transportation systems (Kang et al., 2018), water distribution systems (Abokifa et al., 2019), and spacecraft (Yu et al., 2021). A potentially complicating factor is process volatility and randomness, which, for example, can result from unpredictable weather or user influences. While CyPhERS is conceptually applicable to most CPS types, its implementation requires some case specific adaptations given the heterogeneity of processes and network architectures. These include selection or definition of target features, anomaly types, and known event signatures. Among instances of the same system type (e.g., health monitoring system of a specific provider), the implementation of CyPhERS is fully transferable.

3. Demonstration case description

This section describes the considered demonstration case, which is introduced by Faramondi et al. in Faramondi et al. (2021). The underlying CPS is detailed in Section 3.1. Thereafter, included attack and fault scenarios, and the associated dataset are described in Section 3.2. The demonstration case was selected as the only complete cyber–physical dataset which describes various types of both cyber attacks and physical faults affecting different system components. The physical process represents a typical CPS laboratory setup.

3.1. Cyber-physical structure of demonstration case

Fig. 4 provides a simplified overview of the investigated CPS. The system distributes water between several tanks, where one process cycle is defined by a filling/emptying process of all tanks. This procedure is continuously repeated, making it a cyclical process. The physical system comprises eight water tanks (T1-T8), a reservoir, and several sensors and actuators. Actuators include valves (V10-V22) and pumps (P1-P6), which realize the water distribution between the tanks. Note that for better readability not all sensors and actuators are depicted in Fig. 4. Pressure sensors in T1-T8 and flow sensors (FSs) (FS1-FS4) measure tank fill levels H and water flows F, respectively. The process is monitored and controlled by a typical supervisory control and data acquisition (SCADA) architecture consisting of the sensors and actuators (field instrumentation control layer), four PLCs (process control layer), and a SCADA workstation, including a human-machine interface (HMI) and data historian (supervisory control layer). The SCADA workstation, HMI and data historian together are referred to HMI in the following. The communication is conducted via MODBUS TCP/IP protocol. The process consists of four stages, each of which is controlled by one of the four PLCs. The PLCs send sensor values to the SCADA workstation, so that physical process data can be stored centrally on the historian. Moreover, values of tank fill levels H and water flows F are directly exchanged between the PLCs and FSs, which they require to control tank fill levels by (de-)activating pumps and valves. While most sensors and actuators are connected to the PLCs via wired links, FS1 and FS2 are MODBUS TCP/IP sensors with own IP

addresses. Thus, the communication network in total consists of seven devices, which are marked red in Fig. 4. Note that an additional Kali Linux machine was used to launch cyber attacks, which is not depicted in Fig. 4.

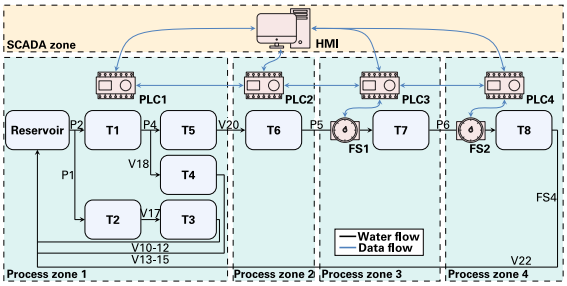


Fig. 4. Simplified overview of the demonstration case CPS based on schematic representations in Faramondi et al. (2021). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.2. Threat scenarios and dataset

The dataset comprises four partitions, each covering multiple process cycles. While the first partition describes a normal operation scenario (S0) the remaining three describe attack and fault scenarios (S1-S3). S1-S3 exhibit an increasing level of event type variety. S1 includes several physical component breakdowns and water leaks as well as man-in-the-middle (MITM) attacks. In MITM attacks a perpetrator positions himself between two victim devices to relay and potentially alter communication while the victims assume a direct communication (Conti et al., 2016). In the present case, the attacker modifies H values send by victim one, which are required by victim two to control fill levels of tanks in the respective process zone. In S2, denial-of-service (DoS) attacks are additionally included. These cause a disconnection of the targeted device from the network by flooding it with requests (Mahjabin et al., 2017). In the investigated dataset, several DoS attack variants are used to disconnect specific PLCs or the SCADA workstation. Finally, S3 adds scanning attacks. Scanning is a reconnaissance method used by attackers to determine possible vulnerabilities by searching for services and service identifiers in a target network or host (Bou-Harb et al., 2014). The given case considers several scanning attacks, which are used to gather information about various PLCs. Note that S1-S3 comprise of unique attack and fault events affecting various components and communications links in the system, so that each scenario represents an entirely new case. In total, eight MITM, five DoS, and seven scanning attacks as well as three water leaks and six sensor or pump breakdowns are included. The considered attack types are among the most relevant for CPSs (Hasan et al., 2023; Cao et al., 2020; Li et al., 2019b; Yaacoub et al., 2020). Table 2 lists the raw physical and network features of the dataset. The physical data has a constant one-second resolution, resulting in 3429 (S0), 2421 (S1), 2105 (S2), and 1255 (S3) samples. The network data during normal operation (S0) on average contains 2265 packets per second, and in total comprises ∼7.8 (S0), ∼5.5 (S1), ∼5.2 (S2), and ∼5.9 (S3) $\times 10^6$ packets. For a more detailed explanation of the demonstration case the reader is referred to Faramondi et al. (2021).

Table 2
Raw network and process features within the study case.

No.	Physical features	No.	Network features
1	Timestamp	1	Timestamp
2-9	Fill level H of T1-T8	2-3	IP address (src. & dst.)[a]
10-15	Activation state S of P1-P6	4-5	MAC address (src. & dst.)
16-19	Flow value F measured by FS1-FS4	6-7	Port (src. & dst.)
20-41	Activation state S of V1-V22	8	Protocol
		9	TCP flags
		10	Packet size
		11	MODBUS function code
		12	MODBUS response value
		13-14	No. of packets (src. & dst.)

[a]Src. and dst. refer to source and destination, respectively.

4. Methodology and implementation of CyPhERS

This section first provides details on the methodology and case-specific implementation of the online event signature creation (Stage 1). This includes extraction of target features (Section 4.1) and covariates (Section 4.2) as well as the signature extraction system (Section 4.3). Thereafter, methodology and case-specific implementation of the event signature evaluation (Stage 2) is detailed in Section 4.4.

4.1. Target feature extraction

Methodology. The landscape of CPSs is characterized by a pronounced heterogeneity, resulting from factors such as the diversity of physical components (e.g., tanks, engines or batteries) and communication protocols (e.g., MODBUS, UDP or DNP). Thus, a general set of target features cannot be defined. Nevertheless, guidelines for extraction of relevant features can be provided.

CyPhERS considers sensor measurements and actuator states as raw physical process data. Monitoring such data has two motivations, namely the identification of (i) true physical events and (ii) manipulation of process-relevant data. The former requires features which represent the behavior (e.g., state or output) of all physical components of a CPS to (1) localize affected devices and (2) infer the impact on them. For the latter, readings exchanged among devices for automated process control need to be monitored. For processes exhibiting low randomness and noise levels, raw sensor readings can directly be used as physical target features. Otherwise, further processing is required to extract the available information. Strategies include resampling (e.g., moving average) or derivation of features which describe a component's behavior on a simplified level (e.g., on/off state).

In CyPhERS, network target features are extracted from OT network traffic[2] of a CPS. Traffic monitoring is considered to retain information for (i) localizing affected digital devices, and (ii) concluding on attack types. For the former, traffic of each network device is monitored separately. The latter requires extraction of several features for each device in order to provide sufficient information for distinguishing attack types. Detecting and differentiating attacks is challenged by the fact that some solely concern individual packets (e.g., sending a malicious control command), while others are only visible from the context of multiple packets (e.g., replaying valid data transmission). Therefore, CyPhERS considers both extraction of network features which (i) are sensible to values of single packets (e.g., count of packets send from unknown source IP or MAC addresses), and (ii) set multiple packets into context (e.g., average number of received packets within a time period).

Case-specific implementation. In the present demonstration case, noise in the raw physical features is comparatively low, which allows direct use as target features with original per-second resolution. For the sake of clarity, not all available sensor readings and actuators states are taken into account. Instead, fill levels H of the water tanks are considered, as they allow to monitor all four process steps, and describe the behavior of the most important components. The network traffic is separately monitored for PLC1-4, FS1-2 and the HMI. For this purpose, it is first filtered by the destination MAC addresses and then evaluated for each second through several features. An overview of the resulting set of physical $\mathcal{I}$ and network target features $\mathcal{J}$ can be found in Table 3.

[2] Other potential data sources include system logs and key performance indicators of digital devices (e.g., memory usage).

Table 3
Overview of target features extracted in the study case.

No.	Physical target features $\mathcal{I}$	No.	Network target features[a] $\mathcal{J}$
1-8	Fill level H of T1-T8	1-7	Average packet size s_{packet}
		8-14	Packet count n_{packet}
		15-21	New src. IP/MAC count $n_{\text{IP/MAC}}$
		22-28	New TCP flags count n_{TCP}
		29-35	Mean of encoded TCP flags[b] μ_{TCP}
		36-42	Different src. ports count n_{ports}

[a]For PLC1-4, FS1-2 and HMI, each considered as destination device.
[b]Each flag type is encoded as a specific integer.

4.2. Covariate extraction

Methodology. As for target features, extraction of covariates is dependent on the respective CPS. In any case, however, they should represent process-relevant context such as environmental conditions or human interactions. Examples are irradiation for solar plants or user intervention in case of self-driving cars. Such information allows models to learn whether an event is normal or abnormal given the current context. For example, irradiation facilitates differentiation of normal weather- or malicious attacker-induced drops of solar feed-in power. CPSs often exhibit repeating processes. In such cases, covariates which inform models about the current position in a cycle (e.g., process stage or time of the day) provide valuable context information. CyPhERS considers sine and cosine transformation (Chakraborty and Elzarka, 2019) of such cyclical covariates, which is schematically represented in Fig. 5. Let $Z_c = \{z_1^c, z_2^c, \dots, z_N^c \mid z_i^c \in \mathbb{R}\ \forall i\}$ be a cyclical covariate time series of length N for a target feature c. Within one cycle, values of Z_c are linearly increasing on the range $z_i^c \in [\min(Z_c), \max(Z_c)]$. Due to the jump discontinuity between two cycles, a linear representation cannot properly describe the continuity of cyclical processes. To eliminate the discontinuity, values of Z_c are transformed according to

$$z_{\text{sin},i}^c = \sin\left(\frac{2\pi z_i^c}{\max(Z_c)}\right), \text{ and } z_{\text{cos},i}^c = \cos\left(\frac{2\pi z_i^c}{\max(Z_c)}\right), \tag{1}$$

$\forall i \in [1, N]$, resulting in the two new covariate time series Z_{sin}^c and Z_{cos}^c. The use of both sine and cosine transformation is required as they individually are not bijective, which would lead to ambiguity in the transformed covariate (see Fig. 5).

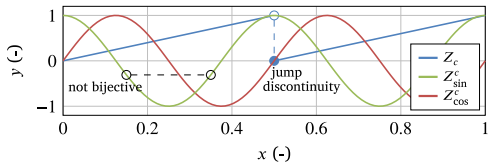


Fig. 5. Illustration of sine and cosine transformation of cyclical covariates. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Case-specific implementation. In the investigated study case, the normal progress of a process cycle is considered as covariate time series $P_c = \{p_1^c, p_2^c, \dots, p_N^c \mid p_i^c \in \mathbb{R}\ \forall i\}$ of length N, $\forall c \in \mathcal{I}$. Using S0 (normal operation scenario) the usual duration d_c of a process cycle is determined. Based on the duration, values of P_c are defined on the range $p_i^c \in [0, d_c]$. The additional covariate time series P_{sin}^c and P_{cos}^c are extracted $\forall c \in \mathcal{I}$ by applying sine and cosine transformation on the values of P_c according to (1) with $\max(P_c) = d_c$.

4.3. Signature extraction system

The signature extraction system (see Fig. 2) follows the idea of applying individual anomaly detection and classification pipelines to each target feature. Compared to the joint processing in one large model, several advantages exist: (1) The complexity of time series models can be adjusted to individual target features. As features in CyPhERS originate from very different sources (process data and network traffic), they exhibit strong variations in characteristics such as observation rates and noise levels. (2) Independent definition of abnormal behavior for each system component. By selecting covariates for individual target features, it is possible to define which context models should consider when deciding whether a component is behaving abnormally. (3) Promotes a distributed implementation of CyPhERS on edge devices. As attackers can manipulate data to hide induced physical impact from centralized monitoring (Giraldo et al., 2018), this facilitates detection of hidden process manipulations.

The anomaly detection and classification pipelines are explained in Section 4.3.1. After that, Section 4.3.2 addresses the forecasting models which are applied within the pipelines. Finally, the procedure for automated implementation of the signature extraction system is detailed in Section 4.3.3.

4.3.1. Anomaly detection and classification pipelines

Methodology. The anomaly detection and classification pipeline of a target feature c is schematically depicted in Fig. 6. A pipeline consists of two fundamental and consecutive steps, namely a time-series forecasting model and an anomaly detector. Given $X_c = \{x_1^c, x_2^c, \dots, x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$ and $Z_{c,1}, \dots, Z_{c,n} = \{\{z_{1,1}^c, z_{1,2}^c, \dots, z_{1,N}^c\}, \dots, \{z_{n,1}^c, z_{n,2}^c, \dots, z_{n,N}^c\} \mid z_{j,i}^c \in \mathbb{R} \forall (j,i)\}$ of a target feature c, the forecasting model predicts the expected value $\hat{x}_t^c$ at time t based on lag values $x_{t-w}^c, \dots, x_{t-1}^c$ and covariates $z_{1,t}^c, \dots, z_{n,t}^c$ according to

$$\hat{x}_t^c = \Phi\left([x_{t-w}^c, \dots, x_{t-1}^c], [z_{1,t}^c, \dots, z_{n,t}^c]\right), \tag{2}$$

where w is the length of the history window and n the number of covariates. Depending on the target feature, $x_{t-w}^c, \dots, x_{t-1}^c$ and $z_{1,t}^c, \dots, z_{n,t}^c$ are only partially used as model input, which is specified in Section 4.3.2.

Next, the expected value $\hat{x}_t^c$ and ground truth x_t^c are forwarded to the anomaly detector. In CyPhERS, anomalies are flagged based on multiple consecutive observations instead of only the most recent one, aiming at reducing noise-induced false positives (FPs). For that purpose, the detector first calculates the average of the distances of the last l observations to their respective expected values according to

$$\varepsilon_t^c = \frac{\sum_{j=0}^{l-1} |x_{t-j}^c - \hat{x}_{t-j}^c|}{l}. \tag{3}$$

Based on ε_t^c and further characteristics of the current target feature observations, the detector then differentiated several anomaly types. While the definition of meaningful anomaly types is facilitated by taking process specificities of a CPS into account, some widely applicable ones exist. Table 4 defines some of them. The listed types can be transferred into the detector's anomaly flag decision function according to

$$v_t^c = \begin{cases} 2 & \text{if } \overbrace{(\varepsilon_t^c > \tau_c)}^{\text{Detection}}, \overbrace{(x_t^c > \hat{x}_t^c)}^{\text{Direction}} \text{ and } \overbrace{[\exists (x_i^c = x_{i-1}^c \text{ or NaN}) \in X_J^c]}^{\text{Disruption}} \\ 1 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c > \hat{x}_t^c) \text{ and } [\nexists (x_i^c = x_{i-1}^c \text{ or NaN}) \in X_J^c] \\ -1 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c < \hat{x}_t^c) \text{ and } [\nexists (x_i^c = x_{i-1}^c \text{ or NaN}) \in X_J^c] \\ -2 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c < \hat{x}_t^c) \text{ and } [\exists (x_i^c = x_{i-1}^c \text{ or NaN}) \in X_J^c] \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where τ_c is a target feature-specific threshold, and $X_J^c = x_{t-l+1}^c, \dots, x_t^c$ the ground truth values within the distance averaging window. The automated adaption of τ_c to individual target features is described in Section 4.3.3. The differentiation between static and non-static behavior is neglected for target features exhibiting static values during normal operation, which in this event reduces (4) to

$$v_t^{c*} = \begin{cases} 2 & \text{if } \overbrace{(\varepsilon_t^c > \tau_c)}^{\text{Detection}}, \overbrace{(x_t^c > \hat{x}_t^c)}^{\text{Direction}} \text{ and } \overbrace{[\exists (x_i^c = \text{NaN}) \in X_J^c]}^{\text{Disruption}} \\ 1 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c > \hat{x}_t^c) \text{ and } [\nexists (x_i^c = \text{NaN}) \in X_J^c] \\ -1 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c < \hat{x}_t^c) \text{ and } [\nexists (x_i^c = \text{NaN}) \in X_J^c] \\ -2 & \text{if } (\varepsilon_t^c > \tau_c),\ (x_t^c < \hat{x}_t^c) \text{ and } [\exists (x_i^c = \text{NaN}) \in X_J^c] \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

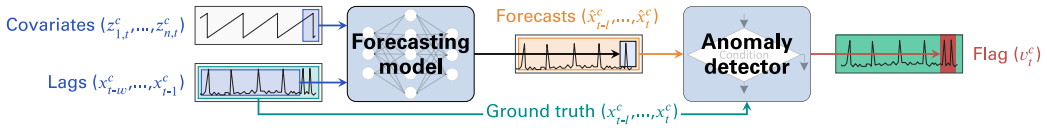


Fig. 6. Schematic overview of the anomaly detection and classification pipeline of a target feature c.

Case-specific implementation. For the investigated study case, a distance averaging window $l = 10$ is selected for calculation of ϵ^c_t according to (3), $\forall c \in \mathcal{I}$ and $\mathcal{J}$. Considered anomaly types correspond to the ones listed in Table 4. For H_{T7} and H_{T8} as well as $n_{\text{IP/MAC}}$ and n_{TCP} of all network devices, the detector's decision function reduces to (5) as they contain static values during S0.

Table 4
Description of some general anomaly types.

Flag v	Anomaly type	Description	Schematic
2	Positive disrupted	Target feature positively differentiates from normal behavior, exhibiting static or NaN values.	
1	Positive undisrupted	Target feature positively differentiates from normal behavior, not exhibiting static or NaN values.	
-1	Negative undisrupted	Target feature negatively differentiates from normal behavior, not exhibiting static or NaN values.	
-2	Negative disrupted	Target feature negatively differentiates from normal behavior, exhibiting static or NaN values.	
0	No anomaly	No abnormal behavior.	

4.3.2. Forecasting models

Methodology. As some CPSs come with computational constraints, keeping model complexity at a required minimum is favorable. The forecasting models used within the anomaly detection and classification pipelines (see Fig. 6) are interchangeable, which allows adapting their complexity to the specific characteristics of a target feature. Four target feature property classes are differentiated in CyPhERS. These are listed in Table 5 together with feature examples and recommended forecasting model types. Class A comprises target features exhibiting unchanged values during normal operation. In these cases, a trivial constant-value forecast is sufficient, which simplifies (2) to $\hat{x}^c_t = a_c$, where a_c corresponds to the constant value of the respective target feature during normal operation. In class B, continuous target features with covariate availability are summarized, which typically includes physical sensor measurements. Due to the additional information covariates provide, less complex forecasting models can be considered. The use of simple regression models (e.g., linear regression) for modeling class B features according to (2) is recommended. Target features may also exhibit discrete values (Class C) as in the case of actuator states. In this event, the use of ensemble models such as a random forest (RF) regressor (Breiman, 2001) is proposed for modeling features according to (2). The rationale behind using ensemble models for class C features is their internal process of discretizing continuous variables, which facilitates prediction of sudden steps. Moreover, they are known for robustness, few parameters to tune and good performance compared to many other standard methods on a variety of prediction problems (Scornet et al., 2015; Bojer and Meldgaard, 2021). For a detailed theoretical description of ensemble models, the reader is referred to Hastie et al. (2009). Finally, Class D comprises continuous target features without availability of covariates. Since covariates are neglected, (2) reduces to

$$\hat{x}^c_t = \Phi([x^c_{t-w}, \dots, x^c_{t-1}]). \tag{6}$$

Due to lack of additional information through covariates, more advanced models are required, which are capable of exploiting short and long-term temporal dependencies within a target feature. Thus, deep-learning-based forecasting models are suggested for class D features. Prominent representatives are long short-term memory (LSTM) networks (Hochreiter and Schmidhuber, 1997). LSTMs constitute a special architecture of neural networks capable of capturing complex long-term temporal dependencies in sequential data, which makes them well suited for time series forecasting. Many works have demonstrated their superior performance in various areas (Siami-Namini et al., 2018; Nelson et al., 2017; Srivastava and Lessmann, 2018). For a detailed theoretical description of LSTM networks, the reader is referred to Goodfellow et al. (2016).

Case-specific implementation. In the investigated demonstration case, $n_{\text{IP/MAC}}$ and n_{TCP} constitute constant target features (class A), which hence are modeled with trivial constant value forecasts. All considered physical target features of the study case ($H_{\text{T1}} - H_{\text{T8}}$) fall into class B. Thus, RF is selected as the regression model. As P_c, $P^c_{\sin}$ and $P^c_{\cos}$ are available as supporting covariate time series, a comparatively short history window of $w = 10$ is considered. Table 6 lists the tuned hyperparameters and their respective search spaces. The underlying model selection procedure is detailed in Section 4.3.3. The RF models are implemented in Python using the forecasting-library *Darts* (Herzen et al., 2021).

Table 5
Target feature property classes and proposed forecasting models.

Class	Condition[a]	Model type	Feature example
A	Target features with solely constant values	Constant value	Occurrence of new IP address
B	Continuous target features with covariates	Simple regression (e.g., linear or RF)	Sensor measurements
C	Target features with discrete levels	Ensemble regression (e.g., RF)	States of actuators
D	Continuous target features without covariates	Deep-learning (e.g., LSTM)	Number of transmitted network packets

[a]Constant, continuous and discrete behavior relates to normal operation.

Table 6
Hyperparameters and search spaces for RF and LSTM.

No.	Hyperparameter[a]	Search space
RF (physical target features $\mathcal{I}$)		
1	Number of trees	1, 50, 100, 250, 500, 1000
2	Nr. of features for best split determination	1, 3, 5, 7, 9, 11
LSTM (non-constant network target features $\mathcal{J}_{\text{nc}}$)		
1	Number of LSTM layers	1, 2, 3
2	Batch size	32, 64
3	Number of epochs	100, 200, 500
4	Dropout rate	0, 0.2
5	Number of nodes in LSTM layers	20, 50, 100

[a]For other hyperparameters, default values from Herzen et al. (2021) are used.

For forecasting the network target features, no supporting covariates are available. Thus, the set of non-constant network target features $\mathcal{J}_{\text{nc}}$ falls into property class D. Consequently, concerned features are modeled using LSTM networks. To allow a LSTM to capture long-term dependencies, the history window is extended to $w = 300$, covering an entire process cycle. In Table 6, the tuned hyperparameters and their

respective search spaces are given. Implementation of the LSTM models is realized in Python using the forecasting-library *Darts* (Herzen et al., 2021).

4.3.3. Automated model and detector tuning procedure

Methodology. Implementing the signature extraction system requires performing the same automated tuning procedure for the detection and classification pipelines of all target features, which is schematically represented in Fig. 7. First step is the selection of forecasting models, which is concerned with the determination of appropriate target feature-specific hyperparameters. The hyperparameters are selected based on a grid search using time series cross-validation on a training/validation set $X^c_{\text{train/val}}$ of normal operation data. For each fold $X^c_{\text{fold}} = \{x^c_1, x^c_2, \dots, x^c_{N_{\text{fold}}} \mid x^c_i \in \mathbb{R}\ \forall i\}$ of the cross-validation, data is scaled to $\bar{x}^c_i \in [0, 1]$ before training by

$$\bar{x}^c_i = \frac{x^c_i - \min\left(X^c_{\text{train}}\right)}{\max\left(X^c_{\text{train}}\right) - \min\left(X^c_{\text{train}}\right)}, \tag{7}$$

$\forall x^c_i \in X^c_{\text{fold}}$, where X^c_{train} corresponds to the first 75 % of the respective fold. In case a covariate time series $Z^c_{\text{train/val}}$ or multiple of them are used, they are scaled in the same way as target features in (7). To finalize model selection, the resulting forecasting models are retrained on the respective full training/validation set $X^c_{\text{train/val}}$, again on values scaled according to (7), where now $X^c_{\text{train}} = X^c_{\text{train/val}}$.

After selecting appropriate forecasting models, the anomaly detectors (see Fig. 6) of all pipelines are fitted to the respective target feature. For that purpose, a target feature-specific threshold τ_c is determined for each pipeline as follows. First, the associated forecasting model is used to predict the expected values of a normal operation test set X^c_{test} based on a rolling one-step ahead forecast. The expected values are required to calculate all averaged distances of the test set $E^c_{\text{test}} = \{\epsilon^c_1, \epsilon^c_2, \dots, \epsilon^c_{N_{\text{test}}} \mid \epsilon^c_i \in \mathbb{R}\ \forall i\}$. Based on E^c_{test} and a threshold factor f, the feature-specific threshold is determined according to

$$\tau_c = f \cdot \max\left(E^c_{\text{test}}\right). \tag{8}$$

If a target feature exhibits a high noise level or if non-optimal hyperparameters are selected, $\max\left(E^c_{\text{test}}\right)$ increases due to a weaker forecasting performance. Thus, according to (8), τ_c automatically adapts to the prediction performance of the respective forecasting model. Objective of the proposed adaptive threshold is the reduction of FPs.

Next, to fully exploit available historical data, the selected forecasting models are retrained on the respective totality of normal operation observations ($X^c_{\text{train/val}} + X^c_{\text{test}}$) and potentially covariates ($Z^c_{\text{train/val}} + Z^c_{\text{test}}$).

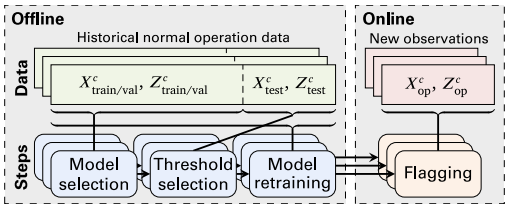


Fig. 7. Automated tuning procedure of the anomaly detection and classification pipelines constituting the signature extraction system.

After tuning the forecasting models and anomaly detectors, the detection and classification pipelines are ready for operation and can be applied to create anomaly flags for newly incoming observations X^c_{op}, which again are scaled according to (7) with $X^c_{\text{train}} = X^c_{\text{train/val}}$. Finally, the resulting anomaly flags of individual pipelines are grouped for each system zone of a CPS to obtain event signatures as output of CyPhERS' Stage 1.

Case-specific implementation. In the considered study case, $X^c_{\text{train/val}}$ and $Z^c_{\text{train/val}}$ are taken from the first 75 % and X^c_{test} and Z^c_{test} from the remaining 25 % of S0, $\forall c \in \mathcal{I}$ and $\mathcal{J}$. X^c_{op} and Z^c_{op} are provided by the three attack and fault scenarios S1-S3, $\forall c \in \mathcal{I}$ and $\mathcal{J}$. Moreover, the threshold factor f in (8) is specified to $f = 1.5$. Consequently, an anomaly within a target feature c only is flagged if the average distance over the last 10 observations exceeds the biggest average distance during normal operation by at least 50 %. The target features are grouped according to the system zones depicted in Fig. 4.

4.4. Signature evaluation (Stage 2)

Methodology. Signature evaluation in CyPhERS is based on manually or automatically matching anomaly flags provided by Stage 1 with a database of known event signatures. The possible attack and fault types that can affect a CPS depend on the system's physical and digital components and architectures. Consequently, a set of event signatures which is valid across all systems cannot be defined. However, some general anomaly flag interpretation rules which hold for most CPSs have been identified, and are listed in Table 7. These general rules can be applied to create signature databases. They range from simple principles, such as indication of affected system components through appearance of anomaly flags in the associated target features (rule

Table 7
General anomaly flag interpretation rules for the definition of event signatures.

No.	Rule description
1	Appearance of anomaly flags in a target feature indicates that the underlying physical or network component is affected (localization).
2	Anomaly flags exclusively in physical target features points towards a physical failure.
3	Anomaly flags exclusively in network target features, including flags which indicate malicious activities[a], points towards a cyber attack.
4	Anomaly flags exclusively in network target features, without flags indicating malicious activities[a], points towards a network (device) failure.
5	Flags in both physical and network target features, including flags indicating malicious activities[a], points towards a cyber–physical attack.
6	Flags in physical and network target features, without flags indicating malicious activities[a], indicate a network (device) failure entailing physical impact.
7	Anomaly flags exclusively in physical target features of one component indicates a local failure without impact on other components.
8	Physically plausible and coherent flags in physical target features of multiple components indicates a problem of their physical connection.
9	Anomaly flags exclusively in network target features of one device indicates a local device problem without impact on the rest of the system.
9.1	Rule 9 together with flags indicating malicious activity[a] point towards a reconnaissance attack (e.g., scanning).
10	Flags simultaneously and exclusively in network target features of two network devices indicates a problem of their bilateral communication.
10.1	Rule 10 together with flags indicating malicious activities[a] for both devices point towards a MITM attack.
10.2	Rule 10.1 with flags in physical target features indicate a MITM attack which manipulates process relevant data entailing physical impact.
11	Flags in physical target features indicating data disruption point towards disconnection of the network device which sends the data.
11.1	Rule 11 together with flags indicating malicious activities[a] indicates a DoS attack against the disconnected device.
11.2	Rule 11.1 with flags in physical target features which indicate true physical impact point towards DoS attack interrupting process relevant data.
12	Flags only in network target features of a device X and the ones connected to it indicate disconnection of device X.
12.1	Rule 12 together with flags indicating malicious activities[a] point towards a DoS attack against device X.
12.2	Rule 12.1 with flags in physical target features which indicate true physical impact point towards DoS attack interrupting process relevant data.

[a]Malicious activities are, for example, communication with unknown devices or untypical connection requests from known devices.

1), to more complex anomaly flag patterns which point to specific attack or fault types (e.g., rule 12.2). Therefore, also signatures of different information detail can be defined, ranging from unclassified signatures, e.g. `Unknown event type affecting device X`, to specific event hypotheses, e.g. `DoS attack against device X entailing physical impact Y on component Z`.

Case-specific implementation. Based on the general flag interpretation rules in Table 7, a set of known event signatures is defined for the demonstration case. Fig. 8 visualizes them for selected victim devices, and on a reduced version of the system. A description of the signatures is provided in Table 8. For the sake of clarity, flags in network target features are grouped for each network device according to

$$\tilde{v}_t = \begin{cases} 1\ (anomaly) & \text{if } \exists(v_t^c = 2, 1, -1 \text{ or } -2) \in V_t^{\mathcal{J}} \\ 0\ (no\ anomaly) & \text{otherwise,} \end{cases} \tag{9}$$

where $V_t^{\mathcal{J}}$ is the set of anomaly flags of one device at time t. However, note that the grouped network target features of the victim devices during attacks must include flags indicating malicious activities (e.g., connection to unknown external device) to fulfill the attack signatures. The differentiation between anomaly types is not applicable for the grouped anomaly flags $\tilde{v}_t$ and thus neglected. For the demonstration of CyPhERS on the given study case in Section 5, manual recognition of the defined event signatures is considered.

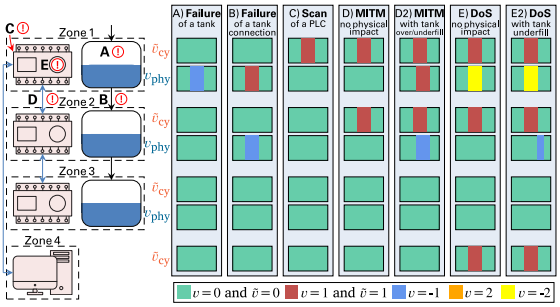


Fig. 8. Event signatures for the study case. The signatures are depicted for selected victim devices and physical impacts. Grouped network features $\tilde{v}_{cy}$ of the victim devices include flags indicating malicious activities during attacks. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 8
Description of the event signatures depicted in Fig. 8, and applied anomaly flag interpretation rules.

Event signature	Description	Applied rules
(A) **Failure** of a tank	Abnormal high/low level of individual tank, no network anomaly. Thus, leak, in- or outflow failure[a].	1,2,7
(B) **Failure** of tank connection	Parallel over-/underfill of linked tanks, no network anomaly. Thus, pump/valve failure or leak between them[a].	1,2,8
(C) **Scan** of a PLC	Individual PLC affected, communication to unknown device with unusual TCP flags, no physical anomalies.	1,3,9.1
(D) **MITM** w/o physical impact	Simultaneous and exclusive network anomalies in two connected PLCs, both communicate with unknown device, no physical anomalies.	1,3,10.1
(D2) **MITM** w. over-/underfill	Signature D with over-/underfill of tanks controlled by the victim PLCs. Manipulated fill levels distract pumps.	1,5,10.2
(E) **DoS** w/o physical impact	Network anomalies only for a device X (e.g., a PLC) and the ones connected to it, physical data communicated by device X disrupted, connection of device X to unknown device, no physical plausible anomalies.	1,3,11.1,12.1
(E2) **DoS** w. tank underfill	Signature E with underfill of a tank controlled by a PLC which receives data from the disconnected victim device.	1,5,11.2,12.2

[a]Specific failure type can be concluded from anomaly flag directions and/or actuator type between tanks.

5. Demonstration

In this section, results of applying CyPhERS on the three attack and fault scenarios (S1-S3) of the demonstration case are presented. In preparation of that, Section 5.1 explains how the considered alternative approaches are represented for benchmarking, and Section 5.2 demonstrates attack signatures within ungrouped network target features. Thereafter, S1-S3 are successively evaluated in the Sections 5.3–5.5. In that context, examination of S2 includes comparison with the three benchmarks. The investigated dataset contains some wrong ground truth event lengths and labels as well as further unlabeled anomalies. Thus, focus of this section is on a qualitative demonstration of CyPhERS since a meaningful quantitative assessment is impractical under these circumstances.

5.1. Benchmark concepts

As part of the following demonstration of CyPhERS, a qualitative comparison to the existing event identification concepts introduced in Section 1.1 is conducted (group A-C). Group A is represented by considering only physical target features of CyPhERS, and grouping associated flags to provide system-wide monotypic anomaly flags v_{CPS}. CyPhERS' physical target features without grouping their flags are considered for representing monotypic anomaly flags on feature level (group B). For group C, anomaly flags in both physical and network target features together are grouped, providing cyber–physical system-wide monotypic anomaly flags.

5.2. Attack signatures in ungrouped network target features

The subsequent evaluation of S1-S3 considers grouped network target features for clarity (see Section 4.4). To demonstrate the appearance of the ungrouped flags that CyPhERS' Stage 1 provides during MITM, DoS and scanning attacks, they are depicted in Fig. 9 for selected network devices.

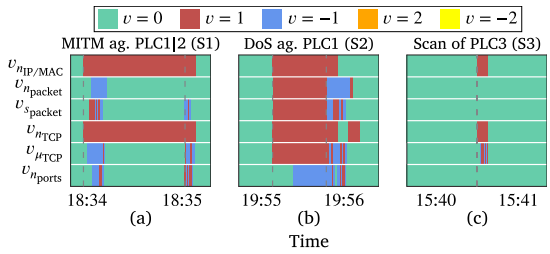


Fig. 9. Flags in ungrouped network target features provided by CyPhERS' Stage 1 for (a) PLC1 during MITM, (b) PLC1 during DoS and (c) PLC3 during scanning attack. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

During all three attack types malicious activities are flagged, which is a requirement of the pre-defined attack signatures (see Fig. 8). These activities comprise communication with an unknown device ($v_{n_{\text{IP/MAC}}} = 1$) and use of unusual TCP flags ($v_{n_{\text{TCP}}} = 1$). Fig. 9 further indicates that the different attack types also express in distinctive signatures within the ungrouped network features. For example, DoS attacks result in pronounced anomaly flags on all features in contrast to the others. Fig. 10 showcases this on a comparison to MITM attacks. While the DoS attack entails a global anomaly, the MITM attack only results in

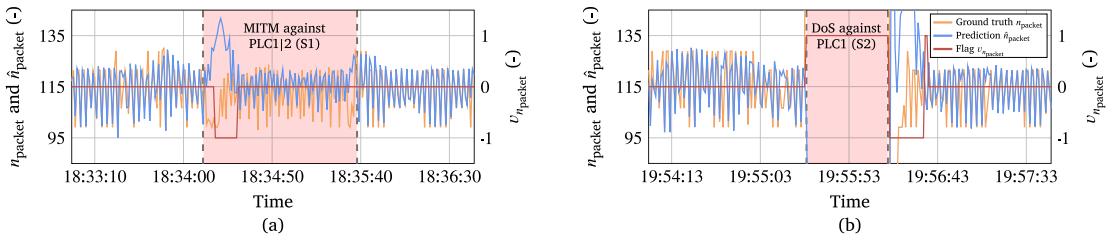


Fig. 10. Anomalies in n_{packet} of PLC1 induced by a (a) MITM and (b) DoS attack. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

a local one, primarily in the beginning and end of the attack.[3] While the differences within ungrouped network features are not taken into account in the following demonstration of CyPhERS, they potentially can be integrated to provide further information for distinction of different attack types.

5.3. Evaluation of attack and fault scenario S1

The event signatures provided by CyPhERS' Stage 1 during S1 are depicted in Fig. 11. The anomaly flags of tank fill levels ($v_{H_{T1}} - v_{H_{T8}}$) are located next to the grouped network feature flags $\tilde{v}$ of the PLC controlling the respective process zone (see Fig. 4). The scenario comprises five MITM attacks and three physical faults, affecting several network devices and physical components.

5.3.1. MITM attacks

From Fig. 11 it is visible that anomaly flags during MITM attacks either follow event signature D or D2 (see Fig. 8), which allows to conclude on the attack type, victim devices, attacker location and physical impact.[4] For example, during the third MITM attack (~18:45), signature D2 indicates a MITM attack against PLC3 and PLC4 from an external device entailing overfill of T7 ($v_{H_{T7}} = 1$) and underfill of T8 ($v_{H_{T8}} = -1$) due to failed pump activation as a result of manipulated fill levels exchanged between the victim PLCs. Note that in all cases MITM-induced anomalies are flagged in network features before the physical process is impacted, potentially allowing incident response mechanisms to take timely countermeasures.

The modification of the physical process during the first and third MITM attack is comparatively strong. As a result, the impact also extends over the next process cycle as well as consecutive tanks, which explains the two additional anomalies indicated in Fig. 11. As an example, Fig. 12 depicts the abnormal behavior of T7 in the next process cycle after the first MITM attack. The comparison with the subsequent process cycle clearly indicates that T7 remains filled for an unusually long period.

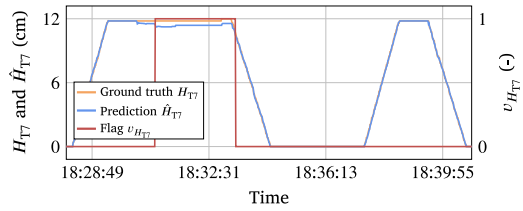


Fig. 12. Unlabeled anomaly of T7 during S1 as a result of the MITM attack in the previous process cycle. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.3.2. Physical faults

Anomaly flags during physical faults follow event signature B (see Fig. 8), allowing to localize affected components, and infer fault types and physical impact. For example, during the second fault (~18:40) the provided signature indicates a leak between T1 and T4, resulting in a simultaneous underfill of T1 ($v_{H_{T1}} = -1$) and overfill of T4 ($v_{H_{T4}} = 1$). During the first and third fault, parallel anomalies from the previous MITM attacks complicate recognition of signature B. However, as the

[3] Note that the detection of the MITM attack-induced local anomaly in Fig. 10 demonstrates the advantage of incorporating temporal information through time series models, as motivated in Section 2.1.3.

[4] According to Faramondi et al. (2021), the last MITM attack is supposed to affect PLC3 and FS2. However, the anomaly flags point towards FS1 instead of FS2. A look on Fig. 4 shows that PLC3 and FS2 in fact do not communicate, proving successful identification of a wrong label.

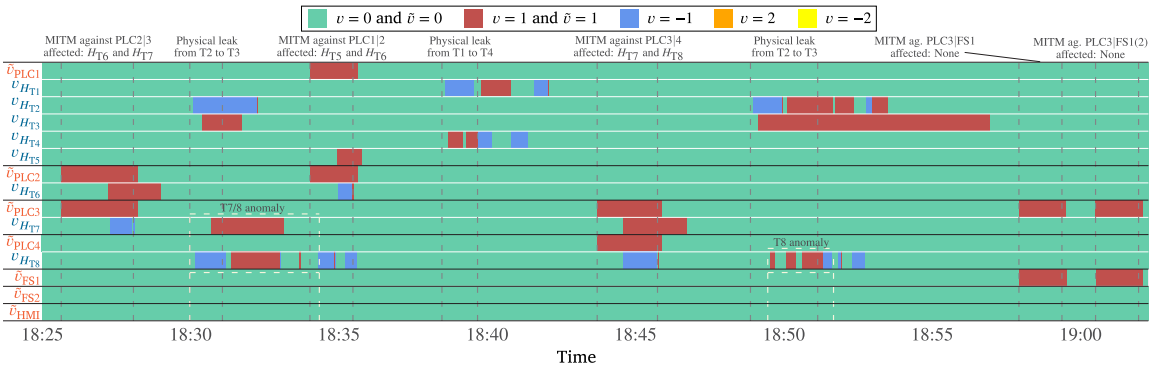


Fig. 11. Event signatures of CyPhERS' Stage 1 during scenario S1. Additional anomalies not labeled by Faramondi et al. (2021) are marked using beige boxes, wrong ground truth labels are given in parenthesis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

flags for T7 and T8 can be explained by the preceding attacks, while a plausible connection to the abnormal behavior of T2 and T3 cannot be derived, these two events can be disentangled.

Fig. 11 indicates that CyPhERS' Stage 1 detects events at an early stage, and reliably differentiates anomaly types in the beginning of events. However, the flags overrun the events for two reasons: Firstly, the CPS does not immediately recover after an attack or fault, thus, anomalies naturally persist. Secondly, the anomaly detection and classification pipelines exhibit a *recovering phase* after detected events. As the detector evaluates the average distance of several consecutive observations, anomalies are flagged for some more steps even though the system behavior is already normal. Moreover, while passing an event, abnormal observations become the new model input, which manipulates the predictions entailing longer anomaly flags and unreliable flag types. A concept improvement tackling this shortcoming is discussed in Section 6.2.

5.4. Evaluation of attack and fault scenario S2

In the second scenario, DoS attacks are performed in addition to MITM attacks and physical faults. The event signatures provided by CyPhERS' Stage 1 during S2 are depicted in Fig. 13 together with the ones of the three benchmarks.

5.4.1. MITM attacks and physical faults

Anomaly flags provided by CyPhERS during faults either follow signature A or B. Thus, the affected tanks are localized, and pump or valve failures together with resulting physical impact inferred. The benchmarks providing system-wide flags (Fig. 13(a) and (c)) indicate fault-induced anomalies, however, do not provide information on affected components, event types and physical impact. In contrast, flagging anomalies in individual physical features (Fig. 13(b)) additionally allows for localizing affected tanks. Nevertheless, the lack of network target features and anomaly type differentiation renders identification of event types and physical impact infeasible.

Flags of CyPhERS during the two MITM attacks follow signature D2, allowing to conclude attack type, victims, attacker location, and physical impact. During the last attack, no physical impact should exist according to Faramondi et al. (2021). However, as can be seen from Fig. 14, T8 in fact exhibits abnormal behavior.

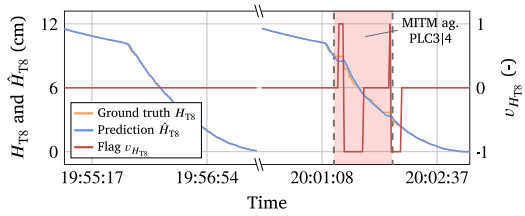


Fig. 14. Unlabeled anomaly of H_{T8} in S2 during the MITM attack against PLC3 and PLC4. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The benchmarks of group A and B only detect the physical impact of MITM attacks (see Fig. 13(a) and (b)). As a result, they exhibit high detection delays during the first one. Moreover, none of the three benchmarks can indicate that anomalies are caused by a cyber attack.

5.4.2. DoS attacks

The flags provided by CyPhERS' Stage 1 during the two DoS attacks correspond to signature E or E2, allowing to conclude on attack type, victim device, attacker location, and physical impacts. For example, in the second case, signature E2 indicates a DoS attack against PLC1 from an external device, resulting in a slight underfill of T6 ($v_{H_{T6}} = -1$) due to interrupted communication of H_{T5} values from PLC1 to PLC2.

The three benchmarks neither indicate disconnection of a network device nor localize it, since they either only detect DoS-induced anomalies in physical process data, or provide non-interpretable system-wide

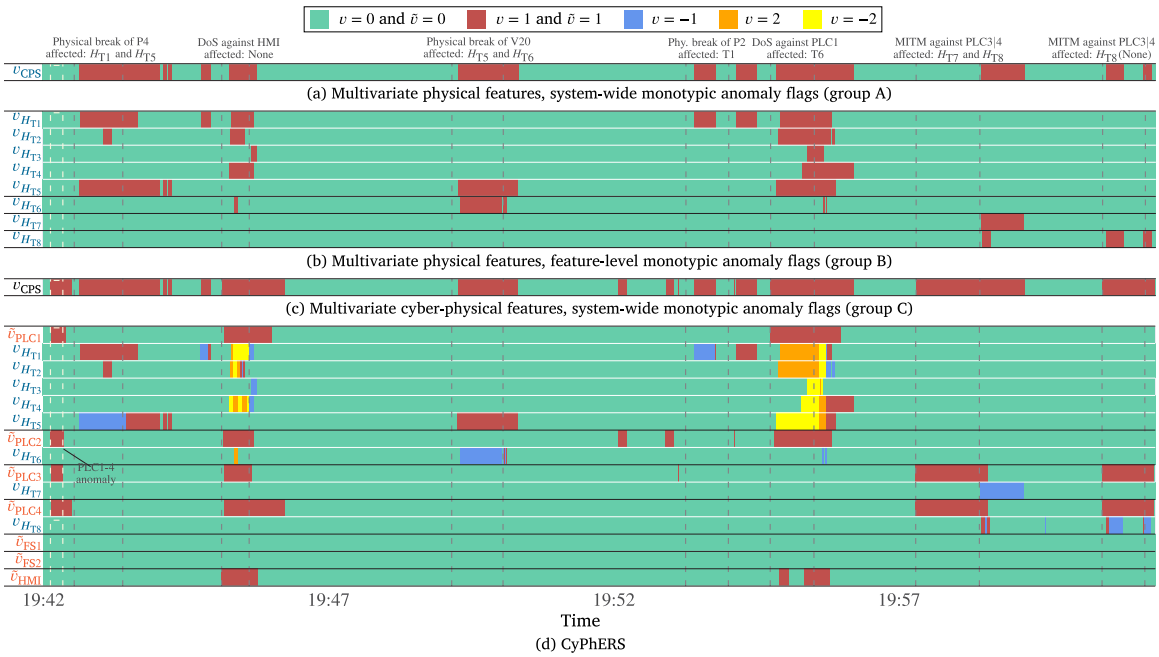


Fig. 13. Event signatures of CyPhERS' Stage 1 and the three benchmarks during scenario S2. Additional anomalies not labeled by Faramondi et al. (2021) are marked using beige boxes, wrong ground truth labels are given in parenthesis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

flags. Moreover, as they only output monotypic flags, it cannot be inferred that most anomalies result from data disruption instead of true physical events.

5.4.3. Unlabeled event

CyPhERS' Stage 1 detects an unlabeled event in the beginning of S2, which is depicted in Fig. 15 on the example of PLC2's μ_{TCP}. Although the event signature is not known, fulfillment of flag interpretation rules 1 and 4 (see Table 7) indicates a network failure only affecting PLC1-4 without physical impact. This example demonstrates how CyPhERS provides real-time information including occurrence, affected devices, physical impact, and differentiation between network failure and cyber attack also for unknown event types. The alternative detection strategies either entirely miss this event (see Fig. 13(a) and (b)) or cannot provide any more information than its occurrence (see Fig. 13(c)).

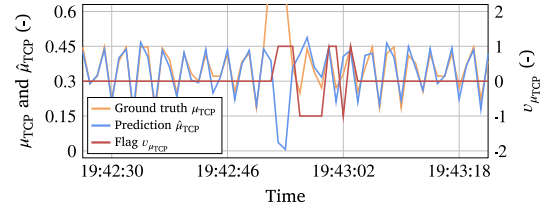


Fig. 15. Anomaly in μ_{TCP} of PLC2 induced by an unlabeled event in S2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.5. Evaluation of attack and fault scenario S3

The third scenario adds scanning attacks to the previously evaluated attack and fault types. The event signatures provided by CyPhERS' Stage 1 during S3 are depicted in Fig. 16.

5.5.1. Unlabeled events

Event 1. S3 is characterized by a fundamental process modification not indicated by Faramondi et al. (2021). Throughout the entire scenario, anomalies are regularly flagged in $v_{H_{\text{T2}}}$, $v_{H_{\text{T3}}}$ and $v_{H_{\text{T4}}}$, as depicted in Fig. 17 for H_{T2}. While the event signature is not known, fulfillment of flag interpretation rules 1 and 2 (see Table 7) points towards a physical failure only affecting the two sub-strings comprising T2-4 (see Fig. 4). For the sake of clarity, $v_{H_{\text{T2}}}$, $v_{H_{\text{T3}}}$ and $v_{H_{\text{T4}}}$ are faded in Fig. 16.

Event 2. CyPhERS' Stage 1 indicates two further unlabeled anomalies in S3, which likely result from the same event. The anomaly in network traffic of PLC1 is depicted in Fig. 18. Shortly after, a consecutive underfill of H_{T6}, H_{T7} and H_{T8} is indicated. Compliance with flag interpretation rules 1 and 6 suggest a network device failure of PLC1 potentially entailing the underfill of T6-T8 due to interrupted communication of H_{T5} values from PLC1 to PLC2.

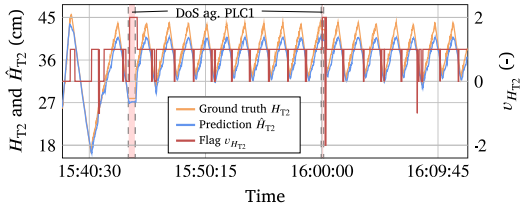


Fig. 17. Unlabeled event of H_{T2}, H_{T3} and H_{T4} throughout S3 on the example of H_{T2}. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

These two examples again showcase how CyPhERS provides real-time information for unknown event types including occurrence, affected devices, physical impact, and differentiation between physical failures, network failures, cyber attacks, and cyber–physical attacks.

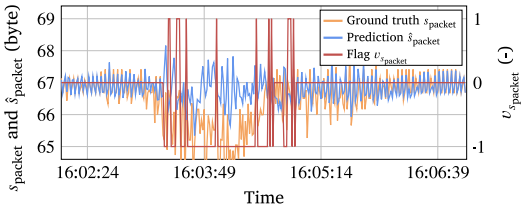


Fig. 18. Unlabeled anomaly in s_{packet} of PLC1 in S3. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.5.2. Scanning attacks

Fig. 16 illustrates that anomaly flags during all scanning attacks follow signature C, which gives insights regarding attack type, victim, attacker location, and impact on the physical process.[5] For example, during the first scan (~15:38), signature C, which includes flagging of communication with an unauthorized external device ($v_{n_{\text{IP/MAC}}} = 1$) containing unusual TCP flags ($v_{n_{\text{TCP}}} = 1$), indicates a scan of PLC1 by an external device not impacting the physical process.

[5] In this context, a wrong label is identified for the last scanning attack.

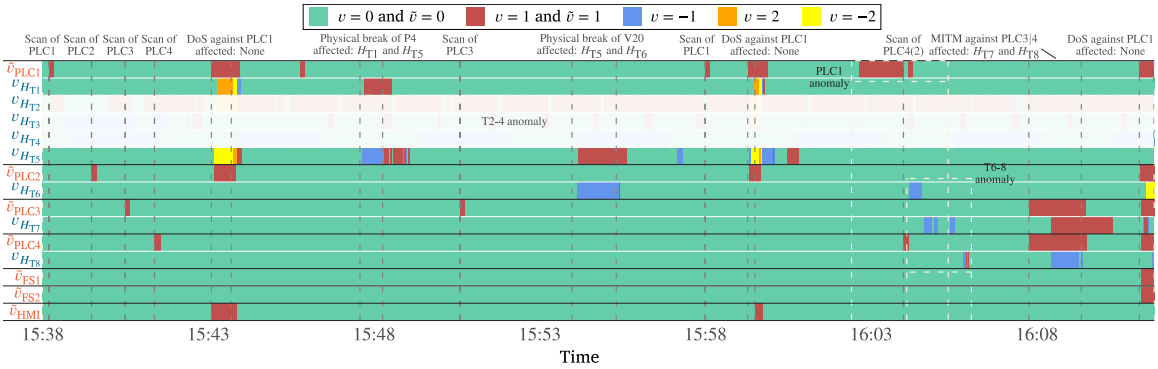


Fig. 16. Event signatures of CyPhERS' Stage 1 during scenario S3. Additional anomalies not labeled by Faramondi et al. (2021) are marked using beige boxes, wrong ground truth labels are given in parenthesis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6. Discussion

This section discusses key findings of the concept demonstration in Section 5. In Section 6.1, it is analyzed whether CyPhERS fulfills its intended purpose. Section 6.2 addresses possible concept improvements. Finally, Section 6.3 reflects on the transferability of the demonstration case results to other CPSs.

6.1. Proof of concept

The aim of CyPhERS is to provide CPSs operators with relevant information on unknown and known types of attacks and faults for real-time incident response, while being independent of historical event observations. The results in Section 5 proof this thesis. All considered attack and fault types are identified, including localization of victim devices and attacker location as well as determination of the impact on the physical process, through matching anomaly flags provided by CyPhERS' Stage 1 with known event signatures. Moreover, information on further unknown events, which are not officially labeled by the authors of the investigated dataset, are provided, including event occurrence, affected components, physical impact, and differentiation between physical failure, cyber attack, cyber–physical attack and network device failure.

6.2. Concept improvements

An open issue identified by the concept demonstration is the recovering phase of the anomaly detection and classification pipelines. Primary reason is the modification of the ground truth model inputs $x^c_{t-w}, \dots, x^c_{t-1}$ in (2) during and through anomalies, which affects the model's capability to predict the normal behavior of a target feature. One approach is to replace $x^c_{t-w}, \dots, x^c_{t-1}$ in (2) with the associated normal behavior predictions $\hat{x}^c_{t-w}, \dots, \hat{x}^c_{t-1}$ during flagged anomalies.

Another improvement is seen in automating the process of creating the event signature database. As of now, CyPhERS requires to define signatures by manually applying anomaly flag interpretation rules (see Table 7) on the specific system at hand. In the future, this process should be automated through an application which generates event signatures by providing it with interpretation rules and specifications about the physical and digital components and architectures of a CPS.

6.3. Result transferability to other CPSs

As pointed out in Section 2.3, process volatility and randomness may complicates application of CyPhERS to some CPSs, including power system applications. Since the study case considered in this work exhibits comparatively simple repeating process patterns, a feasibility demonstration of CyPhERS for systems with pronounced volatility and randomness is required. Moreover, to proof applicability for smaller CPSs without dedicated human operator and for more complex processes where manual recognition of signatures would require a high cognitive effort, the automated signature evaluation in Stage 2 needs to be demonstrated.

7. Conclusion and future work

This work introduces CyPhERS, a cyber–physical event reasoning system that provides real-time information about known and unknown types of attacks and faults in CPSs, independent of historical event observations. CyPhERS uses a two-stage process to infer event information, including occurrence, location, root cause, and physical impact. In Stage 1, informative event signatures are created using methods such as cyber–physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. In Stage 2, the event signatures are evaluated either automatically by matching with a signature database of known events or through manual interpretation by the operator. CyPhERS is demonstrated on a cyber–physical water distribution system, where it successfully identifies various attack and fault types, which includes localization of victim devices and attacker location as well as determination of attack or failure type and impact on the physical process. Additionally, CyPhERS provides information on unknown event types such as occurrence, affected components, physical impact, and differentiation between physical failure, cyber attack, and network failure. Future work will focus on demonstrating CyPhERS for systems with pronounced volatility and randomness under consideration of the automated signature evaluation in Stage 2.

CRediT authorship contribution statement

Nils Müller: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. **Kaibin Bao:** Conceptualization, Methodology, Writing – review & editing. **Jörg Matthes:** Writing – review & editing, Supervision, Funding acquisition. **Kai Heussen:** Conceptualization, Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data used for concept demonstration is available under the following link: https://ieee-dataport.org/open-access/hardware-loop-water-distribution-testbed-wdt-dataset-cyber-physical-security-testing.

Acknowledgments

This work is partly funded by the Innovation Fund Denmark (IFD) under File No. 91363, and by the Helmholtz Association, Germany under the program 'Energy System Design'.

References

Abokifa, A.A., Haddad, K., Lo, C., Biswas, P., 2019. Real-time identification of cyber-physical attacks on water distribution systems via machine learning–based anomaly detection techniques. J. Water Resour. Plan. Manag. 145 (1), http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0001023.

Alguliyev, R., Imamverdiyev, Y., Sukhostat, L., 2018. Cyber-physical systems and their security issues. Comput. Ind. 100, 212–223. http://dx.doi.org/10.1016/j.compind.2018.04.017.

Ayodeji, A., Liu, Y.-k., Chao, N., Yang, L.-q., 2020. A new perspective towards the development of robust data-driven intrusion detection for industrial control systems. Nucl. Eng. Technol. 52 (12), 2687–2698. http://dx.doi.org/10.1016/j.net.2020.05.012.

Barbosa, R.R.R., Sadre, R., Pras, A., 2012. Towards periodicity based anomaly detection in SCADA networks. In: Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation. ETFA 2012, pp. 1–4. http://dx.doi.org/10.1109/ETFA.2012.6489745.

Bezemskij, A., Loukas, G., Anthony, R.J., Gan, D., 2016. Behaviour-based anomaly detection of cyber-physical attacks on a robotic vehicle. In: Proceedings of 2016 15th International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security. IUCC-CSS, pp. 61–68. http://dx.doi.org/10.1109/IUCC-CSS.2016.017.

Bojer, C.S., Meldgaard, J.P., 2021. Kaggle forecasting competitions: An overlooked learning opportunity. Int. J. Forecast. 37 (2), 587–603. http://dx.doi.org/10.1016/j.ijforecast.2020.07.007.

Bou-Harb, E., Debbabi, M., Assi, C., 2014. Cyber scanning: A comprehensive survey. IEEE Commun. Surv. Tutor. 16 (3), 1496–1519. http://dx.doi.org/10.1109/SURV.2013.102913.00020.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Cai, X., Wang, Q., Tang, Y., Zhu, L., 2019. Review of cyber-attacks and defense research on cyber physical power system. In: Proceedings of 2019 IEEE Sustainable Power and Energy Conference. ISPEC, pp. 487–492. http://dx.doi.org/10.1109/iSPEC48194.2019.8975131.

Cao, L., Jiang, X., Zhao, Y., Wang, S., You, D., Xu, X., 2020. A survey of network attacks on cyber-physical systems. IEEE Access 8, 44219–44227. http://dx.doi.org/10.1109/ACCESS.2020.2977423.

Chakraborty, D., Elzarka, H., 2019. Advanced machine learning techniques for building performance simulation: a comparative analysis. J. Build. Perform. Simul. 12 (2), 193–207. http://dx.doi.org/10.1080/19401493.2018.1498538.

Colabianchi, S., Costantino, F., Di Gravio, G., Nonino, F., Patriarca, R., 2021. Discussing resilience in the context of cyber physical systems. Comput. Ind. Eng. 160, http://dx.doi.org/10.1016/j.cie.2021.107534.

Conti, M., Dragoni, N., Lesyk, V., 2016. A survey of man in the middle attacks. IEEE Commun. Surv. Tutor. 18 (3), 2027–2051. http://dx.doi.org/10.1109/COMST.2016.2548426.

Cook, A.A., Mısırlı, G., Fan, Z., 2020. Anomaly detection for IoT time-series data: A survey. IEEE Internet Things J. 7 (7), 6481–6494. http://dx.doi.org/10.1109/JIOT.2019.2958185.

Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., Barbosa, J., 2020. Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. Comput. Ind. 123, http://dx.doi.org/10.1016/j.compind.2020.103298.

Faramondi, L., Flammini, F., Guarino, S., Setola, R., 2021. A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing. IEEE Access 9, 122385–122396. http://dx.doi.org/10.1109/ACCESS.2021.3109465.

Feng, C., Tian, P., 2021. Time series anomaly detection for cyber-physical systems via neural system identification and Bayesian filtering. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. KDD '21, pp. 2858–2867. http://dx.doi.org/10.1145/3447548.3467137.

Frattini, F., Giordano, U., Conti, V., 2019. Facing cyber-physical security threats by PSIM-SIEM integration. In: Proceedings of 15th European Dependable Computing Conference. EDCC, pp. 83–88. http://dx.doi.org/10.1109/EDCC.2019.00026.

Giraldo, J., Urbina, D., Cardenas, A., Valente, J., Faisal, M., Ruths, J., Tippenhauer, N.O., Sandberg, H., Candell, R., 2018. A survey of physics-based attack detection in cyber-physical systems. ACM Comput. Surv. 51 (4), http://dx.doi.org/10.1145/3203245.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Hallac, D., Vare, S., Boyd, S., Leskovec, J., 2017. Toeplitz inverse covariance-based clustering of multivariate time series data. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 215–223. http://dx.doi.org/10.1145/3097983.3098060.

Hasan, M.K., Habib, A.A., Shukur, Z., Ibrahim, F., Islam, S., Razzaque, M.A., 2023. Review on cyber-physical and cyber-security system in smart grid: Standards, protocols, constraints, and recommendations. J. Netw. Comput. Appl. 209, http://dx.doi.org/10.1016/j.jnca.2022.103540.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.

Heartfield, R., Loukas, G., Bezemskij, A., Panaousis, E., 2021. Self-configurable cyber-physical intrusion detection for smart homes using reinforcement learning. IEEE Trans. Inf. Forensics Secur. 16, 1720–1735. http://dx.doi.org/10.1109/TIFS.2020.3042049.

Herzen, J., Lässig, F., Piazzetta, S.G., Neuer, T., Tafti, L., Raille, G., Pottelbergh, T.V., Pasieka, M., Skrodzki, A., Huguenin, N., Dumonal, M., Kościsz, J., Bader, D., Gusset, F., Benheddi, M., Williamson, C., Kosinski, M., Petrik, M., Grosch, G., 2021. Darts: User-friendly modern machine learning for time series. arXiv:2110.03224.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Hsieh, R.-J., Chou, J., Ho, C.-H., 2019. Unsupervised online anomaly detection on multivariate sensing time series data for smart manufacturing. In: 2019 IEEE 12th Conference on Service-Oriented Computing and Applications. SOCA, pp. 90–97. http://dx.doi.org/10.1109/SOCA.2019.00021.

Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T., 2018. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 387–395. http://dx.doi.org/10.1145/3219819.3219845.

Huong, T.T., Bac, T.P., Long, D.M., Luong, T.D., Dan, N.M., Quang, L.A., Cong, L.T., Thang, B.D., Tran, K.P., 2021. Detecting cyberattacks using anomaly detection in industrial control systems: A Federated Learning approach. Comput. Ind. 132, http://dx.doi.org/10.1016/j.compind.2021.103509.

Kang, S., Sristi, S., Karachiwala, J., Hu, Y.-C., 2018. Detection of anomaly in train speed for intelligent railway systems. In: Proceedings of 2018 International Conference on Control, Automation and Diagnosis. ICCAD, pp. 1–6. http://dx.doi.org/10.1109/CADIAG.2018.8751374.

Khoshnevisan, F., Fan, Z., 2019. RSM-GAN: A convolutional recurrent GAN for anomaly detection in contaminated seasonal multivariate time series. arXiv preprint arXiv:1911.07104.

Li, D., Chen, D., Jin, B., Shi, L., Goh, J., Ng, S.-K., 2019a. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In: Proceedings of Artificial Neural Networks and Machine Learning. pp. 703–716. http://dx.doi.org/10.1007/978-3-030-30490-4_56.

Li, F., Yan, X., Xie, Y., Sang, Z., Yuan, X., 2019b. A review of cyber-attack methods in cyber-physical power system. In: Proceedings of 2019 IEEE 8th International Conference on Advanced Power System Automation and Protection. APAP, pp. 1335–1339. http://dx.doi.org/10.1109/APAP47170.2019.9225126.

Lindemann, B., Maschler, B., Sahlab, N., Weyrich, M., 2021. A survey on anomaly detection for technical systems using LSTM networks. Comput. Ind. 131, http://dx.doi.org/10.1016/j.compind.2021.103498.

Luo, Y., Xiao, Y., Cheng, L., Peng, G., Yao, D.D., 2021. Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities. ACM Comput. Surv. 54 (5), http://dx.doi.org/10.1145/3453155.

Maglaras, L.A., Kim, K.-H., Janicke, H., Ferrag, M.A., Rallis, S., Fragkou, P., Maglaras, A., Cruz, T.J., 2018. Cyber security of critical infrastructures. ICT Express 4 (1), 42–45. http://dx.doi.org/10.1016/j.icte.2018.02.001.

Mahjabin, T., Xiao, Y., Sun, G., Jiang, W., 2017. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. Int. J. Distrib. Sens. Netw. 13 (12), http://dx.doi.org/10.1177/1550147717741463.

Müller, N., Ziras, C., Heussen, K., 2022. Assessment of cyber-physical intrusion detection and classification for industrial control systems. In: Proceedings of 2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids. SmartGridComm, pp. 432–438. http://dx.doi.org/10.1109/SmartGridComm52983.2022.9961010.

Navarro, J.M., Rossi, D., 2020. HURRA! human readable router anomaly detection. In: Proceedings of 2020 32nd International Teletraffic Congress. ITC 32, pp. 19–28. http://dx.doi.org/10.1109/ITC3249928.2020.00011.

Nelson, D.M.Q., Pereira, A.C.M., de Oliveira, R.A., 2017. Stock market's price movement prediction with LSTM neural networks. In: Proceedings of 2017 International Joint Conference on Neural Networks. IJCNN, pp. 1419–1426. http://dx.doi.org/10.1109/IJCNN.2017.7966019.

Niu, X., Li, J., Sun, J., Tomsovic, K., 2019. Dynamic detection of false data injection attack in smart grid using deep learning. In: Proceedings of 2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference. ISGT, pp. 1–6. http://dx.doi.org/10.1109/ISGT.2019.8791598.

Scornet, E., Biau, G., Vert, J.-P., 2015. Consistency of random forests. Ann. Statist. 43 (4), 1716–1741. http://dx.doi.org/10.1214/15-AOS1321.

Siami-Namini, S., Tavakoli, N., Siami Namin, A., 2018. A comparison of ARIMA and LSTM in forecasting time series. In: Proceedings of 2018 17th IEEE International Conference on Machine Learning and Applications. ICMLA, pp. 1394–1401. http://dx.doi.org/10.1109/ICMLA.2018.00227.

Song, D., Xia, N., Cheng, W., Chen, H., Tao, D., 2018. Deep r -th root of rank supervised joint binary embedding for multivariate time series retrieval. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 2229–2238. http://dx.doi.org/10.1145/3219819.3220108.

Srivastava, S., Lessmann, S., 2018. A comparative study of LSTM neural networks in forecasting day-ahead global horizontal irradiance with satellite data. Sol. Energy 162, 232–247. http://dx.doi.org/10.1016/j.solener.2018.01.005.

Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., Pei, D., 2019. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 2828–2837. http://dx.doi.org/10.1145/3292500.3330672.

Tuli, S., Casale, G., Jennings, N.R., 2022. TranAD: Deep transformer networks for anomaly detection in multivariate time series data. arXiv preprint arXiv:2201.07284.

Xi, L., Wang, R., Haas, Z.J., 2022. Data-correlation-aware unsupervised deep-learning model for anomaly detection in cyber–physical systems. IEEE Internet Things J. 9 (22), 22410–22421. http://dx.doi.org/10.1109/JIOT.2022.3150048.

Yaacoub, J.-P.A., Salman, O., Noura, H.N., Kaaniche, N., Chehab, A., Malli, M., 2020. Cyber-physical systems security: Limitations, issues and future trends. Microprocess. Microsyst. 77, http://dx.doi.org/10.1016/j.micpro.2020.103201.

Yu, J., Song, Y., Tang, D., Han, D., Dai, J., 2021. Telemetry data-based spacecraft anomaly detection with spatial–temporal generative adversarial networks. IEEE Trans. Instrum. Meas. 70, 1–9. http://dx.doi.org/10.1109/TIM.2021.3073442.

Zhang, J., Pan, L., Han, Q.-L., Chen, C., Wen, S., Xiang, Y., 2022. Deep learning based attack detection for cyber-physical system cybersecurity: A survey. IEEE/CAA J. Autom. Sin. 9 (3), 377–391. http://dx.doi.org/10.1109/JAS.2021.1004261.

Zhang, C., Song, D., Chen, Y., Feng, X., Lumezanu, C., Cheng, W., Ni, J., Zong, B., Chen, H., Chawla, N.V., 2019. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 1409–1416. http://dx.doi.org/10.1609/aaai.v33i01.33011409.

PAPER E

Nils Müller, Kaibin Bao, Kai Heussen

Cyber-physical event reasoning for distributed energy resources on the case of a PV-battery system

Müller, N., K. Bao, and K. Heussen, "Cyber-physical event reasoning for distributed energy resources on the case of a PV-battery system," under review at *Sustainable Energy, Grids and Networks*.

Cyber-physical event reasoning for distributed energy resources on the case of a PV-battery system

Nils Müller[a,*], Kaibin Bao[b], Kai Heussen[a]

[a] *Wind and Energy Systems Department, Technical University of Denmark, Building 330, Risø campus, 4000 Roskilde, Denmark*
[b] *Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, Building 445, Campus North, 76344 Eggenstein-Leopoldshafen, Germany*

Abstract

The widespread adoption of internet-connected and remotely controllable solar plants, battery storages, and other electric devices renders coordinated cyber-physical attacks against such distributed energy resources (DERs) an emerging risk for power systems. Effective incident response can be facilitated by online DER monitoring providing real-time information on event root causes and physical impacts. Such online event identification is challenged by the lack of historical attack observations, and emergence of new attack strategies. The cyber-physical event reasoning system CyPhERS provides real-time information on both known and unknown attack types in form of informative and interpretable event signatures, without need for historical attack samples. However, CyPhERS has been demonstrated on a generic cyber-physical laboratory testbed, considering human evaluation of event signatures. This work extends applicability of CyPhERS to DER monitoring on the case of a real PV-battery system affected by several cyber and cyber-physical attack types. In this context, an automated signature evaluation system is realized, and the concept adapted to specificities of DERs, such as weather and consumer-induced volatility. The results demonstrate that CyPhERS can be applied for online DER monitoring, providing information on root causes and physical impacts of both known attack strategies and unknown event types in an automated fashion.

Keywords: Distributed energy resources, PV-battery systems, Attack detection, Cyber-physical monitoring, Machine learning

1. Introduction

The transformation towards widespread use of sustainable energy sources is driven by decentralization and electrification. Both the replacement of centralized fossil power plants with renewable generation, as well as the electrification of the mobility and heating sectors are boosting the deployment of distributed energy resources (DERs) such as rooftop solar panels, electric vehicles, batteries and heat pumps. The large-scale adoption of DERs provides benefits beyond decarbonizing energy consumption, including lower transmission costs and improved grid stability through provision of ancillary services [1]. Harnessing this potential requires integration with information and communication technology (ICT) for continuous coordination and management of numerous geographically distributed devices. However, the associated connection to public networks and remote control capability, combined with often low security standards [2], render DERs promising targets for cyber criminals. Incidents such as the Mirai botnet attack have demonstrated how a fleet of internet of things (IoT) devices can be simultaneously seized [3]. Malicious control of multiple DERs can provoke grid instability by switching the devices simultaneously on or off, rendering coordinated attacks on DERs a serious threat for power system operation [4]. In this context, the increasing number of attacks on critical infrastructure demonstrates the need to support the large-scale deployment of DERs with adequate security mechanisms [5, 6].

Attack detection is among the most frequently suggested security measures for DERs [5, 7]. Once an attack is detected and identified, affected systems and network zones can be isolated, and incident response mechanisms activated. In the light of cyber-physical attacks, timely and appropriate counteractions require real-time information on both root causes and physical impact. Most existing detection concepts exclusively monitor either cyber network or physical process data [7]. While cyber network attack detection potentially allows to distinguish several attack types, physical impacts are not identified. In contrast, physical attack detection can determine the attack impact, but not the underlying attack vector. Consequently, some works propose the combined evaluation of OT network traffic and process data [7], and demonstrate the superior performance of such cyber-physical attack identification applying supervised machine learning (ML) [8]. However, due to the dependency on historical samples of typically rarely occurring attacks, supervised methods lack practical relevance [9, 10]. In [11], the authors propose CyPhERS, a cyber-physical event reasoning system which exploits advantages of cyber-physical monitoring while being independent of attack observations. However, CyPhERS was demonstrated on a laboratory testbed exhibiting simple repeating process patterns. Moreover, the demonstrated version of the concept requires active involvement of human operators. In the context of DER monitoring, the problem is more complicated due to the pronounced volatility and randomness resulting from dependency on weather and consumer behavior [10]. Moreover, especially for small scale DERs, active operator participation is impractical. Thus, this work addresses the following research question: *How can real-time information about cyber attacks against DERs such as occurrence, type, victim devices, attacker location, and physical impact be provided*

*Corresponding author. E-mail address: nilmu@dtu.dk (N. Müller).

in an automated fashion, while being independent of historical attack observations?

For this purpose, the present study adapts CyPhERS to the behavioral and functional specificities of DERs, and demonstrates it on data of a real photovoltaic (PV)-battery system targeted by a various cyber and cyber-physical attack types.

1.1. Related work

The literature on attack detection and identification methods for DERs is rich as existing reviews suggest [9, 10, 12]. These can be broadly divided into methods monitoring the cyber network, physical process or both.

1.1.1. Physical attack detection and identification

Many works propose attack detection applying physics-based models [13–15]. The models are used to emulate a DER under normal condition. By evaluating the residual between the model and actual measurements against a threshold, attacks can be detected [16]. An advantage of this approach is the independency from attack samples [9]. However, accurate modeling might be challenging for DERs with complex architectures (e.g., virtual power plants) leading to imprecise detection. Moreover, the consideration of a binary detection problem (*normal* vs. *attack*) neglects insights on root causes and impacts. Data-driven methods constitute another widely considered approach for physical attack detection and identification [7]. One argument is generalizability as process representations are automatically learned from data, avoiding expensive manual model development. The majority of works considers supervised approaches such as multi-class classification [10, 17–19]. The explicit learning from attack samples, on the one hand, allows them to detect and differentiate various attack types. On the other hand, it renders them impractical due to the natural scarcity of such data. Other works apply regression models to learn the normal behavior of a DER, and detect attacks by comparing the predictions to the actual measurements [20]. Similar to the approaches based on physical models, the consideration of a binary detection problem makes them of limited use for incident response.

1.1.2. Cyber network attack detection and identification

Among the classical approaches for monitoring DER network data are signature-based intrusion detection systems applying tools such as Snort [21] or Suricata. These can detect and differentiate attacks in case of known attack signatures. Related but newer approaches include supervised detection of attack patterns, for example, firmware modifications in inverter-based microgrids [22]. Neither the traditional signature-based nor the newer supervised ML-based methods can detect new attack strategies. Furthermore, they provide no information about the physical impact of an attack on the operation of DER. Another approach is detection of anomalies in network traffic, known as behavior-based intrusion detection [7]. In the recent years, an increasing focus is on ML-based normal behavior reference models, which are compared to actual traffic, allowing to detect anomalies [23]. Although such approaches can potentially detect both known and unknown types of attacks, they do not provide any information other than the occurrence of abnormal network behaviour.

1.1.3. Cyber-physical attack detection and identification

As exclusive monitoring of a DER's cyber or physical domain neither can identify both attack root causes and impact nor differentiate cyber-physical attacks from process failures, many works suggest investigation of cyber-physical detection [7, 8, 24]. Nevertheless, literature on the combined evaluation of process and network data of DERs or other power system applications is rare. The authors of [25] propose joint evaluation of synchrophasor measurements and properties of network traffic applying a multi-class decision tree classifier. In [26], unsupervised anomaly detection is applied to both network traffic and physical process features of a DER. A comparison of cyber, physical and cyber-physical detection in power systems is conducted in [27] by applying both supervised and unsupervised methods for the binary detection problem (*normal* vs. *attack*). The listed works all indicate that the joint monitoring of cyber and physical DER data improves detection performance. However, none of them combines root cause and impact identification with independence of historical attack samples. The authors in [11] propose the cyber-physical event reasoning system CyPhERS (see Fig. 1) to close this gap. CyPhERS utilizes a two-stage process to deduce event information, including the occurrence, type, location, and physical impact, from joint processing of network traffic and physical process data in real-time. The first stage generates informative event signatures for both unknown and known types of cyber attacks and physical faults. This is achieved through a combination of several methods including cyber-physical data fusion, unsupervised multivariate time series anomaly detection, and anomaly type differentiation. In the second stage, the event signatures are evaluated either through automated matching with a signature database of known events or through manual interpretation by the operator. Although the authors claim that the evaluation of event signatures can be automated, only manual interpretation is demonstrated to date. Moreover, the concept demonstration is conducted on a water distribution system. Thus, feasibility for DERs monitoring first needs to be demonstrated.

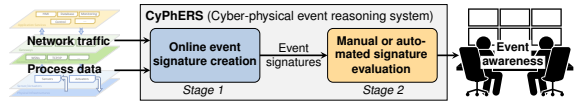


Figure 1: Schematic representation of CyPhERS adapted from [11].

1.2. Contribution and paper structure

The main contributions of this work are as follows:

- Methodological adaptation of CyPhERS to DER monitoring including probabilistic time series modeling and anomaly detection.
- Realization of the automated event signature identification in CyPhERS' Stage 2.
- Feasibility demonstration of applying CyPhERS for automated real-time identification of cyber(-physical) attacks against DERs on the case of a real PV-battery system targeted by several attack types.

The remainder of the paper is structured as follows: In Section 2, CyPhERS is conceptually introduced. Section 3 presents the considered PV-battery system study case. The methodology of CyPhERS and its adaption and implementation to the investigated DER case is detailed in Section 4. In Section 5, results of applying CyPhERS on the PV-battery system study case are presented. Finally, key findings of the demonstration are discussed in Section 6, followed by a conclusion in Section 7.

2. Introduction of the CyPhERS concept

This section summarizes CyPhERS, which was introduced in [11]. The online event signature creation (Stage 1) is explained in Section 2.1. Thereafter, Section 2.2 provides details on the signature evaluation (Stage 2).

2.1. Online event signature creation (Stage 1)

CyPhERS' Stage 1 (see Fig. 2) employs a range of methods to produce real-time event signatures for known and unknown types of attacks and failures. The signatures encompass information including event occurrence, type, location, and physical impact. The applied methods are introduced in the following.

2.1.1. Cyber-physical information

Stage 1 jointly evaluates physical process and cyber network data to detect and differentiate cyber attacks and physical failures. In context of cyber-physical attacks, this further allows to determine whether physical impact already happened or detect the attack at an early stage in network traffic to mitigate damage.

2.1.2. Feature-level monitoring

The second strategy is the individual monitoring of multiple system variables, covering both variables of multiple devices and multiple variables of the same device. While the former allows to localize affected devices or sub-systems of a DER, the latter further specifies abnormal behavior of the concerned device. The monitored variables are derived from sensor readings and OT network traffic, and in the following denoted *target features*, where $\mathcal{I}$ and $\mathcal{J}$ represent the physical and network feature subset, respectively. For a target feature c, its time series is given as $X_c = \{x_1^c, x_2^c, ..., x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$. The extraction of target features in case of the investigated PV-battery system is addressed in Section 4.1.

2.1.3. Unsupervised time series anomaly detection using covariates

Attacks and faults are detected by Stage 1 through unsupervised time series anomaly detection. First, a normal behavior reference model is derived for each target feature. Their predictions are then compared to actual observations to detect abnormal behavior of a DER. The key argument for: the independence of historical event observations, which enables the detection of both known and unknown event types. The benefit of monitoring target features as time series is the detection of deviations from normal behavior which are only abnormal under a specific temporal context (local anomalies) [28]. Additionally, covariates are used to provide the models with further DER internal or external information, allowing detection of situational anomalies which are only abnormal in the context of the provided covariates (e.g., detecting abnormal PV feed in context of irradiation). A covariate time series associated with a target feature c is formally denoted $Z_c = \{z_1^c, z_2^c, ..., z_N^c \mid z_i^c \in \mathbb{R}\ \forall i\}$ in the following. A description of target features used for the given study case is provided in Section 4.1.

2.1.4. Differentiation of anomaly types

The fourth element of Stage 1 pertains to the differentiation of multiple anomaly types. Once an anomaly is flagged for a target feature c, it is further classified using characteristics such as the direction of the deviation (e.g., abnormally *low* PV feed). This differentiation of anomaly types facilitates identification of attack root causes and physical impacts. The series of flags produced by the signature extraction system for a target feature c is represented as $v_c = \{v_1^c, v_2^c, ..., v_N^c \mid v_i^c \in \{-2, -1, 0, 1, 2\}\ \forall i\}$. An explanation of the signature extraction system, including anomaly types, and its adaption and implementation to the evaluated PV-battery system follows in Section 4.2.

2.1.5. Event signature visualization

Covering multiple domains, system variables, and anomaly types enables Stage 1 to provide dense event information in

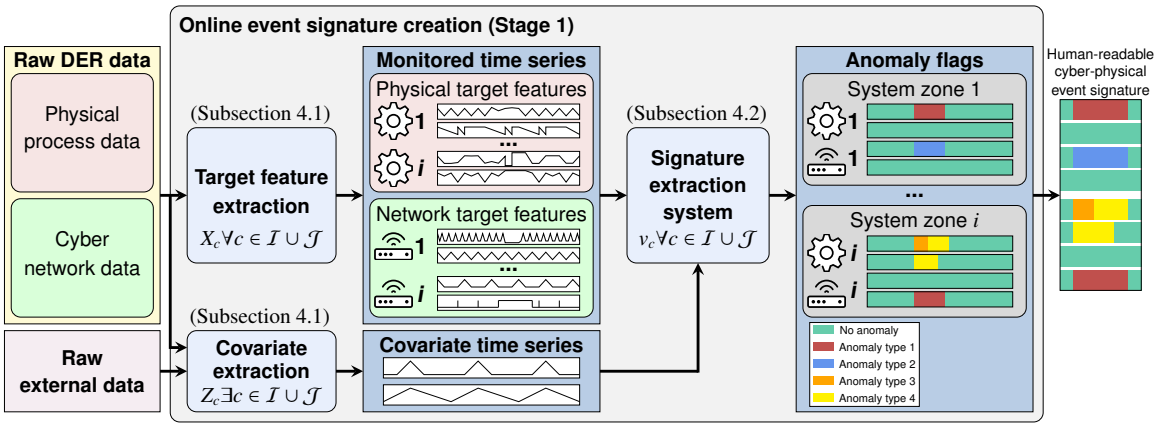


Figure 2: Schematic overview of CyPhERS' online event signature creation (Stage 1) based on [11].

form of anomaly flag series of a set of target features. These flag series are re-organized by grouping them for each system zone of a DER to ease information extraction. A system zone is defined as a collection of physical components and network devices that are directly linked (e.g., a battery stack and the associated battery inverter). Their logical relation facilitates relating anomaly flags of different target features. Consequently, Stage 1 of CyPhERS generates event signatures that are both rich in information and can be read and interpreted by humans.

2.2. Signature evaluation (Stage 2)

Fig. 3 illustrates the concept of CyPhERS' Stage 2. Stage 2 involves the evaluation of event signatures from Stage 1, which can be done by human operators or automated evaluation systems. The signatures are specific to event types and distinguishable. For known attacks or faults, they can be pre-defined and stored in a database. After Stage 1 identifies an event, its signature can be compared to the database. If a match is found, information such as the type of event, the affected component, the root cause, and/or the physical impact can be retrieved to form a hypothesis about the event. Matching of signatures can be done by visual comparison or an automated evaluation system. One automation approach is the transformation of a signature into a set of rules in the following form: ***flagging of*** *anomaly type 1* ***in*** *target feature X* ***and*** *type 2* ***in*** *target feature Y* ***indicates*** *device A* ***being targeted by*** *attack type B* ***causing*** *physical impact C*. Signatures can also be defined for unknown event types based on partial knowledge, for example, ***flagging of*** *anomaly type 1* ***in*** *target feature X* ***indicates*** *device A failure [type unknown]* ***causing*** *physical impact B*.

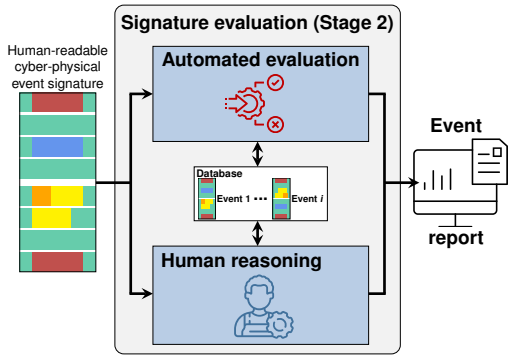


Figure 3: Schematic overview of CyPhERS' signature evaluation (Stage 2) taken from [11].

In case that Stage 1 provides a signature which is unknown or undefined, event information cannot be inferred in an automated fashion. However, human operators may still derive information based on process expertise. At a minimum, CyPhERS informs about the occurrence of abnormal system behavior in any event case.

3. Demonstration case description

This section introduces the investigated demonstration case. The PV-battery system is detailed in Section 3.1. Thereafter, Section 3.2 describes the threat model. Finally, Section 3.3 provides details on the conducted experiments and resulting dataset.

3.1. PV-battery system

The cyber-physical structure of the investigated PV-battery system is schematically depicted in Fig. 4. The system is located at the Karlsruhe Institute of Technology (KIT), Germany. The system comprises four PV inverters, each with four dedicated solar panel string (PV1-4), four battery stacks with associated battery inverters (BAT1-4), four energy meters (M1-4), a data manager (DM), and a data server (DS). Each battery stack has a capacity of 10.24 kWh and a maximum dis-/charge power of 5 kW. The connected solar panel strings of each inverter have a peak power between 15.50 kW_p and 16.74 kW_p. While PV1-4 are connected to all phases (L1-L3), BAT1-4 are linked to individual lines (see Fig. 4). The three phases are measured individually by M1-3. In addition, M4 measures all three phases and thus provides measurement redundancy. The PV and battery inverters are connected to the same grid connection point (GCP) as the building load. The load is characterized by typical office building patterns (higher load during working hours, lower on weekends). An additional characteristic are periodic load peaks due to activation of an air compressor. The control objective of the system is to minimize active power exchange with the grid. Consequently, batteries are charged if PV production exceeds the load, and discharged in the opposite case, provided an appropriate state of charge (SOC). As BAT1-4 are connected to individual phases, power flows P^{L1}-P^{L3} are controlled separately by a dedicated battery. An exception is P^{L1}, which is connected to BAT1 and BAT4. In case that batteries reach their maximum dis-/charging power, the offset on the respective phase is compensated by the other batteries on their phase, which is coordinated by communication among BAT1-4. The battery controllers receive the required measurements of P^{L1}-P^{L3} through subscription to the multicast of the respective energy meter (M1, M2 or M3). The DM collects measurements from solar panels and batteries such as panel and cell temperatures. The DS hosts a custom data visualization software for users and builds the interface to the external network.

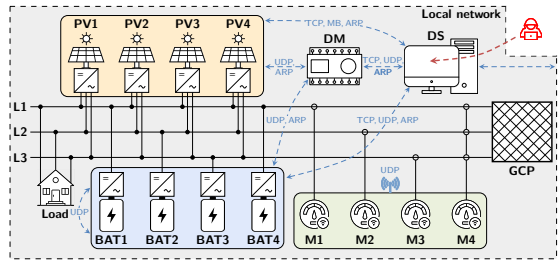


Figure 4: Schematic representation of the PV-battery system.

3.2. Threat model

In the investigated threat scenario, it is assumed that the attacker gained virtual access to the local network by hijacking the DS. From there she/he launches several cyber and cyber-physical attacks targeting different devices of the PV battery

system. The considered attack types are among the most relevant ones for DERs [10, 29, 30]. The cyber attacks comprise SYN scans and HTTPS requests, falling under the category of reconnaissance activities, as well as ARP spoofing used for eavesdropping, which belongs to data collection activities [31]. Among the cyber-physical attacks are false data injection attacks (FDIAs), false command injection attacks (FCIAs), and replay attacks. In case of the FDIAs, false active power readings are injected in the name of the respective meter, causing an abrupt dis-/charging process of the batteries. The FCIAs comprise shut-down of either PV or battery inverters. For the replay attacks, the attacker repeats valid active power readings of the energy meters, which multiplies the control error and thus results in oscillation of the batteries. In the context of a coordinated manipulation of multiple DERs, the considered cyber-physical attacks could be part of either a static (FDIAs and FCIAs) or dynamic (replay attack) load altering attack against power systems [32].

3.3. Dataset

The experiments have been conducted in October 2022. After recording the system under normal operation for approximately two weeks, 15 attacks were launched within one day between 10:00 and 17:00 local time (see Table 1).

Table 1: Schedule of the attack experiments.

No.	Attack type	Victim	Start	End
1	FDIA	M3	10:01	10:17
2	ARP spoof	PV3/DM	10:31	10:48
3	HTTPS request	BAT1	11:06	11:06
4	SYN Scan	PV3	11:22	11:34
5	FCIA	PV2	11:47	12:01
6	FDIA	M1	12:14	12:32
7	HTTPS request	DM	12:47	12:47
8	Replay attack	M1	13:05	13:07
9	FCIA	BAT3	13:26	13:39
10	FCIA	PV1	13:56	14:09
11	ARP spoof	PV4/DM	14:30	14:44
12	FCIA	BAT4	15:00	15:19
13	FDIA	M2	15:39	15:48
14	SYN Scan	PV4	16:04	16:08
15	Replay attack	M2	16:20	16:23

The data was recorded using port mirroring (SPAN) in form of a passive packet capture of the local network. Both physical process and network traffic features are extracted from the resulting pcap file. The set of considered raw features is listed in Table 2. The physical data have a resolution between one second and one minute, depending on the respective feature. Network data on average comprise 7539 packets per minute.

4. Adaption and implementation of the CyPhERS methodology

This section details the methodology of CyPhERS, and its adaption and implementation to the considered PV-battery system. First, the online event signature creation (Stage 1) is addressed, which includes target feature and covariate extraction (Section 4.1), as well as the signature extraction system (Section 4.2). Thereafter, the implementation of the signature evaluation (Stage 2) is explained in Section 4.3.

Table 2: Raw network and process features considered in the study case.

No.	Physical features	No.	Network features
1	Timestamp	1	Timestamp
2	Solar irradiation Ir	2-3	IP (source & destination)
3-14	Act. power $P^{\text{PV1-4}}$, $P^{\text{BAT1-4}}$, $P^{\text{M1-4}}$	4-5	MAC (source & destination)
15-18	Battery state of charge $SOC^{\text{BAT1-4}}$	6	Protocol
19-22	Battery voltage $V^{\text{BAT1-4}}$	7	TCP flags
23-26	Battery temperature $T^{\text{BAT1-4}}$	8	Modbus (MB) function code

4.1. Target feature and covariate extraction

In this section, the extraction of target features and covariates from raw data of the considered DER is presented. While Section 4.1.1 is concerned with physical process features, Section 4.1.2 addresses network traffic features.

4.1.1. Physical process features

According to [11], physical target features are monitored to identify both true physical events and manipulations of process-relevant data. The former requires features which represent the operation of all physical components of the system in question in order to localize the affected ones, and derive the physical impact on them. The latter necessitates monitoring of sensor readings used for process control. Therefore, this work considers physical target features which i) represent the physical operation of PV1-4, BAT1-4, and M1-3, and ii) monitor the multicasted active power readings required for controlling the batteries. A specificity of DERs is that attacks can directly target the functionality of a component (e.g., switch battery off) or exploit the normal functionality to achieve an abnormal behavior (e.g., control battery to create load oscillation). To adapt CyPhERS to DER monitoring, target features should monitor both the technical functionality and behavior of components. Whether a target feature is functional or behavioral depends on the model inputs (target feature lags and covariates). More precisely, even a model of the same feature can represent either the functionality or behavior depending on the input, as demonstrated in Fig. 5. In the depicted example, the attacker creates a sudden change of the PV feed to launch a load altering attack. Instead of directly damaging the plant, the attacker uses the normal functionality (reduced feed if less sun) to launch the attack. The functionality model uses the local Ir measurement as input and predicts the expected feed reduction. Thus, no anomaly is flagged. The behavior model uses an external Ir measurement

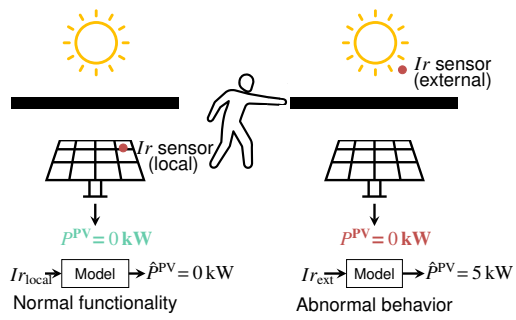


Figure 5: Comparison of functionality- and behavior-describing target features on the example of a physical "attack" where PV panels are covered.

Table 3: Overview of extracted physical target features.

Target feature	Model input	Type	Description
$P^{\mathrm{PV}i}_{\mathrm{fmean}}$	Ir (local)	Functional	For every 60s time step τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{PV}i}$ within τ_{60} according to $P^{\mathrm{PV}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{PV}i}_p / N_{\tau_{60}}$. Anomalies in $P^{\mathrm{PV}i}_{\mathrm{fmean}}$ can indicate disfunction of the i-th solar panel string or its inverter.
$P^{\mathrm{BAT}i}_{\mathrm{fmean}}$	$P^{\mathrm{M}i}_{\mathrm{mean}}$, $SOC^{\mathrm{BAT}i}_{\mathrm{mean}}$, $V^{\mathrm{BAT}i}_{\mathrm{mean}}$, $T^{\mathrm{BAT}i}_{\mathrm{mean}}$	Functional	For every τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{BAT}i}$ within τ_{60} according to $P^{\mathrm{BAT}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{BAT}i}_p / N_{\tau_{60}}$. Anomalies in $P^{\mathrm{BAT}i}_{\mathrm{fmean}}$ can indicate a disfunction of the i-th battery stack or its associated inverter.
$P^{\mathrm{M1\text{-}3}}_{\mathrm{fmean}}$	Redundant measurement of M4	Functional	For every 5s time step τ_5 the mean value is determined as average over the N_{τ_5} data packets carrying $P^{\mathrm{M}i}$ within τ_5 according to $P^{\mathrm{M}i}_{\mathrm{mean},\tau_5} = \sum_{p=1}^{N_{\tau_5}} P^{\mathrm{M}i}_p / N_{\tau_5}$. Anomalies can indicate a disfunction of the i-th meter.
$P^{\mathrm{BAT}i}_{\mathrm{osc}}$	$P^{\mathrm{BAT}i}_{\mathrm{osc}}$ lag values	Behavioral	For every 15s time step τ_{15} the absolute sum of power changes is calculated according to $P^{\mathrm{BAT}i}_{\mathrm{osc},\tau_{15}} = \sum_{f=0s}^{15s} \lvert P^{\mathrm{BAT}i}_{\mathrm{mean},\bar{\tau}+f+1s} - P^{\mathrm{BAT}i}_{\mathrm{mean},\bar{\tau}+f} \rvert$, where $\bar{\tau}$ is the start time of τ_{15}. Anomalies in $P^{\mathrm{BAT}i}_{\mathrm{osc}}$ can indicate oscillation of the i-th battery.
$S^{\mathrm{BAT}i}$	$P^{\mathrm{PV1\text{-}4}}_{\mathrm{mean}}$, time of day	Behavioral	For every τ_{60} the on-off state ($S^{\mathrm{BAT}i}_{\tau_{60}} \in [0, 1]$) is determined. Anomalies may indicate that BATi is unexpectedly online/offline given the current time of day and PV feed.
$P^{\mathrm{BAT}i}_{\mathrm{bmean}}$	$P^{\mathrm{PV1\text{-}4}}_{\mathrm{mean}}$, time of day	Behavioral	For every τ_{60} the mean value is determined as average over the $N_{\tau_{60}}$ data packets carrying $P^{\mathrm{BAT}i}$ within τ_{60} according to $P^{\mathrm{BAT}i}_{\mathrm{mean},\tau_{60}} = \sum_{p=1}^{N_{\tau_{60}}} P^{\mathrm{BAT}i}_p / N_{\tau_{60}}$. Anomalies can indicate abnormal behavior of BATi given the current time of day and PV feed.
$\lvert P^{\mathrm{M1\text{-}3}}_{\mathrm{sum}} \rvert$	$P^{\mathrm{M}i}_{\mathrm{mean}}$	Behavioral	For every τ_5 the absolute sum is determined as $\lvert P^{\mathrm{M}i}_{\mathrm{sum},\tau_5} \rvert = \sum_{p=1}^{N_{\tau_5}} P^{\mathrm{M}i}_p$. Anomalies can indicate abnormal behavior of the i-th meter given the current $P^{\mathrm{M}i}_{\mathrm{mean}}$ (sending abnormally many or few packets carrying $P^{\mathrm{M}i}$).

which results in a deviation between the predicted and actual feed. As a result, an anomaly is flagged. In case of only monitoring the functionality, the attack impact would not be detected. By solely monitoring the behavior, it could not be determined whether the anomaly stems from device disfunction or misuse.

Table 3 lists the extracted target features and associated model inputs. The average active load P_{fmean} is selected as target feature representing the functionality of PV1-4, BAT1-4, and M1-3. In these cases, model inputs exclusively stem from the component's variables and immediate inputs, or data of a redundant device. Three features are included which represent the behavior of the batteries. $P^{\mathrm{BAT}i}_{\mathrm{bmean}}$ is the average active load of the i-th battery modeled based on the time of the day and PV feed. Thus, anomalies in $P^{\mathrm{BAT}i}_{\mathrm{bmean}}$ indicate abnormal battery behavior given the current time and PV feed. Battery operation is influenced by weather and consumer behavior, thus exhibiting pronounced volatility and randomness. For such cases, the authors of CyPhERS suggest extracting additional features that break down a component's behavior to simpler abstractions such as the on/off state. $P^{\mathrm{BAT}i}_{\mathrm{osc}}$ describes how much power changes the i-th battery conducts in a 15s period. An unusual high value can be an indicator for abnormal load oscillation. Finally, $S^{\mathrm{BAT}i}$ describes the on-off state of a battery modeled under use of the current time of the day and PV feed. This target feature is supposed to indicate whether a battery is activated during times and PV feeds where it usually is deactivated and vice versa. In addition, the absolute sum of the i-th meters active load readings $|P^{\mathrm{Mi}}_{\mathrm{sum}}|$ modeled based on $P^{\mathrm{M}i}_{\mathrm{fmean}}$ is considered. As the meters multicast on constant frequency, a sum/mean-mismatch can be a sign for abnormally many or few multicasts, potentially indicating misuse.

Note that also data manipulations within the physical target features are detected. Differentiation to real physical events is facilitated by parallel network traffic monitoring. An example follows in Section 5.

4.1.2. Network traffic features

Following the descriptions in [11], the purpose of traffic monitoring is twofold: 1) localization of compromised network devices, and 2) determination of attack types. To achieve the former, the traffic of each network device should be monitored individually. The latter necessitates extracting several informative features per device. Consequently, this work considers multiple network features for the PV and battery inverters, energy meters, DM, and DS (see Table 4). The features comprise counts of packets with specific protocols, TCP flags and modbus function codes for periods of 15s. Within the local network of the considered PV-battery system, a variety of protocols are used, which enable certain functionalities, such as communicating process-relevant data (UDP packets from energy meters) or sending control commands (MB packets to battery inverters). Thus, unusual deviations in the packet count of certain protocols can point to specific attack types. Abnormal numbers of packets with SYN flags can be an indicator for adverse connection attempts and is thus taken into account. Finally, packets with function code 4 (read registers) and 16 (write registers) are counted. In particular, abnormal high numbers of packets with function code 16 can be a sign for adverse control commands. In all cases, network target features are modelled using lag values and the time of day as model input. Time of day is an important covariate in OT networks, as certain processes often are regularly conducted at specific times, e.g. every full hour.

Table 4: Overview of extracted network target features. Indices s and d refer to source and destination, respectively. Packets are counted for 15s periods.

Target feature	Description
$n^{\mathrm{PV}i}_{\mathrm{UDP_d}}, n^{\mathrm{BAT}i}_{\mathrm{UDP_d}}, n^{\mathrm{DM}}_{\mathrm{UDP_d}}, n^{\mathrm{DS}}_{\mathrm{UDP_d}}, n^{\mathrm{M}i}_{\mathrm{UDP_s}}, n^{\mathrm{DS}}_{\mathrm{UDP_s}}$	UDP packets send to/from the device
$n^{\mathrm{PV}i}_{\mathrm{TCP_d}}, n^{\mathrm{BAT}i}_{\mathrm{TCP_d}}, n^{\mathrm{DM}}_{\mathrm{TCP_d}}, n^{\mathrm{DS}}_{\mathrm{TCP_d}}, n^{\mathrm{DS}}_{\mathrm{TCP_s}}$	TCP packets send to/from the device
$n^{\mathrm{PV}i}_{\mathrm{MB_d}}, n^{\mathrm{BAT}i}_{\mathrm{MB_d}}, n^{\mathrm{DM}}_{\mathrm{MB_d}}, n^{\mathrm{DS}}_{\mathrm{MB_d}}, n^{\mathrm{DS}}_{\mathrm{MB_s}}$	MB packets send to/from the device
$n^{\mathrm{PV}i}_{\mathrm{ARP_d}}, n^{\mathrm{BAT}i}_{\mathrm{ARP_d}}, n^{\mathrm{DM}}_{\mathrm{ARP_d}}, n^{\mathrm{DS}}_{\mathrm{ARP_d}}, n^{\mathrm{DS}}_{\mathrm{ARP_s}}$	ARP packets send to/from the device
$n^{\mathrm{PV}i}_{\mathrm{TLS_d}}, n^{\mathrm{BAT}i}_{\mathrm{TLS_d}}, n^{\mathrm{DM}}_{\mathrm{TLS_d}}, n^{\mathrm{DS}}_{\mathrm{TLS_d}}, n^{\mathrm{DS}}_{\mathrm{TLS_s}}$	TLS packets send to/from the device
$n^{\mathrm{PV}i}_{\mathrm{SYN_d}}, n^{\mathrm{BAT}i}_{\mathrm{SYN_d}}, n^{\mathrm{DM}}_{\mathrm{SYN_d}}, n^{\mathrm{DS}}_{\mathrm{SYN_d}}, n^{\mathrm{DS}}_{\mathrm{SYN_s}}$	Packets with SYN flag send to/from the device
$n^{\mathrm{PV}i}_{16_{\mathrm{d}}}, n^{\mathrm{BAT}i}_{16_{\mathrm{d}}}$	Packets with write register code send to the device
$n^{\mathrm{PV}i}_{4_{\mathrm{d}}}, n^{\mathrm{BAT}i}_{4_{\mathrm{d}}}$	Packets with read register code send to the device

Note that a broad spectrum of further useful features can be extracted. For example, the number of ports used within a certain period would provide information to identify port scans. Other features could be derived, among others, from IP/MAC addresses, packet sizes or checksums. However, for the sake of comprehensibility of the results in Section 5, only features which are considered most relevant are taken into account. Due to the same reason, only packets sent *to* a device are taken into consideration in most cases. Exceptions are M1-3 and the DS. As the meters M1-3 multicast process-relevant data, monitoring UDP packets send *from* them is of importance. Moreover, as the DS is available for users and thus directly connected to the external network, it constitutes a likely target for attackers. Thus, both packets send to and from the DS are monitored.

4.2. Signature extraction system

The signature extraction system generates the anomaly flags for the set of target features that eventually form the event signatures (see Fig. 2). The authors of CyPhERS argue for using individual models per target feature. Among the reasons is the independent selection of covariates, which is particularly relevant in context of the previously introduced differentiation between functionality- and behavior-describing target features. Consequently, the signature extraction system comprises a set of individual anomaly detection and classification pipelines.

The methodology of the anomaly detection and classification pipelines, and their adaption to DER monitoring are explained in Section 4.2.1. After that, Section 4.2.2 addresses the time series models which are used within the pipelines. Finally, the procedure for automated implementation of the signature extraction system and its realization for the given case is detailed in Section 4.2.3.

4.2.1. Anomaly detection and classification pipelines

As explained in [11], a pipeline comprises a target feature model and consecutive anomaly detector (see Fig. 6). While the model predicts the normal behavior of the respective target feature, the detector compares the predictions with the ground truth observations to decide whether to flag an anomaly or not. In [11], both predictions and the detector's decision function are deterministic. Due to the weather- and consumer-induced randomness and variability of DER operation, modeling of some features might be subject to pronounced uncertainties, rendering anomaly detection more challenging. Thus, for the adaptation of CyPhERS to monitoring of DERs, probabilistic time series predictions and decision functions are considered. For that purpose, the lower quantile q_L, median q_M, and upper quantile q_U are predicted for each target feature. Given $X_c = \{x_1^c, x_2^c, ..., x_N^c \mid x_i^c \in \mathbb{R}\ \forall i\}$ and $Z_{c,1}, ..., Z_{c,n} = \{\{z_{1,1}^c, z_{1,2}^c, ..., z_{1,N}^c\}, ..., \{z_{n,1}^c, z_{n,2}^c, ..., z_{n,N}^c\} | z_{j,i}^c \in \mathbb{R} \forall (j,i)\}$ of a target feature c, the expected quantile $\hat{x}_t^{q,c}$ at time t is predicted using lag values $x_{t-w}^c, ..., x_{t-1}^c$ and covariates $z_{1,t}^c, ..., z_{n,t}^c$ according to

$$\hat{x}_t^{q,c} = \Phi\left([x_{t-w}^c, ..., x_{t-1}^c], [z_{1,t}^c, ..., z_{n,t}^c]\right), \forall q \in \{q_\mathrm{L}, q_\mathrm{M}, q_\mathrm{U}\}, \quad (1)$$

where n is the number of covariates, and w the length of the history window. Note that, depending on the target feature, $x_{t-w}^c, ..., x_{t-1}^c$ and $z_{1,t}^c, ..., z_{n,t}^c$ are only partially used (see Section 4.1). A model is trained to predict $\hat{x}^{q,c}$ by minimizing the quantile loss function [33]

$$L_q(\hat{x}_i^c, x_i^c) = \max\left[q(x_i^c - \hat{x}_i^c), (q-1)(x_i^c - \hat{x}_i^c)\right] \quad (2)$$

over a training set $X_\mathrm{train}^c = \{x_1^c, x_2^c, ..., x_{N_\mathrm{train}}^c \mid x_i^c \in \mathbb{R}\ \forall i\}$.

The quantile predictions of the target feature are forwarded to the anomaly detector. In the original concept proposal in [11], the detector decides whether to flag an anomaly or not based on the distance between ground truth observations and *deterministic* predictions. The present work proposes to extend to the distance between the ground truth and the *probabilistic* prediction interval (PI) $[\hat{x}^{q_\mathrm{L},c}, \hat{x}^{q_\mathrm{U},c}]$. In this way, the calculation of distances between ground truth and model prediction is dynamically adapted to the current model's confidence. When a model is certain about its predictions, even small deviations are accounted for. On the other hand, low model confidence (larger PI) will reduce the calculated distance. The distances are averaged over the last l observations according to

$$\varepsilon_t^c = \frac{\sum_{j=0}^{l-1} \begin{cases} x_{t-j}^c - \hat{x}_{t-j}^{q_\mathrm{U},c} & \text{if } x_{t-j}^c > \hat{x}_{t-j}^{q_\mathrm{U},c} \\ \hat{x}_{t-j}^{q_\mathrm{L},c} - x_{t-j}^c & \text{if } x_{t-j}^c < \hat{x}_{t-j}^{q_\mathrm{L},c} \end{cases}}{l}. \quad (3)$$

For the investigated PV-battery case, $l = 5$, $q_\mathrm{L} = 0.01$, and $q_\mathrm{U} = 0.99$ is selected $\forall c \in \mathcal{I}$ and $\mathcal{J}$. Based on ε_t^c and further characteristics of the current target feature observation, different anomaly types are distinguished, which is expressed by the decision function

$$v_t^c = \begin{cases} 2 & \text{if } \overbrace{(\varepsilon_t^c > \tau_c)}^{\text{Detection}}, \overbrace{(x_t^c > \hat{x}_t^{q_\mathrm{M},c})}^{\text{Direction}} \text{ and } \overbrace{(x_t^c = 0)}^{\text{Is zero}} \\ 1 & \text{if } (\varepsilon_t^c > \tau_c), (x_t^c > \hat{x}_t^{q_\mathrm{M},c}) \text{ and } (x_t^c \neq 0) \\ -1 & \text{if } (\varepsilon_t^c > \tau_c), (x_t^c < \hat{x}_t^{q_\mathrm{M},c}) \text{ and } (x_t^c \neq 0) \\ -2 & \text{if } (\varepsilon_t^c > \tau_c), (x_t^c < \hat{x}_t^{q_\mathrm{M},c}) \text{ and } (x_t^c = 0) \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

with τ_c being a target feature-specific threshold. The anomaly types are further explained in Table 5. Both the direction of an abnormally large deviation and the information about a target feature being zero provides valuable information for identification of event root causes and physical impact. For example, an abnormally high number of UDP packets send by an energy meter may indicate a FDIA, while a PV feed of zero during daytime may points towards a switched off inverter.

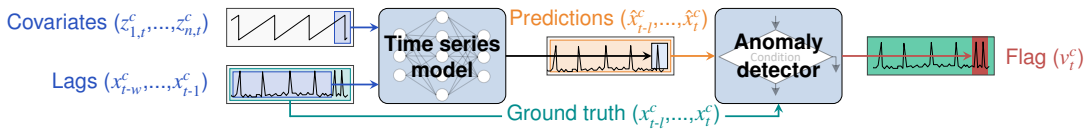


Figure 6: Schematic representation of the anomaly detection pipeline of a target feature c based on [11].

Table 5: Definition of considered anomaly types.

Flag v	Anomaly type	Description	Schematic
2	Positive zero	Target feature abnormally high and zero.	
1	Positive non-zero	Target feature abnormally high and non-zero.	
−1	Negative non-zero	Target feature abnormally low and non-zero.	
−2	Negative zero	Target feature abnormally low and zero.	
0	No anomaly	No abnormal behavior.	

4.2.2. *Time series models*

The authors of CyPhERS suggest the use of specific models for different target feature classes, which includes the use of long-short term memory (LSTM) networks for network traffic features. However, given the computational limitations of small DERs, the use of resource intensive models is impractical. Thus, this work proposes to apply gradient-boosted decision trees (GBDT) [34] for predicting the quantiles $\forall c \in \mathcal{I}$ and $\mathcal{J}$. GBDT is a frequently applied ML technique, popular for simultaneous high accuracy and efficiency, which renders them a good fit for the given problem [35]. GBDT consists of a set of simple decision trees, which are connected in series. Thus, each of them minimizes the prediction error of the preceding tree. For a detailed explanation the reader is referred to [34].

4.2.3. *Automated model and detector tuning procedure*

In [11], the authors of CyPhERS suggest an automated implementation procedure for the signature extraction system, which comprises independent tuning of the individual detection and classification pipelines (see Fig. 7). First step is the training and hyperparameter selection of the GBDT models of all target features. Selection of hyperparameters is conducted on 75 % of the

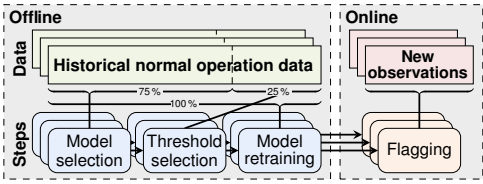


Figure 7: Procedure for tuning the anomaly detection pipelines based on [11].

two weeks normal operation data. The tuned hyperparameters and associated search spaces are summarized in Table 6.

Table 6: Hyperparameters and search spaces of the GBDT models.

No.	Hyperparameter	Search space
1	w^{a}	[0,...,60]
2	Max. depth of a tree	[3,...,21]
3	Number of decision trees	[100,...,1000]
4	Learning rate	[0.001,...,0.3]

[a]Only for target features considering lag values.

Next, the anomaly detectors of all pipelines are tuned. For that purpose, the fitted models are used to predict the remaining 25 % of normal operation data. Based on the predictions, the distances $E^{c}_{\text{test}} = \{\varepsilon^{c}_{1}, \varepsilon^{c}_{2}, ..., \varepsilon^{c}_{N_{\text{test}}} \mid \varepsilon^{c}_{i} \in \mathbb{R}\ \forall i\}$ are calculated according to (3), $\forall c \in \mathcal{I}$ and $\mathcal{J}$. From E^{c}_{test} and a threshold factor f, the feature-specific thresholds are determined according to

$$\tau_{c} = f \cdot \max\left(E^{c}_{\text{test}}\right), \tag{5}$$

where the threshold factor is selected as $f = 1.1$ in this work. Therefore, anomalies within a target feature c are flagged if ε^{c}_{i} exceeds the largest distance during normal operation by at least 10 %. Before the anomaly detection pipelines are applied for online flagging of new observations, the models are retrained on the entire set of historical normal operation data (see Fig. 7).

Finally, the resulting anomaly flag series provided by the individual pipelines are grouped for each system zone to obtain human-readable event signatures as output of CyPhERS' Stage 1. For the investigated PV-battery system, the defined system zones comprise PV1-4, BAT1-4, M1-4, DM, and DS.

4.3. *Signature evaluation (Stage 2)*

This section details the implementation of CyPhERS' signature evaluation for the studied PV-battery system case, which includes the definition of a signature database (Section 4.3.1), and automated signature evaluation system (Section 4.3.2).

4.3.1. *Signature database*

Signature evaluation in CyPhERS is based on manually or automatically matching the anomaly flags provided by Stage 1 with a database of known event signatures. The signatures of the attack types considered in this work are depicted in Fig. 8 on the example of selected victim devices and physical impacts. The associated signature descriptions are provided in Table 7.

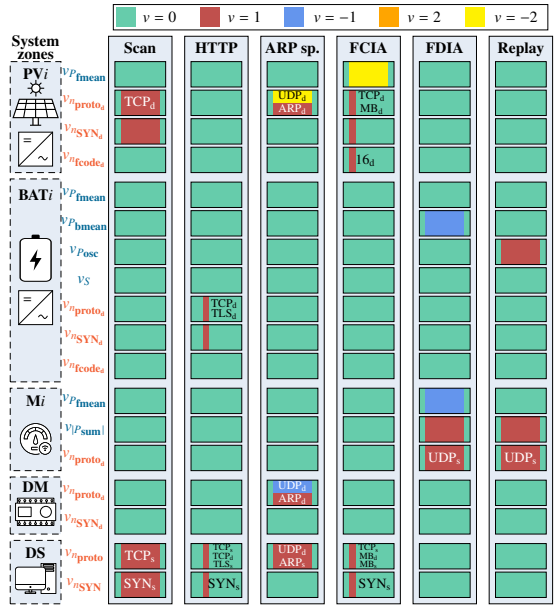


Figure 8: Event signatures of investigated attack types. The signatures are depicted for selected victim devices and physical impacts.

Table 7: Description of the event signatures of the attack types considered in this work (see Fig. 8).

Attack	Event signature description
Scan	A device (e.g., PV inverter) receives an abnormally large number of TCP packets ($v_{n_{\mathrm{TCP_d}}} = 1$) with connection request ($v_{n_{\mathrm{SYN_d}}} = 1$) over a longer period. Simultaneously, another device (e.g., DS) sends unusually many TCP packets ($v_{n_{\mathrm{TCP_s}}} = 1$) with connection request ($v_{n_{\mathrm{SYN_s}}} = 1$). Together, this points towards scanning of a victim device (here, the PV inverter), where the attacker is located on a local device (here, the DS). The lack of anomaly flags in physical target features indicates a pure cyber attack without physical impact.
HTTP request	A device (e.g., battery inverter) receives abnormally many TLS packets ($v_{n_{\mathrm{TLS_d}}} = 1$) over short period, pointing towards a web service call (HTTP request). Simultaneously, another device (e.g., DS) sends more TLS packets than usual ($v_{n_{\mathrm{TLS_s}}} = 1$), which suggests the attacker being located on this device. Parallel increase of TCP packets ($v_{n_{\mathrm{TCP_d}}} = 1$) and packets with SYN flags ($v_{n_{\mathrm{SYN_d}}} = 1$) due to connection establishment between attacker and victim device. The absence of anomaly flags in physical target features indicates a cyber attack without any physical impact.
ARP spoof	Two devices (e.g., PV inverter and DM) receive abnormally many ARP packets ($v_{n_{\mathrm{ARP_d}}} = 1$), while another (e.g., DS) sends more than expected ($v_{n_{\mathrm{ARP_s}}} = 1$). This points towards ARP spoofing where the attacker is located on a local device (here, the DS). The two victim devices receive less (or no) UDP packets ($v_{n_{\mathrm{UDP_d}}} = -1$ or -2), while the device occupied by the attacker receives more ($v_{n_{\mathrm{UDP_d}}} = 1$), which suggests that the communication between the victims is successfully redirected via the occupied device[a]. Lack of flags in physical features imply eavesdropping instead of manipulation of process-relevant data.
FCIA	A device (e.g., PV inverter) receives an abnormally large number of MB packets ($v_{n_{\mathrm{MB_d}}} = 1$) with write register function code 16 ($v_{n_{16_\mathrm{d}}} = 1$). In parallel, another device (e.g., DS) sends more MB packets than usual ($v_{n_{\mathrm{MB_s}}} = 1$). Together, this indicates an attacker sending false control commands to a victim device (here, the PV inverter) from the occupied local device (here, the DS). Parallel increase of TCP packets and packets with SYN flags because of connection establishment between occupied and victim device. Abnormally low and zero PV feed ($v_{P_{\mathrm{fmean}}} = -2$) indicate that the attacker switched off the PV inverter[b].
FDIA	An energy meter Mi sends unusually many UDP packets[c] ($v_{n_{\mathrm{UDP_s}}} = 1$) while the absolute sum of its active power readings is too high ($v_{|P_{\mathrm{sum}}|} = 1$). Together this points towards unusual frequent broadcasting of active power readings. The parallel abnormally low mean ($v_{P_{\mathrm{fmean}}} = -1$) indicates false $P^{\mathrm{M}i}$ injection imitating grid exports. For the battery which uses Mi readings, an unusually low mean active power given the current time and PV feed ($v_{P_{\mathrm{bmean}}} = -1$) suggests reaction with charging[b]. Absence of anomalies in $P^{\mathrm{BAT}i}_{\mathrm{fmean}}$ underlines that the battery accepts the false data and reacts to them in an expected way.
Replay attack	An energy meter Mi sends abnormally high numbers of UDP packets[c] ($v_{n_{\mathrm{UDP_s}}} = 1$), and the absolute sum of its active power measurements is higher than expected ($v_{|P_{\mathrm{sum}}|} = 1$). Together this indicates unusually frequent broadcasting of active power readings. As the mean is normal ($v_{P_{\mathrm{fmean}}} = 0$), no false data is injected, and instead, a replay of valid $P^{\mathrm{M}i}$ readings is likely. Abnormally high power changes ($v_{P_{\mathrm{osc}}} = 1$) of one or more batteries indicates load oscillation due to multiplication of the control error through replaying $P^{\mathrm{M}i}$ values.

[a] Parallel network anomalies for other devices which communicate with the victims possible as victim functionality can be affected by the attack.
[b] Physical impact depends on the malicious control command/injected false data, and the victim device.
[c] Parallel network anomalies for other devices possible due to UDP traffic overloading of those.

For the sake of conciseness, target features of the same kind are jointly represented in one row in Fig. 8. More precisely, for individual system zones, the various protocol count flags are combined in $v_{n_{\mathrm{proto}}}$, and flags in counts of different MB function codes ($v_{n_{16_\mathrm{d}}}$ and $v_{n_{4_\mathrm{d}}}$) in $v_{n_{\mathrm{fcode}}}$. Signatures can also be defined for unknown event types, which then carry reduced information such as indication of affected system zones. For the sake of clarity, however, these are not included in Fig. 8.

4.3.2. Automated signature evaluation

In case of larger plants such as medium voltage level-connected wind and solar parks, visual recognition of event signatures in control rooms may be feasible. However, for smaller resources such as residential PV-battery systems, manual evaluation is impractical. In [11], transforming known event signatures into a set of decision rules to automate signature evaluation is suggested, however, not realized and demonstrated. For the sake of proofing feasibility of automated signature evaluation, this work applies a simple rule-based signature evaluation system which jointly evaluates the most recent flags of all target features within a moving time window T_{eval} of five minutes. A simplified representation is provided in Algorithm 1. In case that no anomaly is detected (i.e., $v_c = 0$, $\forall c \in \mathcal{I}$ and $\mathcal{J}$) within

Algorithm 1 Simplified representation of the rule-based signature evaluation system.

$T_{\mathrm{eval}} \leftarrow$ Last 5 minutes
if all flags in T_{eval} are zero **then**
  prediction $\leftarrow$ Normal operation
else if $v^X_{n_{\mathrm{TCP_d}}} = 1$, $v^X_{n_{\mathrm{SYN_d}}} = 1$, $v^Y_{n_{\mathrm{TCP_s}}} = 1$, and $v^Y_{n_{\mathrm{SYN_s}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ Scan of device X from device Y
else if $v^X_{n_{\mathrm{TLS_d}}} = 1$, $v^X_{n_{\mathrm{TCP_d}}} = 1$, $v^X_{n_{\mathrm{SYN_d}}} = 1$, $v^Y_{n_{\mathrm{TLS_s}}} = 1$, $v^Y_{n_{\mathrm{TCP_d}}} = 1$, $v^Y_{n_{\mathrm{TCP_s}}} = 1$, and $v^Y_{n_{\mathrm{SYN_s}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ HTTPS request of device X from device Y
else if $v^X_{n_{\mathrm{ARP_d}}} = 1$, $v^Y_{n_{\mathrm{ARP_d}}} = 1$, $v^Z_{n_{\mathrm{ARP_s}}} = 1$, $v^X_{n_{\mathrm{UDP_d}}} = -1$ or -2, $v^Y_{n_{\mathrm{UDP_d}}} = -1$ or -2, and $v^Z_{n_{\mathrm{UDP_d}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ ARP spoof against devices X,Y from device Z
else if $v^X_{n_{\mathrm{TCP_d}}} = 1$, $v^X_{n_{\mathrm{MB_d}}} = 1$, $v^X_{n_{\mathrm{SYN_d}}} = 1$, $v^X_{n_{16_\mathrm{d}}} = 1$, $v^Y_{n_{\mathrm{TCP_s}}} = 1$, $v^Y_{n_{\mathrm{MB_d}}} = 1$, $v^Y_{n_{\mathrm{MB_s}}} = 1$, $v^Y_{n_{\mathrm{SYN_s}}} = 1$, and $v^X_{P_{\mathrm{fmean}}} = -2$ within T_{eval} **then**
  prediction $\leftarrow$ FCIA against device X from device Y with physical impact A (here, switch X off)
else if $v^M_{n_{\mathrm{UDP_s}}} = 1$, $v^M_{|P_{\mathrm{sum}}|} = 1$, $v^M_{P_{\mathrm{fmean}}} = -1$, and $v^X_{P_{\mathrm{bmean}}} = -1$ within T_{eval} **then**
  prediction $\leftarrow$ FDIA against meter M with physical impact A on device X (here, battery charging)
else if $v^M_{n_{\mathrm{UDP_s}}} = 1$, $v^M_{|P_{\mathrm{sum}}|} = 1$, $v^M_{P_{\mathrm{fmean}}} = 0$, and $v^X_{P_{\mathrm{osc}}} = 1$ within T_{eval} **then**
  prediction $\leftarrow$ Replay attack against meter M with physical impact A on device X (here, battery oscillation)
else
  prediction $\leftarrow$ Unknown abnormal behavior

T_{eval}, the system predicts normal operation. Given that the flags within T_{eval} match with one of the pre-defined event signatures, the associated event description forms the prediction, for example, *FCIA against PV1 from DS with physical impact "PV1 switched off"*. In case that anomaly flags provided by Stage 1 do not match with any of the defined rules, the evaluation system predicts *Unknown abnormal behavior*. In contrast to the simplified representation in Algorithm 1, additional rules for different variations of specific attack types are defined. For example, *Replay attack against meter M1 with physical impact "oscillation" on BAT1* can be *with* or *without parallel traffic overloading of other devices*. Further rules predict affected system zones in case of unknown event types, for example, *Unknown network anomaly affecting PV2 and DS*. Finally, some rules additionally take previous predictions into account. For example, in case of predicting a battery being switched off by a FCIA, information about the injected false command is stored over a time period larger than T_{eval} in order to avoid switching to prediction of a pure physical failure after five minutes.

5. Demonstration of CyPhERS

This section demonstrates the results of applying CyPhERS on the PV-battery system study case. The considered attack types are successively evaluated in the Sections 5.1-5.6.

5.1. Scanning attacks

The event signatures provided by CyPhERS' Stage 1 during the two scans are depicted in Fig. 9 together with the predictions of the rule-based signature evaluation system (Stage 2). Note that system zones without flagged anomalies are not depicted in the following. In both cases the provided signatures correspond to the signature of scanning attacks (see Fig. 8). Thus, visual recognition of the signature allows identifying the attack type (scan), victim (PV3 or PV4, respectively), and attacker location (DS) manually. The same prediction is provided without human interaction by the rule-based system. During the first scan, the rule for predicting a scan is not immediately fulfilled, since

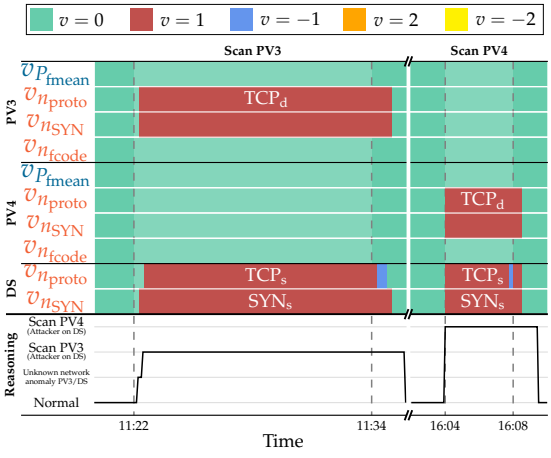


Figure 9: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the scan attacks.

flagging $v_{n^{\text{DS}}_{\text{TCP}_s}} = 1$ is delayed. Therefore, the rule-based system first predicts an unknown network traffic anomaly for PV3 and DS, based on the occurrence of flags in the associated network target features. This example demonstrates how CyPhERS can automatically provide information such as event occurrence and affected devices also for unknown event types.

Fig. 9 also confirms a known shortcoming of CyPhERS: Anomaly flags at the end of detected events are less reliable due to a *recovering phase* of the underlying detection and classification pipelines [11]. Fig. 10 depicts this behavior on the example of $n^{\text{DS}}_{\text{TCP}_s}$ during the scan of PV4. Since distances between ground truth and prediction are averaged over the last $l = 5$ observations, anomalies are flagged for some more steps even though distances of new observations are small. Thus, the end of detected events is usually not accurately indicated. Moreover, within the recovering phase, predictions are often already accurate and fluctuate around the ground truth, which translates to frequently changing anomaly flag types. As the less reliable flags within the recovering phase neither affect the manual nor the automated signature evaluation they are considered unproblematic.

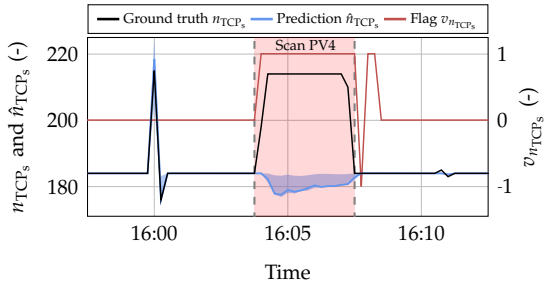


Figure 10: Ground truth, prediction (98 % PI and median) and anomaly flag for $n^{\text{DS}}_{\text{TCP}_s}$ during scan of PV4.

Finally, Fig. 10 also exemplifies the advantage of modeling target features with time series models. Since $n^{\text{DS}}_{\text{TCP}_s}$ exhibits normal peaks at full hours, the increase during scanning of PV4 only constitutes a local anomaly which cannot be detected by static thresholds not taking temporal information into account. In contrast, the applied GBDT model allows to detect the scanning-induced local anomaly by learning that peaks should only occur at full hours.

5.2. HTTPS request attacks

The provided event signatures and rule-based predictions for the two HTTPS requests are depicted in Fig. 11. Due to a match with the signature of HTTPS requests (see Fig. 8), the attack type can be identified together with the victims (BAT1 and DM, respectively), and attacker location (DS), both through visual signature recognition and the rule-based system.

5.3. ARP spoofing attacks

Fig. 12 depicts the provided event signatures, and predictions of the rule-based system for both ARP spoofing attacks. The signatures match the one for ARP spoofing (see Fig. 8). Since the associated rules are fulfilled, Stage 2 predicts ARP spoofing

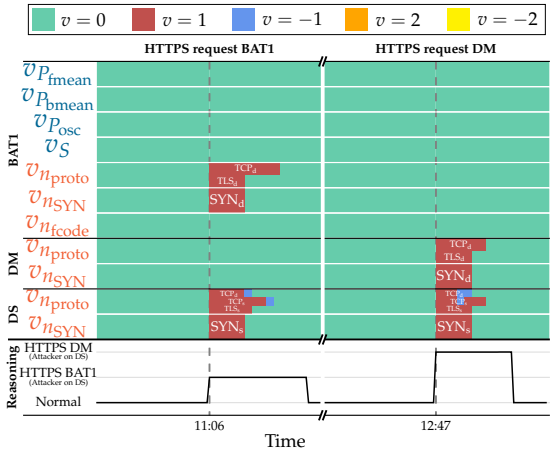


Figure 11: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation (Stage 2) during the HTTPS requests.

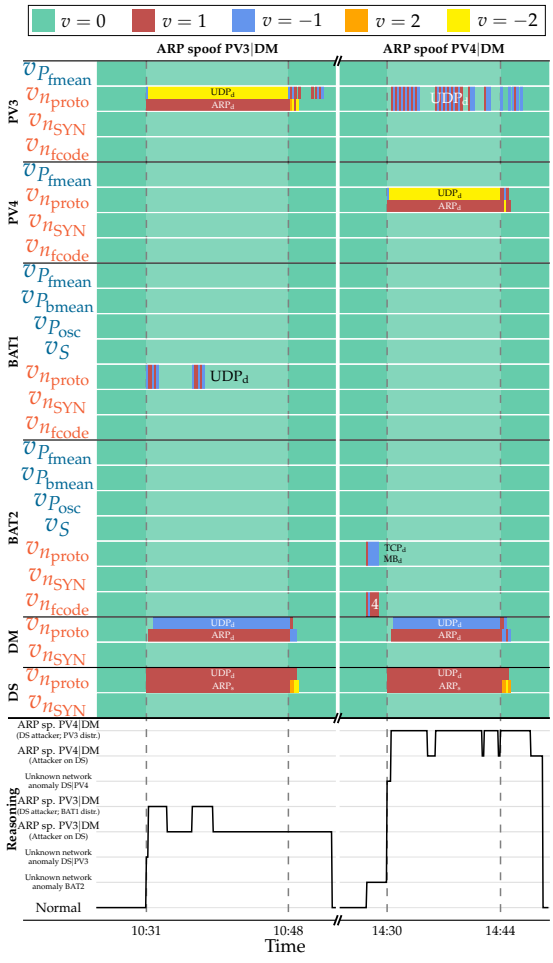


Figure 12: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the ARP spoofs.

attacks from an attacker located on the DS against PV3/DM, and PV4/DM, respectively. Since the attacks distract the DM, its UDP communication pattern to non-victim devices is also affected, resulting in parallel network flags for BAT1 and PV3. As this behavior is considered as a sub-case of the ARP spoofing signature, and integrated as such in the rule-based system, predictions switch between ARP spoofing with and without parallel traffic distraction of other devices (see Fig. 12). Another event is detected shortly before the second ARP spoof. As the provided anomaly flags do not match with the signature of a known attack, reduced event information (occurrence, affected network device, no physical impact) is provided by Stage 2.

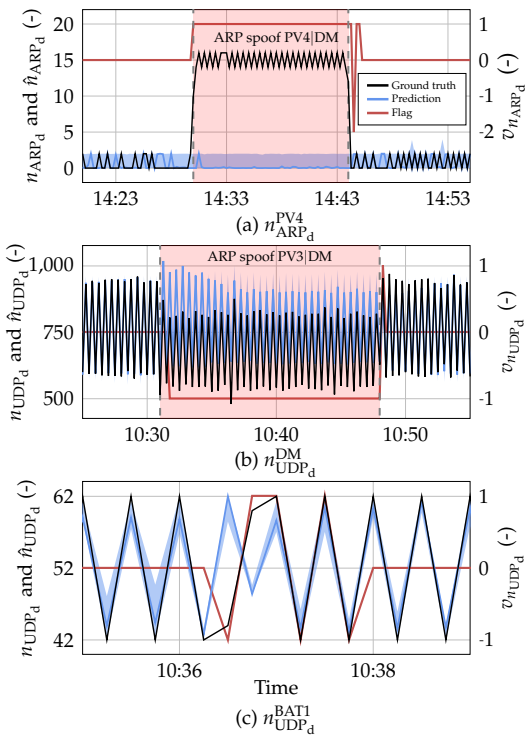


Figure 13: Ground truth, prediction (98 % PI and median), and anomaly flag for (a) $n^{\text{PV4}}_{\text{ARP}_\text{d}}$ during second ARP spoof, (b) $n^{\text{DM}}_{\text{UDP}_\text{d}}$ during first ARP spoof, and (c) $n^{\text{BAT1}}_{\text{UDP}_\text{d}}$ during first ARP spoof.

To illustrate the prediction and flagging process of the underlying pipelines, Fig. 13 depicts some examples. Fig. 13 (a) represents $n^{\text{PV4}}_{\text{ARP}_\text{d}}$ during the second ARP spoofing attack. It can be noticed that the spoofs result in pronounced global anomalies which are immediately detected. As ARP packets during normal operation occur non-deterministically, the small peaks cannot be learned by the GBDT model. Instead, it puts the PI on a constant level to capture those peaks, which illustrates how the GBDT model approximates a static but accurate threshold in cases without learnable pattern. Fig. 13 (b) shows $n^{\text{DM}}_{\text{UDP}_\text{d}}$ during the first ARP spoof. As the DM maintains UDP communication with non-victim devices, the oscillative pattern continuous, however, on a lower level. The level decrease is detected by the underlying pipeline. Fig. 13 (c) depicts an excerpt of the UDP traffic distraction of BAT1 during the first ARP spoof. It can

be seen that the distraction only expresses as a local and short pattern interruption without specific traffic increase or decrease.

5.4. False command injection attacks

Fig. 14 depicts the event signatures of Stage 1, and predictions of Stage 2 during the four FCIAs. In all cases the provided anomaly flags match the FCIA signature (see Fig. 8), allowing to identify the attack type, victims, attacker location, and physical impact, as the rule-based predictions indicate. Fig. 15 illustrates the detection of false commands on the example of $n^{\mathrm{DS}}_{\mathrm{MB_s}}$ during the FCIA against PV1. The underlying GBDT model successfully learned the normal peaks at full hours. Moreover, it understands that small positive peaks are usually followed by negative ones. As the injection of false commands is not followed by a negative peak, the larger distance between prediction and ground truth results in an anomaly flag.

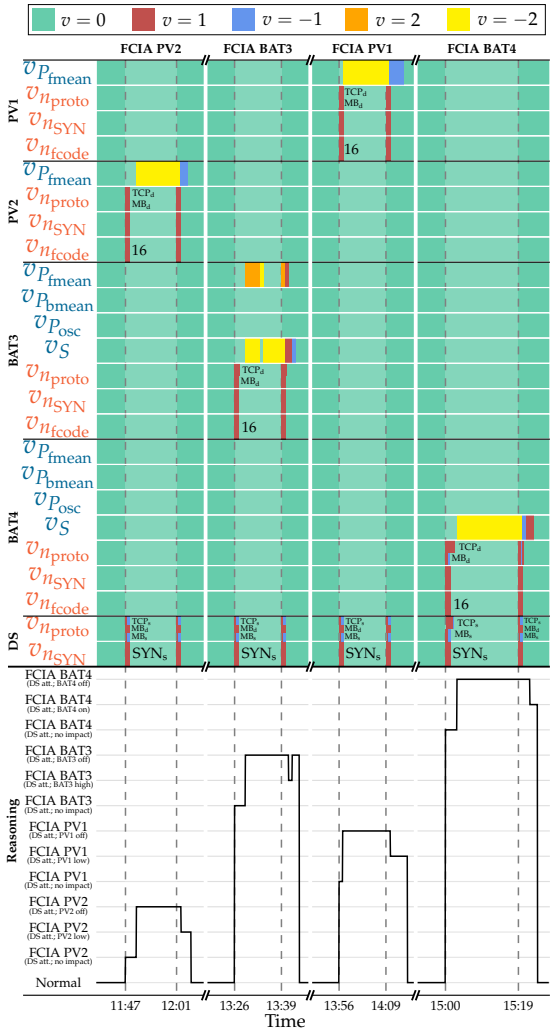


Figure 14: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the FCIAs.

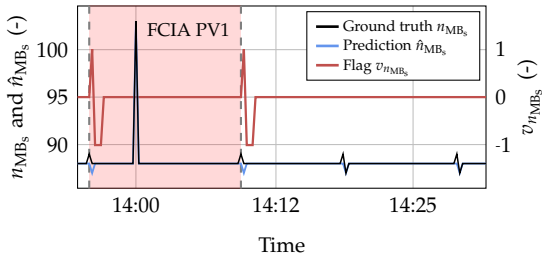


Figure 15: Ground truth, prediction (98 % PI and median) and anomaly flag for $n^{\mathrm{DS}}_{\mathrm{MB_s}}$ during FCIA against PV1.

The yellow or orange flags ($v = -2$ or 2) in the physical target features indicate that the attacker switched off the respective victim device. The detection of the physical impact is exemplified on the FCIAs against PV1 and BAT3[1] in Fig. 16. It can be noticed that modeling of physical target features is subject to larger uncertainties compared to network traffic modeling. As a result, smaller physical impacts may be missed by some features as, for example, the case for $v_{P^{\mathrm{BAT4}}_{\mathrm{fmean}}}$ and $v_{P^{\mathrm{BAT4}}_{\mathrm{bmean}}}$ during the FCIA against BAT4. Note that on the abstraction level of the battery state S, the switch-off is detected which suggests the importance of such abstracting features for applying CyPhERS for DER monitoring.

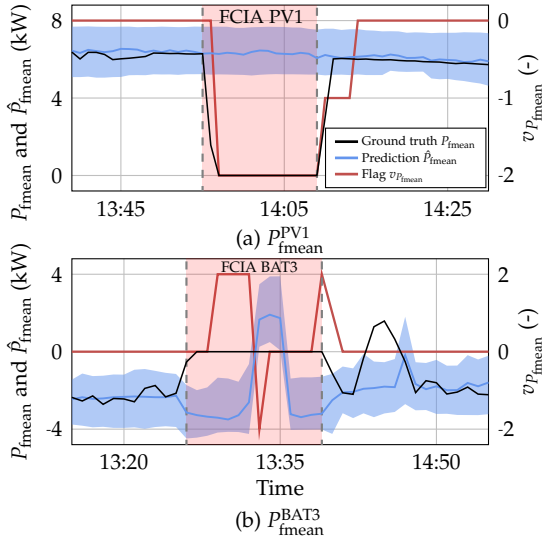


Figure 16: Ground truth, prediction (98 % PI and median) and anomaly flag for (a) $P^{\mathrm{PV1}}_{\mathrm{fmean}}$ during FCIA against PV1, and (b) $P^{\mathrm{BAT3}}_{\mathrm{fmean}}$ during FCIA against BAT3.

5.5. False data injection attacks

Fig. 17 depicts the provided event signatures, and predictions of the rule-based system during the FDIAs. As the signatures

[1] Note that the predicted sudden change from charging to discharging in Fig. 16 (b) results from the compressor load peak that the battery would compensate if not switched off.

during all three attacks correspond to the ones of FDIAs (see Fig. 8), attack type, victim device, and physical impact can be derived through visual signature recognition or rule-based predictions. Since the batteries accept the injected false data and react to them in an expected way, no disfunctionality is flagged

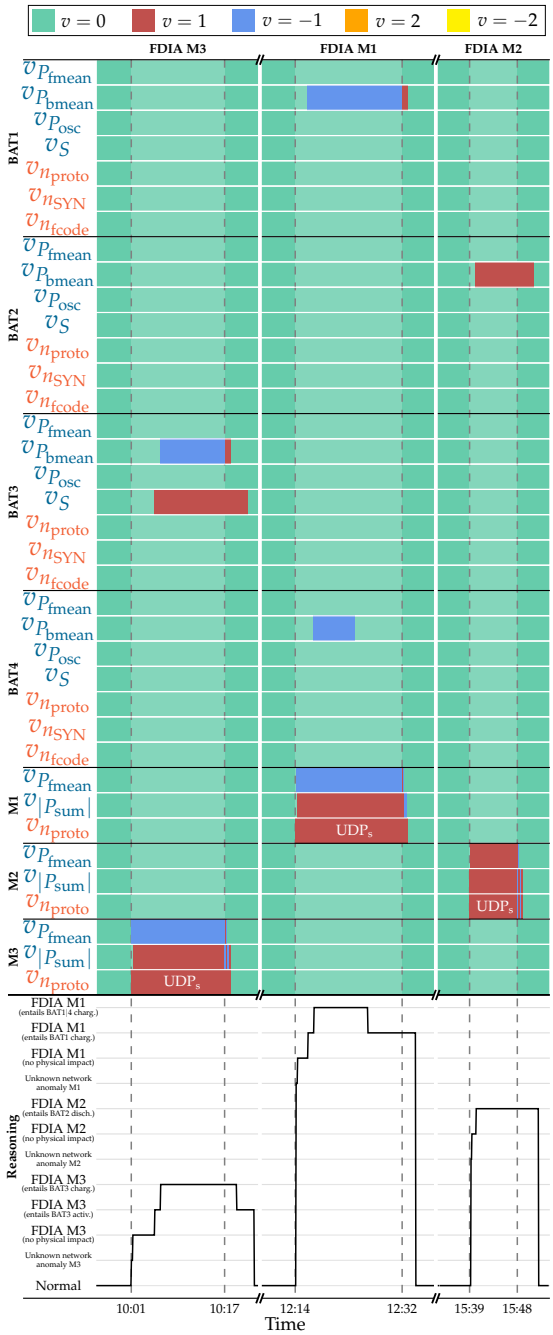


Figure 17: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the FDIAs.

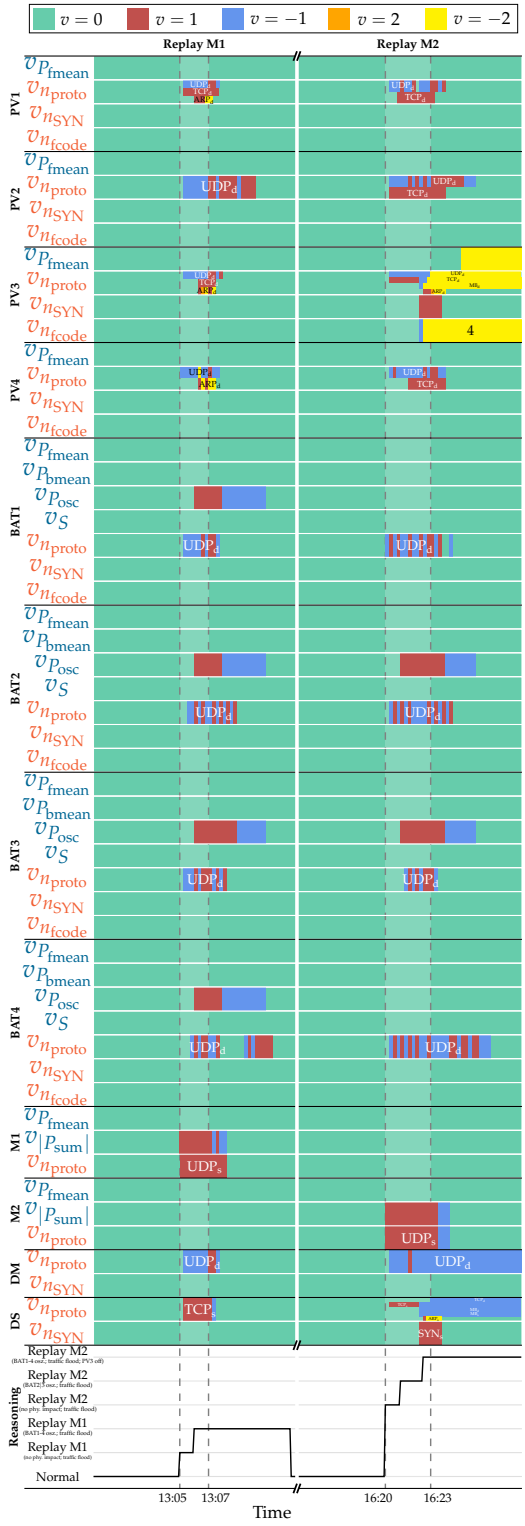


Figure 18: Event signatures provided by CyPhERS' Stage 1, and predictions of the rule-based signature evaluation system (Stage 2) during the replay attacks.

($v_{P^{\mathrm{BAT}i}_{\mathrm{fmean}}} = 0$). At the same time, anomaly flags in $v_{P^{\mathrm{BAT}i}_{\mathrm{bmean}}}$ indicate an untypical battery behavior given the current time of the day and PV feed[2]. While blue flags ($v_{P_{\mathrm{bmean}}} = -1$) indicate abnormal charging, red flags ($v_{P_{\mathrm{bmean}}} = 1$) point towards unusual discharging. This example underlines the importance of behavior-describing target features, and the differentiation of anomaly types for identification of the physical attack impact.

5.6. Replay attacks

The event signatures of Stage 1, and predictions of Stage 2 during the two replay attacks are depicted in Fig. 18. During both attacks, anomalies are flagged in almost all system zones as the network devices are distracted by processing the large number of replayed energy meter multicasts. In this case, visual recognition of specific attack patterns is challenging. In contrast, the rule-based system quickly identifies the signature as the associated rules are still fulfilled. Since parallel traffic flooding is integrated into the rule-based system as a sub-case of the replay attack signature, Stage 2 predicts the attack type, victim device, and physical impact along with network flooding. Due to $v_{P^{\mathrm{BAT}i}_{\mathrm{osc}}} = 1$, the physical impact (load oscillation), and affected batteries can be identified. As BAT1 and BAT4 are fully discharged during the second attack, no oscillation is indicated. Towards the end of the second replay attack, the inverter in PV3 crashes since it cannot process the large number of packets as pointed out by the yellow flags in the network target features. Shortly after, $v_{P^{\mathrm{PV3}}_{\mathrm{fmean}}} = -2$ indicates that also the feed of the associated solar panel string is interrupted.

6. Discussion

In this section, results of the case study are discussed and put into a wider perspective.

6.1. Applicability of CyPhERS for monitoring of DERs and other power system applications

The results in Section 5 demonstrate that CyPhERS can be applied for DER monitoring. During all considered attacks, CyPhERS' Stage 1 provides event signatures which can be manually or automatically evaluated in Stage 2 to conclude on root causes (attack type, victim devices and attacker location), and physical impacts (e.g., load step, as inverter was switched off) in real time. Moreover, it is shown that CyPhERS also infers information about unknown event types including occurrence, affected devices, and physical impact. The applicability is facilitated by five concept adaptions: 1) Consideration of functionality- and behavior-describing physical target features, which allows to detect and differentiate attacks targeting loss of functionality, and attacks exploiting normal device functionality to achieve a physical impact. 2) Use of physical target features, such as the on/off state of a battery, that break down modeling complexity for components exhibiting pronounced variability and uncertainty. 3) Applying probabilistic models and detection rules to cope with the high randomness and volatility of DER operation by dynamically adapting the detection sensitivity to model confidence. 4) Realization of a rule-based system for automated signature evaluation, allowing implementation for small DERs without human supervision. 5) Exclusive use of lightweight models to facilitate application to resource-restricted small DERs. Taking these aspects into account, applying CyPhERS to other power system applications, including substations and energy communities, is considered possible. A potentially limiting factor for resource-constrained systems is the linear dependency between the number of models and target features. Thus, careful selection of monitored features is of high relevance for minimizing the computational burden of CyPhERS.

[2]Note that $v_S^{\mathrm{BAT3}} = 1$ during the first FDIA indicates that BAT3 was first activated by the attack.

6.2. The role of ML in CyPhERS

In CyPhERS, ML is used to model target features and eventually provide the indicator for deciding whether an observation is normal or abnormal. The results in Section 5 demonstrate how ML allows to detect complex local anomalies which are only abnormal in a specific context (see, for example, Fig. 15). In case of some target features, similar detection results could be achieved with simpler methods. For example, the global anomaly in $n_{\mathrm{ARP_d}}$ during an ARP spoof (see Fig. 13 (a)) could be detected with a pre-defined static threshold. ML allows to generalize and automate modeling of target features and definition of detection rules. Thus, even in cases where simpler methods can achieve the same performance, ML is advantageous as it avoids manual effort, which is particularly relevant for larger numbers of target features. Furthermore, through regular retraining, the models automatically adapt to changes such as new consumer behaviour.

6.3. Uniqueness of event signatures

For the sake of conciseness and readability, the number of target features (in particular network features) is limited in this work. Many other relevant features which are, for example, based on port numbers or MAC and IP addresses are neglected. Moreover, other information sources are fully excluded. These include human interactions with the system (e.g., maintenance activities), and system logs. Consequently, some of the event signatures may be explainable by other incidents as well. For example, the pattern of a FCIA may also result from the rare event of switching off inverters for maintenance. If models are informed about such activities, these events can be distinguished. Thus, for an implementation outside an academic environment, all relevant target features should be taken into account, in order to guarantee uniqueness of the event signatures.

6.4. Integration into a distributed attack detection system

Power system operation is increasingly dependent on DERs, and the associated ICT infrastructure make coordinated attacks against multiple of them more likely. To quickly identify such threats, CyPhERS could be integrated into a bottom-up security architecture for power systems. Attack reports of multiple distributed CyPhERS systems could be aggregated and jointly evaluated by a cyber security incident response team (CSIRT). The CSIRT can then inform affected transmission or distribution system operators about cyber incidents in their area.

7. Conclusion

This work evaluates applicability of the cyber-physical event reasoning system CyPhERS for online DER monitoring on a study case which comprises several cyber and cyber-physical attack types targeting a real PV-battery system. CyPhERS is a two-stage process, where Stage 1 generates informative and interpretable signatures for known and unknown event types in real-time, which are manually or automatically evaluated in Stage 2 to conclude on event root causes and physical impacts. Key strength of CyPhERS is the independence of historical event observations. To facilitate applicability for DER monitoring, several concept adaptions are proposed and realized, including probabilistic models and detection rules, as well as a rule-based system for automated event signature evaluation. The results demonstrate that the adapted version of CyPhERS can be used to jointly evaluate a DER's process and network traffic data for automated inference of information such as attack occurrence, type, victim devices, attacker location, and physical impact in real-time and without need for historical attack observations.

Acknowledgement

This work is partly funded by the Innovation Fund Denmark (IFD) under File No. 91363, and by the Helmholtz Association under the program 'Energy System Design'.

References

[1] I. J. Perez-Arriaga, The transmission of the future: The impact of distributed energy resources on the network, IEEE Power and Energy Magazine 14 (4) (2016) 41–53. doi:10.1109/MPE.2016.2550398.

[2] M. Šarac, N. Pavlović, N. Bacanin, F. Al-Turjman, S. Adamović, Increasing privacy and security by integrating a blockchain secure interface into an iot device security gateway architecture, Energy Reports 7 (2021) 8075–8082. doi:https://doi.org/10.1016/j.egyr.2021.07.078.

[3] A. G. Eustis, The mirai botnet and the importance of iot device security, in: 16th International Conference on Information Technology-New Generations (ITNG 2019), Springer, 2019, pp. 85–89.

[4] S. Lakshminarayana, J. Ospina, C. Konstantinou, Load-altering attacks against power grids under covid-19 low-inertia conditions, IEEE Open Access Journal of Power and Energy 9 (2022) 226–240. doi:10.1109/OAJPE.2022.3155973.

[5] EnergiCERT, Cyber attacks against european energy & utility companies, `https://energicert.dk/wp-content/uploads/2022/09/Attacks-against-European-energy-and-utility-companies-2020-09-05-v3.pdf`, accessed: 2023-05-06.

[6] S. Huntley, Fog of war: how the ukraine conflict transformed the cyber threat landscape, `https://blog.google/threat-analysis-group/fog-of-war-how-the-ukraine-conflict-transformed-the-cyber-threat-landscape/`, accessed: 2023-05-06.

[7] Y. Li, J. Yan, Cybersecurity of smart inverters in the smart grid: A survey, IEEE Transactions on Power Electronics 38 (2) (2023) 2364–2383. doi:10.1109/TPEL.2022.3206239.

[8] N. Müller, C. Ziras, K. Heussen, Assessment of cyber-physical intrusion detection and classification for industrial control systems, in: 2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2022, pp. 432–438. doi:10.1109/SmartGridComm52983.2022.9961010.

[9] J. Ye, A. Giani, A. Elasser, S. K. Mazumder, C. Farnell, H. A. Mantooth, T. Kim, J. Liu, B. Chen, G.-S. Seo, W. Song, M. D. R. Greidanus, S. Sahoo, F. Blaabjerg, J. Zhang, L. Guo, B. Ahn, M. B. Shadmand, N. R. Gajanur, M. A. Abbaszada, A review of cyber–physical security for photovoltaic systems, IEEE Journal of Emerging and Selected Topics in Power Electronics 10 (4) (2022) 4879–4901. doi:10.1109/JESTPE.2021.3111728.

[10] N. D. Tuyen, N. S. Quan, V. B. Linh, V. Van Tuyen, G. Fujita, A comprehensive review of cybersecurity in inverter-based smart power system amid the boom of renewable energy, IEEE Access 10 (2022) 35846–35875. doi:10.1109/ACCESS.2022.3163551.

[11] N. Müller, K. Bao, J. Matthes, K. Heussen, Cyphers: A cyber-physical event reasoning system providing real-time situational awareness for attack and fault response (2023). arXiv:2305.16907.

[12] J. Qi, A. Hahn, X. Lu, J. Wang, C.-C. Liu, Cybersecurity for distributed energy resources and smart inverters, IET Cyber-Physical Systems: Theory & Applications 1 (1) (2016) 28–39.

[13] I. Zografopoulos, C. Konstantinou, Detection of malicious attacks in autonomous cyber-physical inverter-based microgrids, IEEE Transactions on Industrial Informatics 18 (9) (2022) 5815–5826. doi:10.1109/TII.2021.3132131.

[14] A. Y. Fard, M. Easley, G. T. Amariucai, M. B. Shadmand, H. Abu-Rub, Cybersecurity analytics using smart inverters in power distribution system: Proactive intrusion detection and corrective control framework, in: 2019 IEEE International Symposium on Technologies for Homeland Security (HST), 2019, pp. 1–6. doi:10.1109/HST47167.2019.9032978.

[15] Y. Li, P. Zhang, L. Zhang, B. Wang, Active synchronous detection of deception attacks in microgrid control systems, IEEE Transactions on Smart Grid 8 (1) (2017) 373–375. doi:10.1109/TSG.2016.2614884.

[16] S. Tan, J. M. Guerrero, P. Xie, R. Han, J. C. Vasquez, Brief survey on attack detection methods for cyber-physical systems, IEEE Systems Journal 14 (4) (2020) 5329–5339. doi:10.1109/JSYST.2020.2991258.

[17] A. A. Khan, O. A. Beg, M. Alamaniotis, S. Ahmed, Intelligent anomaly identification in cyber-physical inverter-based systems, Electric Power Systems Research 193 (2021) 107024. doi:https://doi.org/10.1016/j.epsr.2021.107024.

[18] D. Mukherjee, A novel strategy for locational detection of false data injection attack, Sustainable Energy, Grids and Networks 31 (2022). doi:https://doi.org/10.1016/j.segan.2022.100702.

[19] Z. Warraich, W. Morsi, Early detection of cyber–physical attacks on fast charging stations using machine learning considering vehicle-to-grid operation in microgrids, Sustainable Energy, Grids and Networks 34 (2023). doi:https://doi.org/10.1016/j.segan.2023.101027.

[20] A. M. Kosek, O. Gehrke, Ensemble regression model-based anomaly detection for cyber-physical intrusion detection in smart grids, in: 2016 IEEE Electrical Power and Energy Conference (EPEC), 2016, pp. 1–7. doi:10.1109/EPEC.2016.7771704.

[21] C. B. Jones, A. R. Chavez, R. Darbali-Zamora, S. Hossain-McKenzie, Implementation of intrusion detection methods for distributed photovoltaic inverters at the grid-edge, in: 2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2020, pp. 1–5. doi:10.1109/ISGT45199.2020.9087756.

[22] A. P. Kuruvila, I. Zografopoulos, K. Basu, C. Konstantinou, Hardware-assisted detection of firmware attacks in inverter-based cyberphysical microgrids, International Journal of Electrical Power & Energy Systems 132 (2021) 107150. doi:https://doi.org/10.1016/j.ijepes.2021.107150.

[23] C. B. Jones, A. Chavez, S. Hossain-McKenzie, N. Jacobs, A. Summers, B. Wright, Unsupervised online anomaly detection to identify cyber-attacks on internet connected photovoltaic system inverters, in: 2021 IEEE Power and Energy Conference at Illinois (PECI), 2021, pp. 1–7. doi:10.1109/PECI51586.2021.9435234.

[24] I. Zografopoulos, C. Konstantinou, N. D. Hatziargyriou, Distributed energy resources cybersecurity outlook: Vulnerabilities, attacks, impacts, and mitigations, arXiv preprint arXiv:2205.11171 (2022).

[25] V. K. Singh, M. Govindarasu, A cyber-physical anomaly detection for wide-area protection using machine learning, IEEE Transactions on Smart Grid 12 (4) (2021) 3514–3526. doi:10.1109/TSG.2021.3066316.

[26] A. Chavez, C. Lai, N. Jacobs, S. Hossain-McKenzie, C. B. Jones, J. Johnson, A. Summers, Hybrid intrusion detection system design for distributed energy resource systems, in: 2019 IEEE CyberPELS (CyberPELS), 2019, pp. 1–6. doi:10.1109/CyberPELS.2019.8925064.

[27] A. Sahu, Z. Mao, P. Wlazlo, H. Huang, K. Davis, A. Goulart, S. Zonouz, Multi-source multi-domain data fusion for cyberattack detection in power systems, IEEE Access 9 (2021) 119118–119138. doi:10.1109/ACCESS.2021.3106873.

[28] A. A. Cook, G. Mısırlı, Z. Fan, Anomaly detection for iot time-series data: A survey, IEEE Internet of Things Journal 7 (7) (2020) 6481–6494. doi:10.1109/JIOT.2019.2958185.

[29] M. K. Hasan, A. A. Habib, Z. Shukur, F. Ibrahim, S. Islam, M. A. Razzaque, Review on cyber-physical and cyber-security system in smart grid: Standards, protocols, constraints, and recommendations, Journal of Network and Computer Applications 209 (2023) 103540.

doi:https://doi.org/10.1016/j.jnca.2022.103540.

[30] F. Li, X. Yan, Y. Xie, Z. Sang, X. Yuan, A review of cyber-attack methods in cyber-physical power system, in: 2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP), 2019, pp. 1335–1339. doi:10.1109/APAP47170.2019.9225126.

[31] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, C. B. Thomas, Mitre att&ck: Design and philosophy, in: Technical report, The MITRE Corporation, 2018.

[32] S. Lakshminarayana, S. Adhikari, C. Maple, Analysis of iot-based load altering attacks against power grids using the theory of second-order dynamical systems, IEEE Transactions on Smart Grid 12 (5) (2021) 4415–4425. doi:10.1109/TSG.2021.3070313.

[33] Q. Wang, Y. Ma, K. Zhao, Y. Tian, A comprehensive survey of loss functions in machine learning, Annals of Data Science (2020) 1–26.

[34] J. H. Friedman, Greedy function approximation: A gradient boosting machine, The Annals of Statistics 29 (5) (2001) 1189–1232.
URL http://www.jstor.org/stable/2699986

[35] C. S. Bojer, J. P. Meldgaard, Kaggle forecasting competitions: An overlooked learning opportunity, International Journal of Forecasting 37 (2) (2021) 587–603. doi:https://doi.org/10.1016/j.ijforecast.2020.07.007.

PAPER F

Nils Müller, Samuel Chevalier, Carsten Heinrich, Kai Heussen, Charalampos Ziras

Uncertainty quantification in LV state estimation under high shares of flexible resources

Müller, N., S. Chevalier, C. Heinrich, K. Heussen, and C. Ziras, "Uncertainty quantification in LV state estimation under high shares of flexible resources," in *Electric Power Systems Research*, vol. 212, 2022, doi: 10.1016/j.epsr.2022.108479.

Electric Power Systems Research 212 (2022) 108479

Contents lists available at ScienceDirect

Electric Power Systems Research

journal homepage: www.elsevier.com/locate/epsr

Uncertainty quantification in LV state estimation under high shares of flexible resources

Nils Müller *, Samuel Chevalier, Carsten Heinrich, Kai Heussen, Charalampos Ziras

Department of Electrical Engineering, Technical University of Denmark, Lyngby, Denmark

ARTICLE INFO

Keywords:
Bayesian neural network
Low-voltage network
Quantile regression
State estimation
Uncertainty quantification

ABSTRACT

The ongoing electrification introduces new challenges to distribution system operators (DSOs). Controllable resources may simultaneously react to price signals, potentially leading to network violations. DSOs require reliable and accurate low-voltage state estimation (LVSE) to improve awareness and mitigate such events. However, the influence of flexibility activations on LVSE has not been addressed yet. It remains unclear if flexibility-induced uncertainty can be reliably quantified to enable robust DSO decision-making. In this work, uncertainty quantification in LVSE is systematically investigated for multiple scenarios of input information availability and flexibility usage, using real data. For that purpose, a Bayesian neural network (BNN) is compared to quantile regression. Results show that frequent flexibility activations can significantly deteriorate LVSE performance, unless secondary substation measurements are available. Moreover, it is demonstrated that the BNN captures flexibility-induced voltage drops by dynamically extending the prediction interval during activation periods, and that it improves interpretability regarding the cause of uncertainty.

1. Introduction

In light of the European effort to reach carbon neutrality by 2050, distribution networks (DNs) must undergo radical changes. Traditionally designed for the supply of consumers based on centralized generation, DNs turn into carriers of volatile and often bidirectional power flows [1]. Key drivers are the increasing deployment of variable distributed generation, as well as the proliferation of electric vehicles (EVs), heat pumps and residential storage. To balance consumption and generation in renewable-based power systems, it is widely acknowledged that more local consumption flexibility is required [2]. However, controllable distributed energy resources (DERs) may react to price signals with a sudden change of power consumption, resulting in higher coincidence factors in DNs dominated by such resources [3]. As a consequence, unacceptably high voltage deviations could occur and transformer or line protections systematically be triggered. Since large parts of the DN are unobserved, distribution system operators (DSOs) require techniques for reliable and accurate low-voltage state estimation (LVSE) in order to (i) improve awareness about the potential negative side effects of flexibility utilization and distributed generation, and (ii) be able to mitigate any potential operational problems.

For conventional state estimation (SE), network topology and line parameters must be known. Moreover, since LVSE constitutes a mathematically underdetermined problem, pseudo measurements are typically required to account for low meter coverage [4]. A widely considered approach to overcome these shortcomings is machine learning [4,5], whose feasibility and high estimation accuracy have been extensively demonstrated [6,7].

A relatively small number of works have investigated the topic of probabilistic LVSE so far. Importantly, the influence of frequent flexibility activations on reliability and accuracy has not been sufficiently addressed. Moreover, it remains unclear whether estimation uncertainty introduced by flexibility can be reliably quantified to support DSO decision making, especially under varying levels of real-time data availability.

1.1. Related work

A number of works propose the use of neural networks (NNs) for real-time LVSE. In [8] an NN is used to generate pseudo measurements for weighted least squares (WLS) SE, and errors are modeled by a Gaussian mixture model. Ref. [6] proposes and validates the use of an NN, utilizing only real-time secondary substation information. The application is seen as a real-time bus voltage estimator, with smart

* Corresponding author.
E-mail addresses: nilmu@elektro.dtu.dk (N. Müller), schev@elektro.dtu.dk (S. Chevalier), cahei@elektro.dtu.dk (C. Heinrich), kh@elektro.dtu.dk (K. Heussen), chazi@elektro.dtu.dk (C. Ziras).

https://doi.org/10.1016/j.epsr.2022.108479
Received 1 October 2021; Received in revised form 12 April 2022; Accepted 2 July 2022
Available online 4 August 2022

meter (SM) data being available with a latency of one day. The authors of [9] also propose the use of an NN for estimating voltage magnitudes based on the same input, but do not use real data. Further, the authors stressed that model performance is sufficient, largely because of the relatively simple topology that lacks long side branches, and the lack of significant amounts of photovoltaic (PV) in-feed.

However, given the high variability of low-voltage (LV) network states and the increasing penetration of DERs, a probabilistic approach seems more suitable compared to a deterministic output. The authors of [10] use quantile NNs to provide probabilistic forecasts of the states of an LV network. However, a high degree of observability is assumed (known voltage and bus injections), and focus is given to the model's forecasting abilities. Further, it is shown that EV charging creates large uncertainty, which the authors mitigate by assuming full knowledge by the DSO of the EV charging start time and duration.

Ref. [11] proposes a data-driven probabilistic LVSE based on an analog search technique and kernel density estimation. This approach relies on finding similar past patterns, and it assumes real-time information from the secondary substation and voltage values from a number of LV buses, while validation is performed by assessing the impact of varying levels of PV penetration. The authors of [12] apply a variation of WLS for probabilistic SE. However, a series of real-time measurements is assumed to be available, and hourly data is used without considerations for flexibility activation.

In [13], a deep learning approach to Bayesian SE is proposed. The authors propose to first learn the distribution of bus injections from SM data. Based on samples drawn from the learned distributions, a feed-forward NN is trained for the minimum mean-squared-error estimation of the system state. Due to the application of a deterministic regression model, the proposed approach is not inherently probabilistic. The work does not consider uncertainty quantification, and it compares results only on deterministic error metrics. Moreover, no flexible resources are considered. The authors in [14] propose a deep belief network for pseudo measurements modeling. Based on an extended WLS estimator the probability density function of system states is inferred. However, the work assumes partial real-time knowledge of voltage states, and only medium voltage (MV) states are estimated.

1.2. Contribution and paper structure

The literature review shows that most works on LVSE do not consider probabilistic approaches. Moreover, effects of varying levels of input information and flexibility usage on the estimation and uncertainty quantification performance are rarely studied. In this work, a Bayesian neural network (BNN) is applied for probabilistic estimation of multiple LV node voltages. The BNN is selected as it constitutes an inherently probabilistic approach, combining benefits of Bayesian uncertainty quantification and the predictive power of NNs. The main contributions of this work are as follows:

- Systematic evaluation of the influence of different levels of flexibility usage and input information to the DSO on the accuracy and uncertainty of LVSE.
- First application of a BNN on LVSE and comparison to a quantile regression (QR) benchmark.
- First work considering and discussing epistemic and aleatoric uncertainty for probabilistic LVSE.

The remainder of the paper is structured as follows. In Section 2, the use of a BNN for LVSE is motivated and a theoretical description of applied models and their implementation is provided. Section 3 presents the experimental setup, including the dataset, flexibility scenarios (FSs) and input scenarios (ISs), as well as the applied performance metrics. In Section 4, results are presented and discussed, followed by a conclusion and view on future work in Section 5.

2. Method

This section first motivates the selection of a BNN. Next, the theoretical background and implementation of both the BNN and QR, which serves as a benchmark, are described.

2.1. Model selection

Accurate SE is the basis for the decision-making process of DSOs. However, even an accurate estimator may result in large inaccuracies under specific or new circumstances, such as a certain time of the day or rare social events. Under these conditions, a deterministic estimator fails silently, affecting the DSO decision-making process and potentially impacting critical decisions. In contrast, a probabilistic estimator can capture prediction uncertainty. By expressing *what*, *when* and *why* it does not know, such an estimator increases interpretability of predictions. Thus, incorporating uncertainty quantification in LVSE allows for risk-aware DSOs network operation.

Uncertainty can be classified into aleatoric (data) and epistemic (model) uncertainty. Aleatoric uncertainty is introduced by randomness in the process. In case of LVSE, this randomness is given by factors such as measurement errors and random consumer behavior. Adding to this type of uncertainty, a given substation measurement can correspond to a variety of load realizations and LV states, making the mapping of measurements to unobserved states non-unique. Epistemic uncertainty comprises of model structure and parameter uncertainty due to lack of knowledge. In the LVSE problem, this is given for example by non-stationarity of data. A model only trained on winter months will encounter high epistemic uncertainty when predicting summer months, since input features are out-of-distribution. A fundamental difference between epistemic and aleatoric uncertainty is the fact that only the former can be reduced through additional information. To account for total uncertainty in LVSE, a model capable of capturing both components is required.

While quantifying estimation uncertainty is seen as an important requirement, accurate predictions are indispensable. As presented in Section 1.1, multiple works have demonstrated the predictive power of NNs for LVSE. However, most of the available models are not able to represent uncertainty.

BNNs constitute a new direction in machine learning [15]. By connecting Bayesian statistics and deep learning, BNNs combine the benefits of Bayesian uncertainty quantification with the predictive power of NNs. In contrast to traditional NNs, model parameters of BNNs are not fixed. Instead, every weight and bias is represented by a conditional probability distribution, representing the uncertainty of the respective parameter. Predictions are generated through posterior inference. By directly sampling from the probabilistic parameters, BNNs are inherently probabilistic, instead of deterministic, models.

2.2. BNN description

Let $X_{\text{train}} = \{x_1, \dots, x_{N_{\text{train}}}\}$ and $Y_{\text{train}} = \{y_1, \dots, y_{N_{\text{train}}}\}$ be the training input and output data, respectively, with N_{train} being the number of training samples. The BNN can be formulated as

$$[\hat{y}, \hat{\sigma}^2] = f_{\text{BNN}}^{\hat{W}}(x), \tag{1}$$

where $\hat{W} = \{\hat{w}_1, \dots, \hat{w}_{N_{\text{L}}}\}$ are model parameters and N_{L} the number of network layers. To consider aleatoric uncertainty, the output of the model is an estimate of both the predictive mean $\hat{y}$ and variance $\hat{\sigma}^2$. To account for epistemic uncertainty, a prior distribution is placed over $\hat{W}$. In this work, a Gaussian prior $\mathcal{N}(0, I)$ is applied since Gaussian priors for BNNs are known to provide the benefit of regularization [16]. The posterior distribution $p(W|X_{\text{train}}, Y_{\text{train}})$ over the model parameters, given the training data $\{X_{\text{train}}, Y_{\text{train}}\}$ is calculated by Bayes' rule.

The predictive distribution for a new observation x is obtained by marginalizing over the posterior distribution [17] according to

$$p(y|x, X_{\text{train}}, Y_{\text{train}}) = \int p(y|x, W)p(W|X_{\text{train}}, Y_{\text{train}})\mathrm{d}W. \tag{2}$$

Due to the non-linearity and non-conjugacy of NNs, the true posterior is typically intractable. By minimizing the Kullback–Leibler (KL) divergence between $p(W|X_{\text{train}}, Y_{\text{train}})$ and a surrogate distribution, the posterior is approximated. In this work variational inference was used as an inference algorithm to minimize the KL divergence [18]. To allow for simultaneous output of $\hat{y}$ and $\hat{\sigma}^2$ (1), and thus include aleatoric uncertainty, the loss function [19] of the BNN is formulated as

$$\mathcal{L}_{\text{BNN}}(\theta) = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \frac{1}{2\hat{\sigma}_i^2} \|y_i - \hat{y}_i\|^2 + \frac{1}{2} \log \hat{\sigma}_i^2. \tag{3}$$

To take epistemic uncertainty into account, multiple predictions of $\hat{y}$ and $\hat{\sigma}^2$ for input x are required. Note that every prediction is based on a new set of sampled model parameters $\hat{W}_t$. The predictive mean is estimated with

$$\widetilde{\mathbb{E}}(y) \approx \frac{1}{N_{\text{sample}}} \sum_{t=1}^{N_{\text{sample}}} f_{\text{BNN}}^{\hat{W}_t}(x), \tag{4}$$

where N_{sample} denotes the number of stochastic forward passes through the BNN. The total predictive uncertainty, composed of an epistemic and aleatoric term, is approximated with

$$\widetilde{\text{Var}}(y) \approx \underbrace{\left[\frac{1}{N_{\text{sample}}} \sum_{t=1}^{N_{\text{sample}}} \hat{y}_t^2 - \left(\frac{1}{N_{\text{sample}}} \sum_{t=1}^{N_{\text{sample}}} \hat{y}_t\right)^2\right]}_{\text{epistemic}} + \underbrace{\frac{1}{N_{\text{sample}}} \sum_{t=1}^{N_{\text{sample}}} \hat{\sigma}_t^2}_{\text{aleatoric}}, \tag{5}$$

with $\{\hat{y}_t, \hat{\sigma}_t^2\}_{t=1}^{N_{\text{sample}}}$ being a set of N_{sample} model outputs for the input x according to (1) with randomly drawn weights $\hat{W}_t \sim q(W)$. For a detailed explanation of BNNs the reader is referred to [19].

In this work all investigated datasets are split into a training, validation and test set. A detailed explanation of the datasets, including input features and output variables, follows in Section 3. Unless explicitly defined differently, the partition is 80/10/10. The selection of model structure and hyperparameters is realized based on the validation loss. For all datasets the smallest validation loss is achieved by using Adam optimizer, tanh activation function, two hidden layers and a batch size of 64. The number of required epochs and units in the hidden layers varies in the range 2000–10 000 and 5-12, respectively. The selected models are retrained on the respective training and validation data. Depending on the dataset and input features, the number of trainable parameters varies between 3780 and 26 766. The BNN is implemented using the *Tensorflow Probability* library [20].

2.3. QR description

QR constitutes an extension of linear regression [21]. In contrast to linear regression which estimates the conditional mean of the target variable, QR calculates the conditional median or any quantile τ for sample $i \in 1, \dots, n$ as

$$Q_\tau(y_i) = \beta_0(\tau) + \beta_1(\tau)x_{i,1} + \cdots + \beta_p(\tau)x_{i,p}. \tag{6}$$

Coefficients $\beta_j(\tau)$ for $j \in 0, \dots, p$ are retrieved by solving

$$\min_{\beta_0(\tau),\dots,\beta_p(\tau)} \sum_{i=1}^{n} \rho_\tau \left(y_i - \beta_0(\tau) - \sum_{j=1}^{p} x_{i,j}\beta_j(\tau)\right), \tag{7}$$

where $\rho_\tau(r) = \tau \max(r, 0) + (1-\tau)\max(-r, 0)$ gives asymmetric weights to the error r with respect to τ and the sign of the error. For the QR the same input features and data partitions as for the BNN are considered. The model is implemented using the *statsmodel* library [22], which does not include any hyperparameters to be selected.

3. Experimental setup

This section first describes the considered real network and load dataset, and explains the creation of the various FSs and ISs. Thereafter, metrics used for evaluating the LVSE performance are introduced.

3.1. Network and used dataset

The network used in this study is a real suburban MV-LV network located in Bornholm, Denmark, and is depicted in Fig. 1. It consists of six 10/0.4 kV secondary substations, all connected to a single primary 60/10 kV substation, which in the following is referred to as SubP and whose high voltage side is taken as the reference voltage and is set to 1 pu. The network serves a total of 564 residential customers. A detailed model of the LV network below secondary substation SubS is shown in Fig. 1, while the remainder are modeled as PQ buses that represent the aggregated active and reactive power of all customers connected to the respective substation.

Real 5 min average active and reactive power measurements from a large number of residential customers on Bornholm are used. Data was collected during the EcoGrid 2.0 project [23], and also includes the flexible operation of heating loads, as a result of experimental demonstration of a local flexibility market [23,24]. Most customers are equipped with a heat pump or electric heater, and typically have an average yearly consumption of 8 MW h. Approximately 10 % of the customers have 6 kW_p PV systems installed. Project participants are randomly assigned to the leaf nodes below SubS, and to the aggregated PQ profiles of the other secondary substations. This approach was followed because during the project only a small portion of participants were connected to the depicted network. Flexibility from heating loads present in the original dataset is rather limited. To simulate scenarios with larger shares of flexible demand and study the impact on LVSE, smart EV charging profiles are added, as described in Section 3.3. The time span of the study is the full year 2018.

3.2. Creating datasets for network states

SM data is assumed to become available to the DSO with a daily delay. Depending on infrastructure and DSO practice, voltage values may also be extracted by SMs. If this is the case, a historical dataset can

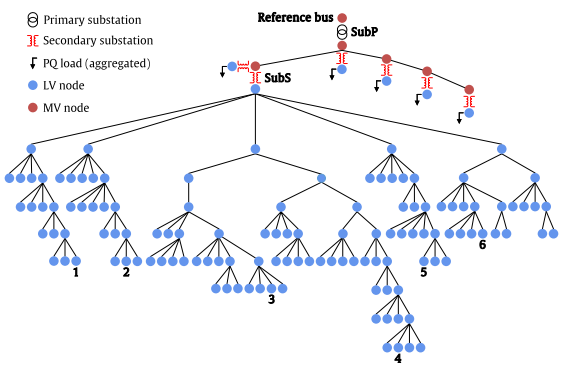


Fig. 1. Real MV-LV network used in the case study.

be created that contains all relevant voltage values. If this information is unavailable, an accurate network model is needed to construct this dataset by using SM consumption data and running power flows.

However, in most cases DSOs have no real-time observability below the secondary substation level. The objective is to provide a probabilistic estimation of such LV states, and more specifically in this study for voltages at nodes 1 to 6, marked on Fig. 1, given different flexibility usage scenarios and varying levels of information availability to the DSO. The network model and its accuracy have been validated with real network measurements. Since no voltage measurements are available from the SMs, AC power flow was used to obtain the network states, which serve as ground truth.

3.3. Flexibility usage scenarios

3.3.1. FS1 — original data

This scenario considers the original residential profiles. Customers are assigned randomly to the leaf nodes of the LV grid and the other five secondary substations. FS1 presents a case with mild flexibility utilization.

3.3.2. FS2 — original data and EVs below adjacent substations

In this scenario the customer placement of FS1 is kept, but EVs are added to each customer below the five adjacent secondary substations of SubS. EV profiles are taken from [25], and it is assumed that users perform smart charging with the objective of minimizing their costs based on retail prices, which follow the day-ahead spot prices of Danish zone DK2. FS2 presents a case with significant use of flexibility, which potentially adds randomness and reduces the correlation between the considered LV nodes and the primary substation.

3.3.3. FS3 — original data and EVs below all substations

In the third scenario, an EV performing smart charging is also added to each customer under SubS. FS3 presents a case with high flexibility-induced load variability in the examined network below SubS, potentially complicating voltage estimation.

3.4. DSO input information scenarios

Observability in LV networks is still rather limited. This work assesses how different levels of DSO information availability affect the performance of probabilistic SE, for each of the aforementioned three FSs. In Fig. 2 an overview of the resulting nine scenarios is given. Below three ISs with increasing data availability are described.

3.4.1. IS1 — low availability

The DSO has access to past customer SM P, Q and V data with a delay of 24 hours, real-time weather data (temperature and solar irradiation) and calendric features, namely weekday vs weekend indicators and time of the day. Additionally, retailer prices are used, which is beneficial for scenarios FS2 and FS3 (smart charging). IS1 assumes zero DSO real-time observability and the used information can be readily available to every DSO without the installation of additional devices.

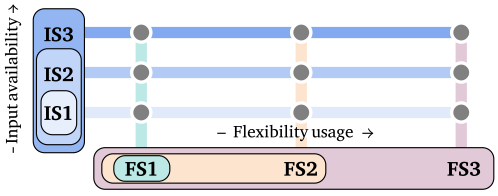


Fig. 2. Overview of the nine scenarios with increasing flexibility usage (x axis) and input availability (y axis). FS1 ⊂ FS2 ⊂ FS3 and IS1 ⊂ IS2 ⊂ IS3.

3.4.2. IS2 — medium availability

In IS2 it is assumed that real-time measurements from the primary substation are available, which is not an unusual operational practice. Therefore, SubP's PQ values are considered as additional features on top of IS1 to construct IS2. Note that adding voltage values from SubP was found to add a negligible benefit to the performance of the models and was thus omitted. However, this may not be the case in the presence of tap-changing transformer actions.

3.4.3. IS3 — high availability

The high availability case IS3 assumes that the DSO also monitors in real time the secondary substation and thus PQ and voltage values from SubS are added as features on top of the ones from IS2.

3.5. Evaluation

In this work, the BNN is benchmarked with QR, which is a straightforward method to conduct probabilistic SE and has shown good performance in many applications [26]. Model performance is evaluated for all flexibility usage scenarios (FS1, FS2 and FS3) and input information scenarios (IS1, IS2 and IS3) on the test dataset. Root mean square error (RMSE) is used for the evaluation of point estimation performance. The popular Pinball and Winkler scores are considered for assessing the reliability, sharpness, and resolution of the probabilistic estimation. These metrics are calculated for each considered network state $j \in \mathcal{J}$ individually. In Section 4 their average, minimum and maximum values are reported. $y_{j,t}$ is the actual value of network state j at the evaluated time step t (out of n steps). $\hat{y}^{\mathrm{m}}_{j,t}$ is the expectation of the predicted value. Note that this expectation corresponds to the median for QR and the mean for the BNN. Finally, $\hat{y}^{q}_{j,t}$ is the predicted qth quantile. RMSE is defined as

$$RMSE_j = \sqrt{\frac{\sum_t^n \left(y_{j,t} - \hat{y}^{\mathrm{m}}_{j,t}\right)^2}{n}}, \quad \forall j \in \mathcal{J}. \tag{8}$$

The Pinball loss function is given by

$$Pinball_j = \begin{cases} (y_{j,t} - \hat{y}^{q}_{j,t})q, & y_{j,t} \geq \hat{y}^{q}_{j,t} \\ (\hat{y}^{q}_{j,t} - y_{j,t})(1-q), & y_{j,t} < \hat{y}^{q}_{j,t}. \end{cases} \tag{9}$$

The average Pinball score for all n steps is calculated for $q = 0.01, \ldots, 0.99$, with a lower value indicating better performance. Finally, the Winkler score is determined for specific prediction intervals (PIs), which constitute the estimate of an interval in which future observations will lie with a certain probability. For a PI $1-\alpha$ the Winkler score is given by

$$Winkler_j = \begin{cases} \delta, & \hat{y}^{-}_{j,t} \leq y_{j,t} \leq \hat{y}^{+}_{j,t} \\ 2(\hat{y}^{-}_{j,t} - y_{j,t})/\alpha + \delta, & y_{j,t} < \hat{y}^{-}_{j,t} \\ 2(y_{j,t} - \hat{y}^{+}_{j,t})/\alpha + \delta, & y_{j,t} > \hat{y}^{+}_{j,t}, \end{cases} \tag{10}$$

where $\hat{y}^{-}_{j,t}$ and $\hat{y}^{+}_{j,t}$ represent the lower and upper PI bounds, respectively. $\delta = \hat{y}^{+}_{j,t} - \hat{y}^{-}_{j,t}$ and $\alpha = 0.1$ because a 90 % PI is considered. A lower Winkler score implies a better PI.

4. Results

In this section the voltage estimation performance of the BNN under varying input availability and flexibility scenarios is evaluated and compared to the QR benchmark. In Section 4.1 a qualitative evaluation is presented, followed by a quantitative assessment in Section 4.2. In Section 4.3 the behavior of aleatoric and epistemic uncertainty is evaluated over a period of 10 months.

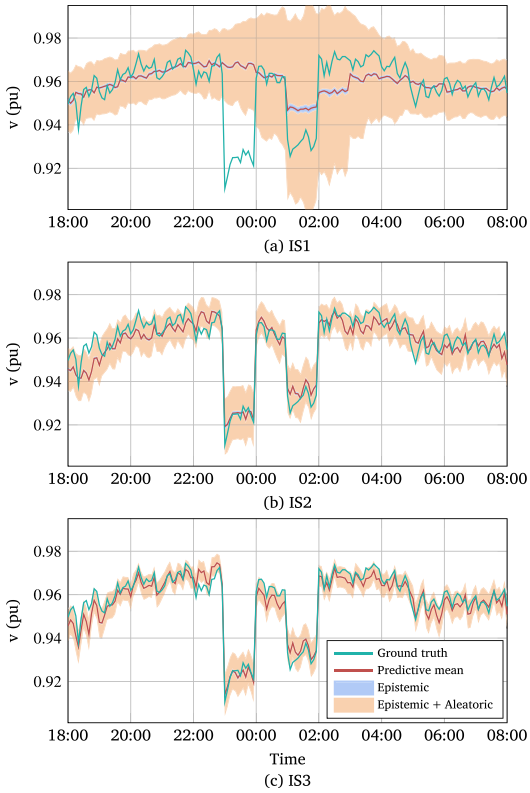


Fig. 3. Comparison of the predictive mean and 90% PI for IS1-IS3 based on a representative excerpt from FS3 and voltage node 4. Selection of node 4 results from proximity to the average RMSE of all predicted node voltages.

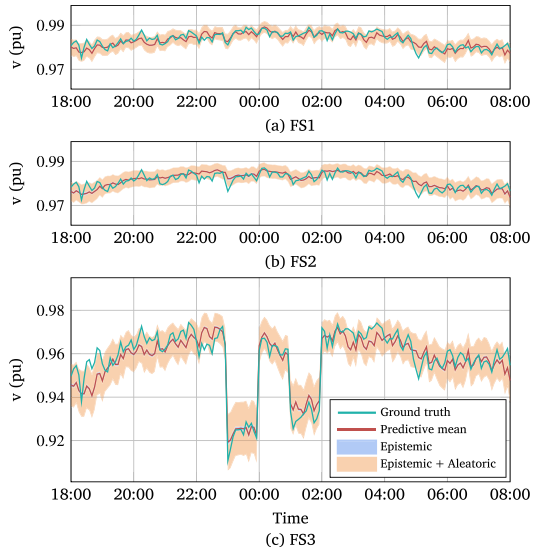


Fig. 4. Comparison of the predictive mean and 90% PI for a representative excerpt of FS1-FS3 based on IS2 and voltage node 4. Selection of node 4 results from proximity to the average RMSE of all predicted node voltages.

4.1. Qualitative evaluation of the voltage estimation

Fig. 3 shows the output of the BNN for a 14-hour period of voltage node 4 under three levels of available input information (IS1-IS3). Case FS3 is considered, where substantial flexibility activations occur in the examined network below SubS. This is evident during periods 23:00-00:00 and 01:00-02:00, where large load increases occur due to EV charging, leading to low voltage values. For all ISs, aleatoric uncertainty is dominating, which can be explained by the comprehensive training dataset. However, an increase of the epistemic uncertainty is noticed during flexibility activations. This can be explained by a lack of knowledge, as they constitute only a small subset of the training data. For all ISs the BNN understands that uncertainty is higher during times of flexibility activations, resulting in an extended PI. Under IS1, no real-time network information is considered. Thus, voltage drops are only estimated based on retailer prices, time of the day and flexibility activations at the previous day, represented in the day-before SM data. The retailer price was found to be a weakly informative feature, due to the difficult-to-capture charging optimization process and customer uncertainty. However, as flexibility was activated on the previous day between 01:00 and 02:00 and EV charging is most likely between 01:00 and 03:00, the model is able to estimate the occurrence of the second voltage drop, while the first one is entirely missed. Due to lack of real-time network information, the model fails at estimating the severity of the voltage drops. It can be concluded that IS1 is not sufficient to estimate flexibility-driven voltage drops. By incorporating primary (IS2) or primary and secondary (IS3) substation measurements, the model provides PIs which successfully capture voltage drops, as the increased consumption which causes them is reflected on the substation's power and voltage. An additional advantage of including secondary substation measurements can be seen from the tighter PI.

Next, only IS2 is considered, as the typical scenario for most DSOs. In Fig. 4 results are depicted for the same period under the three flexibility usage scenarios (FS1-FS3). In all scenarios epistemic uncertainty is almost entirely reduced by the BNN learning process. This leads to the conclusion that 11 months of training data is sufficient to exploit available information. The remaining uncertainty is mainly introduced by randomness. In FS1 load variability and flexibility usage are limited. Thus, the BNN is able to provide good voltage estimation with a very tight PI. Similar results are obtained under FS2. Visual inspection suggests that frequent flexibility activations below adjacent substations have small impact on the estimation. A quantitative evaluation follows in Section 4.2. For FS3, substantial flexibility is active in the examined LV network. The wider PI indicates that estimations are less accurate. Moreover, an increase of the PI occurs during flexibility activations, which allows the BNN to successfully capture the voltage drops.

4.2. Quantitative evaluation and benchmark comparison

4.2.1. Point estimation performance

The results in terms of RMSE accuracy for the different scenarios are shown in Fig. 5. The scaled RMSE values are plotted, with unity corresponding to an error of 0.0106 pu. As assumed, point estimates improve with the level of input information (IS1 to IS3) in all FSs. For FS3 the decrease of RMSE between IS1 and IS2 is comparatively large, as without real-time network measurements both the BNN and QR cannot accurately capture the frequent flexibility activations in the LV network (see Fig. 3(a)). Thus, the inclusion of primary substation information reduces RMSE significantly by 42%. As assumed from the qualitative evaluation, FS1 and FS2 show similar point estimation performance. However, a large relative RMSE increase can be noticed for FS3. The large presence of flexibility activations in the LV network increases load variability and uncertainty, complicating voltage estimation. Another

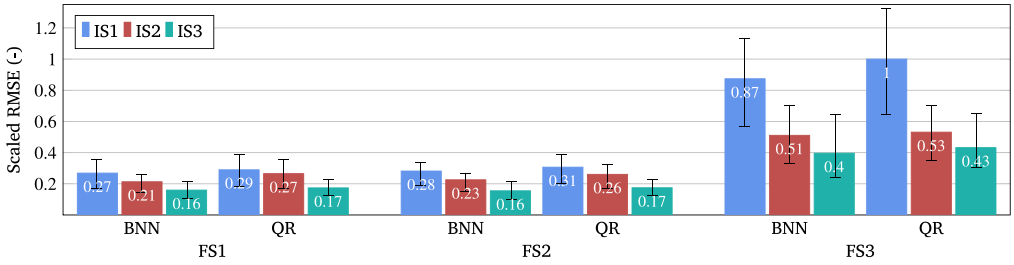


Fig. 5. Scaled RMSE of BNN and QR for FS1-FS3 and IS1-IS3 averaged over all estimated node voltages. Min/max values are indicated by the black bars.

finding is the superior performance of the BNN compared to the QR for all FSs and ISs. On average, the BNN improves RMSE by 10.3 %.

4.2.2. Probabilistic estimation performance

By considering Pinball and Winkler scores, probabilistic estimation performance is evaluated for reliability, sharpness and resolution, and reported in Fig. 6 for FS1-FS3 and IS1-IS3. Values are scaled to a Pinball score of 0.00328 pu and a Winkler (90 %) score of 0.0478 pu, respectively. A comparison between FS1 and FS2 shows that both the BNN and QR for all ISs provide credible PIs also in a scenario with frequent flexibility activation below adjacent secondary substations. Although for FS2 with IS2 both models, compared to FS1, show a performance decrease in the range of 5 %, it can be concluded that frequent flexibility activations below adjacent substations only slightly increase randomness, and thus barely deteriorate the correlation between primary substation measurements and node voltages in the examined LV network. By including secondary substation measurements (IS3) this effect can be fully mitigated. In contrast, for frequent flexibility activations in the LV network (FS3) both scores indicate a strong performance decrease for all ISs. However, for FS3 the performance gain of the BNN by including secondary substation measurements (IS3) is comparatively large. While for FS1 and FS2 Pinball decreases by approximately 10 percentage points, a decrease by 22 percentage points is achieved in FS3. A similar behavior can be derived from Winkler (90 %). As a result, for IS3 the BNN is able to keep both scores in a comparable performance range as for FS1 and FS2 with IS1 and IS2. It can be concluded that, especially under frequent flexibility activations below the same secondary substation, incorporating secondary substation measurements adds great value to probabilistic LVSE. Moreover, although Fig. 6 shows a strong relative performance decrease for FS3, Fig. 3 indicates that already for IS2 the BNN provides credible PIs that successfully capture sudden voltage drops.

Another finding is that the considered deterministic and probabilistic performance metrics (see Figs. 5 and 6) show a similar behavior. Thus, the BNN outperforms QR across all scenarios, also in terms of probabilistic SE. The average performance improvement for Pinball is 17.2 % and for Winkler 13.4 %. In this context, it should be considered that the large size of the training dataset allowed to reduce epistemic uncertainty to a negligible part (see Fig. 3). In cases of less training data or more frequently changing conditions, such as grid topology changes or newly added EV fleets, it can be assumed that the BNN by capturing both aleatoric and epistemic uncertainty has an even stronger advantage.

4.3. Evaluation of uncertainty behavior

To shed light on the behavior of aleatoric and epistemic uncertainty under varying conditions, the BNN was trained on January and February data, and used to estimate voltage node 4 for the remaining 10

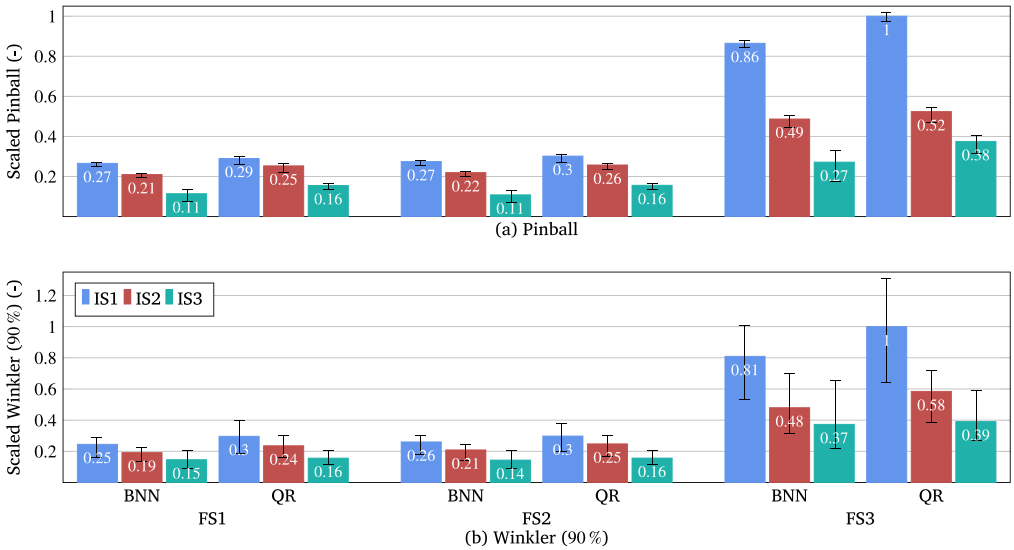


Fig. 6. Scaled Winkler and Pinball of BNN and QR for FS1-FS3 and IS1-IS3 averaged over all estimated voltages. Black bars indicate min/max values.

months of the year. Note that this is a showcase which intentionally does not consider model retraining. In Fig. 7 both uncertainty types are shown for FS1 and IS2. In accordance with the previous findings (see Fig. 3), epistemic uncertainty is low for a recently trained model. However, large peaks can be noticed, which correlate with solar radiation and result from lack of PV generation in the training data. Between April and October an increasing trend of epistemic uncertainty is visible. This is attributed to shifting away from the training data distribution, as a result of higher ambient temperatures. In practice, frequent retraining would keep epistemic uncertainty small during the entire year. From November on, epistemic uncertainty approaches zero again, as data moves towards the distribution of the training set. Due to little PV generation, no peaks occur in December. Aleatoric uncertainty decreases between April and October. In fact, it exhibits a strong correlation with substation loadings, which are lower during this period because of lower heating demand. The fact that aleatoric uncertainty is higher for a recently trained model shows that, in contrast to epistemic uncertainty, frequent retraining cannot reduce it as it stems from randomness rather than lack of knowledge.

The following conclusions can be drawn. For a recently trained model, aleatoric uncertainty is dominant, justifying the application of models capable of capturing it. Without the epistemic counterpart, uncertainty induced by sudden or ongoing changes is not quantified, which can be seen from the PV-induced peaks and trend in Fig. 7, respectively. As such changes are present in LV networks, a model that considers epistemic uncertainty is beneficial for LVSE, both for improved uncertainty quantification and better interpretability. While sudden peaks are an indicator for unknown events, an increasing trend of epistemic uncertainty indicates retraining need. By providing accurate quantification of both uncertainties, the BNN is seen as a valuable approach for LVSE.

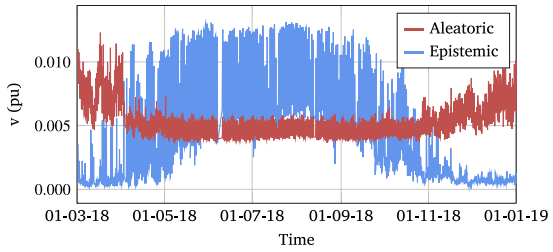


Fig. 7. Epistemic and aleatoric uncertainty of FS1 for a BNN trained on January and February 2018 with IS2. Uncertainties correspond to the 90 % PI. The few gaps are the result of short periods of missing SM data.

5. Conclusion

In this work, uncertainty quantification in LVSE is investigated for various flexibility usage scenarios. For that purpose, a BNN capable of capturing both epistemic and aleatoric uncertainty is implemented and compared to a QR benchmark. Estimation and uncertainty quantification performance are systematically evaluated from a qualitative and quantitative perspective, and for multiple scenarios of input information availability and flexibility utilization.

Results show that flexibility activations below adjacent secondary substations have only minor impact on LVSE, while activations below the same substation decrease performance significantly. However, it is also shown that including secondary substation measurements allows retaining an acceptable degree of performance. By dynamically extending the PI during flexibility activation periods, the BNN is able to capture flexibility-induced voltage drops, enabling a more reliable LVSE. Moreover, the model shows superior performance compared to the QR benchmark for all considered cases. By considering and differentiating between epistemic and aleatoric uncertainty, it improves interpretability, as it provides insights in occurrence of unknown events or retraining need. Future work will be directed towards applying the BNN to scenarios such as uncontrolled EV charging and other flexibility control strategies, and for probabilistic LV state forecasting.

CRediT authorship contribution statement

Nils Müller: Conceptualization, Methodology, Software, Validation, Writing – original draft, Visualization. **Samuel Chevalier:** Writing – review & editing. **Carsten Heinrich:** Methodology, Software. **Kai Heussen:** Writing – review & editing, Supervision, Funding acquisition. **Charalampos Ziras:** Conceptualization, Methodology, Software, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is partly funded by the Innovation Fund Denmark (IFD) under File No. 91363, and by the INTERPRETER project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 864360.

References

[1] C.-H. Lo, N. Ansari, Decentralized controls and communications for autonomous distribution networks in smart grid, IEEE Trans. Smart Grid 4 (1) (2013) 66–77, http://dx.doi.org/10.1109/TSG.2012.2228282.

[2] E. Hillberg, A. Zegers, B. Herndler, S. Wong, J. Pompee, J.-Y. Bourmaud, S. Lehnhoff, G. Migliavacca, K. Uhlen, I. Oleinikova, et al., Flexibility needs in the future power system, ISGAN (2019).

[3] S. Habib, M.M. Khan, F. Abbas, L. Sang, M.U. Shahid, H. Tang, A comprehensive study of implemented international standards, technical challenges, impacts and prospects for electric vehicles, IEEE Access 6 (2018) 13866–13890, http://dx.doi.org/10.1109/ACCESS.2018.2812303.

[4] A. Primadianto, C.-N. Lu, A review on distribution system state estimation, IEEE Trans. Power Syst. 32 (5) (2017) 3875–3883, http://dx.doi.org/10.1109/TPWRS.2016.2632156.

[5] K. Dehghanpour, Z. Wang, J. Wang, Y. Yuan, F. Bu, A survey on state estimation techniques and challenges in smart distribution systems, IEEE Trans. Smart Grid 10 (2) (2019) 2312–2322, http://dx.doi.org/10.1109/TSG.2018.2870600.

[6] M. Pertl, P.J. Douglass, K. Heussen, K. Kok, Validation of a robust neural real-time voltage estimator for active distribution grids on field data, Electr. Power Syst. Res. 154 (2018) 182–192, http://dx.doi.org/10.1016/j.epsr.2017.08.016.

[7] J.-H. Menke, N. Bornhorst, M. Braun, Distribution system monitoring for smart power grids with distributed generation using artificial neural networks, Int. J. Electr. Power Energy Syst. 113 (2019) 472–480, http://dx.doi.org/10.1016/j.ijepes.2019.05.057.

[8] E. Manitsas, R. Singh, B.C. Pal, G. Strbac, Distribution system state estimation using an artificial neural network approach for pseudo measurement modeling, IEEE Trans. Power Syst. 27 (4) (2012) 1888–1896, http://dx.doi.org/10.1109/TPWRS.2012.2187804.

[9] M. Ferdowsi, A. Benigni, A. Löwen, B. Zargar, A. Monti, F. Ponci, A scalable data-driven monitoring approach for distribution systems, IEEE Trans. Instrum. Meas. 64 (5) (2015) 1292–1305, http://dx.doi.org/10.1109/TIM.2015.2398991.

[10] T. Zufferey, S. Renggli, G. Hug, Probabilistic state forecasting and optimal voltage control in distribution grids under uncertainty, Electr. Power Syst. Res. 188 (2020) http://dx.doi.org/10.1016/j.epsr.2020.106562.

[11] R. Bessa, G. Sampaio, V. Miranda, J. Pereira, Probabilistic low-voltage state estimation using analog-search techniques, in: 2018 Power Systems Computation Conference (PSCC), 2018, pp. 1–7, http://dx.doi.org/10.23919/PSCC.2018.8443074.

[12] M. Sun, T. Zhang, Y. Wang, G. Strbac, C. Kang, Using Bayesian deep learning to capture uncertainty for residential net load forecasting, IEEE Trans. Power Syst. 35 (1) (2020) 188–201, http://dx.doi.org/10.1109/TPWRS.2019.2924294.

[13] K.R. Mestav, J. Luengo-Rozas, L. Tong, Bayesian state estimation for unobservable distribution systems via deep learning, IEEE Trans. Power Syst. 34 (6) (2019) 4910–4920, http://dx.doi.org/10.1109/TPWRS.2019.2919157.

[14] Y. Huang, Q. Xu, C. Hu, Y. Sun, G. Lin, Probabilistic state estimation approach for AC/MTDC distribution system using deep belief network with non-Gaussian uncertainties, IEEE Sens. J. 19 (20) (2019) 9422–9430, http://dx.doi.org/10.1109/JSEN.2019.2926089.

[15] L.V. Jospin, W. Buntine, F. Boussaid, H. Laga, M. Bennamoun, Hands-on Bayesian neural networks–a tutorial for deep learning users, 2020, arXiv preprint arXiv:2007.06823.

[16] M. Vladimirova, J. Verbeek, P. Mesejo, J. Arbel, Understanding priors in Bayesian neural networks at the unit level, in: International Conference on Machine Learning, PMLR, 2019, pp. 6458–6467.

[17] A.G. Wilson, P. Izmailov, Bayesian deep learning and a probabilistic perspective of generalization, 2020, arXiv preprint arXiv:2002.08791.

[18] M.D. Hoffman, D.M. Blei, C. Wang, J. Paisley, Stochastic variational inference, J. Mach. Learn. Res. 14 (1) (2013).

[19] A. Kendall, Y. Gal, What uncertainties do we need in Bayesian deep learning for computer vision? Adv. Neural Inf. Process. Syst. 30 (2017).

[20] J.V. Dillon, I. Langmore, D. Tran, E. Brevdo, S. Vasudevan, D. Moore, B. Patton, A. Alemi, M. Hoffman, R.A. Saurous, Tensorflow distributions, 2017, arXiv preprint arXiv:1711.10604.

[21] R. Koenker, K.F. Hallock, Quantile regression, J. Econ. Perspect. 15 (4) (2001) 143–156, http://dx.doi.org/10.1257/jep.15.4.143.

[22] S. Seabold, J. Perktold, Statsmodels: econometric and statistical modeling with Python, in: Proceedings of the 9th Python in Science Conference, 57, (61) Austin, TX, 2010, pp. 10–25080.

[23] C. Heinrich, C. Ziras, A.L. Syrri, H.W. Bindner, EcoGrid 2.0: A large-scale field trial of a local flexibility market, Appl. Energy 261 (2020) http://dx.doi.org/10.1016/j.apenergy.2019.114399.

[24] C. Ziras, C. Heinrich, M. Pertl, H.W. Bindner, Experimental flexibility identification of aggregated residential thermal loads using behind-the-meter data, Appl. Energy 242 (2019) 1407–1421, http://dx.doi.org/10.1016/j.apenergy.2019.03.156.

[25] Department for transport, electric chargepoint analysis 2017, 2021, https://www.gov.uk/government/statistics/electric-chargepoint-analysis-2017-domestics. Accessed: 2021-09-14.

[26] R. Koenker, V. Chernozhukov, X. He, L. Peng (Eds.), Handbook of quantile regression, CRC Press, 2017, http://dx.doi.org/10.1201/9781315120256.

PAPER G

Nils Müller, Carsten Heinrich, Kai Heussen, Henrik W. Bindner

Unsupervised detection and open-set classification of fast-ramped flexibility activation events

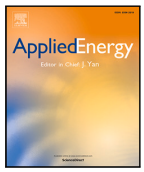
Müller, N., C. Heinrich, K. Heussen, and H. W. Bindner, "Unsupervised detection and open-set classification of fast-ramped flexibility activation events," in *Applied Energy*, vol. 312, 2022, doi: 10.1016/j.apenergy.2022.118647.

Applied Energy 312 (2022) 118647

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy



Unsupervised detection and open-set classification of fast-ramped flexibility activation events

Nils Müller *, Carsten Heinrich, Kai Heussen, Henrik W. Bindner

Center for Electric Power and Energy, Technical University of Denmark, Elektrovej, Building 325, 2800 Kgs. Lyngby, Denmark

ARTICLE INFO

Keywords:
Flexibility
Event detection
Open-set classification
Active distribution networks
Machine learning
Electrification

ABSTRACT

The continuous electrification of the mobility and heating sectors adds much-needed flexibility to the power system. However, flexibility utilization also introduces new challenges to distribution system operators (DSOs), who need mechanisms to supervise flexibility activations and monitor their effect on distribution network operation. Flexibility activations can be broadly categorized to those originating from electricity markets and those initiated by the DSO to avoid constraint violations. Coinciding electricity market driven flexibility activations may cause voltage quality or temporary overloading issues, and the failure of flexibility activations initiated by the DSO might leave critical grid states unresolved. This work proposes a novel data processing pipeline for automated real-time identification of fast-ramped flexibility activation events. Its practical value is twofold: (i) potentially critical flexibility activations originating from electricity markets can be detected by the DSO at an early stage, and (ii) successful activation of DSO-requested flexibility can be verified by the operator. In both cases the increased awareness would allow the DSO to take counteractions to avoid potentially critical grid situations. The proposed pipeline combines techniques from unsupervised detection and open-set classification. For both building blocks feasibility is systematically evaluated and proofed on real load and flexibility activation data.

1. Introduction

Renewable electricity and electrification are key pillars of global efforts to eliminate fossil fuels in the energy supply. The European goal of carbon neutrality in 2050 is reported to require increased shares of renewable energy and continued electrification of the mobility and heating sectors [1]. This trend will further increase uncertainty and volatility in distribution networks (DNs). Thus, active management of DNs based on the emerging smart solutions for monitoring, control, and communication is seen as a requirement for distribution system operators (DSOs) [2]. With the improving capability and affordability of information and communication technology (ICT) the implicit or explicit utilization of local consumption flexibility, commonly referred to as demand response, is becoming more attractive. By requesting a load deviation of flexible units during a certain time period such as peak hours, referred to as flexibility activation (FA) event, DSOs can use local flexibility to avoid or postpone grid reinforcements [3]. However, applying local end user flexibility for mitigation of potentially critical network states makes grid operation and security partly dependent on the reliability of third parties. Thus, real-time detection of FA events is desirable for DSOs to verify successful activation, and initiate other measures in case of a failure. Moreover, FA events are not exclusive to DSOs, due to activations originating from electricity markets. On the one hand, balance responsible parties could request and activate flexibility for portfolio optimization. On the other hand, controllable heat pumps and electric vehicles may simultaneously react to price signals with a sudden change of power consumption [4,5]. As a result, DSOs are not aware of all FA events affecting their network. If not detected at an early stage, such fast-ramped FA events could trigger transformer or line protections due to high coincidence [6,7] or load rebound effects [8–11]. The resulting disconnection of customers could lead to high social and financial cost.

The increasing deployment of measuring devices, such as micro phasor measurement units and smart meters (SMs), increase observability of DNs and thus provide the data basis for identification of FA events. However, real-time identification of FA events is challenged by different practical problems. The infrequent occurrence of FA events limits the available data required for implementation of supervised detection methods. Moreover, the operation of active DNs is influenced by a variety of rare or even unseen event classes such as line faults, topology changes or communication failures [12,13]. Thus, FA event identification also requires differentiation between unknown event classes and FA events. This questions the use of widely applied closed-set

* Corresponding author.
E-mail address: nilmu@elektro.dtu.dk (N. Müller).

https://doi.org/10.1016/j.apenergy.2022.118647
Received 30 November 2021; Accepted 24 January 2022
Available online 24 February 2022

Fig. 1. Schematic overview of the proposed EIP for FA events in active DNs.

(CS) classifiers that will falsely classify unknown event classes, due to their inability of rejecting these. Another challenge for real-time FA event identification is seen in the central data processing, e.g. via cloud computing. Already today the integration of SM data in real-time power system operation is limited by communication instead of meter-recording capability [14]. Upgrading communication networks entails a high economic burden. Moreover, long communication paths increase the possibilities for false data injection attacks and other fraudulent modification of data [15,16]. One approach to overcome the drawbacks of central data processing is seen in a distributed event identification architecture based on edge or fog computing [17].

The described challenges set specific requirements to the approach and implemented techniques for FA event identification. However, a formulation of requirements is missing, complicating the development of appropriate strategies and methods for identifying FA events.

In this work, a novel event identification pipeline (EIP) for fast-ramped FA events is proposed. A schematic overview of the proposed pipeline is depicted in Fig. 1. The data processing pipeline is based on unsupervised detection and open-set (OS) classification algorithms, suitable for application in a distributed event identification architecture. The scheme and algorithm selection are based on a thorough requirements analysis. The core contributions of this work are the systematic selection of processing algorithms and their evaluation on real load and FA event data.

1.1. Related work

To the extent of the authors' knowledge, this is the first work on detection and classification of FA events. Works on thematically-related topics are presented first, followed by a presentation of methodologically-related works. In both cases literature on detection and classification is discussed separately.

1.1.1. Thematically-related works

A frequently studied topic in power system literature is unsupervised anomaly detection in energy time series data. To detect anomalies, most works train models predicting normal behavior. A data point is declared an anomaly if deviation between model prediction and ground truth exceeds a predefined threshold. Various models such as variational autoencoder [18], hierarchical temporal memory (HTM) [19], autoregressive integrated moving average (ARIMA) or long short-term memory [20] are applied. None of these works consider FA events as anomaly. Moreover, most works assume anomaly-free training data for learning of the normal behavior. Some works exist on flexibility detection on building or device level, which try to quantify the flexible load potential in load data of individual devices or buildings [21,22]. Although the name suggests similarity, the problem under investigation is different to the present work, as these works actually investigate flexibility potential identification.

The topic of event classification in DNs has been studied intensely [23]. Most works investigate the multi-class CS classification problem. Literature considering OS classification in a power system context is rare. Lazzaretti et al. [24] first applied one-class classifiers for automatic oscillography classification under existence of unseen event classes in two different approaches. The first one considers a single one-class classifier for modeling the boundary of multiple known classes. A separate multi-class classifier is applied to differentiate between the known classes. In the second approach each class is modeled with a separate one-class classifier. In the following years other works considered one-class classifiers for detection of new classes [25,26]. Although these methods in general can be used for OS classification, the different problem setting results in a comparatively low detection performance [27,28]. To the best of the authors' knowledge, the literature provides no work applying classifiers inherently made for the OS problem to event classification in active DNs. With respect to the proposed EIP, some works on event classification exist that assume an upstream detection step [29,30]. However, none of these works describe how input samples for the classifier are generated from detector results, rather examining event classification for existing samples.

1.1.2. Methodologically-related works

Similar to literature on anomaly detection in energy time series data, multiple works propose forecasting-based unsupervised anomaly detection [31–33]. In most cases, the euclidean distance between point forecast and ground truth is used to flag anomalies based on a defined threshold. In [34], the authors propose the use of a convolutional neural network (CNN) as the time series forecaster. According to the authors, the proposed method can be trained on comparatively few training data and without removing anomalies from the training dataset. A novel approach on anomaly detection in time series data is proposed in [35]. The authors introduce the use of the spectral residual (SR) algorithm from saliency detection in computer vision for unsupervised anomaly detection in time series.

In contrast to the traditional CS classification problem, less literature exists on OS classification. Scheirer et al. [36] first formalized the OS classification problem and proposed the 1-vs-Set machine as a preliminary solution. Since then various methods such as distance-based [37], margin distribution-based [38] or generation-based [39] OS classifiers were proposed. In [27], the authors provide a systematic categorization of OS classification techniques, and compare a number OS classifiers on popular benchmark datasets.

1.2. Contribution and paper structure

The main contributions of this work are as follows:

- Proposal of a novel EIP based on unsupervised detection and OS classification.
- First work on detection and classification of FA events.
- First application of an OS classifier to events in active DNs.
- Introduction of a new performance metric for the evaluation of real-time detection of FA events.
- Systematic demonstration and evaluation of unsupervised detection and OS classification of fast-ramped FA events as the main building blocks of the proposed pipeline based on real load and FA event data.

The remainder of the paper is structured as follows: In Section 2 requirements for FA event identification in active DNs are evaluated and strategies are proposed. Section 3 provides a description of models and methods. In Section 4 the experimental setup is presented, including the dataset under investigation, data preparation for model development and evaluation, and applied performance metrics. In Section 5 results are presented and discussed followed by a conclusion and a view on future work in Section 6.

2. Requirements and strategies for FA event identification

Identifying FA events in active DNs comes with specific requirements not only concerning the general approach, but also the implemented algorithms. Additional requirements at algorithmic level are introduced by the consideration of event identification based on a distributed architecture. In the following, the identified requirements are presented. Based on the requirements analysis, the concept of the proposed EIP as well as the selection of specific models for the main building blocks of the pipeline are justified. A detailed explanation of the implemented models and the proposed pipeline follows in Section 3.

The problem is limited to fast-ramped load reduction and load increase FA events with a length of up to 3 hours. Moreover, the aggregated active power load profile is assumed to represent averaged active power measurements on secondary substation level or aggregated SM data collected in a data hub on neighborhood level. In both scenarios a large low-voltage feeder is considered. The core of the concept is the separation of the event identification task into an unsupervised event detection and a supervised classification task, resulting in the proposed EIP. Besides the two main building blocks, an event sampler is required to prepare event observations for the classifier based on the results of the event detector.

2.1. Real-time identification

A key requirement for identification of FA events is real-time capability. Real-time identification allows DSOs to take immediate counteractions in cases where FAs could result in critical situations, such as congestions and under or over-voltages. Supervised detection or classification of time series events usually requires as input the entire time series sample [40]. For real-time event identification this becomes a fundamental problem. Existing early classification techniques come at the cost of decreased accuracy [40], and are not applicable to an OS classification problem. In the proposed pipeline the problem of prediction delay is addressed by separating event identification into two consecutive steps. The use of an unsupervised, point-wise event detector allows for immediate flagging of abnormal data points in real-time. Although this cannot solve the intrinsic problem of supervised classification being dependent on multiple data points of an event, information extraction is improved. Instead of identifying an event at the end of its occurrence, with the presented EIP DSOs will immediately be aware of the existence of a deviation from normal operation, followed by an ex-post classification of the event.

2.2. Model development based on limited and partly-labeled training data

FAs in active DNs constitute rare events. Thus, comprehensive datasets of FA events for supervised methods are difficult to obtain. The heterogeneity of DNs and flexibility portfolios brings additional challenges for acquiring datasets, since characteristics of FAs will differ for different networks. In contrast, unsupervised event detection does not require datasets of FA events. Instead, most works on unsupervised event detection in energy time series data, such as [18], assume event-free training data to learn a representation of the normal behavior. However, existing training data will most likely contain events, since manual removing is a time-consuming and impractical process [41] and DSOs might not be aware of all FAs (see Section 1). In the proposed pipeline a persistence forecast-based detector is considered, which is not dependent on event-free training data. By applying an unsupervised detector with no demand for event-free training data the dependency on labeled training data is reduced to the classification step. Therefore, compared to an one-step event identification approach, the proposed pipeline maintains event detection capability also in scenarios without labeled training data, maximizing information extraction.

2.3. Lightweight models for event identification

Distributed event identification in an edge or fog computing scheme requires models and methods to be lightweight. The vast number and limited processing power of edge devices as well as the continuously growing amount of data sets time and resource constraints to the development, operation and maintenance of models and methods. Therefore, a key requirement for FA event identification is seen in keeping computational and maintenance efforts, such as periodical re-training, at a minimum. This is considered for the proposed concept in several ways. First of all, the proposed pipeline entirely works with delta encoded data. Delta encoding is a technique for data compression based on differencing sequential data, reducing data communication and storage load [42]. By working with differenced data the proposed pipeline can directly be applied in a system which uses delta encoding for data compression. Both the detection and classification model within the pipeline only retrieve features from the univariate load time series. No additional information such as weather or market price data are considered, reducing the requirement for data communication and the dimensionality of the detection and classification problem. The use of a simple persistence forecast keeps size and computational effort of the proposed detector at a minimum, and avoids the need for frequent re-training. In the proposed pipeline the classifier only gets activated if the detector has detected a deviation from normal behavior above a predefined threshold. This event-triggered scheme avoids continuous running of the classifier which reduces the required processing power.

For OS classification the extreme value machine (EVM) model is selected as it comes with several features, making it a comparatively lightweight classifier [38]. EVM is capable of incremental learning which allows for efficient model updating without time and computation intensive re-training. Moreover, the model reduction strategy of EVM discards redundant data points within a class of training points, allowing for limitation of model size and classification time as dataset size increases.

2.4. Handling multiple and unknown event classes

In active DNs a large variety of events with various backgrounds, such as faults and switching actions, can occur. While in this work the detection performance is evaluated on the basis of fast-ramped FA events, in principle an *unsupervised* detector also allows for detection of other fast-ramped events.

Although traditional CS classifiers can differentiate between multiple known event classes, introducing new unknown classes will lower classification performance drastically, since observations of unknown classes are wrongly assigned to one of the classes the classifier was trained on [27]. Given that many event classes only occur rarely and new ones might emerge, e.g. due to changes in grid topology, assuming that training data includes sufficient observations to describe all existing events is considered an unrealistic assumption. An important requirement for FA event identification is therefore seen in the capability to differentiate between FA events and other event classes by either recognizing known or rejecting unknown event classes. For that purpose, an OS classifier is specifically selected.

2.5. Extension to new event classes

For many other events, such as high-impedance faults or sensor failures, real-time identification would add additional value to DSOs. However, adding additional identification models for every event would again violate the aforementioned time and resource constraints. For this reason, a central requirement is seen in the capability of a model to be extended to identification of additional events while respecting computational and maintenance effort limitations. The use of an unsupervised event detector allows for the detection of other fast-ramped events beyond the considered FA events. To extend the event

identification problem to slow-ramped events, the persistence forecast-based detector needs to be replaced or extended. However, due to the modular fashion of the proposed pipeline an extension to slow-ramped events can be achieved without affecting the subsequent classification step. With regard to the classification step, the implementation of an OS classifier with rejection and incremental learning capability facilitates the extension to new event classes: indicating observations of unknown classes allows for automated collection, facilitating the manual preparation of new event classes. Once sufficient observations of a new class are collected, the EVM model enables efficient incorporation under an incremental update mechanism. The model reduction strategy makes EVM a sparse OS classifier with limited model size also under extension with new classes.

3. Model description

This section first formulates the problem of unsupervised event detection and OS classification and describes the implemented models. Thereafter, the proposed EIP is explained.

3.1. Unsupervised detection of FA events

In this subsection, the unsupervised event detection problem is formulated. Subsequently, the implemented models for unsupervised event detection are described, namely HTM, ARIMA, CNN, SR and Persistence detector.

3.1.1. Problem formulation

The unsupervised detection of FA events is formulated as a point anomaly detection problem in univariate time series data. A point anomaly is considered a data point that significantly deviates from its expected value. Given a univariate time series $\boldsymbol{X} = \{x_1, x_2, \dots, x_N \mid x_i \in \mathbb{R} \forall i\}$, a data point x_t at time t is declared an anomaly if the anomaly score s_t, defined as the distance to the expected value $\hat{x}_t$, exceeds a predefined threshold τ:

$$s_t = |x_t - \hat{x}_t| > \tau \tag{1}$$

Although all detectors within this work follow different strategies to calculate s_t, they are all either explicitly or implicitly based on fitting a model to the normal behavior. Given the univariate time series $\boldsymbol{X}$, all detection models either aim at learning a mapping function $\boldsymbol{\Phi}$ from historical time steps to the next time step

$$\hat{x}_t = \boldsymbol{\Phi}([x_{t-w}, \dots, x_{t-1}]), \tag{2}$$

or a direct mapping function Θ from historical time steps to the anomaly score of the next time step

$$s_t = \Theta([x_{t-w}, \dots, x_{t-1}]), \tag{3}$$

where w is the size of the history window, which can vary for the different detectors.

3.1.2. HTM detector

HTM is a machine learning technique that is based on the structural and algorithmic properties of the neocortex [43]. Compared to many other methods, HTM comes with several advantages that simplify handling of the anomaly detection problem. These include continuous online learning capability, robustness to noise, and applicability without case-specific hyperparameter tuning. Thus, HTM can be applied without model training and selection on separate training and validation datasets and frequent re-training. For a detailed description of HTM the reader is referred to [44].

The implementation of HTM includes an internal calculation of anomaly scores such that the HTM detector overall follows (3). HTM comes with approximately 30 model configuration parameters. For the anomaly detection problem, a set of optimal parameters is provided in the supplementary material of [31], which is applied in this work.

Table 1
Summary of hyperparameters of the CNN model.

Hyperparameter	Search space	Selected value
History window size	144, 288, 576, 1152	288
Forecasting horizon	1	1
Learning rate	[0.00001, 0.1]	1e−5
L2 weight regularization	[0.0001, 0.1]	0.01
Dropout rate	[0, 0.2]	0.2
Batch size	10, 50, 100, 500	50
Maximum number of epochs	2000	2000
Number of filters	10, 30, 50, 70	50
Kernel size	2, 3, 4	3
Neurons in the fully connected layer	10, 50, 100, 150	100
Early stopping patience	10, 50, 100	50
Activation function	ReLU, Tanh	ReLU

3.1.3. ARIMA detector

ARIMA models are widely known and applied for time series forecasting [45]. As they use lagged values to forecast future behavior, ARIMA models can be applied to learn a mapping function according to (2). To enable application to seasonal data, this work considers modeling of seasonal patterns based on Fourier terms as proposed in [46]. Model training and selection is based on the first 10 days of the dataset. In a pre-processing step, the distribution of the training data is centered on a mean of $\mu = 0$ and a standard deviation of $\sigma = 1$. Based on the Akaike information criterion (AIC) a $(p, d, q) = (8, 1, 2)$ ARIMA model is chosen. After the initial training, the autoregressive and moving average parameters are updated with every new incoming observation. Every 14 days an entirely new ARIMA model is selected based on the AIC, resulting in a new selection of p, d and q.

3.1.4. CNN detector

CNNs [47] are a specialized class of artificial neural networks that can capture temporal patterns in time series data. Thus, they can be applied to provide a mapping function according to (2). An advantage of CNNs is the good performance also for small training datasets (as opposed to many other deep learning techniques) even without removing anomalies from the training data [34].

To define the architecture and hyperparameters of the CNN, extensive empirical experiments are conducted based on the first 10 days of the dataset. The model is trained on predicting the difference $x_{\Delta,t} = x_t - x_{t-1}$ instead of x_t to avoid learning a local minimum given by the persistence forecast $\hat{x}_t = x_{t-1}$. To enable faster convergence the training data is centered on a mean of $\mu = 0$ and a standard deviation of $\sigma = 1$. While the first 7 days are used as an initial training dataset, the remaining 3 days are used for validation. The resulting CNN architecture consists of three convolutional/max-pooling pairs followed by a fully connected layer. An overview of the main hyperparameters is given in Table 1. After the initial training and model selection phase the CNN is re-trained every 14 days based on the previous data. To avoid overfitting, a combination of early stopping, L2 regularization and dropout is applied.

3.1.5. SR detector

SR [48] is an unsupervised algorithm for visual saliency detection in computer vision. Recently, Ren et al. [35] proposed the use of SR for anomaly detection in time series data, motivated by the similarity to visual saliency detection. An advantage of SR is the comparatively small number of hyperparameters. SR performs a mapping of previous data points to the next time step according to (2). However, the mapping is conducted within a saliency map representation of a sliding sequence $\boldsymbol{x} = \{x_1, x_2, \dots, x_N\}$, which is based on Fourier transformation and calculation of the spectral residual. Within the saliency map a local average of previous data points is compared to the actual value to declare anomalies similar to (1). As this work is concerned with real-time detection, only the most recent data point x_N of sequence $\boldsymbol{x}$ is

evaluated. Since detection performance of SR improves for data points located in the center of $\boldsymbol{x}$, Ren et al. [35] propose to add estimated data points following x_N. In this work, the hyperparameters are selected according to the selection in [35], which results from empirical investigation of multiple datasets for time-series anomaly detection. The number of estimated points is set to 5, the size for the local average is 21 and the length of the sequence $\boldsymbol{x}$ is set as 1440. It is worth noting that a parameter study on the investigated dataset revealed low sensitivity of the SR detection performance to hyperparameter selection.

3.1.6. Persistence detector

In this work, the use of a persistence forecast for modeling the expected value $\hat{x}_t$ is proposed and compared to the previously introduced methods. The proposed Persistence detector considers a history window $w = 1$ and determines $\hat{x}_t$ according to

$$\hat{x}_t = x_{t-1}. \tag{4}$$

The triviality of the Persistence detector eliminates the need for model selection, training and re-training.

3.2. OS classification of FA events

This subsection first formulates the OS classification problem. Thereafter, the implemented OS classifier is described.

3.2.1. Problem formulation

OS classification is contrasted with CS methods typically applied in the literature. CS classification requires identical event classes in training and test data and thus assumes full awareness of all existing event classes. For example, a CS classifier trained on observations of line failures and tap change operations will declare every observation of an unknown event class as either a line failure or a tap change. For DSOs it might be impractical to obtain data comprising observations off all existing classes, since events such as failures of transmission elements, large loads or sensors rarely occur. An OS classifier rejects observations of event classes not previously seen and declares them as *unknown*. In Fig. 2 CS classification is compared to OS classification. Making CS assumptions leads to regions of unbounded support, as can be seen from Fig. 2(b). This will result in misclassification of observations from unknown classes which can drastically weaken the performance of the classification.

The problem of OS classification can be formulated as follows [49]. Let $\boldsymbol{D}_{\text{train}} = \{(\boldsymbol{v}_i, y_i)\}_{i=1}^{N_{\text{train}}}$ be a training dataset, with $\boldsymbol{v}_i \in \mathbb{R}^d$ being a feature vector instance and $y_i \in \boldsymbol{Y}_{\text{train}} = \{1, 2, \dots, K\}$ the corresponding event class label. During test or application a classifier needs to predict event classes of the open dataset $\boldsymbol{D}_o = \{(\boldsymbol{v}_i, y_i)\}_{i=1}^{\infty}$, where $y_i \in \boldsymbol{Y}_o = \{1, 2, \dots, K, K+1, \dots, M\}$ with $M > K$. The occurrence of unknown classes requires the classifier to learn a mapping function $f(v) : \boldsymbol{V} \rightarrow \boldsymbol{Y}' = \{1, 2, \dots, K, \textit{unknown}\}$, with the option *unknown* representing the rejection of classes not seen during training.

Similar to the described situation of a DSO not having comprehensive recordings of event classes, in the present dataset the number of FA events is comparatively small. Thus, the size of the input feature vectors $\boldsymbol{v}_i$ is limited to 6. Reducing the dimension of the feature space that needs to be described by the limited number of training observations allows for better determination of the decision boundaries. All features are derived from the delta encoded time series (see Section 3.3) and are listed in Table 2. The definition of the input features is given based on a delta encoded sequence observation $\boldsymbol{x}_\Delta = \{x_{\Delta,1}, \dots, x_{\Delta,N_x}\}$.

Table 2
Overview of features used for OS classification of FA events.

Feature	Definition
Mean μ_{x_Δ}	$\frac{1}{N_x}\left(\sum_{i=1}^{N_x} x_{\Delta,i}\right)$
Standard deviation σ_{x_Δ}	$\sqrt{\frac{1}{N_x-1}\sum_{i=1}^{N_x}(x_{\Delta,i}-\mu_{x_\Delta})^2}$
Minimum value $x_{\Delta,\min}$	$\min(\boldsymbol{x}_\Delta)$
Maximum value $x_{\Delta,\max}$	$\max(\boldsymbol{x}_\Delta)$
Number of zeros n_0	$\text{count}(\boldsymbol{x}_\Delta \overset{!}{=} 0)$
Points between minimum and maximum value n_{minmax}	$\lvert\text{index}(x_{\Delta,\min}) - \text{index}(x_{\Delta,\max})\rvert$

3.2.2. EVM classifier

The EVM is an OS classifier proposed by Rudd et al. [38]. Advantages of the EVM include incremental learning and model reduction capability, which avoids frequent re-training and keeps model size and classification time small. The EVM models known classes within the training dataset by a set of radial inclusion functions (see Fig. 2(c)). Based on the radial inclusion function of a class C_l, the probability $\hat{P}(C_l|v')$ of a new observation v' belonging to C_l can be determined. The decision function of the EVM is given by

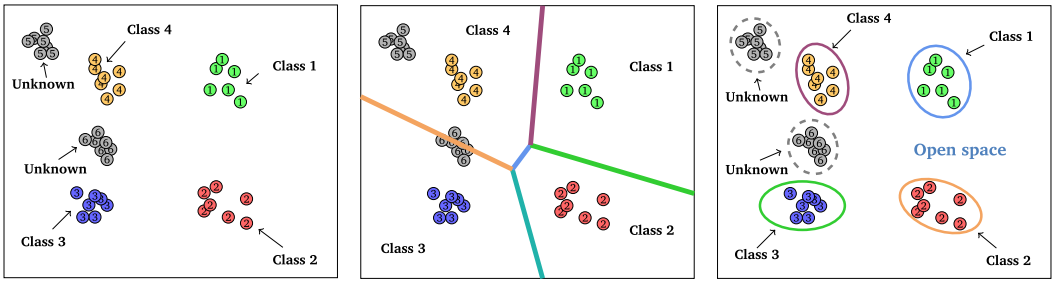
$$y^* = \begin{cases} \arg\max_{l \in \{1,\dots,M\}} \hat{P}(C_l|v') & \text{if } \hat{P}(C_l|v') \geq \rho \\ \textit{unknown} & \text{otherwise} \end{cases}, \tag{5}$$

where ρ is a threshold defining the boundary between the set of known classes and the unknown open space.

The EVM model training and selection is based on 90 % of the available event observations applying 5-fold time series cross-validation. The features are standardized based on training data for every individual split. In order to select an appropriate threshold ρ, a minimum performance requirement on the training dataset is defined based on the F_1 score (Section 4.2). The model with the smallest threshold ρ, which still fulfills the performance requirement $F_1 \geq 0.8$ in the time series cross-validation, is selected. An overview of the selected hyperparameters is given in Table 3.

3.3. EIP

To connect the presented models for unsupervised event detection and OS classification, a data processing pipeline is proposed. In Fig. 1



(a) Distribution of the original dataset. (b) CS classification. (c) OS classification.

Fig. 2. Comparison of CS and OS classification according to [27].

Table 3
Summary of hyperparameters of the EVM model.

Hyperparameter	Search space	Selected value
Tailsize	[1,100]	7
Distance multiplier	[0.1,1.1]	0.9
Distance metric	Canberra, Cosine, Euclidean	Canberra
Threshold	[0.1, 0.99999]	0.9

a schematic overview of the EIP is depicted. The functionality of the main building blocks of the pipeline was described in Sections 3.1 and 3.2. In Section 2.3 the use of delta encoded data for the EIP was motivated. Delta encoding exploits the autocorrelation of time series data. In the simplest version of delta encoding, a time series $\boldsymbol{X} = \{x_1, x_2, \dots, x_N\}$ is encoded as difference between successive samples, resulting in the differenced time series $\boldsymbol{X}_\Delta = \{x_1, x_2 - x_1, \dots, x_N - x_{N-1}\} = \{x_1, x_{\Delta,2}, \dots, x_{\Delta,N}\}$. Delta encoding performs best when the values in the original data contain only small changes between adjacent values [50]. By applying the Persistence detector on the differenced time series $\boldsymbol{X}_\Delta$, the anomaly detection problem reduces to comparison of the amplitudes of $x_{\Delta,i}$ to the predefined threshold τ. To connect point-wise unsupervised event detection with OS classification, an event sampler is interposed, exploiting the characteristics of fast-ramped events. As can be seen in Fig. 4 fast-ramped FA events show peaks in the delta encoded data at the beginning and/or end of an event. Based on this property, the detection of an event at data point $x_{\Delta,t}$ can be used to extract a backward sequence sample $\boldsymbol{x}_{\Delta,\text{bw}} = \{x_{\Delta,t-w_x}, \dots, x_{\Delta,t+e}\}$ and forward sequence sample $\boldsymbol{x}_{\Delta,\text{fw}} = \{x_{\Delta,t-e}, \dots, x_{\Delta,t+w_x}\}$, where w_x is the sample window size and e a window extension, ensuring sampling of the entire event. In case of forward sampling, an early stopping criterion can be introduced which breaks the sampling process in case another event is detected at a data point $x_{\Delta,t+a} \in \{x_{\Delta,t+1}, \dots, x_{\Delta,t+w_x}\}$, resulting in a forward sample $\boldsymbol{x}_{\Delta,\text{fw}} = \{x_{\Delta,t-e}, \dots, x_{\Delta,t+a+e}\}$. Such event-triggered early stopping of the forward sampling process reduces the sampling time and thus the time until a sample can be classified. In the Appendix in Algorithm 1 the general procedure of the proposed EIP is described.

4. Experimental setup

This section presents the experimental setup of this study. The considered dataset as well as the preparation of the dataset for investigation of unsupervised event detection and OS classification of FA events are presented in Section 4.1. Section 4.2 introduces metrics for the evaluation of the detection and classification performance.

4.1. Dataset and data preparation

The first part of this subsection is concerned with presenting key information of the dataset under investigation as well as describing the process of activating flexibility. In the second part the preparation of the dataset is explained, which includes data cleaning and in case of preparation for the investigation of the OS classification the extension of the dataset with two artificial event classes.

4.1.1. Dataset

Within this work, a dataset from EcoGrid 2.0 is used. EcoGrid 2.0 was a demonstration project which examined the use of flexible consumption of residential customers for power system services at transmission system operator and DSO level [9]. The experiments were conducted on the Danish island of Bornholm. The residential customers were equipped with SMs and ICT infrastructure for participating in demand response experiments. The flexible load corresponded to electric heaters and heat pumps that were controlled by adjusting room temperature setpoints or by sending a throttle signal, respectively. An increase in the setpoints results in higher consumption, while lowering the setpoints leads to a reduction of consumption. The throttle signal blocks the operation of the heat pump until the signal is released. Besides flexible load, the installed SMs also capture household consumption and photovoltaic production, when present. As can be seen in Fig. 3(a), the dataset used consists of six and a half months of aggregated load data, beginning from 15th of September 2017. The aggregated active power profile consists of 450 household loads with a 5 min time resolution. The FA events comprise load reduction and load increase experiments (see Fig. 3(b)) realized by two different aggregators and customer portfolios. Activation periods are in the range of 30 – 120 min. Different numbers of customer loads participated in each FA, and activations were conducted under varying conditions of temperature, time of the day and photovoltaic production. In Fig. 3(a) the trend and (b) seasonality of the non-stationary load time series can be noticed.

4.1.2. Dataset preparation for investigation of unsupervised event detection

The dataset contains 325 FA events. Start and end time as well as type of FA event is known for every experiment. In 75 cases two flexibility portfolios were activated simultaneously. To avoid double counting of either true positives (TPs) or false negatives (FNs), parallel experiments are considered as one FA event, reducing the number of FA events to 250. In most cases, experiments result in either load reduction or load increase of a subset of customer loads. However, in some cases little to no flexibility was activated. This can be due to exclusive testing of connectivity, failed activation of flexibility assets, or low flexibility potential due to high temperatures and thus low heating demand. For this work, such FA events are not considered in the performance evaluation, reducing the number of FA event samples to 205.

4.1.3. Dataset preparation for investigation of OS event classification

In Fig. 4 examples of all event classes are depicted in absolute and delta encoded values. Note that only for the OS classification problem all introduced event classes are considered. For the FA event detection problem only FA events are taken into account. The classifier is trained on two known event classes, namely FA and normal operation (NO)

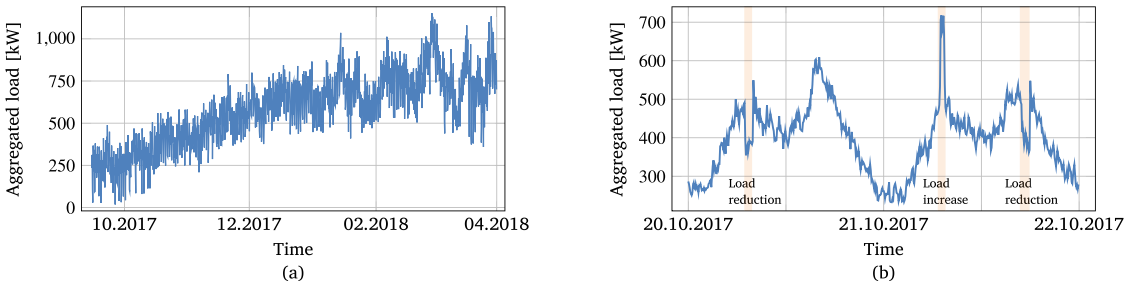


Fig. 3. Aggregated load dataset (a) and aggregated load of two representative days with frequent FAs (b). Periods of FAs are marked with an orange background.

events. All 205 FA events are sampled, including 3 timestamps (15 min) that precede and succeed each event. As the duration of FA events varies, the length of FA event samples varies as well. An equal amount of NO events are randomly sampled from the remaining dataset. The length of a NO event is randomly selected from the distribution of FA event lengths. With this approach, the classifier is prevented from differentiating between FA and NO events based on the sample length. As the sample length of both FA and NO events can vary in the proposed EIP (see Section 3.3), learning a constant sample length is not considered a valid approach.

The problem of OS classification, as formulated in Section 3.2.1, requires additional event classes within the test dataset to investigate the capability of rejecting unknown classes. In this work, 3 unknown event classes are considered. Besides the FA and NO event classes, the EcoGrid 2.0 dataset includes another event class, which in the course of this work will be called Monday peak (MP). On every Monday within the dataset, a load peak occurs at around 8 am. The load peak results from short, collective heating of electric water boilers to 80 °C to inhibit the growth of bacteria. Only MP events that could be manually detected in an extensive ex-post evaluation of the dataset are considered. MP events are not considered a normal operation and thus no overlapping of NO and MP events exists. In total the dataset includes 15 MP events. In order to extend the OS classification problem, two additional artificial event classes are introduced, namely the frozen value (FV) and data unavailability (DU) event class. The FV event class models a data transmission or processing failure in which a measurement at time t_0 remains constant for N_{cons} consecutive steps. At $t_{N_{\text{cons}}+1}$ recording of true measurements is reestablished. The length of FV events is randomly selected from the distribution of FA event lengths. 205 FV events are randomly introduced into the subset of the dataset which is not influenced by FA, NO and MP events. In a DU event a subset of individual measurements, e.g. SM readings, is considered to be unavailable due to device or data transmission failure. The fraction of available measurements is randomly selected from the uniform distribution $\mathcal{U}(0.4, 0.8)$. As for NO and FV events, the length of DU events is drawn from the distribution of FA event lengths. 205 DU events are introduced into the subset of the dataset not affected by any of the previously described events.

4.2. Performance metrics

The performance evaluation of both the unsupervised detection and OS classification of FA events is based on a labeling of each instance of the respective dataset. The definition of an instance for the detection and classification task follows in Sections 4.2.1 and 4.2.2, respectively. One performance metric used for evaluation of both the detection and classification part is the F_1 score. The F_1 score represents the harmonic mean between precision and recall, and is a widely applied performance metric for detection and classification problems with imbalanced classes. Let TP_l, FP_l, TN_l, and FN_l be the number of TPs, false positives (FPs), true negatives (TNs), and FNs for the lth event class, respectively, where $l \in \{1, 2, \dots, M\}$. For a multi-class problem, the macro F_1 score is calculated by

$$F_1 = \frac{1}{M}\sum_{l=1}^{M} F_{1,l} = \frac{1}{M}\sum_{l=1}^{M} 2\frac{Pr_l \cdot Re_l}{Pr_l + Re_l}, \tag{6}$$

where precision Pr and recall Re of the lth event class are defined as

$$Pr_l = \frac{TP_l}{(TP_l + FP_l)}, \text{ and } Re_l = \frac{TP_l}{(TP_l + FN_l)}. \tag{7}$$

4.2.1. Unsupervised event detection metrics

In this work, the problem of unsupervised event detection is considered an anomaly detection problem, reducing the number of classes to $M = 2$. Since anomalies constitute the primary class of interest, the multi-class formulation of the F_1 score in (6) reduces to

$$F_1 = F_{1,1} = 2\frac{Pr_1 \cdot Re_1}{Pr_1 + Re_1}. \tag{8}$$

Besides the F_1 score, another widely applied performance metric for anomaly detection is the area under the precision–recall curve (*AUCPR*). Precision–recall curves summarize the trade-off between precision and recall for different thresholds τ. While the consideration of TNs in traditional receiver-operating-characteristic curves may lead to an overly optimistic view on the performance in case of highly imbalanced classes, *AUCPR* is specifically tailored to problems with imbalanced classes or rare events.

In this work, the entire sequence of an FA event, referred to as event window $\boldsymbol{\omega}_{\text{FA}}$, is considered for labeling as TP or FN. The event window of a FA event is defined by the FA start time $t_{\text{FA,start}}$ and end time $t_{\text{FA,end}}$. Since the dataset under investigation is a real-world dataset it contains some inaccuracies in the event labeling. In some cases flexibility was activated before the official start time $t_{\text{FA,start}}$. Since in these cases an early detection would result in falsely FNs, the event window $\boldsymbol{\omega}_{\text{FA}}$ is extended by 10 min, such that $\boldsymbol{\omega}_{\text{FA}} = \{x_{t_{\text{FA,start}}-2}, \dots, x_{t_{\text{FA,end}}}\}$. The first detection that falls into $\boldsymbol{\omega}_{\text{FA}}$ is considered as TP, while further detections within the same event window are ignored. If no point of $\boldsymbol{\omega}_{\text{FA}}$ is detected the event label will be considered a FN. In most cases,

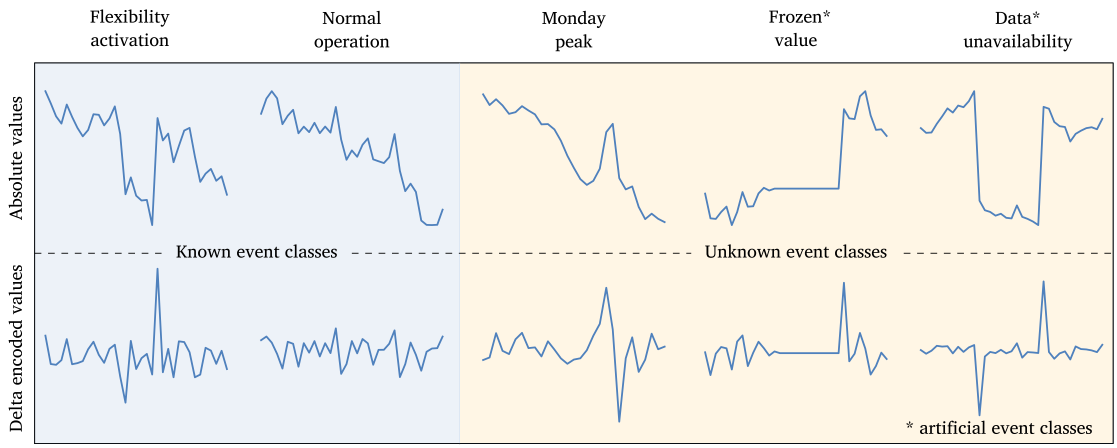


Fig. 4. Exemplary representation of event classes considered in this work.

FAs result in a subsequent load rebound, which can be considered a deviation from the normal load behavior outside of the activation period. While regarding a detection within the rebound area as TP would introduce a positive bias to detection performance, considering them as FP would result in overly pessimistic performance results, as the detector indeed has detected an anomaly. For this reason, a rebound window $\boldsymbol{\omega}_{\mathrm{R}}$ is introduced in which detections are ignored and are thus neither considered a TP nor FP. The length of a rebound window is defined as three times the FA event length, resulting in $\boldsymbol{\omega}_{\mathrm{R}} = \{x_{t_{\mathrm{FA,end}}+1}, \dots, x_{t_{\mathrm{FA,end}}+3\times N_{\mathrm{FA}}}\}$. In contrast to calculation of TPs and FNs, the calculation of FPs and TNs is conducted point-wise. Thus, all detections outside the event and rebound windows are considered FP.

To evaluate the early detection capability, the average detection delay $\bar{\delta}_{\mathrm{det}}$ is introduced as the average time between FA start time $t_{\mathrm{FA,start}}$ and the first detection time t_{det} in minutes according to

$$\bar{\delta}_{\mathrm{det}} = \frac{1}{N_{\mathrm{det}}} \sum_{i=1}^{N_{\mathrm{det}}} \delta_{\mathrm{det},i} = \frac{1}{N_{\mathrm{det}}} \sum_{i=1}^{N_{\mathrm{det}}} (t_{\mathrm{det},i} - t_{\mathrm{FA,start},i}), \tag{9}$$

where N_{det} is the number of detected FA events and $\delta_{\mathrm{det},i}$ the detection delay of a detected FA event. Note that the detection delay of detections is assumed to be zero within the subset $\{x_{t_{\mathrm{FA,start}}-2}, x_{t_{\mathrm{FA,start}}-1}, x_{t_{\mathrm{FA,start}}}\}$.

The use of widely applied performance metrics allows for an easy understanding and comparison of the results to other studies. However, in order to take performance requirements of a specific scenario into account an individual performance metric is required. For this purpose, the flexibility activation detection score (*FAD* score) is proposed. In the scenario of real-time detection of FAs in active DNs, the cost of FN is considered to be higher compared to the cost of FP. While a missed critical FA could lead to violation of power or voltage boundaries, a false alarm would result in a moderate additional manual inspection effort. Moreover, in the proposed pipeline, a FP will lead to a sample of normal behavior (NO event class) which can be classified as such by the OS classifier. In this way the classifier relativizes the FP of the unsupervised event detector. For these reasons, the *FAD* score puts more weight on FNs than on FPs. Further, early detection capability is an important requirement in the considered scenario. The earlier a potentially critical FA event is detected, the greater the scope for countermeasures. On the contrary, detection near the end of a critical FA results in almost no benefit. The *FAD* score takes these considerations into account and expresses the performance for the specific scenario of real-time FA event detection in one score. Besides an easier evaluation and comparison of detection models, the *FAD* score also allows for easy selection of an optimal threshold τ for unsupervised detection of FA events. The *FAD* score is constituted by 3 scoring functions ζ_{TP}, ζ_{FN} and ζ_{FP}, representing the contribution of TPs, FNs and FPs, respectively:

$$FAD = \zeta_{\mathrm{TP}} - \zeta_{\mathrm{FN}} - \zeta_{\mathrm{FP}}. \tag{10}$$

Let $x_{t_{\mathrm{FA,start},i}}$, $x_{t_{\mathrm{FA,end},i}}$ and $x_{t_{\mathrm{det},i}}$ be the start point, end point and first detection within the event window $\boldsymbol{\omega}_{\mathrm{FA},i}$ of the ith FA event, respectively. Then ζ_{TP} is given by

$$\zeta_{\mathrm{TP}} = \sum_{i=1}^{N_{\mathrm{det}}} \sigma_i(x_{t_{\mathrm{det},i}}), \tag{11}$$

where $\sigma_i(x_{t_{\mathrm{det},i}})$ is the positive score of one TP detection. Between $\sigma_i(x_{t_{\mathrm{FA,start},i}}) = \xi$ and $\sigma_i(x_{t_{\mathrm{FA,end},i}}) = 0$ the score σ_i follows a linear declining function, where ξ is the maximum positive score of one TP detection. For each missed FA event a negative score of η is considered according to

$$\zeta_{\mathrm{FN}} = -\eta \cdot FN_1. \tag{12}$$

Scoring function ζ_{FP} is represented by a moved negative exponential function given as

$$\zeta_{\mathrm{FP}} = -\gamma \cdot \exp\left(\frac{-FP_1}{\upsilon}\right) - \nu, \tag{13}$$

where γ is the maximum negative score of a FP detection. Note that $\gamma \ll \xi$. Parameters υ and ν allow for additional adjustments of the score to the dataset. The negative exponential decline considers that the first FP detections will have a stronger negative impact on performance, while for subsequent FPs the additional negative impact is small. The final *FAD* score is normalized such that $FAD_{\mathrm{norm}} \in [0, 1]$, according to

$$FAD_{\mathrm{norm}} = \frac{FAD - FAD_{\mathrm{null}}}{FAD_{\mathrm{opt}} - FAD_{\mathrm{null}}}, \tag{14}$$

where FAD_{null} is the *FAD* score without any detection and FAD_{opt} the *FAD* score under optimal detection. In this work, the parameters are selected as $\xi = 1$, $\eta = 1$, $\gamma = 0.05$, $\upsilon = 10000$ and $\nu = 0$.

4.2.2. OS event classification metrics

The performance evaluation of the OS classification of FA events is based on the F_1 score according to (6). However, for the OS problem the number of classes should only be determined by the known classes. Considering all of the unknown classes as a single additional class in the test dataset would result in a biased performance result. If the problem is treated in the same way as a CS scenario, rejected samples of unknown classes would be considered as TPs - although no training samples of the unknown classes existed. Instead, the calculation of the F_1 score is given by

$$F_1 = \frac{1}{K} \sum_{l}^{K} F_{1,l} = \frac{1}{K} \sum_{l}^{K} 2 \frac{Pr_l \cdot Re_l}{Pr_l + Re_l}, \tag{15}$$

where K is the number of known classes from the training dataset. In Section 5.2 the influence of the number of unknown event classes on the classification performance will be evaluated, which requires the definition of the *openness* of a test dataset. In [36] the authors introduce a formal definition of the openness O of a dataset according to

$$O = 1 - \sqrt{\frac{2 \times |\text{training classes}|}{|\text{testing classes}| + |\text{target classes}|}}, \tag{16}$$

with $O \in [0, 1]$. Large values for O correspond to a higher number of unknown classes in the dataset, while for the CS problem $O = 0$.

5. Results and discussion

In this section the performance evaluation of the proposed models for unsupervised detection and OS classification of FA events is presented. In Section 5.1 the proposed Persistence detector is compared to various other models, introduced in Section 3.1. Section 5.2 investigates the OS classification of FA events based on the introduced EVM model (Section 3.2.2). The classification performance is compared to a CS classifier benchmark. The performance evaluation is conducted based on the dataset and performance metrics from Section 4.

5.1. Unsupervised FA event detection

In Fig. 5 the maximum F_1 score $F_{1,\mathrm{max}}$ and *FAD* score FAD_{max} at the optimal threshold $\tau_{\mathrm{opt,F1}}$ and $\tau_{\mathrm{opt,FAD}}$, respectively, are depicted together with the *AUCPR* for Persistence, HTM, ARIMA, SR and CNN detector. From the comparison of $F_{1,\mathrm{max}}$ and *AUCPR* it can be derived that Persistence, ARIMA, SR and CNN detector roughly lie in the same performance range. However, the HTM detector shows a significantly poorer performance. While according to the F_1 score the Persistence, ARIMA and CNN detector achieve the best detection results, with $F_{1,\mathrm{max}} = 0.71$, in accordance with the *AUCPR* the SR detector outperforms all other detectors with $AUCPR = 0.69$. Interestingly, with a difference of 4 percentage points the F_1 score shows a poorer detection performance for the SR detector.

Although both the F_1 score and *AUCPR* are performance metrics specifically tailored to scenarios with highly imbalanced classes and higher emphasis on the positive class, they suggest different results. This again motivates the need for a scenario-specific performance

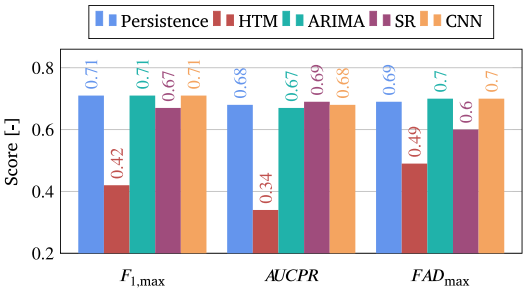


Fig. 5. Maximum F_1 score ($F_{1,\max}$), *AUCPR* and maximum *FAD* score ($FAD_{\max}$) for Persistence, HTM, ARIMA, SR and CNN detector.

metric. Based on the comparison of the maximum *FAD* score $FAD_{\max}$ in Fig. 5 it can be concluded that both the ARIMA and CNN detector achieve the best result for the problem of real-time detection of FA events in aggregated load data with $FAD_{\max} = 0.7$. On the contrary, with $FAD_{\max} = 0.6$ the SR detector shows a significant poorer performance in the given case of FA event detection. However, according to $F_{1,\max}$ and *AUCPR* the SR detector can potentially keep up with or even outperform other detection methods in scenarios with other requirements. With $FAD_{\max} = 0.69$ the performance of the Persistence detector is only slightly below the best *FAD* scores achieved by the ARIMA and CNN detector. As can be seen in Fig. 4 FA events in the given dataset are fast-ramped events that are characterized by a steep slope at the beginning and end of an event, resulting in a large deviation between x_t and x_{t-1}. In case of the Persistence detector this large deviation directly translates to a large anomaly score according to (1). Given that the Persistence detector can keep up with the more complex detectors regardless of the considered performance metrics, it can be concluded that the Persistence detector constitutes a trivial but effective method for detection of fast-ramped FA events.

In Fig. 6 the average detection delay $\bar{\delta}_{\text{det}}$ is shown as a function of the threshold τ for all detectors. As for $\tau = 0$ all data points are declared an event, the average detection delay is $\bar{\delta}_{\text{det}} = 0$ for all detectors. It can be seen that the HTM detector has the lowest detection delay for thresholds $\tau > 0.1$. The comparatively high early detection capability also explains the reduced performance discrepancy between the HTM and the other detectors for $FAD_{\max}$ compared to $F_{1,\max}$ and *AUCPR* (Fig. 5). While for the *FAD* score the contribution of TPs is weighted based on the detection delay, $F_{1,\max}$ and *AUCPR* do not take early detection into account.

The Persistence, ARIMA, SR and CNN detector show a similar $\bar{\delta}_{\text{det}}$ for $\tau < 0.4$. For thresholds $\tau > 0.4$ the SR detector shows a significantly higher detection delay, while Persistence, ARIMA and CNN detector continuously show a similar detection delay. Fig. 7 compares the

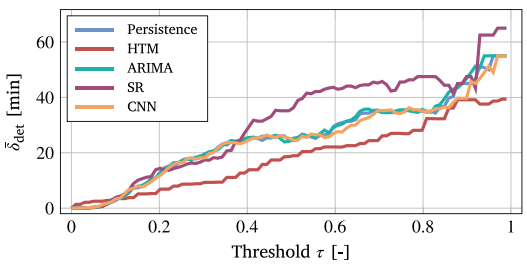


Fig. 6. Average detection delay $\bar{\delta}_{\text{det}}$ for Persistence, HTM, ARIMA, SR and CNN detector.

average detection delay $\bar{\delta}_{\text{det}}$ of all detectors at the optimal threshold $\tau_{\text{opt,FAD}}$ corresponding to $FAD_{\max}$. Although the HTM detector has the lowest average detection delay over the largest range of τ (Fig. 6), at an operating point relevant for the considered scenario (i.e. at $FAD_{\max}$), it has a higher detection delay compared to most other detectors. The HTM detector requires a higher threshold compared to the other detectors (see Fig. 8(c)), due to the particularly strong vulnerability to high FP numbers for low thresholds. The higher threshold in turn explains the higher average detection delay of the HTM detector. The SR detector with $\bar{\delta}_{\text{det}} = 11.42$ min has by far the highest detection delay. Persistence, ARIMA and CNN detector show similar delays. In fact, the proposed Persistence detector shows the lowest detection delay with $\bar{\delta}_{\text{det}} = 7.41$ min. However, it has to be considered, that the calculation of the average detection delay is only based on detected events according to (9). Thus, detecting additional events close to the end of an event (as done by ARIMA and CNN detector) results in an improved *FAD* score, as negative FN scores are avoided, even though the average detection delay increases.

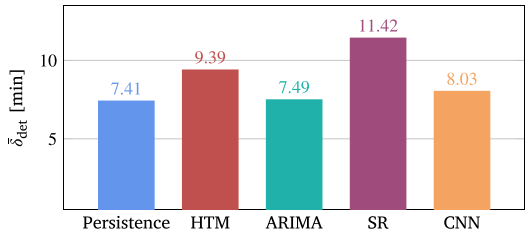


Fig. 7. Average detection delay $\bar{\delta}_{\text{det}}$ at $FAD_{\max}$ for Persistence, HTM, ARIMA, SR and CNN detector.

Fig. 8(a) and (b) show the F_1 score over the threshold τ and the precision–recall curve for the different detectors, respectively. Both on the F_1 score and precision–recall curve a strong similarity of the Persistence, ARIMA and CNN detector can be noticed. This can be explained by the signal-to-noise ratio of the dataset. Although aggregated, the load data under investigation show a comparatively low signal-to-noise ratio due to fluctuations, introduced by unforeseeable customer behavior, and the high resolution of the data. Because of the low signal-to-noise ratio it is difficult for more complex methods, such as the applied ARIMA and CNN model, to extract additional information from the dataset compared to the trivial persistence forecast. Thus, the explainability of the dataset can be exploited by the persistence forecast to a large extend, explaining the similarity of the Persistence, ARIMA and CNN detector. However, the SR detector clearly shows a different behavior. This is due to the different mathematical approach of transforming the dataset from time into the frequency domain.

Fig. 8(b) shows that, compared to all other detectors, the SR detector is able to keep the precision on a higher level for an increasing recall. While a high precision is not seen as an important requirement for the considered scenario, the SR detector may have advantages over the other detectors in scenarios with different requirements. Interestingly, the HTM detector clearly shows a different behavior compared to the Persistence, ARIMA and CNN detectors, even though it is also based on a time series forecast. This can partly be explained by the internal calculation of the anomaly score, which differs from the external calculation used for Persistence, ARIMA and CNN detector (see Section 3.1.2). However, the comparatively poor detection performance also indicates a poor underlying forecast that is even outperformed by a trivial persistence forecast. A potential reason could be insufficient adaption of the various model parameters to the dataset and scenario. Although the authors of HTM claim the provided set of parameters to be the best for anomaly detection, it may not be sufficiently appropriate for the given scenario. However, as explained before, due to the low signal-to-noise ratio, it can be expected that even extensive parameter

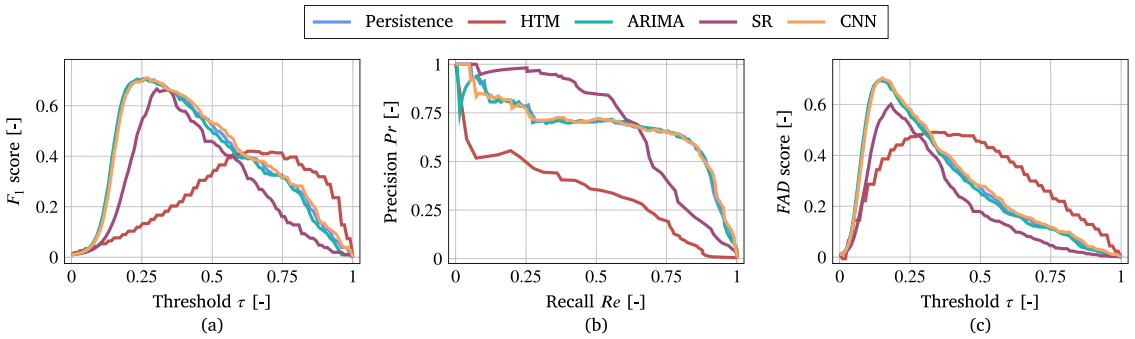


Fig. 8. (a): F_1 score, (b): Precision–recall curve, (c): *FAD* score for Persistence, HTM, ARIMA, SR and CNN detector.

tuning will not result in a significantly better forecast compared to the persistence forecast.

Fig. 8(c) shows the *FAD* score for all detectors over the threshold τ. By comparing the F_1 score with the *FAD* score, a shift between the optimal threshold $\tau_{\text{opt,F1}}$ and $\tau_{\text{opt,FAD}}$ towards smaller values can be noticed. A smaller threshold increases the number of TPs and results in earlier detection of an event. At the same time, the number of FPs increases as well. However, as described in Section 4.2.1 the *FAD* score emphasizes early event detection and weights FPs low compared to FNs, explaining the decrease of the optimal threshold. Based on the *FAD* score an optimal threshold for the proposed Persistence detector of $\tau_{\text{opt,FAD}} = 0.16$ is determined. At $\tau_{\text{opt,FAD}}$ 191 of 205 FA events (93 %) are detected by the Persistence detector, while 498 data points of all 39827 data points outside the event and rebound windows (1.25 %) are falsely declared a FA event.

As previously described, for the dataset and scenario under investigation, more sophisticated models such as ARIMA and CNN only achieve minor improvements of the forecast compared to the persistence forecast. This also translates to a similar characteristic of the *FAD* score curve over the threshold τ. It can be inferred, that for the detection of fast-ramped FA events an upper performance limit should exist at an *FAD* score of roughly $FAD \approx 0.7$. This performance limit can approximately be reached with the proposed Persistence detector ($FAD_{\text{max}} = 0.69$). More advanced detection methods, such as the ARIMA and CNN detectors, only slightly improve the detection performance, but require a significantly higher maintenance and computational effort. The Persistence detector is therefore proposed to avoid frequent time and computation intensive model re-training. As described in Section 2 this is considered a great advantage in a scenario of edge computing-based distributed event detection with time and resource constraints.

5.2. OS classification of FA events

In Fig. 9 the confusion matrix for the EVM model applied on the OS test dataset is depicted. Besides the two known event classes FA and NO, three unknown event classes are included in the test dataset, namely MP, FV and DU, which are summarized as *unknown*. The test dataset in total contains 63 observations with $N_{\text{FA}} = 21$, $N_{\text{NO}} = 21$, $N_{\text{MP}} = 7$, $N_{\text{FV}} = 7$, $N_{\text{DU}} = 7$. The test dataset has an openness of $O = 24.7\,\%$. From Fig. 9 it can be derived that the EVM is able to correctly classify 90 % of the FA events and 76 % of the NO events. Moreover, the EVM successfully rejects 71 % of all observations of the unknown classes. It can be concluded that the EVM in principle is able to differentiate between FA and NO observations also in an OS scenario with an acceptable performance. However, the EVM also erroneously classifies 29 % of the observations from the unknown classes as FA events, which reduces precision for FA events. Precision for NO events is not affected by the unknown event classes. Also the recall of the FA and NO event classes is negatively influenced, since 10 % of the FA events and 19 % of the NO events, respectively, are rejected. In general, Fig. 9 demonstrates that the influence of the unknown classes on precision and recall of a class is higher compared to the influence of the other known class.

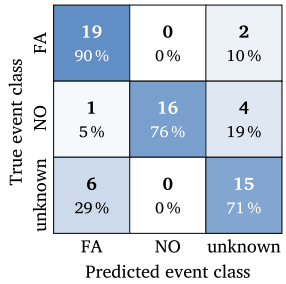


Fig. 9. Confusion matrix of the EVM model on the OS classification test dataset with an openness $O = 24.7\,\%$.

In order to investigate the benefit of applying an OS classifier in the more realistic scenario with presence of unknown classes, performance is compared to a CS classifier. For this purpose, the EVM model is applied on the test dataset as both an OS and CS classifier. In the CS setting the rejection of observations with $\hat{P}(C_l|v') < 0.9$ is deactivated and observations are classified according to $\hat{P}(C_l|v')$. Moreover, in order to investigate the influence of the number of unknown classes, the comparison is conducted on a test dataset with increasing fractions of unknown classes. In Fig. 10 the comparison of the OS and CS EVM for a varying openness O of the test dataset is depicted. Performance is evaluated based on the F_1 score. For the CS problem ($O = 0$) the performance of the CS classifier is slightly better compared to the OS classifier, since the OS classifier wrongly rejects some of the observations of the known classes. By adding MP events as unknown class to the test dataset the openness of the test dataset increases to $O = 10.56\,\%$. In this scenario the OS classifier outperforms the CS classifier. While the OS classifier is capable of rejecting observations from unknown classes, the CS classifier assigns all observations of unknown classes to one of the known classes, resulting in a decreased precision. However, the performance of the OS EVM decreases as well. This is due to two reasons. First, not all observations from the unknown classes are successfully rejected. Second, being capable of rejecting observations can also lead to falsely rejected observations of known classes. Nevertheless, for the investigated scenario of FA event classification the rejection capability improves the performance compared to the CS classifier already for the existence of only one unknown class. Extending the test dataset with the unknown FV event class ($O = 18.35\,\%$) has no influence

on the performance of the OS EVM. This can be explained by the specific characteristic of FV events. The number of zeros n_0 constitutes a strong differentiator for observations of the FV event class, making it comparatively easy for the OS EVM to differentiate between FV and the known FA and NO events. Nevertheless, the F_1 score of the CS classifier further decreases from $F_1 = 0.839$ to $F_1 = 0.792$ since all observations of the FV event class are assigned to either the FA or NO event class. In the final scenario ($O = 24.7\,\%$) all unknown event classes are added to the test dataset, corresponding to the scenario described by Fig. 9. Adding the unknown DU event class further decreases performance for both the OS and CS classifier. However, while for the CS EVM the F_1 score decreases by 6.31 %, the OS classifier only shows a decrease of 3.12 % percent. In summary, the performance of the OS EVM decreased from $F_1 = 0.896$ ($O = 0\,\%$) to $F_1 = 0.837$ ($O = 24.7\,\%$), while the CS classifier performance decreased from $F_1 = 0.905$ ($O = 0\,\%$) to $F_1 = 0.742$ ($O = 24.7\,\%$). This demonstrates that the rejection capability of the OS classifier allows maintaining the classification performance on a higher level, for increasing fractions of unknown classes. Nevertheless, also for the OS classifier the performance deteriorates with additional unknown classes.

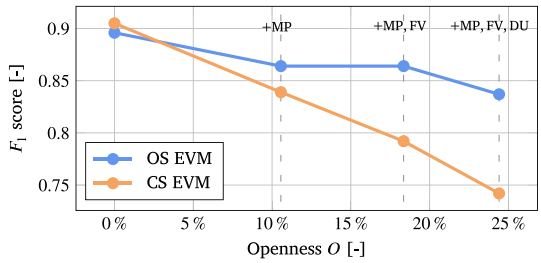


Fig. 10. Comparison of the F_1 score between OS and CS EVM model on the OS classification problem with a varying openness O.

To summarize, FA events can be classified also in the more realistic OS scenario and applying OS classifiers can significantly improve performance under these conditions. However, the classification performance of the EVM is expandable, due to the very limited training data. The size of the dataset constitutes a limitation of the presented study, since the small number of observations and classes prevent more comprehensive investigations. Nevertheless, this study proves the fundamental feasibility of OS classification of FA events on real data.

6. Conclusion and future work

This work demonstrates the fundamental feasibility of unsupervised detection and OS classification of fast-ramped FA events. A data processing pipeline for FA event identification is suggested, which combines both steps. For unsupervised FA event detection, a simple Persistence detector is proposed and implemented. The comparison with more complex and computationally expensive detection models demonstrates a similar performance and the existence of an upper performance limit. Results indicate that the Persistence detector is particularly suitable for the specific class of FA events. As OS classifier the EVM is used. It is shown that the use of an OS classifier significantly improves classification performance in the more realistic OS scenario compared to a traditional CS classifier. Both the Persistence detector and EVM classifier are selected with a view to an application in a distributed event detection architecture with time and resource constraints due to edge computing. Their good performance demonstrates that main building blocks of the proposed pipeline can be realized with comparatively simple and lightweight methods that fulfill important requirements for an application in a distributed event detection architecture.

Given the fundamental proof of the main building blocks, a logical next step is the investigation of the coupling of the Persistence detector and EVM classifier in the proposed EIP for FA events. Moreover, for both the detection as well as classification step, several possible improvements could be investigated. One direction could be the integration of additional regressors for unsupervised event detection such as temperature and solar radiation. For the OS classification problem principle component analysis or other methods for dimensionality reduction could be applied to reduce the feature space dimension while retaining the majority of the information. Finally, the proposed pipeline could be extended from FA event identification to identification of multiple relevant events in active DNs.

CRediT authorship contribution statement

Nils Müller: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. **Carsten Heinrich:** Conceptualization, Resources, Data curation, Writing – review & editing. **Kai Heussen:** Conceptualization, Resources, Writing – review & editing, Supervision. **Henrik W. Bindner:** Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is partly funded by the Innovation Fund Denmark (IFD) under File No. 91363 and by the INTERPRETER project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 864360.

Appendix

Algorithm 1 General procedure of the EIP for FA events.

for new incoming data point $x_{\Delta,t}$ **do**:
  PERSISTENCE_FORECAST($x_{\Delta,t}$) ▷ **Start of unsupervised event detection**
    return $\hat{x}_{\Delta,t}$
  if $|\hat{x}_{\Delta,t} - x_{\Delta,t}| < \tau$ **then**:
    Declare $x_{\Delta,t}$ normal behavior
  else:
    Declare $x_{\Delta,t}$ an event
    BACKWARD_SAMPLING($x_{\Delta,t}$) ▷ **Start of event sampling**
      return $\boldsymbol{x}_{\Delta,\mathrm{bw}} = \{x_{\Delta,t-w_x}, ..., x_{\Delta,t+e}\}$
    FORWARD_SAMPLING($x_{\Delta,t}$)
      if another event at $x_{\Delta,t+a} \in \{x_{\Delta,t+1}, ..., x_{\Delta,t+w_x}\}$ **then**:
        return $\boldsymbol{x}_{\Delta,\mathrm{fw}} = \{x_{\Delta,t-e}, ..., x_{\Delta,t+a+e}\}$
      else:
        return $\boldsymbol{x}_{\Delta,\mathrm{fw}} = \{x_{\Delta,t-e}, ..., x_{\Delta,t+w_x}\}$
    for $\boldsymbol{x}$ in $[\boldsymbol{x}_{\Delta,\mathrm{bw}}, \boldsymbol{x}_{\Delta,\mathrm{fw}}]$ **do**:
      Calcul. feature vector $v = [\mu_{\boldsymbol{x}}, \sigma_{\boldsymbol{x}}, x_{\min}, x_{\max}, n_0, n_{\mathrm{minmax}}]$
      EXTREME_VALUE_MACHINE(v) ▷ **Start of OS classification**
        return $P(C_{flexibility}|v)$, $P(C_{normal\_behavior}|v)$
      if $P(C_{flexibility}|v) \geq P(C_{normal\_behavior}|v)$ **and** $\geq \rho$ **then**:
        Declare sample $\boldsymbol{x}$ as flexibility activation event
      else if $P(C_{flexibility}|v) \leq P(C_{normal\_behavior}|v)$ **and** $\geq \rho$ **then**:
        Declare sample $\boldsymbol{x}$ as normal behavior
      else:
        Declare sample $\boldsymbol{x}$ as unknown event

References

[1] European Commission. Impact assessment on stepping up Europe's 2030 climate ambition. investing in a climate-neutral future for the benefit of our people. 2020, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52020SC0176. [Online; accessed: 23/07/2021].

[2] Zhao J, Wang C, Zhao B, Lin F, Zhou Q, Wang Y. A review of active management for distribution networks: current status and future development trends. Electr Power Compon Syst 2014;42(3–4):280–93. http://dx.doi.org/10.1080/15325008.2013.862325.

[3] Spiliotis K, Ramos Gutierrez AI, Belmans R. Demand flexibility versus physical network expansions in distribution grids. Appl Energy 2016;182:613–24. http://dx.doi.org/https://doi.org/10.1016/j.apenergy.2016.08.145.

[4] Muratori M, Rizzoni G. Residential demand response: Dynamic energy management and time-varying electricity pricing. IEEE Trans Power Syst 2015;31(2):1108–17. http://dx.doi.org/10.1109/TPWRS.2015.2414880.

[5] Schey S, Scoffield D, Smart J. A first look at the impact of electric vehicle charging on the electric grid in the EV project. World Electr Veh J 2012;5(3):667–78. http://dx.doi.org/10.3390/wevj5030667.

[6] Richardson P, Flynn D, Keane A. Impact assessment of varying penetrations of electric vehicles on low voltage distribution systems. In: IEEE PES General Meeting, 2010, 1–6. http://dx.doi.org/10.1109/PES.2010.5589940.

[7] Lehnhoff S, Krause O, Rehtanz C. Dezentrales autonomes energiemanagement. Automatisierungstechnik 2011;59(3):167–79. http://dx.doi.org/doi:10.1524/auto.2011.0906.

[8] Sperstad IB, Degefa MZ, Kjølle G. The impact of flexible resources in distribution systems on the security of electricity supply: A literature review. Electr Power Syst Res 2020;188. http://dx.doi.org/https://doi.org/10.1016/j.epsr.2020.106532.

[9] Heinrich C, Ziras C, Syrri AL, Bindner HW. EcoGrid 2.0: A Large-scale field trial of a local flexibility market. Appl Energy 2020;261. http://dx.doi.org/10.1016/j.apenergy.2019.114399.

[10] Ziras C, Heinrich C, Pertl M, Bindner HW. Experimental flexibility identification of aggregated residential thermal loads using behind-the-meter data. Appl Energy 2019;242:1407–21. http://dx.doi.org/10.1016/j.apenergy.2019.03.156.

[11] Mishra A, Irwin D, Shenoy P, Zhu T. Scaling distributed energy storage for grid peak reduction. In: Proceedings of the Fourth International Conference on Future Energy Systems. 2013, p. 3–14. http://dx.doi.org/10.1145/2487166.2487168.

[12] Zhang H, Liu B, Wu H. Smart grid cyber-physical attack and defense: A review. IEEE Access 2021;9:29641–59. http://dx.doi.org/10.1109/ACCESS.2021.3058628.

[13] Labrador Rivas AE, ao TA. Faults in smart grid systems: Monitoring, detection and classification. Electr Power Syst Res 2020;189. http://dx.doi.org/10.1016/j.epsr.2020.106602.

[14] Kemal M, Sanchez R, Olsen R, Iov F, Schwefel H-P. On the trade-off between timeliness and accuracy for low voltage distribution system grid monitoring utilizing smart meter data. Int J Electr Power Energy Syst 2020;121. http://dx.doi.org/10.1016/j.ijepes.2020.106090.

[15] Müller N, Afzal Z, Eliasson P, Ekstedt M, Heussen K. Threat scenarios and monitoring requirements for cyber-physical systems of energy flexibility markets. 2021, arXiv preprint arXiv:2111.03300.

[16] Singh D, Banyal RK, Sharma AK. Cloud computing research issues, challenges, and future directions. In: Rathore VS, Worring M, Mishra DK, Joshi A, Maheshwari S, editors. Emerging trends in expert applications and security. Singapore: Springer Singapore; 2019, p. 617–23.

[17] Yousefpour A, Fung C, Nguyen T, Kadiyala K, Jalali F, Niakanlahiji A, Kong J, Jue JP. All one needs to know about fog computing and related edge computing paradigms: A complete survey. J Syst Archit 2019;98:289–330. http://dx.doi.org/10.1016/j.sysarc.2019.02.009.

[18] Pereira J, Silveira M. Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention. In: 2018 17th IEEE international conference on machine learning and applications (ICMLA). IEEE; 2018, p. 1275–82. http://dx.doi.org/10.1109/ICMLA.2018.00207.

[19] Barua A, Muthirayan D, P. Khargonekar P, Al Faruque MA. Hierarchical temporal memory based one-pass learning for real-time anomaly detection and simultaneous data prediction in smart grids. IEEE Trans Dependable Secur. Comput. 2020. http://dx.doi.org/10.1109/TDSC.2020.3037054.

[20] Hollingsworth K, Rouse K, Cho J, Harris A, Sartipi M, Sozer S, Enevoldson B. Energy anomaly detection with forecasting and deep learning. In: 2018 IEEE international conference on big data (big data). IEEE; 2018, p. 4921–5. http://dx.doi.org/10.1109/BigData.2018.8621948.

[21] Neupane B, Pedersen TB, Thiesson B. Towards flexibility detection in device-level energy consumption. In: International workshop on data analytics for renewable energy integration. Springer; 2014, p. 1–16.

[22] Mocanu E, Nguyen PH, Gibescu M. Energy disaggregation for real-time building flexibility detection. In: 2016 IEEE power and energy society general meeting (PESGM). IEEE; 2016, p. 1–5. http://dx.doi.org/10.1109/PESGM.2016.7741966.

[23] Liu Y, Wu L, Li J. D-PMU based applications for emerging active distribution systems: A review. Electr Power Syst Res 2020;179. http://dx.doi.org/10.1016/j.epsr.2019.106063.

[24] Lazzaretti AE, Ferreira VH, Neto HV, Toledo LFRB, Pinto CLS. A new approach for event classification and novelty detection in power distribution networks. In: 2013 IEEE power & energy society general meeting. IEEE; 2013, p. 1–5. http://dx.doi.org/10.1109/PESMG.2013.6672703.

[25] Lazzaretti AE, Tax DMJ, Vieira Neto H, Ferreira VH. Novelty detection and multi-class classification in power distribution voltage waveforms. Expert Syst Appl 2016;45:322–30. http://dx.doi.org/10.1016/j.eswa.2015.09.048.

[26] Huang N, Fang L, Cai G, Xu D, Chen H, Nie Y. Mechanical fault diagnosis of high voltage circuit breakers with unknown fault type using hybrid classifier based on LMD and time segmentation energy entropy. Entropy 2016;18(9). http://dx.doi.org/10.3390/e18090322.

[27] Geng C, Huang S-J, Chen S. Recent advances in open set recognition: A survey. IEEE Trans Pattern Anal Mach Intell 2021;43(10):3614–31. http://dx.doi.org/10.1109/TPAMI.2020.2981604.

[28] Mahdavi A, Carvalho M. A survey on open set recognition. 2021, arXiv preprint arXiv:2109.00893.

[29] Niazazari I, Livani H. Disruptive event classification using PMU data in distribution networks. In: IEEE Power & Energy Society, editor. 2017 IEEE power & energy society general meeting. New York: IEEE; 2018, p. 1–5. http://dx.doi.org/10.1109/PESGM.2017.8274154.

[30] Phillips D, Overbye T. Distribution system event detection and classification using local voltage measurements. In: 2014 Power and Energy Conference at Illinois (PECI). IEEE; 2014, p. 1–4. http://dx.doi.org/10.1109/PECI.2014.6804576.

[31] Ahmad S, Lavin A, Purdy S, Agha Z. Unsupervised real-time anomaly detection for streaming data. Neurocomput 2017;262:134–47. http://dx.doi.org/10.1016/j.neucom.2017.04.070.

[32] Yaacob AH, Tan IK, Chien SF, Tan HK. Arima based network anomaly detection. In: 2010 Second International Conference on Communication Software and Networks. IEEE; 2010, p. 205–9. http://dx.doi.org/10.1109/ICCSN.2010.55.

[33] Ergen T, Kozat SS. Unsupervised anomaly detection with LSTM neural networks. IEEE Trans Neural Netw Learn Syst 2019;31(8):3127–41. http://dx.doi.org/10.1109/TNNLS.2019.2935975.

[34] Munir M, Siddiqui SA, Dengel A, Ahmed S. DeepAnT: A Deep learning approach for unsupervised anomaly detection in time series. IEEE Access 2019;7:1991–2005. http://dx.doi.org/10.1109/ACCESS.2018.2886457.

[35] Ren H, Xu B, Wang Y, Yi C, Huang C, Kou X, Xing T, Yang M, Tong J, Zhang Q. Time-series anomaly detection service at microsoft. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. ACM Digital Library, New York, NY, USA: Association for Computing Machinery; 2019, p. 3009–17. http://dx.doi.org/10.1145/3292500.3330680.

[36] Scheirer WJ, de Rezende Rocha A, Sapkota A, Boult TE. Toward open set recognition. IEEE Trans Pattern Anal Mach Intell 2013;35(7):1757–72. http://dx.doi.org/10.1109/TPAMI.2012.256.

[37] Bendale A, Boult T. Towards open world recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, 1893–1902.

[38] Rudd EM, Jain LP, Scheirer WJ, Boult TE. The extreme value machine. IEEE Trans Pattern Anal Mach Intell 2018;40(3):762–8. http://dx.doi.org/10.1109/TPAMI.2017.2707495.

[39] Geng C, Chen S. Collective decision for open set recognition. IEEE Trans. Knowl. Data Eng. 2022;34(1):192–204. http://dx.doi.org/10.1109/TKDE.2020.2978199.

[40] Gupta A, Gupta HP, Biswas B, Dutta T. Approaches and applications of early classification of time series: A review. IEEE Trans Artif Intell 2020;1(1):47–61. http://dx.doi.org/10.1109/TAI.2020.3027279.

[41] Ziras C, Heinrich C, Bindner HW. Why baselines are not suited for local flexibility markets. Renew Sustain Energy Rev 2021;135. http://dx.doi.org/10.1016/j.rser.2020.110357.

[42] Suel T. Delta compression techniques. Encycl Big Data Technol 2018;63:1–8. http://dx.doi.org/10.1007/978-3-319-63962-8_63-1.

[43] Hawkins J, Blakeslee S. On intelligence. USA: Macmillan; 2004.

[44] Wu J, Zeng W, Yan F. Hierarchical temporal memory method for time-series-based anomaly detection. Neurocomput 2018;273:535–46. http://dx.doi.org/10.1016/j.neucom.2017.08.026.

[45] Deb C, Zhang F, Yang J, Lee SE, Shah KW. A review on time series forecasting techniques for building energy consumption. Renew Sustain Energy Rev 2017;74:902–24. http://dx.doi.org/10.1016/j.rser.2017.02.085.

[46] Hyndman RJ. Forecasting with long seasonal periods. 2010, https://robjhyndman.com/hyndsight/longseasonality/. [Online; accessed: 23/07/2021].

[47] O'Shea K, Nash R. An introduction to convolutional neural networks. 2015, arXiv preprint arXiv:1511.08458.

[48] Hou X, Zhang L. Saliency detection: A spectral residual approach. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2007, p. 1–8. http://dx.doi.org/10.1109/CVPR.2007.383267.

[49] Yu Y, Qu W-Y, Li N, Guo Z. Open-category classification by adversarial sample generation. 2017, arXiv preprint arXiv:1705.08722.

[50] Smith SW. The scientist and engineer's guide to digital signal processing. USA: California Technical Publishing; 1997.

Department of Wind and Energy Systems
Division for Power and Energy Systems (PES)
Technical University of Denmark
Frederiksborgvej 399
DTU Risø Campus, 4000 Roskilde
Denmark

https://windenergy.dtu.dk/english
Tel: (+45) 46 77 50 85
E-mail: communication@windenergy.dtu.dk