Conformal Prediction of Reinforcement Learning Agents Short-term Operation

Hugo Cardante
Master in Computer Science
Department of Computer Science
Faculty of Sciences of the University of Porto
2025

U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

MSC

Conformal Prediction of Reinforcement Learning Agents Short-term Operation

Hugo Cardante

Dissertation carried out as part of the Master in Computer Science

Department of Computer Science

2025

Supervisor

Prof. Pedro Ferreira, Assistant Professor with Habilitation, Faculdade de Ciências

Co-supervisor

Prof. Ricardo Bessa, Center Coordinator and Senior Researcher, INESC TEC

Acknowledgements

I would like to thank Professor Pedro Ferreira, my advisor, for the opportunity to participate in this project.

To Professor Ricardo Bessa, who was extremely helpful throughout the project, providing valuable guidance and always offering new ideas to explore, I extend my gratitude for his expertise and input.

To Carla Gonçalves, who was always ready to discuss the next steps and help resolve any doubts that would suddenly arise, I thank you for your patience and thoughtful feedback. Your help was really invaluable.

To Margarida Costa, who helped me get started with the Grid2Op platform, my heartfelt thanks.

To my friends: Marco Gonçalves and Duarte Nóbrega: These were five years in which I had the privilege to meet both of you. Five years in which we learned core computer science concepts, but also five years in which we learned how to work as a team. And what a team we made. Together, we are unstoppable.

To my sister, Sónia Cardante, and my brother from another mother, Guilherme Lima, for all the support and strength given to me throughout these five years, I thank from the bottom of my heart. Nothing would be possible without your support. I would also like to thank my father, and the rest of my family, whose support was vital for the conclusion of my academic studies.

This work is part of the AI4REALNET (AI for REAL-world NETwork operation) project, which received funding from European Union's Horizon Europe Research and Innovation programme under the Grant Agreement No 101119527, and from the Swiss State Secretariat for Education, Research and Innovation (SERI). This project is funded by the European Union and SERI. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union and SERI. Neither the European Union nor the granting authority can be held responsible for them.

To my mother, the pillar of love

Resumo

As redes eléctricas são infraestruturas essenciais para a sociedade moderna. Não apenas como um negócio, mas sobretudo como uma necessidade crítica sem a qual o nosso quotidiano não é compatível. Com a integração e transição contínua para fontes de energia renovável, há uma necessidade constante de quantificar a incerteza das previsões nas redes eléctricas. Esta quantificação é extremamente importante para apoiar a tomada de decisões por parte dos operadores da rede elétrica, e assim garantir que a rede permanece sempre operacional de modo a melhorar a confiança nos sistemas de IA quando usados neste contexto. Eventos recentes mostraram que falhas na rede podem impactar milhares de vidas e deixar sectores essenciais como a educação e a saúde em situações muito difíceis.

Para abordar estes desafios, ambientes de simulação como o Grid2Op permitem o desenvolvimento de agentes de RL que podem escolher a melhor ação a tomar quando a rede está em risco. O Grid2Op modela a rede eléctrica como um grafo onde as arestas representam linhas eléctricas e os nós representam barramentos que ligam geradores, cargas e outros componentes da rede. Durante um episódio do Grid2Op, o *fluxo de potência* é continuamente simulado, e observações são recolhidas a cada intervalo temporal para múltiplas varíaveis. Uma variável de particular importância para esta tese é o valor ρ (rho), que representa a capacidade de cada linha eléctrica e é para cada linha i, a razão entre o fluxo de potência observado na linha i e o limite térmico da mesma linha. Este valor é muito importante porque valores elevados de ρ indicam risco iminente de desconexão da linha o que pode resultar num potencial colapso da rede eléctrica.

Nesta tese, utilizamos um previsor pré-treinado que prevê a carga e geração da rede com base no contexto histórico, executando depois simulações de fluxo de potência para obter os valores ρ, tendo em atenção as ações que um agente baseado em RL tomaria considerando essas previsões. A nossa principal contribuição é a criação de uma *framework* que utiliza um *forecaster* pré-treinado e permite aos utilizadores implementar qualquer algoritmo de Previsão Conforme (PC), tornando possível avaliar sistematicamente o desempenho destes algoritmos ao quantificar a incerteza. A Previsão Conforme é um método estatistico que, dado um parámetro $\alpha \in [0,1]$ que pode ser ajustado pelo utilizador, permite transformar uma previsão de um ponto num intervalo, com garantias estatísticas de que o ponto real vai estar contido no intervalo com probabilidade $1-\alpha$, dando assim mais informação sobre a incerteza da previsão.

Implementamos e adaptamos quatro algoritmos de PC com múltiplas variantes, e fornecemos uma análise detalhada do seu desempenho em contextos de previsão de redes eléctricas. Os resultados mostram os benefícios práticos e também algumas limitações dos métodos de *CP* para quantificação de incerteza em redes eléctricas.

Palavras-chave: Previsão Conforme (PC), Quantificação de Incerteza, Robustez, Resiliência, Redes Elétricas, Reinforcement Learning (RL), Adversarial Attacks, Grid2Op, CurriculumAgent

Abstract

Power grids are essential infrastructure for modern society, and are not just a business but also a critical necessity without which our society cannot function. With the ongoing integration and transition to renewable energy sources, there is a growing need to quantify the uncertainty of power grid forecasts. This quantification is extremely important for supporting human operators in decision-making and in this way guarantee continuous grid operation, therefore enhancing our trust in the AI forecasters in this context. Recent events have demonstrated that grid failures that can affect the lives of thousands of people and leave essential sectors such as education and healthcare in dire straits.

To address these challenges, simulation environments like Grid2Op enable the development of RL agents that can select an action to take when the grid faces operational risks. Grid2Op models the power grid as a graph where edges represent power lines and nodes represent buses that connect generators, loads, and other grid components. During a Grid2Op episode, power flow is continuously simulated, and observations are collected at each time step. One observation of particular importance to this thesis is the ρ (rho) value, which represents the capacity utilization of each power line and is defined for line i as the ratio between the observed current flow of line i, divided by the termal limit of that line. This value is very important because elevated ρ values indicate imminent risk of line disconnection and potential power grid blackout.

In this thesis we use a pre-trained forecaster that predicts grid load and generation based on historical time steps, subsequently running power flow simulations to obtain the ρ values, considering the actions taken by a pre-trained RL-based agent for those forecasts. Our main contribution is the development of a *framework* that integrates pre-trained forecasters and pre-trained agents with Conformal Prediction (CP) algorithms (that can be implemented by the user), enabling the user to predict the grid operation as well as the risk of congestion. CP is a statistical method that, given a parameter $\alpha \in [0, 1]$, which can be ajusted by the user, can transform a point forecast in an uncertainty interval, with statistical guarantees that the real point will be contained in the interval with probability $1 - \alpha$, providing more information about the uncertainty of the forecast.

We implement and adapt four CP algorithms with multiple variations, providing detailed analysis of their performance in power grid uncertainty forecasting. The results show the practical applicability and limitations of CP methods for power grid uncertainty quantification.

Keywords: Conformal Prediction (CP), Uncertainty Quantification, Robustness, Resilience, Power Grids, Reinforcement Learning (RL), Grid2Op, CurriculumAgent

Table of Contents

List of Figures

List of Abbreviations

ACI Adaptive Conformal Inference. xi, xii, 15, 34–36, 42, 46, 48–52, 54–60, 63–66, 68, 77, 78, 80–87, 90–98, 100–104, 106

AI Artificial Intelligence. v, 1, 2, 4, 15

CP Conformal Prediction. iv–vi, viii, xi, 1–4, 7–9, 11–15, 22–30, 34, 37, 38, 43–45, 47, 48, 50, 54, 56, 59, 66, 68, 75, 77, 86, 88

CQR Conformalised Quantile Regression. 23, 37, 38, 41, 107

HBGB Histogram-based Gradient Boosting Regressor. 4, 21, 22, 27

IA Inteligência Artificial. iii

k-NN K-Nearest Neighbours. viii, xi, xii, 23, 31–34, 48–52, 54–56, 60, 62–64, 76–78, 88–90, 100–102

MDP Markov Decision Process. 15

RB-CQR Residual-Based Conformalised Quantile Regression. xi, xii, 38–40, 42, 46, 48–50, 52–54, 56–60, 63–66, 80–87, 91–98, 101, 103–106

RL Reinforcement Learning. iii, v, vi, 1, 2, 4, 15, 17, 20–22, 26

1. Introduction

The inherent challenges of power grid operations, accentuated by the ongoing energy transition from the fossil energy sources to renewable ones, as well as their integration, introduces a lot of uncertainty. Even with advanced simulation tools, power system engineers face an increasing cognitive load due to the sheer volume and complexity of decisions required for real-time grid management. To address these challenges, there is growing interest in using Reinforcement Learning (RL) agents to assist grid operators. These Artificial Intelligence (AI) assistants have the objective of helping to manage, and prevent network congestions. This is because when transmission lines become overloaded, they can be deactivated for security which can trigger failures that propagate throughout the entire network, which may result in blackouts that can affect millions of people. RL agents can reconfigure the grid topology dynamically, for example by connecting and disconnecting some power lines to alleviate these congestions.

The agents generate the actions required for managing the grid (congestion management), but it is also very important to understand when these AI systems might fail or make bad decisions, especially if data quality can be compromised (for example by adversarial agents), or when the operating conditions are different from the ones that were encountered during agent training, which is common, for example, when going from a simulation environment to the real world. Grid operators should be able to assess not only what actions an agent will take, but also what is the confidence level associated with these decisions. Without reliable uncertainty quantification, operators cannot validate agent recommendations or determine when human intervention is necessary, which leads to a potential compromise in grid safety and reliability.

Conformal Prediction (CP) [1] provides statistically rigorous prediction regions for forecasts, providing quantification of uncertainty for each point forecast. Using CP may support the design of AI assistants that can issue “binary alerts” with an associated “level of confidence”, giving operators a clearer basis for action and enabling more effective decisions by using AI.

Applied to the electricity sector and RL-based grid management, CP holds a lot of promise. By providing reliable and quantifiable uncertainty measures, it can help grid operators to make more informed decisions, which in turn results in a safer environment, especially when we consider the dynamics of modern power grids, that can suffer extreme weather events or sudden shifts in renewable energy generation. This uncertainty quantification can also directly contribute to minimising operational costs and facilite the energy transition by reducing renewable energy curtailment, improving the carbon intensity of grid operations.

Equally important, uncertainty-aware forecasts for RL agent predictions foster human trust in AI systems. By showing the uncertainty associated with both system forecasts and agent recommendations, these methods can help to prevent over-reliance as well as distrust, which can lead to more effective human-AI teaming, reduce the human operator's cognitive workload and stress, and enhance the resilience of power grids to extreme events and potential cyberattacks. In essence, embracing uncertainty quantification techniques like CP can be important for building a more reliable, secure, and sustainable energy future.

1.1. Research Questions

In this thesis, we focus on the following research questions (RQ):

RQ1: How can we predict the operational state of power grids, specifically the risk of congestion, while accounting for the impact of Reinforcement Learning (RL) agents' actions?

A framework based on conformal prediction is used to forecast the lines loading in a power grid (i.e., detect potential congestions in the lines) in the short-term (1-hour ahead), considering a deterministic forecast obtained with a pre-trained load and generation time series forecasting model, and power flow calculations*, and the impact of actions generated by an RL-based agent. The output is an uncertainty forecast of the future power grid operating conditions, considering the aleatoric uncertainty associated with load and generation forecasts, as well as the epistemic uncertainty of the RL agent. From these uncertainty forecasts it is possible to obtain a probabilistic indicator about the risk of congestion in the power grid. *A congestion problem in power grids occurs when the power flow on one or more transmission lines approaches or exceeds their thermal or stability limits, risking equipment damage or system instability. It typically arises from high demand, generation patterns, or network constraints that overload specific lines.*

*Power flow calculation determines the steady-state voltages, currents, and power flows throughout a power grid, given the network topology, line parameters, and specified power injections (generation and demand). It solves nonlinear algebraic equations (typically via Newton–Raphson or Gauss–Seidel methods) to find the operating point of the system.

RQ2: How can we select an appropriate Conformal Prediction (CP) method and evaluate its skill (calibration, coverage) in short-term forecast horizons?

Four different CP algorithms and their variants were implemented and adapted to forecast the short-term operation of power grids (i.e., up to one hour ahead). These forecasts explicitly account for the impact of actions taken by a reinforcement learning (RL)-based agent on the grid state. The algorithms were evaluated in terms of the coverage and efficiency of their prediction intervals over a multi-period time horizon. The ultimate objective is to produce reliable forecasts of congestion risk in grid branches.

1.2. Thesis Structure

The remainder of this thesis is organised as follows:

- Chapter 2 reviews the background and related work, covering power grid operations, uncertainty quantification and Conformal Prediction (CP). It also presents the State of the Art for using CP in the power systems sector.
- Chapter 3 introduces the methodology, including the integration of pre-trained forecasters with CP algorithms and the use of the Grid2Op simulation environment.
- Chapter 4 presents some of the implementation details, including parallelisation of the framework, and some of the tactics used for storage efficiency.
- Chapter 5 presents the experimental results obtained from our work. It presents three different Case Studies, each of them building on the concept of uncertainty quantification, and presenting the effects of adversarial attacks in uncertainty quantification.
- Chapter 6 concludes the thesis, summarising what was done, and giving directions for future work.

2. Background and State-of-the-art

This chapter introduces the background for the thesis.

We begin by presenting the project in which this thesis is integrated. Following this, we provide an overview of the problem we are trying to tackle before introducing a possible mitigation: Conformal Prediction (CP), the core concept upon which this thesis is built.

We then present the AI-friendly digital environment used for our work, *Grid2Op* [2], which allows the development, training and performance evaluation of RL agents that act in a simulated power grid. We proceed to describe the experimental setting in which our studies were conducted. Finally, we introduce two essential tools used for this research: the pre-trained CurriculumAgent [3] and the pre-trained Histogram-based Gradient Boosting Regressor (HBGB) [4] forecaster.

2.1. The AI4REALNET Project

The AI4REALNET [5] project aims to improve the real-time and predictive operational capabilities of critical network infrastructures. It does so by integrating Artificial Intelligence (AI) algorithms into open-source, AI-friendly digital environments, and by developing human-centred AI-based decision systems that prioritise effective human-machine interaction. The project targets three vital European sectors: electricity networks, railways, and air traffic management.

This thesis focuses on the **electricity networks** sector.

For this purpose, we use Grid2Op [2], a friendly digital platform developed by the French Transmission Network (RTE), and adopted by the AI4REALNET project for its electricity sector use cases. Grid2Op enables the generation of high-quality powergrid data (chronics) which can be used to train AI agents that make decisions on the grid, and allows the evaluation of these agents (e.g, RL agents). We present the digital platform in more detail in Section 2.5.

2.2. Point Forecasts

A point forecast is a single numerical prediction of a random variable y_{t+h}, made at time t for horizon h [6]. The horizon is the time period in the future, for which the forecast is made.

The performance of point forecaster is formally defined as:

$$\overline{S} = \frac{1}{n}\sum_{i=1}^{n} S\left(\hat{y}_i, y_i\right), \tag{2.1}$$

where we have n forecast cases with corresponding point forecasts $\hat{y}_1, ..., \hat{y}_n$ and observations $y_1, ..., y_n$ [7]. We refer to S as the *scoring function*.

Gneiting and Tilmann [7] demonstrate that relying on measures such as the mean absolute error, or the (root) mean squared error can lead to grossly misguided inferences, unless the scoring function and the forecasting task are carefully matched.

2.3. Probabilistic Forecasting

Traditional forecasting methods produce single-valued predictions $\hat{y}$ (point forecasts) of the target variable Y. While useful, point forecasts fail to capture the uncertainty in many real-world problems, particularly in volatile domains such as power systems, in which a single point forecast does not provide enough assurance that the system is going to behave as intended.

Probabilistic forecasting extends point forecasting by characterizing the uncertainty of future outcomes. Instead of a single prediction, the forecaster provides a full or partial description of the predictive distribution of Y. This can be represented in different but complementary forms [8]:

Conditional probability functions

The predictive distribution functions (PDF) or cumulative distribution functions (CDF) are estimated for the target variable Y_{t+h} given all the information available at time t, Ω_t. These functions model the entire distribution of the conditional variable $\hat{Y}_{t+h}|\Omega_t$.

Conditional quantile forecast

For a given level $\alpha \in [0,1]$, the conditional quantile forecast $\hat{q}^{\alpha}_{t+h}$ is defined such that $P(Y_{t+h} \leq \hat{q}^{\alpha}_{t+h}|\Omega_t) = \alpha$, where Ω_t denotes the information available at time t. This means, for example, that a 0.95-quantile forecast represents a value that the future realization will fall below with probability 95%.

Prediction intervals

For a given probability level $\alpha \in [0,1]$, the prediction interval for the future observation Y_{t+h} is defined as the interval I^{α}_{t+h} that satisfies

$$P\left[Y_{t+h} \in I^{\alpha}_{t+h|t} \,\middle|\, \Omega_t\right] = \alpha$$

Such intervals are typically defined by combining two conditional quantile forecasts. For example, a 90% forecast interval is obtained from the 5% and 95% conditional quantiles.

Scenario forecasts

A finite set of representative trajectories (scenarios) is generated, often used in power systems to capture dependencies across time or space [9].

Each representation has strengths and limitations: quantiles are compact and interpretable, intervals are directly linked to risk levels, scenarios are useful for downstream optimization, and full distributions are the most expressive but often require strong assumptions.

Common approaches to probabilistic forecasting include: i) parametric methods, which assume a specific distributional form (e.g., Gaussian regression); ii) non-parametric methods, which estimates e.g. conditional quantiles without requiring a parametric form. Examples include linear quantile regression, GBR, KDE; and iii) semi-parametric methods like the combination of non-parametric method for central quantiles and extreme value theory for quantile with nominal level $\alpha \in]0, 0.05] \vee [0.95, 1[$.

Probabilistic forecasting methods provide useful ways to quantify uncertainty, but they often rely on strong distributional assumptions or asymptotic arguments. Even quantile regression, though non-parametric, only guarantees asymptotic validity and may lead to miscalibrated intervals in finite samples. In power system time series, these issues are critical because data are limited, distributions evolve over time, and temporal dependence breaks the exchangeability assumption required by standard conformal prediction.

Despite this, conformal prediction offers distribution-free, finite-sample guarantees under exchangeability and can be combined with quantile regression (CQR) to yield adaptive and calibrated intervals. Recent research has also proposed extensions of CP that relax exchangeability, making it applicable to dependent data such as time series. This thesis is motivated by the need to investigate these methods in the Grid2Op environment, where robust and efficient uncertainty quantification is essential for risk-aware decision-making.

2.4. Conformal Prediction

Conformal Prediction (CP), also known as conformal inference, is a statistical method that can be used with any method of point prediction for classification or regression [1, 10]. Originally developed by Vovk, Gammerman, and Shafer [11], CP operates on top of any pre-trained predictive model to provide uncertainty quantification.

In the regression setting, where $\hat{y}$ is the point forecast, CP provides prediction intervals around $\hat{y}$ that are guaranteed to contain the true value with a user-specified probability. As noted in [1], CP "can be seen as a method for taking **any heuristic notion of uncertainty** from **any model** and converting it to a rigorous one." This means that CP can transform the traditional point forecasting problem into an interval prediction problem with rigorous statistical guarantees, without making assumptions about the underlying model or data distribution.

2.4.1 Overview and Assumptions

Given a user-specified confidence level α, the goal of CP is to produce a prediction set $C(X_{test})$ such that the true outcome is contained within the set with probability at least $1-\alpha$. This is formalized as:

$$P(Y_{test} \in C(X_{test})) \geq 1 - \alpha$$

A key strength of CP is that this coverage guarantee holds distribution-free, meaning that it does not require any assumptions about the form of the underlying data distribution, but instead CP relies on a much weaker assumption: exchangeability of the calibration and test examples.

Exchangeability means that the joint distribution of the data is invariant under permutations. While all i.i.d. data are exchangeable the converse is not always true, making exchangeability a slightly more general assumption. Under this condition, CP ensures finite-sample validity, i.e. the coverage guarantee holds exactly for any sample size. This property distinguishes CP from many probabilistic forecasting methods, which often rely on asymptotic arguments or parametric assumptions.

At the same time, the exchangeability requirement highlights an important limitation: in many real-world applications, especially in time series, observations are temporally dependent and therefore not exchangeable. In such cases, the guarantees of CP no longer hold exactly. This has motivated the development of extensions of CP designed for dependent or non-exchangeable data, which typically yield asymptotic rather than exact coverage. These extensions will be discussed in Subsection 2.4.4.

2.4.2 Core Concepts

This subsection introduces the mathematical foundations of Conformal Prediction (CP) and discusses some of the core concepts that are central to its theoretical guarantees and practical applications.

2.4.2.1 Conformity Scores

The conformity score function $S(\hat{f}(x), y)$ is the central mechanism through which conformal prediction quantifies uncertainty, because it is used to calculate the conformity scores [10]. This conformity scores represent how well new data conforms to already seen data. Therefore, the design of this function directly influences the efficiency of the resulting prediction intervals. Common choices for regression include the absolute residual $S(\hat{f}(x), y) = |\hat{f}(x) - y|$ and the squared residual $S(\hat{f}(x), y) = (\hat{f}(x) - y)^2$, although more advanced scores can incorporate additional information about prediction uncertainty [12] (e.g, asymmetric score functions, such as $S(\hat{f}(x), y) = \hat{f}(x) - y$ can be incorporated when over-prediction and under-prediction have different costs).

2.4.2.2 Distribution-Free Guarantees

A distinctive feature of CP is its distribution-free nature: it provides valid coverage guarantees without requiring knowledge of the true data-generating process. This contrasts with other uncertainty quantification approaches. For example, Bayesian methods [13] rely on priors and likelihood assumptions, while bootstrap methods [14, 15] often depend on asymptotic approximations. In contrast, CP guarantees finite-sample validity under minimal assumptions.

This property means that CP can provide reliable uncertainty estimates even when the underlying model is unreliable or the data distribution is unknown. The method essentially provides a safety net that ensures valid coverage regardless of model choice and distribution, having as the only requirement that the data is exchangeable. We later see that CP can be applied to time-series, which does not satisfy exchangeability.

2.4.2.3 Marginal vs. Conditional Coverage

CP provides strong marginal coverage guarantees, which means that it guarantees $(1-\alpha)$ coverage for all test points on average, but does not provide conditional coverage. Conditional coverage [16] ensures that the coverage probability holds not just on average, but for specific regions of the input space:

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}})|X_{\text{test}} = x) \geq 1 - \alpha \tag{2.2}$$

for most values of x.

When considering Vanilla CP, which uses a single global quantile to construct prediction intervals for all test points, achieving conditional coverage is impossible without additional assumptions [17], because Vanilla CP gathers information across the entire input space and therefore leads to coverage guarantees that are only marginal.

2.4.3 Classical Algorithms

Classical CP algorithms provide the foundation for modern approaches to uncertainty quantification. They rely on the assumption of exchangeability and use calibration data to construct prediction sets with guaranteed coverage. The two most widely studied variants are full conformal prediction and split conformal prediction. We now describe each method in detail.

2.4.3.1 Full Conformal Prediction

The original conformal prediction framework, known as **full conformal prediction** [10], provides strong theoretical guarantees through a procedure that tests every possible prediction value. Despite being computationally expensive, this approach establishes the theoretical foundation upon which all subsequent conformal prediction methods are built.

The idea is to evaluate how well each candidate prediction value **conforms** to the patterns observed in the training data. For each potential output value, the algorithm augments the training dataset with the test input paired with that candidate value, retrains the model, and computes a conformity score that measures how typical (or atypical) this augmented example appears relative to the training data. This conformity score is calculated by using the conformity score $S(\hat{f}(x), y)$ function.

The algorithm for Full Conformal Prediction [11, 18] is presented next (Algorithm 1).

Algorithm 1 Full Conformal Prediction for Regression (Grid-based)

Require: Training set $\{(X_i, Y_i)\}_{i=1}^{n}$, test inputs X_{test}, significance level α, Grid of candidate values $\mathcal{G} = \{y_1, y_2, \ldots, y_m\}$ spanning plausible range, Conformity score function $S(\hat{y}, y)$

for $x \in X_{\text{test}}$ **do**
    Initialize prediction set $C(x) = \emptyset$
    for each candidate value $y \in \mathcal{G}$ **do**
        Augment training set: $\mathcal{D}_{\text{aug}} = \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$
        Train model $\hat{f}$ on $\mathcal{D}_{\text{aug}}$
        for $i = 1, \ldots, n$ **do**
            Compute prediction $\hat{y}_i = \hat{f}(X_i)$
            Compute conformity score $s_i = S(\hat{y}_i, Y_i)$
        end for
        Compute prediction for candidate: $\hat{y} = \hat{f}(x)$
        Compute conformity score for candidate: $s_{\text{test}} = S(\hat{y}, y)$
        Calculate rank r of s_{test} among $\{s_1, \ldots, s_n, s_{\text{test}}\}$
        Compute threshold: $\tau = \lceil (n+1)(1-\alpha) \rceil$
        if $r \leq \tau$ **then**
            $C(x) = C(x) \cup \{y\}$
        end if
    end for
end for
return $C(x), \forall x \in X_{\text{test}}$ ▷Prediction interval: $[\min(C(x)), \max(C(x))]$ for each x

Full conformal prediction provides exact finite-sample coverage guarantees under the exchangeability assumption discussed in the previous sections. Specifically, the prediction sets satisfy:

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}})) = 1 - \alpha \ . \tag{2.3}$$

This coverage probability holds exactly for any finite sample, making the method particularly valuable in scenarios with limited data.

Because of the distribution-free nature of full conformal prediction, the coverage guarantee holds regardless of the quality of the underlying predictive model, the choice of conformity score function, or the data distribution which makes it particularly appealing for practical applications where distributional assumptions are difficult to verify or may be violated [12, 19].

However, the computational cost of full conformal prediction is prohibitive for most practical applications because the model must be retrained for each candidate prediction value which means that computational complexity scales with the size of the output space, making it impractical for continuous regression problems.

2.4.3.2 Split Conformal Prediction

To address the computational limitations of full conformal prediction, **split conformal prediction** [10, 20] (also known as Vanilla CP) provides a practical approximation that maintains valid coverage guarantees with reduced computational cost. This approach has the disadvantage in what regards coverage, because it does not have the exact coverage guarantee of full conformal prediction but improves the computational tractability, which makes it applicable to real-world problems.

The idea of split conformal prediction is that the model can be trained once on a portion of the data, and then a separate calibration dataset can be used to determine the appropriate threshold for prediction interval construction. This separation of model training and calibration eliminates the need for repeated model retraining while preserving the essential properties of CP.

The algorithm for Split Conformal Prediction [10, 20, 21] is presented next (Algorithm 2):

Algorithm 2 Split Conformal Prediction

Require: Training set $\mathcal{D}_{\text{train}} = \{(X_{train}, Y_{train})\}$, calibration set $\mathcal{D}_{\text{cal}} = \{(X_{cal}, Y_{cal})\}$, test input X_{test}, significance level α

1: Train predictive model $\hat{f}$ on $\mathcal{D}_{\text{train}}$
2: **for** $(x_i, y_i) \in (X_{cal}, Y_{cal})$ **do**
3: Compute forecast $\hat{y}_i = \hat{f}(x_i)$
4: Compute conformity score $s_i = S(\hat{y}_i, y_i)$
5: **end for**
6: Compute $(1-\alpha)$ empirical quantile: $\hat{q} = \text{Quantile}_{1-\alpha}(\{s_1, \ldots, s_n\})$
7: **for** $x \in X_{test}$ **do**
8: $\hat{y} = \hat{f}(x)$
9: $C(x) = \{y : S(\hat{y}, y) \leq \hat{q}\}$
10: **end for**
11: **return** $C(x), \forall x \in X_{test}$

After training the predictive model on the training set, conformity scores are computed for all examples in the held-out calibration set, using the conformity score function. These scores form an empirical distribution that approximates the distribution of the conformity scores observed when the model makes predictions. The $(1-\alpha)$ quantile of this distribution serves as a threshold that determines the boundary of the prediction interval.

Split conformal prediction provides marginal coverage guarantees that are weaker than the ones of full conformal prediction:

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}})) \geq 1-\alpha \ . \tag{2.4}$$

The inequality rather than equality reflects the approximation involved in using a single global quantile rather than the point-specific ranking procedure of full conformal prediction. However, this coverage probability is computed marginally over the randomness in both the calibration procedure and the test point selection, providing a finite-sample guarantee that holds exactly for any sample size.

In sum, the computational efficiency of split conformal prediction represents a trade-off: by training the model only once, the method becomes practical for large-scale applications, even though this efficiency comes at the cost of weaker theoretical guarantees and the need to set aside data for calibration, which reduces the amount of data available for model training. The exchangeability assumption in split conformal prediction is also more relaxed than in the full version: the method requires only that the calibration set and test data are exchangeable, not that the entire dataset satisfies this property. In practise, this weaker assumption makes split conformal prediction more applicable, particularly in scenarios where the training data may come from a slightly different distribution than the test data, as long as the calibration and test data have the same distribuition.

2.4.4 Extensions and Adaptations

These classical CP methods have some important limitations: Full CP is computationally prohibitive, and split CP, though efficient, is not adaptive to heteroskedastic data and still relies on exchangeablitiy assumptions that rarely hold in time series. These challenges have motivated a rich line of research on extensions and adaptations of CP. Broadly, these developments aim either to improve the efficiency and adaptivity of prediction intervals, or to relax the exchangeablity requirement in order to handle dependent or distribution-shifted data. In the following subsections, we review representative approaches from both directions.

2.4.4.1 Adaptive and Efficient Intervals

Standard conformal prediction faces limitations when dealing with varying uncertainty across the input space. To understand these limitations, it is important to distinguish between two related but distinct concepts: adaptivity and heteroskedasticity.

Heteroskedasticity [20, 22, 23] is a property of the data where the conditional variance of the target variable changes across the input space. For example, if we consider financial forecasting's market volatility, it may be low during stable periods but high during times of economic uncertainty. This would mean that the market volatility data would present heteroskedascticity. Adaptivity [21, 24, 25], on the other hand, refers to the ability of the CP method to adjust its interval sizes based on local uncertainty patterns. When data is heteroskedastic, we want our CP method to be adaptive. However, standard split conformal prediction is not naturally adaptive.

As described previously, split conformal prediction uses a single global quantile $\hat{q}$ to construct prediction intervals for all test points, regardless of their input characteristics. This global threshold means that the method does not adapt to local uncertainty patterns. The same threshold determines the intervals everywhere. If heteroskedasticity is present, the global approach creates problems. The conformity scores computed during calibration reflect the varying levels across different regions, but the global quantile means that high-variance regions will make intervals wider everywhere, while low-variance regions contribute to making them narrower everywhere, ignoring local differences in conformity scores. This can result in prediction intervals that can be too large or to narrow unecessarily, leading to bad conditional coverage properties.

Despite the global quantile limitation, prediction intervals can still vary in size across different regions of the input space, but this depends on the design of the conformity score function $S(\hat{f}(x), y)$. If the conformity score captures input-dependent uncertainty patterns, for example, by incorporating model confidence estimates or local prediction accuracy, then the resulting intervals may exhibit adaptive behaviour.

For example, Weighted conformal prediction [26] provides a method, where conformity scores are weighted during calibration based on estimated local difficulty or variance. Another approach also uses locally-weighted residuals as conformity scores: $S(\hat{f}(x), y, \hat{\rho}(x)) = \frac{|\hat{f}(x)-y|}{\hat{\rho}(x)}$, where $\hat{\rho}(x)$ denotes an estimate of the conditional mean absolute deviation at input x [12]. With this design, regions with higher estimated uncertainty will tend to produce prediction intervals that are wider relative to the local noise level, providing some degree of adaptivity to heteroskedastic patterns. However, this adaptivity through conformity score design is not automatic and requires calculating local uncertainty patterns.

Advanced Adaptive Methods: More advanced conformal prediction variants have been developed specifically to address the limitations of global quantiles. Conformalised quantile regression [22] models conditional quantiles of the target distribution and then applies conformal prediction to these quantile estimates. This approach can naturally adapt to heteroskedastic patterns by modeling the conditional distribution directly rather than relying on a global threshold.

Other locally adaptive conformal prediction methods aim to improve prediction sets by using different approaches. Romano et al. develop specialized conformity scores for classification that account for local uncertainty patterns through generalized inverse quantile scores [27]. Guan et al. [28, 29] propose localized conformal prediction that weights training samples based on their proximity to the test point using localizer functions.

While conformal prediction guarantees valid coverage regardless of the underlying model quality, the efficiency of the resulting prediction intervals depends on the model performance. **Efficiency**, in the context of CP, particularly in the case of regression, refers to the width of the prediction intervals: more efficient methods produce smaller intervals while maintaining the same coverage guarantee. Also, better predictive models lead to more efficient conformal intervals because they produce more consistent conformity scores during calibration: when a model makes accurate predictions, the conformity scores are normally small and concentrated, resulting in a smaller calibration quantile and consequently tighter prediction intervals. One the other hand, poor-performing models produce varying conformity scores, leading to larger quantiles and, consequently, wider intervals. The coverage guarantee remains valid regardless of model quality, with even poorly performing models being able to provide reliable uncertainty quantification, but this comes at the cost of reduced efficiency. Conformal prediction efficiency can be improved through conformity score design, conditional modelling approaches [22], and ensemble methods [30].

2.4.4.2 Beyond Exchangeability

The exchangeability assumption underlying CP can be problematic in several practical settings, limiting the theoretical validity of standard CP methods. Understanding these limitations is important for proper application of CP.

Distribution Shift: When test data comes from a different distribution than calibration data, exchangeability is violated. This scenario is common in real-world applications where models are deployed in environments that differ from the training conditions. Adaptive Conformal Inference (ACI) [31] was a method developed precisely for this setting.

Time Series Data: Sequential data violates exchangeability due to temporal dependencies, where the order of observations must not be ignored: latter observations may be more informative about future values than distant ones, violating exchangeability. Recent work has developed adaptive conformal inference methods to handle this case, updating the conformal quantile dynamically based on recent observed coverage rates [31–33]. Other line of work as been established, a method called EnbPi (and its variants), which wraps around any bootstrap ensemble estimators, in order to construct distribution free prediction intervals for dynamic time-series [30, 34].

These limitations reinforce the importance of selecting appropriate variants when standard methods do not work.

CP is an active area of research, with new advances emerging each year.

2.5. Grid2Op: The AI-friendly Digital Environment

The modern complexity of power systems creates a growing need for the development and testing of control strategies [35]. Because experimentation on real power grids is not possible due to the high cost and risk associated, there is a need for simulation environments which are essential for advancing research in power system operations. Grid2Op [2] is an open-source platform designed for applying AI to power grid management, and provides a realistic simulation of power grid operations while abstracting away most of the complexity that would make research intractable. The framework adopts a Markov Decision Process (MDP) [36, 37] formulation, making it particularly well-suited for RL approaches to power grid control, and is used for the L2RPN [38] (Learning To Run a Power Network) competition series.

2.5.1 Graph-Based Control

In Grid2Op a dynamic graph is controlled where both the observation space and action space involve graph modifications. The power grid is defined as follows:

Definition 2.1 (Power Grid Graph)**.** A power grid in Grid2Op is represented as a graph $G = (V, E)$ where V represents the set of buses (nodes) and E represents the set of transmission lines (edges). Each edge $e \in E$ has an associated power flow and thermal limit, and each node $v \in V$ has either generators or loads connected to it.

Power grid control requires dynamically reconfiguring the network topology to satisfy operational constraints. A visual representation of a topology power grid graph in Grid2Op can be observed in Figure 2.1.

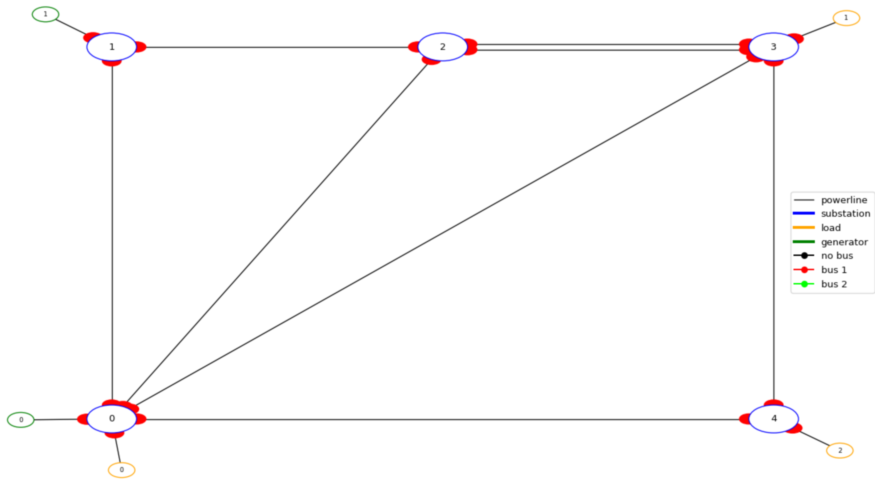

Figure 2.1: Example of a Grid2Op powergrid visual representation. Each edge in this graph represents a powerline. The substations are the nodes in the graph, and allow power to flow from one place to another, connecting two or more powerlines. The loads (circles in yellow) represent power drawn from a grid, for example a small city or a big industrial firm. The generators, in green, represent power that goes into the grid, for example power plants such as coal, nuclear or wind farms. The buses split a substation, i.e if two or more lines share a bus, they connect by a direct electrical path.

All the transmission lines must be below thermal limits while ensuring energy balance through Kirchhoff's current laws [39]. Power flows are computed using physical laws implemented in the backend solver and cannot be directly controlled: they result from the network topology and power balance.

2.5.2 Temporal Dynamics and Time Series Structure

Grid2Op operates on discretized time steps, which happen in 5 minute intervals. This granularity aligns with practical power system operations where major control decisions are made in the order of minutes rather than seconds. The framework focuses on control actions that could realistically be performed by human operators rather than automatic controls. The time-series data driving the environment are referred to as chronics:

Definition 2.2 (Chronics). Grid2Op time series data, called "chronics", are functions $\mathbf{c} : \mathbb{N} \rightarrow \mathbb{R}^d$ that define the evolution of d variables over discrete time steps. The dimension d depends on the environment and may include, for example, loads, generations, and potentially other variables such as renewable injections, storage setpoints, or line statuses.

Grid2Op models real-time power grid behaviour by assuming that these time series values are known to the environment but hidden from the agent, creating a partially observable setting where decisions must be made with partial information.

2.5.3 Core Components

The Grid2Op platform is structured around the standard RL [3, 40, 41] paradigm, which defines the interaction between an agent and the power grid environment through observations, actions, and rewards, as illustrated in Figure 2.2.

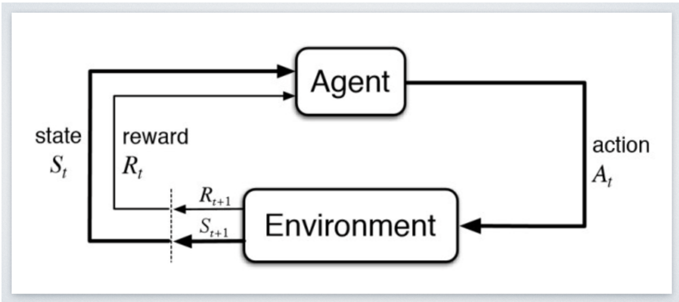

Figure 2.2: Schematic representation of Agent-Environment interaction in RL. Source: [42]. The agent acts on the environment, which generates a reward and a new state, which are fed back to the agent.

In the following subsections we present more information about these components.

2.5.3.1 Agents

In Grid2Op, an agent represents any entity that can take actions based on grid observations. The framework supports diverse agent implementations, from simple heuristic controllers to machine learning models.

Definition 2.3 (Grid2Op Agent)**.** A Grid2Op agent is a function $\pi : \mathcal{O} \times \mathbb{R} \times \{0,1\} \to \mathcal{A}$ that maps an observation $\mathbf{o}_t \in \mathcal{O}$, reward $R_t \in \mathbb{R}$, and termination flag $\text{done}_t \in \{0,1\}$ to an action $a_t \in \mathcal{A}$.

All agents must make control decisions that will affect future grid states without knowledge of how loads and generation will evolve.

2.5.3.2 Observation Space

The observation space defines the information available to agents at each time step. In Grid2Op, observations are high-dimensional vectors containing information about the current grid state.

A Grid2Op observation $\mathbf{o}_t$ at time step t contains several categories of information. The capacity of a powerline ρ_i, that is the current flow of line i divided by the thermal limit of line i. Line status information provides binary indicators $\mathbf{s} \in \{0,1\}^{N_{\text{line}}}$ of whether each transmission line is operational. Power flow data includes active and reactive power flows $(P_{\text{or}}, Q_{\text{or}}, P_{\text{ex}}, Q_{\text{ex}})$ at both ends of each line, while voltage information has voltage magnitudes $(V_{\text{or}}, V_{\text{ex}})$ at line endpoints and voltage setpoints at generators. Generator and load information includes production levels $(P_{\text{gen}}, Q_{\text{gen}})$ and consumption levels $(P_{\text{load}}, Q_{\text{load}})$ respectively. The topological vector $\mathbf{t} \in \{-1,1,2\}^{N_{\text{topo}}}$ indicates which bus each element is connected to (-1 meaning not connected, 1 meaning it is connected to bus 1, and 2 meaning it is connected to bus 2), and temporal features provide current time step, day of week, hour of day, and calendar information. For more detailed information, we refer the official Grid2Op documentation [35].

The dimensionality of the observation space is dependent on the specific environment. Each component (eg. nodes or edges) adds multiple features to the overall observation vector, creating high-dimensional state representations.

2.5.3.3 Action Space

The Grid2Op action space defines the control mechanisms available to the RL agents. The agents have available a set of actions that reflect real-world power system control and can modify multiple aspects of the grid simultaneously. For example, topology changes allow modifying a substation configuration by connecting powerlines to different buses, line status control enables connecting or disconnecting transmission lines, generator redispatching permits adjusting active power production, and renewable curtailment allows limiting renewable generation. Each action must satisfy operational constraints and respect equipment limitations such as cooldown periods (number of timesteps to wait before being able to change the topology of a substation again) and ramp rates (maximum values of the power that can vary between two consecutive time steps).

2.5.3.4 Reward Functions

Grid2Op environments include configurable reward functions that encourage grid stability. The standard reward function focuses on maintaining transmission capacity within safe limits, by using the capacity of each powerline. The ρ value (which represents line capacity) of a line i at time t is defined as:

$$\rho_{i,t} = \frac{|F_{i,t}|}{L_{i,t} + 10^{-1}} \quad , \tag{2.5}$$

where $|F_{i,t}|$ is the observed current flow, and $L_{i,t}$ represents the thermal limit of line i at time t.

Definition 2.4 (L2RPN Reward Function)**.** The L2RPN reward function is defined as:

$$R_t = \sum_{i=1}^{N} \max\left(1 - \min(\rho_{i,t}, 1)^2, 0\right) \tag{2.6}$$

where N is the number of transmission lines.

Although we are presenting this reward, due to the chosen agent for this thesis using this reward, we point out that multiple alternative rewards are available in Grid2Op.

2.5.4 Environment Setup

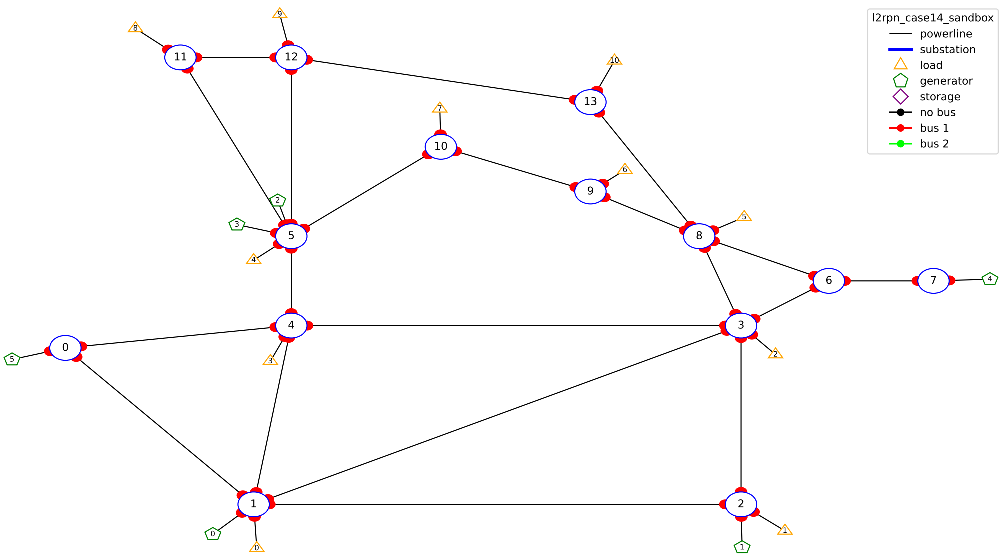

Figure 2.3: Layout of the l2rpn_case14_sandbox environment.

For this thesis, experiments are conducted in the `l2rpn_case14_sandbox` environment, based on the IEEE 14-bus system, slightly modified. This system is illustrated in Figure 2.3 and includes 14 electrical substations connected by 20 power lines, 6 power generation units and 11 electrical loads representing consumer demand. Although simplified, it preserves the essential complexity of real-world power systems and provides a tractable yet realistic power system complexity.

2.5.5 Environment Dynamics and Information Flow

The Grid2Op environment simulates power system physics through a backend that solves power flow equations. Each environment step $t \rightarrow t+1$ follows the sequence: first, the system is updated with new load and generation values from the time series; second, agent actions, for example topology changes or dispatch modifications are applied; third, the backend computes resulting flows and voltages using power flow; fourth, system performance is evaluated based on the reward function; and finally, updated system state information is provided to the agent through a new observation. This information flow is simplified, and many aditional checks are put in place to guarantee that no violations are made.

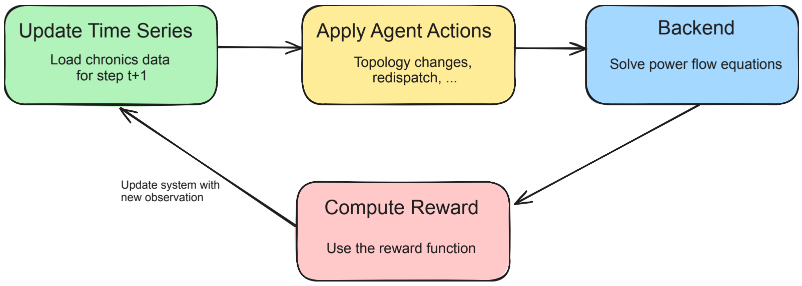

Figure 2.4: Environment Dynamics: A visual representation of the information flow for the interaction between the Agent and the Grid2Op environment.

2.5.6 Forecasting Challenge

One of the challenges in Grid2Op environments is predicting the operational conditions of the power network considering both agent actions and their impact on grid stability. Instead of focusing on predicting load or generation patterns, our objective is to forecast the risk of agent failure and the subsequent emergence of grid congestions in future time steps.

When RL agents make topology decisions, such as switching transmission lines or reconfiguring substations, the effects propagate throughout the network, affecting power flows and potentially creating or alleviating congestions somewhere else in the system. The problem that we want to solve is, given the current grid state and a proposed agent action, what is the probability that this decision will lead to operational failures or dangerous congestions in the near future.

When forecasting models are integrated into Grid2Op to predict operational risks, their predictions will for sure contain errors due to the complexity of power systems, the unpredictable effects of topology changes (that can have effects on the whole network), and generation and renewable energy control. Traditional point forecasts provide no indication of their reliability when predicting whether an agent's proposed action will maintain grid stability or trigger congestions.

This creates a need for model evaluation methods that can quantify prediction uncertainty around operational risk forecasts and provide reliability guarantees about agent performance predictions. We need to understand when and how agents are likely to fail, and with what confidence level, so that it is possible to develop reliable prediction systems that can anticipate grid instabilities before they occur.

2.5.7 Platform for Forecasting Research

Grid2Op's architecture makes it particularly well-suited for forecasting and uncertainty quantification research, because the platform provides extensive historical data for robust statistical analysis, modular architecture that supports external forecasting systems, and realistic complexity. The reproducible environments enable fair comparison across different forecasting approaches and uncertainty quantification methods. Furthermore, the availability of realistic power system constraints creates motivation for forecasting methods that can provide reliability guarantees and uncertainty quantification, precisely the capabilities that conformal prediction may offer for enhancing power grid operational decision-making.

2.5.8 Pre-trained Models

For this thesis, we utilise two pre-trained models that are essential components of our experimental framework. The *CurriculumAgent* serves as our reinforcement learning agent whose operational behaviour and decision-making we want to predict and analyze, while the *Histogram-based Gradient Boosting Regressor (HBGB)* forecaster provides the load and generation forecasts, as detailed in Subsubsections 2.5.8.1 and 2.5.8.2. While our methodology is demonstrated using these specific pre-trained models, the conformal prediction framework we develop is generalizable and can be applied to any RL agent/forecasting model pair operating within power grid management scenarios.

2.5.8.1 The CurriculumAgent

In the Grid2Op project a CurriculumAgent [3], which is an agent based on RL, was developed specifically for power grid management in order to work within the simulation environment.

It operates by making topology decisions within the Grid2Op environment, including actions such as modifying substation configurations and connecting or disconnecting transmission lines. The agent learns to maintain power system stability by keeping the lines within thermal limits while ensuring supply-demand balance through continuous interaction with realistic power grid simulations.

The training of this agent also occurred in the L2RPN Case14 Sandbox environment (the same we use in this thesis), using the first 900 episodes (chronics) data.

2.5.8.2 The Forecaster - HistGradientBoosting Regressor

The HBGB [4] is the forecasting model for predicting loads and active power generation in this thesis.

The forecaster uses time-series data that includes temporal variables such as hour and day of week, as well as historical load and generation values to generate load and forecast predictions at multiple time horizons, typically ranging from 5 minutes to an hour ahead.

Within the broader conformal prediction framework we will introduce in the following sections, the forecaster serves as the base predictor we use to obtain forecasted ρ values influenced by the CurriculumAgent's actions, so we can evaluate the uncertainty using CP methods in order to provide rigorous coverage guarantees.

2.6. State of the Art in Conformal Prediction for Power Systems

The application of CP to power systems, has been gaininaing more and more attention. In this Section, we present some of the recent studies that apply CP methods to the energy sector.

2.6.1 Conformal Prediction for Electricity Price

One of the first applications of CP in energy systems was introduced by Kath and Ziel [43] who apply conformal prediction to electricity markets. The authors combine CP with several point forecasting models and compare the performance of CP with state-of-the-art electricity price forecasting models, such as quantile regression averaging, and show that CP yields reliable prediction intervals in short-term power markets.

O'Connor et al. [44] propose an ensemble approach that combines the efficiency of quantile regression models with the robust coverage properties of time series techiques, such as the EnbPI [30] method. Their study demonstrates that CP not only guarantees reliable coverage but also leads to improved financial returns in energy trading in both the Day-Ahead and Balancing Markets [44].

Another study by De la Torre et al. [45], applies four different forecast algorithms for electricity pricing, and posteriorly CP to obtain uncertainty quantification. They find that "The conformal prediction intervals provide a more comprehensive view of the possible future scenarios, enhancing economic efficiency, risk management, and decision-making processes" [45].

2.6.2 Conformal Prediction for Renewable Energy

Recent work by Renkema et al. [24] creates a framework that uses CP to enhance participation in electricity markets. They apply several variants of CP to quantify the uncertainty of their point forecasts. They show that CP, particularly using K-Nearest Neighbours (k-NN) with Mondrian binning performs better than linear quantile regressors and conclude that the use of CP with one of their bidding strategies, "can yield up to 93% of the potential profit with minimal energy imbalance". Another study by Renkema et al. [25], also applies CP on top of point forecasting models, to predict PV power on a day-ahead basis. The framework is evaluated using large datasets of weather predictions and PV power output in the Netherlands. They conclude that CP with k-NN and Mondrian binning, using a Random Forests regressor as the point forecaster produces the best results.

A 2023 study by Wang et al. [46] developed a conformal asymmetric multi-quantile generative transformer prediction method for wind-power day-ahead prediction. They select two of the quantiles that satisfy nominal confidence as the upper and lower bounds of the interval, which can be naturally asymmetrical, and use the Conformalised Quantile Regression (CQR) method to calibrate the bounds of the prediction interval, to which they confirm obtaining valid empirical coverage.

2.6.3 Emerging Directions and Gaps

Taken together, these studies illustrate the versatility of CP in the energy domain, being used in electricity price forecasting, renewable generation prediction, and market decision support and consistently proving superior to other methods of uncertainty quantification.

To the best of our knowledge, the work in this thesis represents the first application of CP to power grid simulation environments, specifically for uncertainty quantification of the ρ parameter, an observation metric that directly impacts grid robustness and resilience in the Grid2Op digital platform. While previous studies have successfully applied CP to various power system forecasting tasks, including electricity price prediction, renewable energy generation forecasting, and load forecasting, our approach focuses on a fundamentally different and critical aspect of grid operations:

We use a pre-trained forecaster (HBGB) that predicts load and generation values of future grid states. We run power-flow calculations using the Grid2Op backend, which provides the value of ρ, given those forecasts, considering the action that the CurriculumAgent would take given the forecasted grid state. This means that we are not using CP directly on top of a point forecaster. We apply CP to the obtained ρ parameter, with the intuition of understanding if, given the current grid state, and current forecasts of the near-future time steps, the environment will be at risk. High values of ρ on transmission lines can lead to line disconnections and potentially result in system-wide blackouts. The goal of our methodology is to provide a framework for quantifying the uncertainty and reliability of this parameter, in order to enable better assessment of grid vulnerability, which can lead to more informed decision-making regarding preventive actions. This represents the first study to examine power grid robustness and resilience through the lens of CP in simulation environments.

3. Methodology

The developed framework consists of a modular Python-based system that couples a pre-trained forecaster (HistGradientBoostingRegressor, HBGB) with CP to produce prediction intervals for each transmission line capacity (ρ). The system inputs a forecaster (HBGB) and a control agent (CurriculumAgent), and outputs uncertainty-aware predictions that aim to maintain coverage under diverse operating conditions. Figure 3.1 summarises the pipeline.

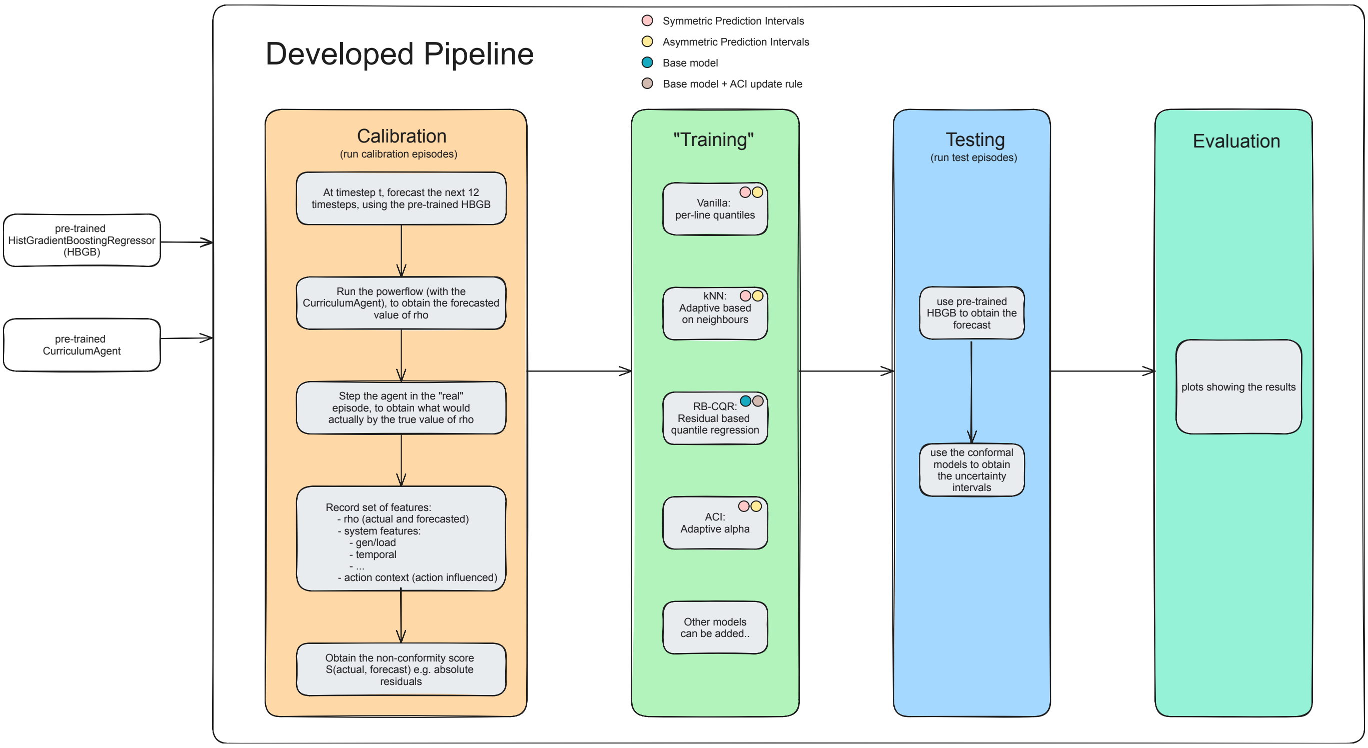

Figure 3.1: Framework Overview: The four phases implemented. In the calibration phase, for each time step, we run the powerflow considering the actions of the CurriculumAgent.

The framework operates through four interconnected phases. The *calibration phase* collects operational data from simulation episodes that are independent from the forecaster's training data. The *model training phase* fits CP models using this calibration data, and support both single model mode and an ensemble approach. The *testing phase* rolls forecasts in real time operational scenarios, while the *evaluation phase* measures coverage and interval width, globally and under action-influenced periods, and visualizes results.

Action influenced periods are also considered because they can impact forecast accuracy. As mentioned in Subsection 2.6.3, instead of assuming static forecasting conditions, our methodology includes RL agent actions which continuously affect the power system and, therefore, prediction accuracy, to enable more realistic uncertainty quantification.

3.1. Calibration Phase

The calibration phase forms the foundation of the conformal prediction framework, and must ensure data independence from the forecaster's training data. To achieve this, we start calibration with chronic 901, since the HBGB's training ends at chronic 900. This separation is essential for maintaining the theoretical foundations of CP, as overlapping data could compromise coverage guarantees and produce optimistic interval estimates.

Each calibration episode uses a rolling forecast evaluation process that generates predictions for twelve future timesteps at five-minute intervals, corresponding to one hour of operational forecasting. At each timestep t, the forecaster generates predictions for the following 12 timesteps $(t+1, \ldots, t+12)$, after which the simulation advances step-by-step to collect the actual ρ values, as explained in Figure 3.2. This way, forecasted data is collected for 12 horizons, and only after the actual data is introduced, which results in more realistic uncertainty quantification. This *multi-horizon approach* reveals how uncertainty propagates over time by allowing the analysis of horizon-specific coverage.

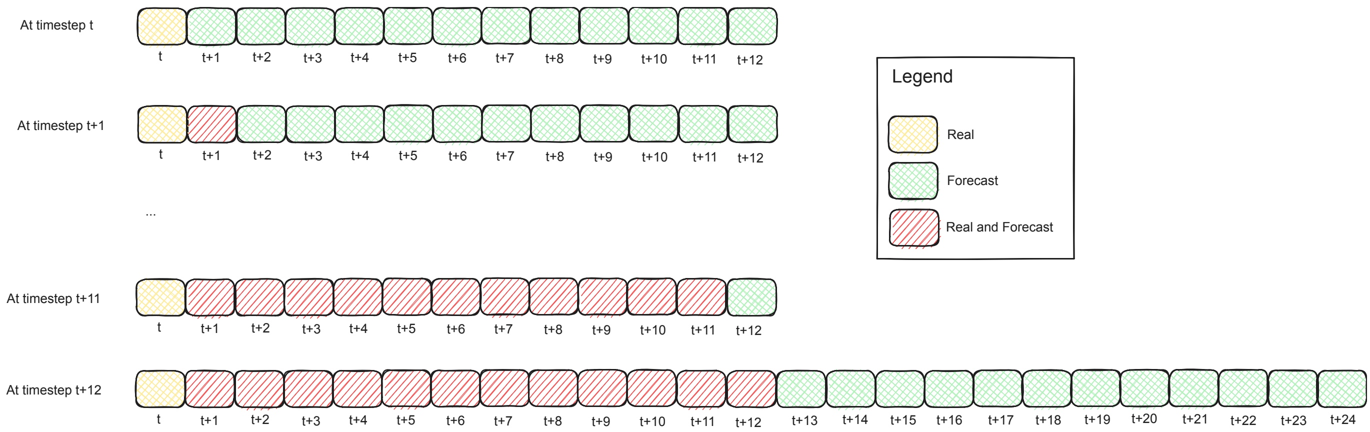

Figure 3.2: Rolling forecast process. At each time step t, the forecaster predicts the next 12 steps; relized outcomes are recorded as the system advances.

Beyond simple prediction/reality ρ pairs, the calibration process records extensive operational data. In each timestep, we also collect the forecasted values that incorporate agent control actions, complete system observations including loads and generation, agent action timestemps, and contextual information such as forecast horizons and periods influenced by recent actions. This data can then be used to fit conformal predictors that can use it as features, in order to adapt to various operational contexts and system states.

The multi-step forecasting process begins with *feature engineering*. The forecaster uses temporal patterns (hourly, daily, seasonal encodings), historical load/generation (from one hour, one day and one week earlier), and the current system state (flows, dispatch, load distribution and recent control actions).

As aforementioned, the forecasting system operates through an *indirect prediction mechanism*. Rather than directly predicting line capacity values, the forecaster generates predictions for active and reactive power at loads and generators, which are then converted to line capacity (our target) through power flow simulation. This approach also integrates control actions that may change for example the system topology or generation dispatch. It also marks a departure from the classical idea of CP, in which the forecaster is trained to predict the target variable. At each forecast timestep, the system configures forecasted power injections that include both predicted load and generation values and the anticipated effects of agent control actions. The agent then acts upon this forecasted system state, and the resulting modified state provides the final prediction that accounts for the dynamic interaction between forecasting and control. This process is depicted in Figure 3.3.

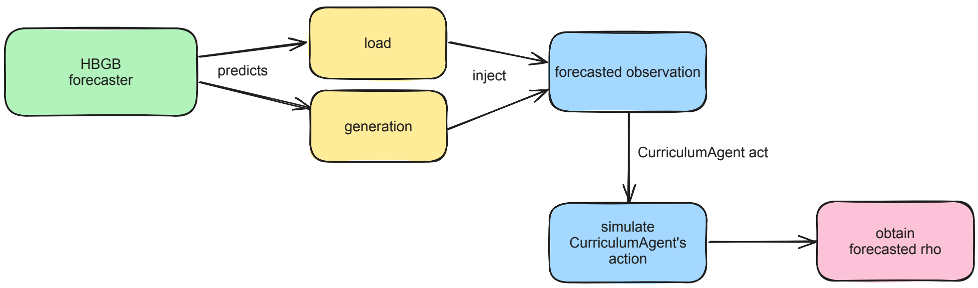

Figure 3.3: Method used to obtain the ρ value: The HBGB forecaster predicts load and generation values, and these forecasted values are injected into an observation in Grid2Op. The CurriculumAgent chooses an action on that observation and we simulate this action to obtain the forecasted value of ρ.

3.2. Training Phase

The training phase fits CP models using the data collected in the calibration phase, supporting both individual line-specific models and ensemble approaches.

3.2.1 Single Model Mode

A single conformal predictor is trained using calibration data collected under two different operational conditions:

Under normal operation, the single model mode collects calibration data from episodes without external disturbances or line attacks. The resulting conformal predictor learns to quantify uncertainty under standard operational variations, including natural load fluctuations and generation dispatch changes. This baseline configuration establishes how our implemented CP methods perform under typical power system operations. We call this scenario *Case Study 1*.

When considering scenarios with line vulnerabilities, the single model mode can be configured so that the grid suffers adversarial attacks. In this configuration, specific power lines are subjected to attacks during calibration episodes, and the same attack pattern is maintained during testing. The conformal predictor tries to learn to account for the uncertainty introduced by these attacks, providing prediction intervals that are calibrated for the particular vulnerability scenario under consideration. This mode where some lines are attacked, could be reflected in the real world, in which some particular lines are more prone to suffer adversarial attacks [47] or are more susceptible to weather conditions. We call this scenario *Case Study 2*.

3.2.2 Ensemble Model Mode

Instead of training a single model where some lines are attacked, this mode constructs an ensemble of twenty specialized conformal predictors, each trained to handle uncertainty quantification when a specific power line experiences stress or attack. In our Grid2Op setup, the ensemble contains 20 models, one per transmission line. If applied to a different environment with another forecaster, the framework automatically adapts to the number of lines in that environment. During prediction, the ensemble mode aggregates results from all twenty models to produce the prediction intervals. It uses worst-case aggregation where the prediction intervals are constructed by taking the minimum lower bounds and maximum upper bounds across all line models, which ensures conservative coverage that accounts for uncertainty under any single-line attack scenario.

Each ensemble component focuses on a particular line vulnerability scenario. Model 0 trains exclusively on episodes where line 0 is subjected to attack, Model 1 specializes in scenarios where line 1 is attacked, and this pattern continues across all twenty power lines in the system. The ensemble training process creates a group of models that understand how different line attacks affect system uncertainty. For example, when line 3 is attacked the corresponding ensemble component (model where only line 3 is attacked) learns how this specific stress propagates throughout the network, affecting not only line

3's capacity but also the other lines through power flow. As we will see in further sections, this robustness comes at the cost of the loss of efficiency. We call this scenario *Case Study 3*.

3.3. Testing and Evaluation Framework

The testing phase simulates realistic power system operations through continuous forecasting and decision-making cycles. During each testing episode, the framework keeps track of the rolling forecast horizons, generates prediction intervals for the current forecasts using the calibrated conformal models, and evaluates coverage performance as actual system outcomes (the real observations, with the real value of ρ) become available.

The evaluation methodology uses multiple performance metrics that characterize conformal prediction effectiveness: coverage metrics assess whether actual outcomes fall within predicted intervals at the specified confidence level, measured globally across all predictions and by forecast horizon, for every transmission line; efficiency metrics measure the global average width of the intervals, and the variation of average width by forecast horizon; action-influenced time steps, which comprise the short-term time-steps after the Agent takes an action, are also recorded and presented using the same method, because in action-influenced periods, the system behaviour may differ significantly from normal operations due to the recent control actions. This is because most of the time, the agent chooses to do nothing, which we interpret as a "Do-Nothing" action. Every time that a non-empty action is performed, we record the next 12 time steps (the next hour), as action-influenced. We separately evaluate coverage performance and efficiency during these periods, an indicator of how well the CP agents behave in periods where the agent was forced to act.

The framework generates multi-model comparison visualisations (if more than one CP model is selected), horizon-based performance analysis, and individual lines plots. These visualisations highlight action-influenced periods in yellow, the hour the follows after an Agent's action.

3.4. Implemented CP Methods

This section describes the (adapted) CP methods implemented and evaluated within the framework. Each method represents a different approach to uncertainty quantification, ranging from classical split conformal prediction to more advanced adaptive techniques, such as the ACI method, that are more adequate for time-series. All implementations follow the standardized interface described in the modular architecture (Section 4.1).

3.4.1 Vanilla Conformal Prediction

Vanilla conformal prediction is the baseline within the split conformal prediction paradigm [11, 12, 48]. This method provides the simplest implementation of conformal prediction by computing nonconformity scores directly from prediction residuals on a held-out calibration set. The method provides exact finite-sample coverage guarantees when calibration and test data are exchangeable. Although our data are time-series and thus violate this assumption, we evaluate Vanilla CP empirically as a benchmark for more advanced methods. As we explained in Section 3.1, we make sure that the calibration data is independent from training data, maintaining the held-out requirement. We implement two variants as detailed next. An overview is presented in Figure 3.4.

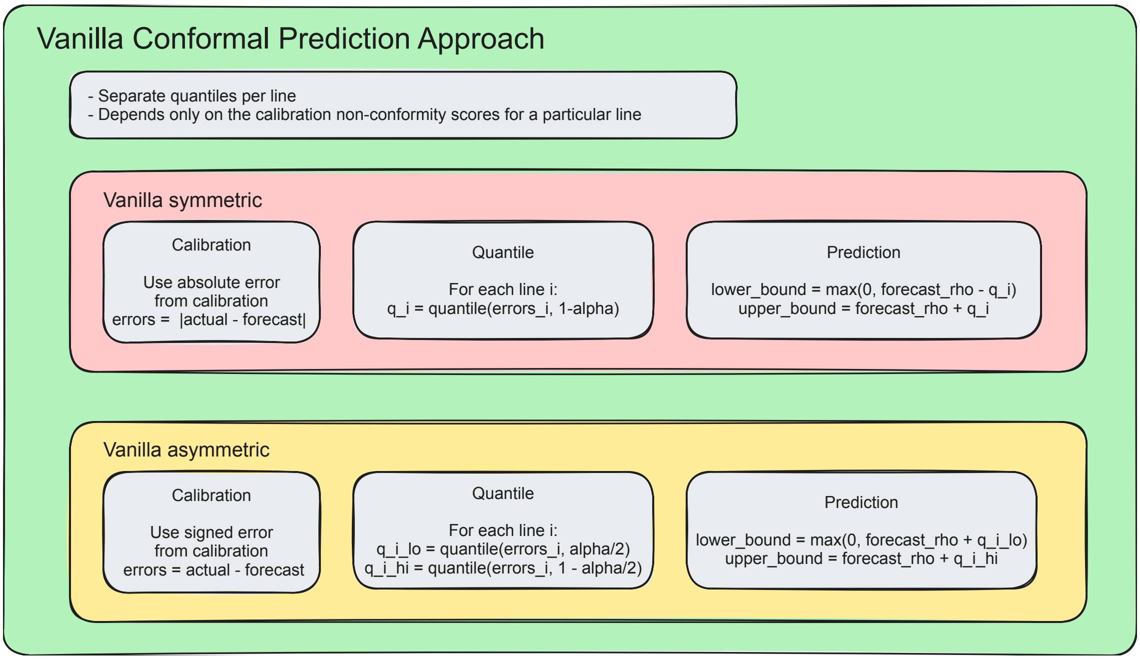

Figure 3.4: Vanilla CP symmetric and asymmetric methods: The symmetric variant uses the absolute error for each point in the calibration dataset, and then calculates the quantile of the absolute errors, posteriorly subtracting and adding the quantiles to obtain the intervals. The asymmetric variant uses the signed error instead, calculates a lower and upper quantile, and adds these quantiles to obtain the intervals.

3.4.1.1 Vanilla Symmetric

The vanilla symmetric implementation uses absolute residuals of the calibration set to calculate the threshold (quantile). For each line i, we compute:

$$\hat{q}_{i,1-\alpha} = quantile(\underbrace{\cup_{t\in\mathcal{T}_{cal}}\{|\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)|\}}_{\text{abs residuals in calibration set}}, 1-\alpha) \tag{3.1}$$

The prediction interval is then $\hat{C}_{i,1-\alpha}(\mathbf{x}) = [\hat{\rho}_i(\mathbf{x}) - \hat{q}_{i,1-\alpha}; \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha}]$, ensuring that each power line receives appropriately calibrated prediction intervals based on the non-conformity scores of the calibration set. Since a global quantile of all the non-conformity scores is used, the Vanilla method produces, for every episode, intervals of the same width in each line, in which the forecasted value of ρ lies in the middle of the interval.

3.4.1.2 Vanilla Asymmetric

The vanilla asymmetric variant extends the symmetric approach by using signed residuals rather than the absolute non-conformity scores, allowing for asymmetric prediction intervals that can better capture directional prediction biases. This implementation computes separate quantiles for upper and lower bounds:

$$\hat{q}_{i,\alpha/2} = quantile(\cup_{t\in\mathcal{T}_{cal}}\{\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)\}, \alpha/2) \tag{3.2}$$

$$\hat{q}_{i,1-\alpha/2} = quantile(\cup_{t\in\mathcal{T}_{cal}}\{\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)\}, 1-\alpha/2) \tag{3.3}$$

The prediction interval is then $\hat{C}_{i,1-\alpha}(\mathbf{x}) = [\hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,\alpha/2}; \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha/2}]$

When the distribuition is asymmetric, this variant can provide improved efficiency. Although the intervals also have fixed width for each line, the point forecast no longer stays in the exact middle of the interval, and the method responds better when over (or under) prediction patterns exist.

3.4.2 K-Nearest Neighbours Conformal Prediction

k-NN conformal prediction leverages the local nature of nearest neighbor methods to create context-aware nonconformity measures that reflect local prediction difficulty [49]. Unlike global conformal methods that use the entire calibration dataset, k-NN-based approaches focus on the k most similar historical observations, with the objective of forming prediction intervals that are appropriate to the local operating conditions. The key idea is that prediction uncertainty should be informed by how well the model has performed on similar past observations rather than by overall calibration performance. By identifying similar situations in the calibration data using feature similarity, k-NN-based conformal predictors are particularly valuable in power system applications where prediction difficulty varies significantly across different operational regimes, load patterns, and system configurations.

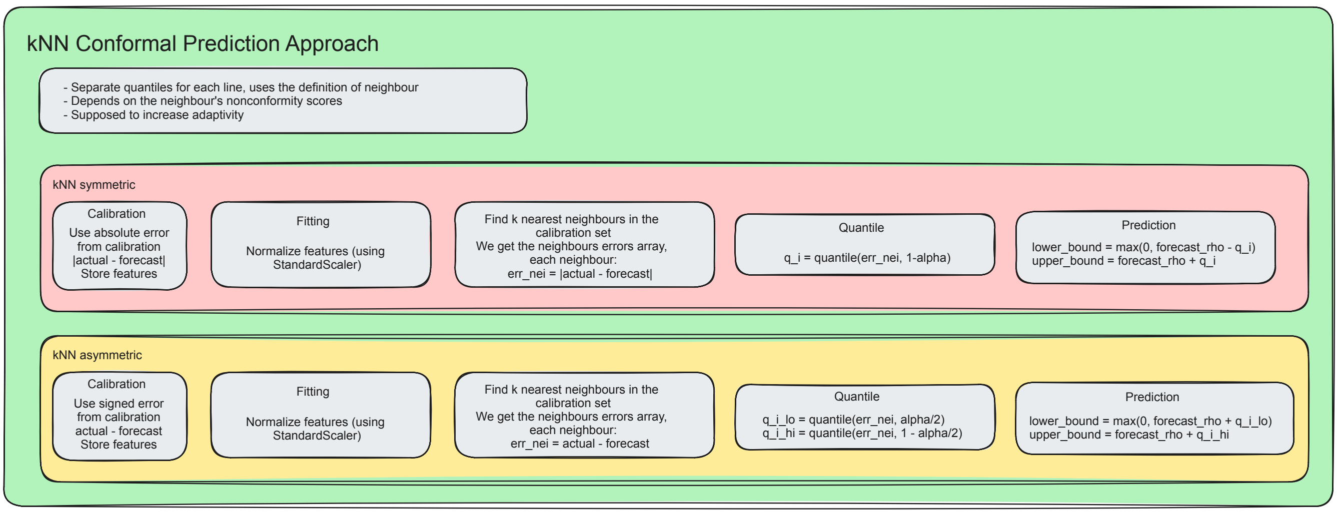

Figure 3.5: kNN-based methods: As in the Vanilla method, the symmetric variant uses absolute errors and the asymmetric variant uses signed errors. The quantiles are calculated for each test point, considering only the portion of the calibration set that corresponds to their neighbours.

3.4.2.1 k-NN Symmetric

The symmetric k-NN implementation computes nonconformity scores using the distribution of absolute errors among the k nearest neighbors in feature space. The prediction intervals are based on the empirical error distribution within this local neighborhood:

$$\hat{q}_{i,1-\alpha}(\mathbf{x}) = \text{quantile}\Big(\underbrace{\cup_{t\in\mathcal{N}_k(\mathbf{x}|\mathcal{T}_{cal})}|\rho_{i,t}-\hat{\rho}_i(\mathbf{x}_t)|}_{\substack{\text{set of k closest neighbours}\\ \text{in calibration set}}}, 1-\alpha\Big) \tag{3.4}$$

The prediction interval is then $\hat{C}_{i,1-\alpha}(\mathbf{x}) = \left[\hat{\rho}_i(\mathbf{x}) - \hat{q}_{i,1-\alpha}(\mathbf{x}); \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha}(\mathbf{x})\right]$. This approach adapts to local prediction difficulty, producing tighter intervals in well-modeled regions and wider intervals where prediction is more challenging.

3.4.2.2 k-NN Asymmetric

The asymmetric k-NN variant uses signed errors from the k nearest neighbors and calculates separate quantiles for upper and lower bounds, which enables the method to capture local bias patterns and adjust interval asymmetry, acording to these patterns.

$$\hat{q}_{i,\alpha/2}(\mathbf{x}) = \text{quantile}\Big(\underbrace{\cup_{t\in\mathcal{N}_k(\mathbf{x}|\mathcal{T}_{cal})}(\rho_{i,t}-\hat{\rho}_i(\mathbf{x}_t))}_{\substack{\text{set of k closest neighbours}\\ \text{in calibration set}}}, \alpha/2\Big) \tag{3.5}$$

$$\hat{q}_{i,1-\alpha/2}(\mathbf{x}) = \text{quantile}\big(\underbrace{\cup_{t\in\mathcal{N}_k(\mathbf{x}|\mathcal{T}_{cal})}}_{\substack{\text{set of k closest neighbours}\\ \text{in calibration set}}}(\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)), 1-\alpha/2\big) \tag{3.6}$$

The prediction interval is then $\hat{C}_{i,1-\alpha}(\mathbf{x}) = \left[\hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,\alpha/2}(\mathbf{x}); \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha/2}(\mathbf{x})\right]$.

3.4.3 k-NN Normalised Conformal Prediction

The normalised k-NN variants implement the normalisation approach used in Article [24], which scales nonconformity scores by a difficulty estimate: the average value of the residuals of the k most similar neighbours of the calibration dataset. This approach is different from the first one we presented: Instead of finding the most similar neighbours at test time, and using the quantile of those neighbours, this approach relies on the difficulty estimate. Every point in the calibration set is normalised by this difficulty estimate. At test time, the difficulty is calculated for each individual test point, and the whole normalised calibration set is used, which is then denormalised using the difficulty estimate.

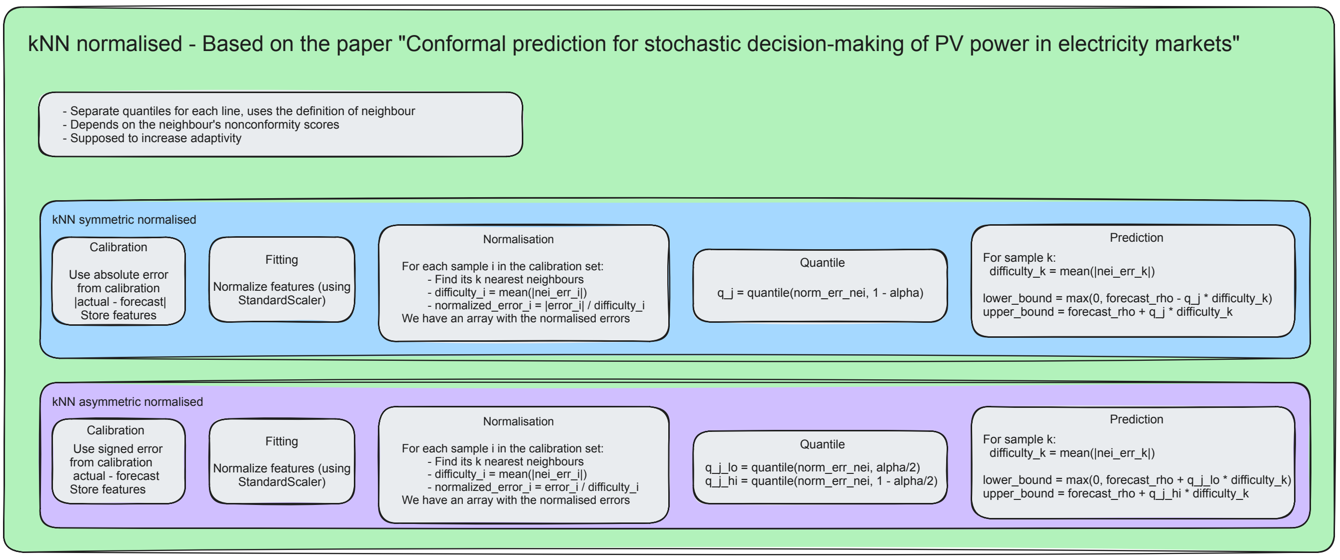

Figure 3.6: k-NN normalised methods: For each data point in the calibration dataset, a difficulty measure is calculated and used to construct a normalised calibration set. This normalisation happens by dividing each non-conformity score by the difficulty measure of each point in question. At test time, the difficulty measure of the test point is calculated, and then multiplied by the quantile of the whole calibration dataset, which makes this approach different from the one in Figure 3.5

The normalisation procedure estimates prediction difficulty using the mean absolute error of nearest neighbors:

$$\text{difficulty}_i(\mathbf{x}) = \max\big(\text{mean}\big(\underbrace{\cup_{t\in\mathcal{N}_k(\mathbf{x}|\mathcal{T}_{cal})}}_{\substack{\text{set of k closest neighbours}\\ \text{in calibration set}}}|\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)|\big), \epsilon\big) \tag{3.7}$$

The conformity scores are then normalised by these difficulty estimates:

$$\text{normalised\_score}_{i,t}(\mathbf{x}_t) = \frac{|\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)|}{\text{difficulty}_i(\mathbf{x}_t)} \tag{3.8}$$

This enhances interval efficiency by accounting for varying levels of local prediction uncertainty. Both symmetric and asymmetric normalised variants are implemented using this idea, with the asymmetric version using signed residuals in the normalisation process.

At test time, the same procedure is followed as in the Vanilla CP, using α and the normalised conformity scores, and calculating the global quantile (but this time of the normalised scores).

$$\hat{q}^{(norm)}_{i,1-\alpha} = quantile(\underbrace{\cup_{t\in\mathcal{T}_{cal}}\{\text{normalised\_score}_{i,t}(\mathbf{x}_t)\}}_{\text{normalised residuals}}, 1-\alpha) \tag{3.9}$$

Then we do the inverse: for each new test point $\mathbf{x}$, we calculate its difficulty, and construct the interval:

$$\hat{C}_{i,1-\alpha}(\mathbf{x}) = \left[\hat{\rho}_i(\mathbf{x}) - \hat{q}^{(norm)}_{i,1-\alpha} \times \text{difficulty}_i(\mathbf{x}); \hat{\rho}_i(\mathbf{x}) + \hat{q}^{(norm)}_{i,1-\alpha/2} \times \text{difficulty}_i(\mathbf{x})\right]$$

The authors in Article [24] use this method to increase adaptivity in order to approximate conditional coverage, to enhance the reliability of probabilistic day-ahead PV power forecasting. This approach, although it also uses k-NN, is fundamentally different from the k-NN approach presented in Subsubsections 3.4.2.1 and 3.4.2.2: notice that as in the Vanilla method, the quantile calculation does not depend on the specific datapoint, but rather is scaled by the difficulty measure.

3.4.4 Adaptive Conformal Inference

Adaptive Conformal Inference (ACI) addresses one limitation of the vanilla method: the assumption of data exchangeability, which often fails in real-world applications where the data generating distribution changes over time. ACI methods were originally designed for online settings where the underlying distribution is allowed to vary over time [31], making static (non-adaptive) calibration inadequate. The key insight of ACI is to make the significance level α adaptive. Rather than using a fixed α, ACI continuously updates it using a gradient descent-like procedure in which: i) if intervals are too narrow (miss too often) α decreases, producing wider intervals; and ii) if intervals are too wide (coverage above target) α increases, shrinking intervals. This adaptation mechanism enables the method to maintain long-term coverage guarantees regardless of the true data generating process, making it particularly suitable for power system applications where operational conditions evolve continuously. We implement two variants as explained next.

3.4.4.1 ACI Symmetric

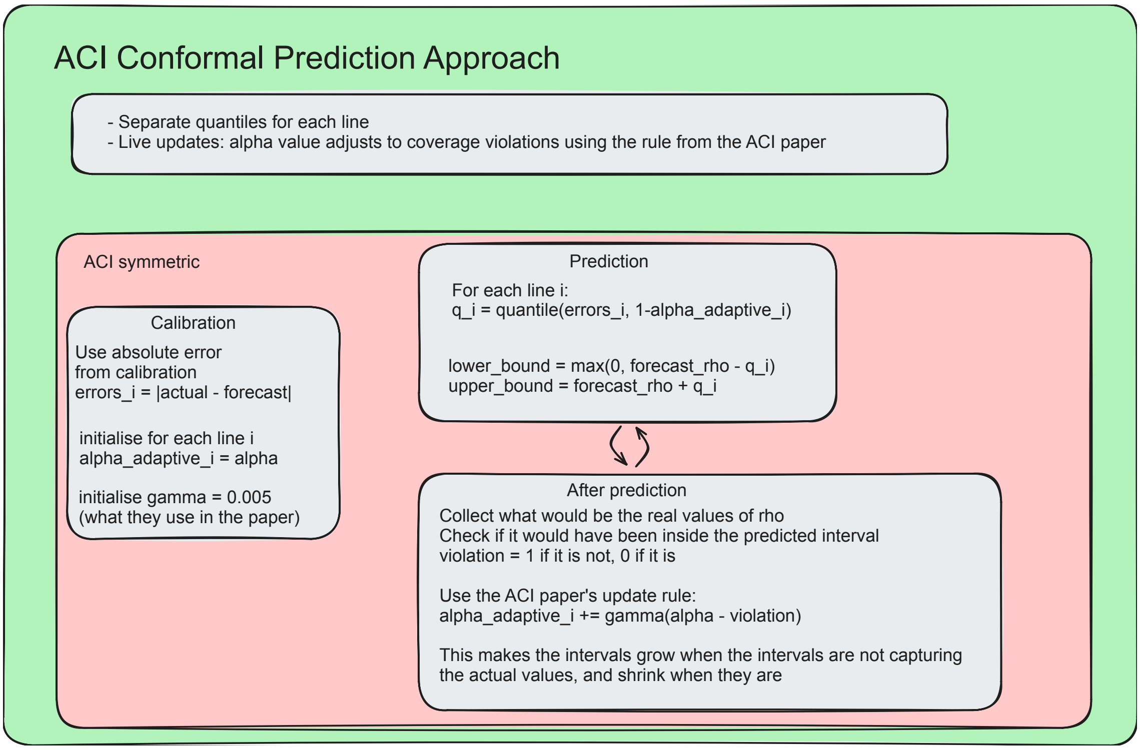

Figure 3.7: Symmetric Adaptive Conformal Inference Adaptation: This adaptation of the ACI method uses the same calibration set (stored during calibration phase), but instead of relying on a single quantile like the Vanilla method, continuously calculates an appropriate quantile with an adaptive alpha at each test step, which is updated using the update rule based on recent performance.

The symmetric ACI implementation adapts the significance level per power line, updating these levels based on recent coverage on that line. For line i, at timestep t_s, the update rule follows:

$$\alpha_{i,t_s} = \alpha_{i,t_s-1} + \gamma\left(\alpha - \mathbf{1}\left(\rho_{t_s-1} \notin \hat{C}_{i,t_s-1,1-\alpha_{t_s-1}}(\mathbf{x}_{t_s-1})\right)\right) \tag{3.10}$$

where γ is the step size parameter. ACI starts with $\alpha_1 = \alpha$. The method computes quantiles using the current adaptive significance level, enabling dynamic adjustment to changing operational conditions while preserving theoretical coverage guarantees.

$$\hat{q}_{i,1-\alpha_{i,t_s}} = quantile(\underbrace{\cup_{t\in\mathcal{T}_{cal}}\{|\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)|\}}_{\text{abs residuals in calibration set}}, 1-\alpha_{i,t_s}) \tag{3.11}$$

The prediction interval is then $\hat{C}_{i,t_s,1-\alpha}(\mathbf{x}) = [\hat{\rho}_i(\mathbf{x}) - \hat{q}_{i,1-\alpha_{i,t_s}}; \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha_{i,t_s}}]$.

The γ value is of particular importance, since its choice gives a tradeoff between adaptability and stability. Raising the value of γ makes the method more adaptive to distribution shifts, but it induces greater volatility in the value of α_t. In practise, large fluctations in α_t are undesirable, because they allow the method to oscillate between outputing small conservative and large anti-conservative prediction sets [31].

3.4.4.2 ACI Asymmetric

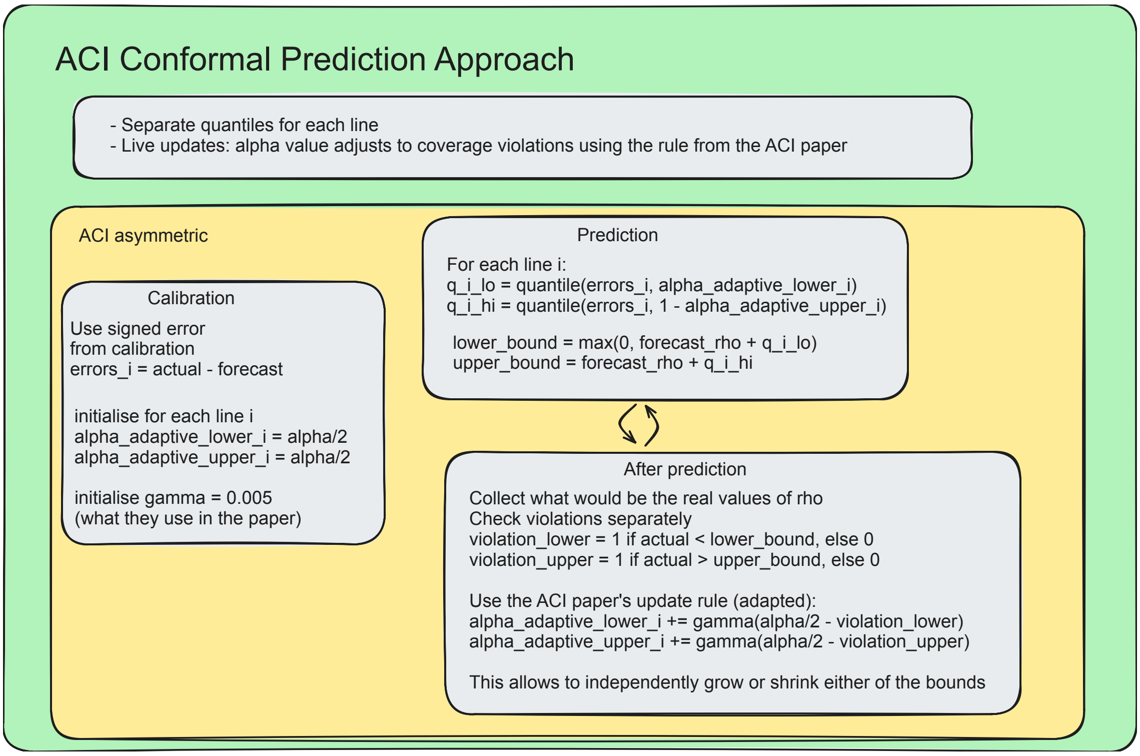

Figure 3.8: Asymmetric Adaptive Conformal Inference Adaptation: The asymmetric variant uses signed errors, and maintains two adaptive α values to control the lower and upper violations independently.

The asymmetric ACI variant maintains separate adaptive parameters for upper and lower bounds, enabling independent adjustment of interval asymmetry based on the directional recent coverage performance, and starts with $\alpha^{(\text{lower})}_{i,t_s} = \alpha^{(\text{upper})}_{i,t_s} = \alpha$. The implementation tracks violations separately for each bound on line i at time step t_s:

$$\alpha^{(\text{lower})}_{i,t_s} = \alpha^{(\text{lower})}_{i,t_s-1} + \gamma \left(\alpha/2 - \mathbf{1}\left(\rho_{t_s-1} < \inf \hat{C}_{i,t_s-1,1-\alpha_{t_s-1}}(\mathbf{x}_{t_s-1})\right)\right) \tag{3.12}$$

$$\alpha^{(\text{upper})}_{i,t_s} = \alpha^{(\text{upper})}_{i,t_s-1} + \gamma \left(\alpha/2 - \mathbf{1}\left(\rho_{t_s-1} > \sup \hat{C}_{i,t_s-1,1-\alpha_{t_s-1}}(\mathbf{x}_{t_s-1})\right)\right) \tag{3.13}$$

where $\inf$ and $\sup$ represent the inferior and superior limits of the prediction interval, respectively.
The quantiles are calculated during prediction at each time step t_s:

$$\hat{q}_{i,\alpha_{i,t_s}^{\text{(lower)}}} = quantile(\cup_{t\in\mathcal{T}_{cal}}\{\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)\}, \alpha_{i,t_s}^{\text{(lower)}}) \tag{3.14}$$

$$\hat{q}_{i,1-\alpha_{i,t_s}^{\text{(upper)}}} = quantile(\cup_{t\in\mathcal{T}_{cal}}\{\rho_{i,t} - \hat{\rho}_i(\mathbf{x}_t)\}, 1-\alpha_{i,t_s}^{\text{(upper)}}) \tag{3.15}$$

Then the prediction interval is $\hat{C}_{i,t_s,1-\alpha}(\mathbf{x}) = \left[\hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,\alpha_{i,t_s}^{\text{(lower)}}}; \hat{\rho}_i(\mathbf{x}) + \hat{q}_{i,1-\alpha_{i,t_s}^{\text{(upper)}}}\right]$. This approach can provide better adaptivity to directional distribution shifts and asymmetric prediction error patterns, by having the ability of only increasing or reducing one side of the intervals in the case that the distribuition of non-conformity scores is asymmetric.

3.4.5 Conformalised Quantile Regression

Conformalised Quantile Regression (CQR) addresses another limitation of the Vanilla conformal method: the production of prediction intervals with constant width across the input space [22], regardless of local prediction difficulty. This leads to overly conservative intervals in well-modeled regions and insufficiently wide intervals in challenging prediction regions. CQR resolves this issue through a two-stage process that combines quantile regression's adaptivity with CP's coverage guarantees.

First, two conditional quantile functions, $\hat{f}_{\alpha/2}$ and $\hat{f}_{1-\alpha/2}$ are fit using a quantile regression algorithm. Then the conformity scores are calculated for the calibration dataset, for the error made by the interval $\hat{C}(\mathbf{x}) = [\hat{f}_{\alpha/2}(\mathbf{x}); \hat{f}_{1-\alpha/2}(\mathbf{x})]$. The scores are calculated as conformity$(\mathbf{x}) = \max(\hat{f}_{\alpha/2}(\mathbf{x}) - \rho, \rho - \hat{f}_{1-\alpha/2}(\mathbf{x}))$ in which ρ represents the actual value for point $\mathbf{x}$, for each $\mathbf{x}$ in the calibration dataset.

The quantile of the conformity scores in the calibration set is calculated:

$$\hat{q}_{1-\alpha} = quantile(\cup_{t\in\mathcal{T}_{cal}}\{\max(\hat{f}_{\alpha/2}(\mathbf{x}_t) - \rho, \rho - \hat{f}_{1-\alpha/2}(\mathbf{x}_t))\}, 1-\alpha) \tag{3.16}$$

The interval constructed for a new test point $\mathbf{x}$ is then:

$$C(\mathbf{x}) = [\hat{f}_{\alpha/2}(\mathbf{x}) - \hat{q}_{1-\alpha}, \hat{f}_{1-\alpha/2}(\mathbf{x}) + \hat{q}_{1-\alpha}]$$

This approach inherits the quantile regression's ability to handle heteroskedasticity and therefore produce locally varying intervals, while CP provides distribution-free finite sample coverage guarantees. The result is more efficient (narrower) prediction intervals that maintain rigorous statistical validity.

3.4.6 Residual-Based Conformalised Quantile Regression

Since our framework depends on an external pre-trained forecaster, we cannot simply replace the forecaster with a quantile regression model. For this reason, we introduce a novel approach called Residual-Based Conformalised Quantile Regression (RB-CQR). The main difference between standard CQR and RB-CQR is that this method trains quantile regressors to estimate the conditional quantiles of prediction residuals $\rho - \hat{\rho}$, obtained using the pre-trained forecaster, as functions of system features, and then add the forecasted value of ρ. In essence, instead of having quantile regression model the distribution of the target variable directly, we model the distribution of the forecaster's prediction errors.

First, we divide the calibration dataset $\mathcal{T}_{cal}$, obtaining a divided calibration set (70/30), obtaining $\mathcal{T}_{cal_{70}}$ and $\mathcal{T}_{cal_{30}}$.

Then, we fit two conditional quantile functions $\hat{f}^{(residuals)}_{\alpha/2}$ and $\hat{f}^{(residuals)}_{1-\alpha/2}$ on the first 70% of the calibration set data, in order to predict the residuals.

Conformity scores are computed as:

$$\text{conformity}_i(\mathbf{x}) = \max\left((\hat{\rho} + \hat{f}^{(residuals)}_{\alpha/2}(\mathbf{x})) - \rho, \rho - (\hat{\rho} + \hat{f}^{(residuals)}_{1-\alpha/2}(\mathbf{x}))\right) \tag{3.17}$$

for every $\mathbf{x}$ on the remaining 30% of the calibration data ($\mathcal{T}_{cal_{30}}$).

The method then applies conformal calibration to ensure the coverage guarantees. We calculate:

$$\hat{q}_{i,1-\alpha} = quantile\left(\left\{\cup_{t\in\mathcal{T}_{cal_{30}}}\{\max\left((\hat{\rho} + \hat{f}^{(residuals)}_{\alpha/2}(\mathbf{x}_t)) - \rho, \rho - (\hat{\rho} + \hat{f}^{(residuals)}_{1-\alpha/2}(\mathbf{x}_t))\right)\}\right\}, 1-\alpha\right) \tag{3.18}$$

The resulting interval is then:

$$\hat{C}_{i,1-\alpha}(\mathbf{x}) = [\hat{\rho}_i + \hat{f}_{\alpha/2}(\mathbf{x}) - \hat{q}_{i,1-\alpha}, \hat{\rho}_i + \hat{f}_{1-\alpha/2}(\mathbf{x}) + \hat{q}_{i,1-\alpha}]$$

All RB-CQR implementations expand the features by appending line indices, so that the quantile models can learn line-specific uncertainty patterns for each of the power lines.

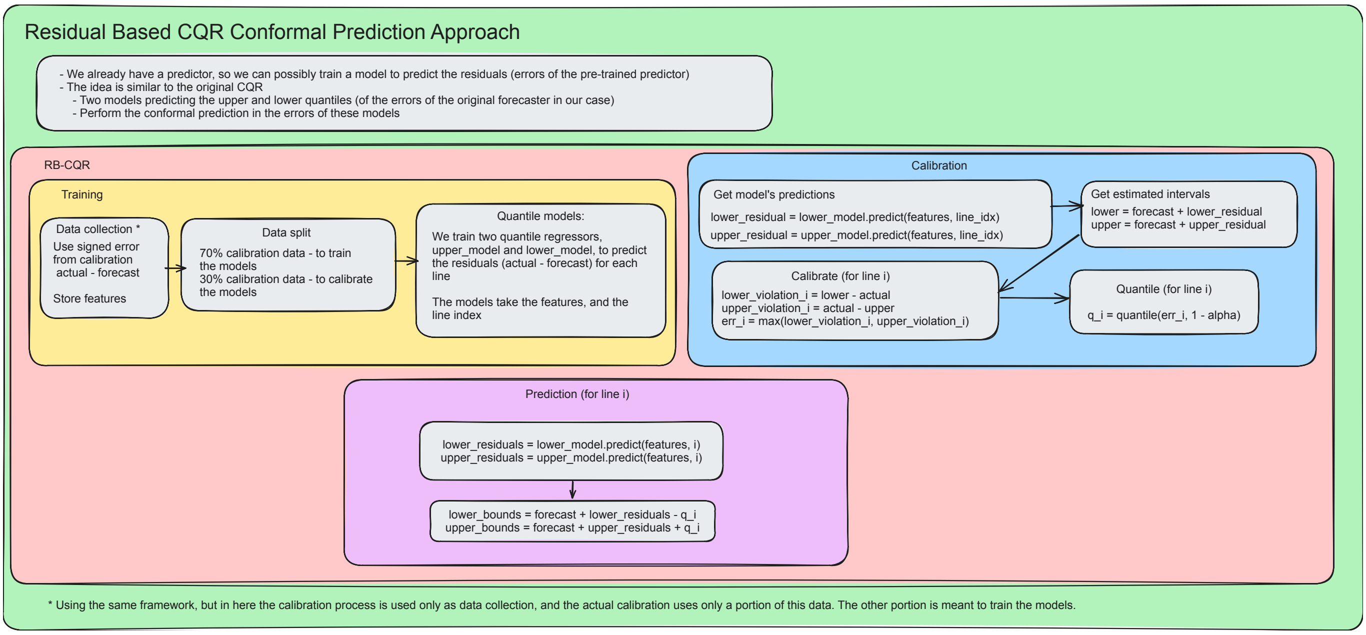

Figure 3.9: Residual-Based CQR approach: The RB-CQR method divides the calibration dataset in two parts, and uses the first part (70% of the data) to train two quantile regressors with the objective of predicting the distribuition of errors of the forecaster. The remaining of the data is used to calibrate the models.

3.4.6.1 Machine Learning Backend Implementations

The RB-CQR approach can be implemented using various machine learning backends for quantile regression, each offering different computational characteristics and modeling capabilities. In this thesis, we explored three:

LightGBM-based RB-CQR

The LightGBM [50] implementations utilise gradient boosting for quantile regression, training separate models for upper and lower quantiles of the residual distribution.

HistGradientBoosting-based RB-CQR

The HistGradientBoosting [4] variants use *scikit-learn*'s optimized histogram-based gradient boosting implementation, providing computational efficiency advantages for large datasets while maintaining CQR's theoretical properties.

Random Forest-based RB-CQR

The Random Forest [51] RB-CQR implementations utilise quantile forests for multi-quantile prediction, using the ensemble nature of random forests to provide robust quantile estimates.

3.4.7 The Inverted Method: An Accidental Discovery

During implementation, a sign error in the conformity score calculation led to the discovery of an alternative formulation that we term the "inverted method". The mistake occurred in the violation computation within the conformity score calculation.

The correct RB-CQR formulation computes conformity scores using:

$$\text{lower\_violation} = (\hat{\rho} + \hat{f}_{\alpha/2}^{(residuals)}) - \rho \tag{3.19}$$

$$\text{upper\_violation} = \rho - (\hat{\rho} + \hat{f}_{1-\alpha/2}^{(residuals)}) \tag{3.20}$$

However, our initial implementation inadvertently used:

$$\text{lower\_violation} = \rho - (\hat{\rho} + \hat{f}_{\alpha/2}^{(residuals)}) \tag{3.21}$$

$$\text{upper\_violation} = (\hat{\rho} + \hat{f}_{1-\alpha/2}^{(residuals)}) - \rho \tag{3.22}$$

This error is mathematically equivalent to computing:

$$\text{conformity} = \max(-\text{lower\_violation}, -\text{upper\_violation}) \tag{3.23}$$

Theorem 3.1. *The inverted conformity scores are always larger than or equal to the original conformity scores, making the inverted method more conservative.*

Proof. Let $L = \hat{\rho} + \hat{f}_{\alpha/2}^{(residuals)}$, $U = \hat{\rho} + \hat{f}_{1-\alpha/2}^{(residuals)}$, and $y = \rho$, with $L \leq U$.

Let $a = L - y$ and $b = y - U$. Then, the original and inverted conformity scores are:

$$C_{\text{orig}} = \max(a, b) \tag{3.24}$$

$$C_{\text{inv}} = \max(-a, -b) = -\min(a, b) \tag{3.25}$$

Using the identity $\max(a, b) + \min(a, b) = a + b$:

$$C_{\text{inv}} = -\min(a, b) = -(a + b - \max(a, b)) \tag{3.26}$$

$$= \max(a, b) - (a + b) = C_{\text{orig}} - (a + b) \tag{3.27}$$

Since $a + b = (L - y) + (y - U) = L - U$:

$$C_{\text{inv}} = C_{\text{orig}} - (L - U) = C_{\text{orig}} + (U - L) \tag{3.28}$$

Since $U \geq L$, we have $C_{\text{inv}} \geq C_{\text{orig}}$. $\square$

Since the quantiles of conformity scores are used to construct prediction intervals, and the inverted conformity scores are always larger, the inverted method produces wider (more conservative) prediction intervals than the standard approach. This mathematical relationship explains why the inverted method often maintains coverage guarantees even when the standard method might struggle under challenging conditions.

Rather than discarding this implementation, we decided to evaluate its performance empirically. Interestingly, this inverted formulation often demonstrated competitive or even superior performance in certain scenarios, suggesting that the additional conservatism may be beneficial for uncertainty quantification in power system applications when we need to ensure coverage. But this improvement in coverage, comes at the cost of the loss of one of the major benefits of the CQR method. The intervals never shrink. Calculating the non-conformity scores in the inverted method always increases the size of the intervals.

We also verified the inverted method empirically in some of the examples of the original CQR paper [22].

One of the authors' experiments is depicted in Figure 3.10:

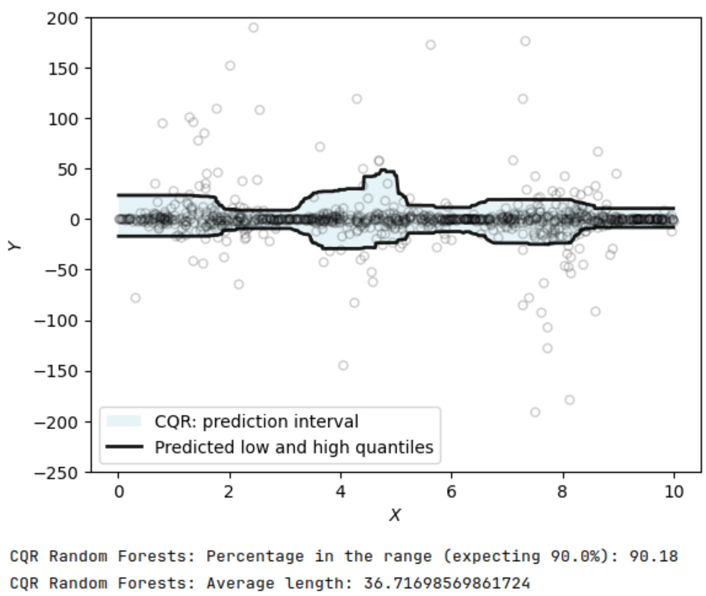

Figure 3.10: Authors' original results. This experiment represents the second synthetic example (heavy-tailed Cauchy distribuition), provided in [22]. The regions highlighted in blue represent the constructed prediction intervals that were obtained by using the CQR method. The two black curves are the lower and upper quantile regression estimates based on quantile random forests.

To which we applied the same logic (see Appendix B), and obtained the adultered results shown in Figure 3.11.

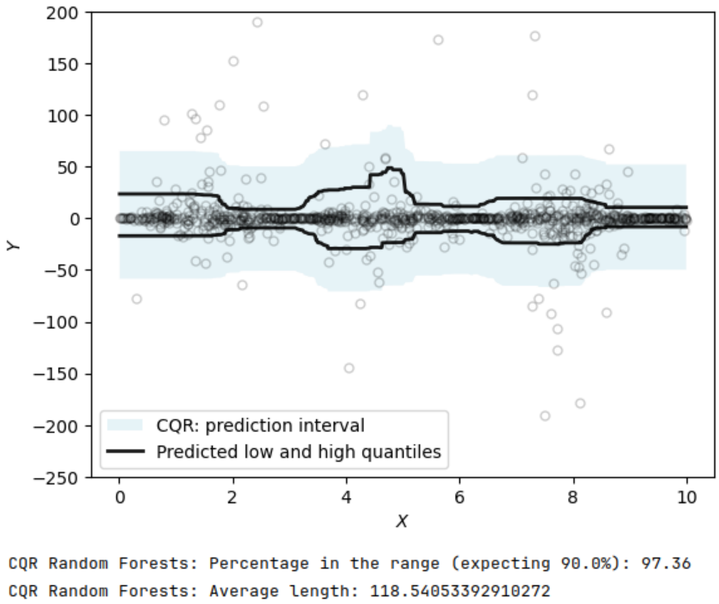

Figure 3.11: Inverted method results: As expected, the blue region increased in size, which results in increased coverage at the cost of increase in width.

The Inverted RB-CQR method, apart from the sign error difference in the conformity score calculation, follows exactly the same implementation as the RB-CQR method.

All machine learning backends (LightGBM, HistGradientBoosting, and Random Forest) were implemented with both the standard and inverted conformity score computations, so that we can compare both approaches.

3.4.7.1 Adaptive Extensions

All RB-CQR implementations have an alternative implementation that incorporates the ACI update rule (3.10) to provide both local adaptivity through quantile regression and temporal adaptivity through adaptive significance (α) levels.

4. Implementation Details

The pipeline shown in Figure 4.1 was implemented as a modular Python-based system designed to easily add and evaluate CP methods. To improve efficiency the calibration, model training, and testing (as described in Chapter 4) phases were parallelized.

4.1. Modular Conformal Prediction Architecture

The framework provides a modular architecture that allows the easy integration of new CP methods.

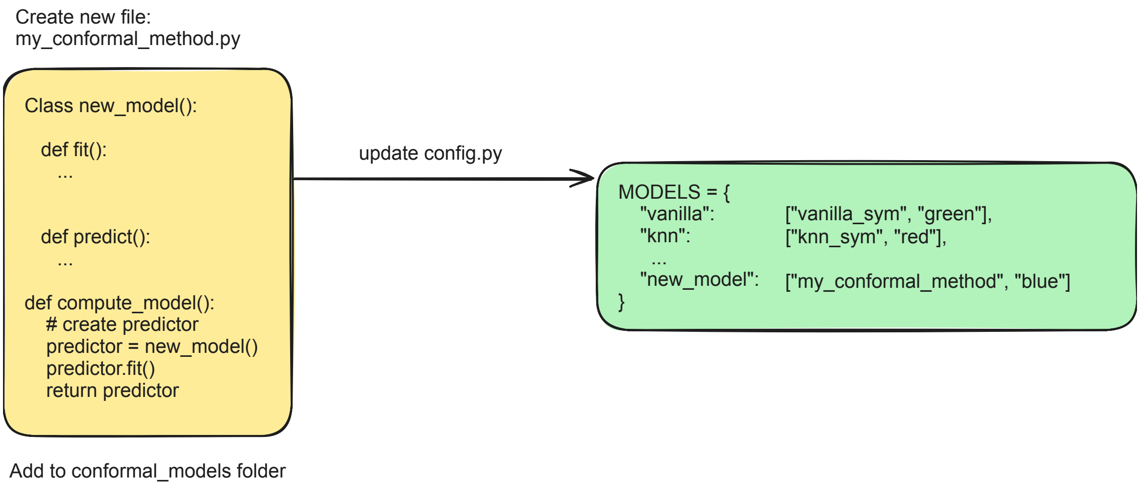

Figure 4.1: Modular architecture. Adding a new conformal method involves implementing fit() and predict() methods in a Python file and registering it in the configuration dictionary.

Each conformal predictor must expose a standardized interface with two mandatory methods: a `fit()` method that accepts calibration data and trains the conformal predictor, and a `predict()` method that generates prediction intervals for new observations. A function `compute_model()` must also be created, in which the class is instantiated, fit, and finally returned. This simple abstraction allows the framework to treat all conformal prediction variants uniformly during training, testing, and evaluation phases.

Model integration occurs through a straightforward configuration system that specifies which implementations to activate and evaluate. New conformal prediction methods can be added by implementing the required interface in a separate Python module (e.g., my_conformal_method.py), then registering the model in the configuration dictionary as pictured in Figure 4.1.

We decided the modularity would be beneficial since it eliminates the need to implement data handling, evaluation metrics, or visualisation components when developing or testing new conformal prediction methods. Instead, only the CP algorithm needs to be implemented and the framework then handles all remaining aspects, such as data flow, from calibration data preparation to final performance analysis, requiring only that new implementations conform to the established interface.

4.2. Parallel Implementation

The framework has available a parallel execution strategy designed to maximize computational efficiency across the three main phases, to use multiple CPU's, while maintaining data integrity, which is only possible because Grid2Op allows the creation of different instances of the same environment.

4.2.1 Memory-Optimized Storage

The ensemble mode can use a lot of memory due to the need to store and manage calibration data for twenty specialized conformal predictors. The framework addresses this by using episode caching, reducing the peak memory used.

4.2.1.1 Simplified Observation

An important optimization involves extracting only essential features from complete Grid2Op observations rather than storing full observation objects. For this, we use a `SimpleObservation` class that contains only the data required for our CP methods implementation: line capacity utilization values (ρ), load active and reactive power, generation active power and voltage, temporal information (hour, day, month, year, day of week, minute), and current simulation step. Extracting the data to a simpler object helped us achieve some memory reduction compared to storing complete Grid2Op observations while preserving all information used in the feature engineering and conformal prediction model training. This class can be altered so that we can save more attributes from a Grid2Op observation, if an implemented method has the need for additional information.

4.2.1.2 Episode Caching

The framework uses caching for calibration episodes to avoid redundant computation for different runs. Each calibration episode is cached using Hierarchical Data Format 5 (HDF5) [52], with filenames that encode all the configuration parameters:

```
calib_ensemble_chronic903_horizon12_actioninf12_warmup288_episodes15_steps8064_
lines111213-121314-133-144-3615-91012_seed2042_line_0_1_0_episode_7.h5
```

This caching system enables:

- **Incremental execution**: Only missing episodes are computed during subsequent runs
- **Configuration-aware caching**: Different parameter combinations maintain separate caches
- **Fault tolerance**: Failed episodes don't prevent reuse of successfully computed episodes

We essentially use this strategy to avoid running the calibration episodes multiple times in the same setting, whey they are deterministically going to produce the same results, which is specially useful when implementing and testing CP models, in order to avoid to re-run the calibration set collection.

Moreover, this strategy is beneficial for the Ensemble method, because it only needs to load to memory the specific episode that it needs when training, instead of having everything in memory at all times, which allows the ensemble training to work even on machines with more limited RAM.

4.2.2 Multi-Level Parallelization

The parameters `MAX_WORKERS_CALIBRATION`, `MAX_WORKERS_MODELS`, and `MAX_WORKERS_TESTING` corresponding to the calibration phase, model training phase, and testing phase respectively, can be set in the configuration file (config.py), to allow optimization based on available computational resources. If these parameters are set to the value of 1, the framework runs in sequential mode. The user is therefore responsible for determining the optimal worker counts for each phase based on their specific hardware, and performance requirements.

In the calibration phase, episodes are distributed across worker processes, with each worker running complete Grid2Op simulations independently. This approach is possible, because Grid2Op guarantees reproducibility and ensures that each simulation maintains its environmental integrity. In the ensemble case, the parallelisation happens at the episode level within each line’s calibration, enabling concurrent execution of multiple episodes.

The model training phase uses parallelization by distributing different conformal prediction models across separate worker processes for concurrent training. This means training multiple model types such as Vanilla, ACI, and RB-CQR variants simultaneously.

The testing phase parallelization also distributes the test episodes across concurrent workers, with each episode running in its own isolated environment. Since this process is *embarrassing parallel*, it enables the simultaneous run of independent episodes, which in turn provides significant speedups in multi-core computers.

5. Experimental Results

This chapter presents an experimental evaluation of the implemented CP methods for power system uncertainty quantification, assessed across the three scenarios we presented in previous Chapters. Our results start by analysing the benign baseline scenario, followed by the adversarial stress testing, and ending with the ensemble defense strategy.

Case Study 1 establishes the performance under stable single-mode operation without adversarial interference. This scenario enables us to understand each method's inherent capabilities, efficiency-coverage trade-offs, and patterns when operating under benign conditions, that is, without the presence of adversarial attacks. The results of this study provide a reference point of comparison when we analyse the performance of the following cases.

Case Study 2 introduces adversarial stress testing, subjecting the same CP methods to coordinated attacks on selected transmission lines during both calibration and testing phases. This scenario evaluates how well methods adapt to hostile operational environments in which the forecaster's training data may not well represent the threats encountered during testing. The stress testing shows some limitations of the static approaches and highlights the value of adaptive mechanisms for maintaining coverage guarantees when facing distribution shifts. We note that in this study, we also subject the same lines to attack during the calibration collection, in order to study what happens when some lines are more susceptible to attacks than others.

Case Study 3 introduces the ensemble defense strategy that emphasises adversarial resilience. This scenario explores preparation through diverse training, with 20 models calibrated under distinct single-line attack scenarios (one model per transmission line). At test time, worst-case aggregation across the ensemble is used to provide robustness and protection against line attacks, which prioritizes being prepared for the worst case in terms of coverage, although at the cost of sacrificing some interval efficiency.

In all three Case Studies, we set the miscoverage rate to $\alpha = 0.1$, establishing a coverage rate of $1 - \alpha = 90\%$, a common choice in the literature that balances practical requirements with statistical rigor. All experiments were run using a MacbookPro (M3 Pro processor, 12 cores, 18Gb RAM).

The calibration phase collects data from 30 episodes (chronics 901 to 930), a configuration chosen to provide statistically significant results while remaining computationally tractable. We test chronic 931, to provide insights to how the models behave in each of the analysed scenarios. For brevity, we only include the analysis of this chronic, which is very similar to the results of the chronics that follow it.

We report comparative performance across scenarios. For each scenario, we examine: i) coverage comparison (globally and in action influenced periods) across all models; ii) width (efficiency) comparison across all models; and iii) coverage and width by forecast horizon. In Appendix A, we present some aditional results, examining for each method the worst-case scenarios and average performance. Visualisations highlight periods in which the control agent intervenes, using the colour yellow for the behaviour during critical operational moments.

The methods described in Chapter 3 were applied to all case studies, namely: Vanilla, k-NN, ACI, RB-CQR and Inverted RB-CQR, and for each method, all the variants were considered.

5.1. Case Study 1: Operating under benign conditions

Case Study 1 provides baseline performance of CP methods under benign conditions: no transmission lines are attacked during calibration or testing. As such, it serves as the reference for subsequent stress and defense analyses.

5.1.1 Line-level Coverage Analysis

Coverage is the fundamental metric for evaluating conformal prediction methods, since it directly measures whether uncertainty estimates respect the nominal 90% target. In this subsection, we first examine overall coverage across all transmission lines under benign conditions, establishing how well each method maintains the desired reliability in a stable environment. We also analyse periods that are followed by the CurriculumAgent's actions. These action-influenced moments, while less frequent in benign scenarios, are an indicator of whether coverage remains stable when the system responds to the agent's actions.

5.1.1.1 Global coverage across lines

Coverage results under benign operating conditions are shown in Figure 5.1. Each panel displays the empirical coverage achieved by all methods across the 20 transmission lines.

The dashed line marks the nominal 90% coverage target. For the bar representations: in the Vanilla, ACI, and kNN methods, dark bars indicate the symmetric variant while lighter bars show the asymmetric variant. For the RB-CQR based models (including inverted variants), dark bars represent the base model and lighter bars represent the base model enhanced with the ACI rule.

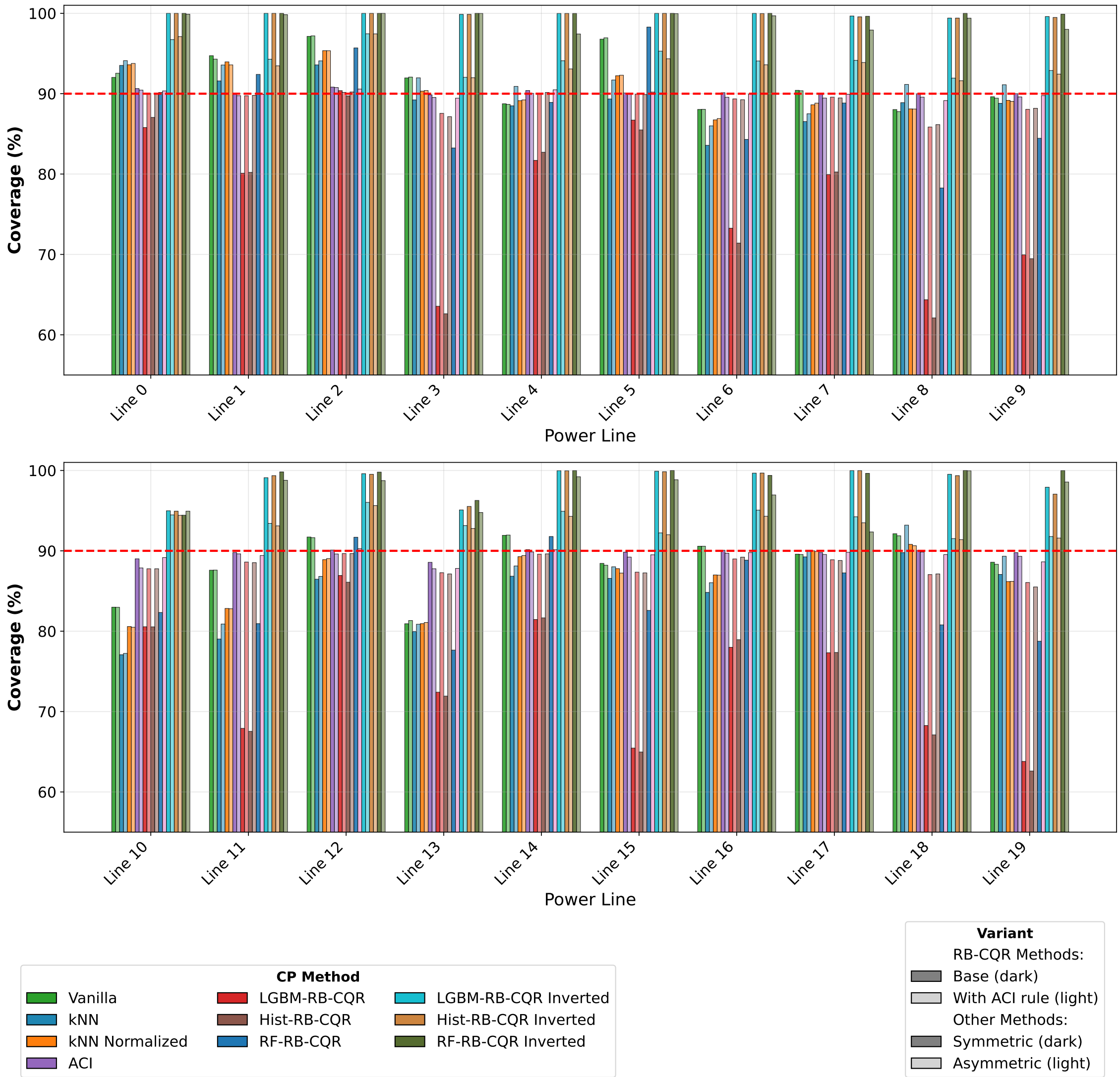

Figure 5.1: Coverage comparison across all conformal prediction methods in Case Study 1, showing empirical coverage achieved on each of the 20 transmission lines under benign operating conditions.

The Vanilla method consistently achieves coverage near or above the required 90% threshold for most lines, and becomes a reliable baseline in benign conditions, by using simple global calibration. The k-NN method delivers comparable coverage performance to Vanilla and the ACI method outperforms both by achieving perfect coverage across all lines.

In contrast, the standard RB-CQR implementations exhibit bad performance and fail to achieve coverage, although the Random Forests seems to be the most reliable option and gets closer to the required coverage. However, incorporating the ACI update rule resolves these coverage deficiencies for all the backends, making the Random Forests + ACI combination the most effective RB-CQR configuration tested.

The Inverted RB-CQR variants demonstrate different behavior, with all inverted implementations exceeding the target nominal level, resulting in more conservative uncertainty quantification intervals. Since every backend in the inverted method achieves nominal coverage levels, their relative performance must be evaluated through efficiency metrics, as detailed in Subsection 5.1.2.

Lines 10 and 13 exhibit the worst performance overall, and are examined in more detail in Appendix A.

5.1.1.2 Action-influenced coverage across lines

To complement the previous analysis, Figure 5.2 shows the coverage performance of the CP methods during action-influenced periods.

The action-influenced analysis is consistent with the overall coverage patterns, but we note that its statistical significance is reduced due to the limited frequency of CurriculumAgent actions under benign conditions. The observed distributional differences result from smaller sample sizes rather than performance degradation. We observe that most models attain the nominal level of coverage during action influenced periods, with line 10 being the most challenging line.

5.1.2 Line-level Interval Width Analysis

The efficiency of each method is presented in Figure 5.3, showing the average interval widths of all transmission lines. We remind the reader that having lower interval widths does not mean that the model performs better, but instead that its efficiency is better, and in order to be a better performing model, it has to first guarantee coverage, and only then can be compared in terms of efficiency.

Once again, the Vanilla method serves as the baseline to which the other methods are compared. The k-NN method achieves efficiency advantages in comparison to the Vanilla method, demonstrating significant reductions in average interval width while having comparable performance. This improvement comes from using the neighbours to construct

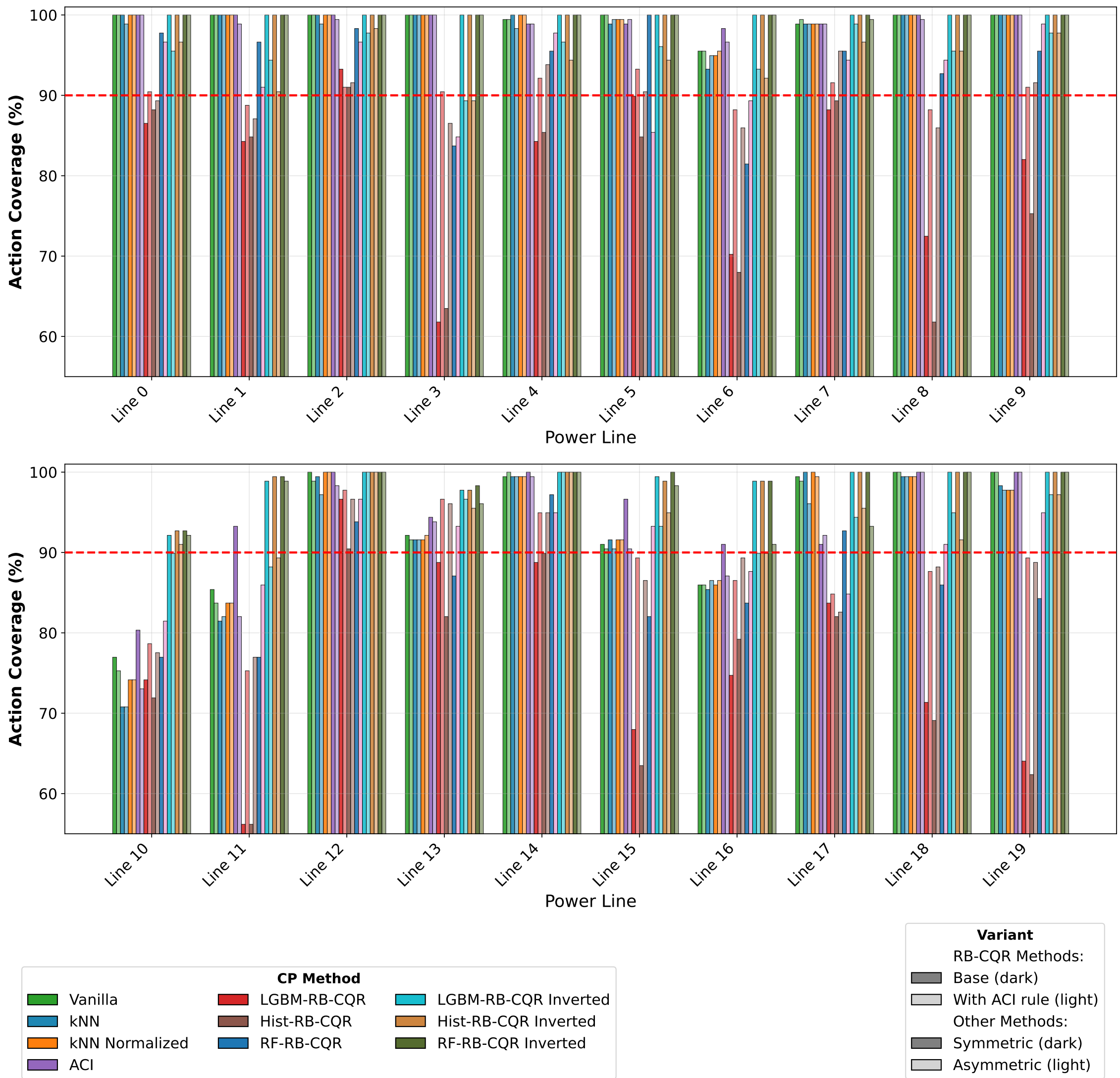

Figure 5.2: Coverage performance during action-influenced periods for selected conformal prediction methods in Case Study 1, showing method coverage in periods affected by the CurriculumAgent actions.

quantiles for each point forecast, which helps in capturing local uncertainty. k-NN is naturally more adaptive to heteroskedasticity, which allows to reduce interval sizes in various regions, leading to lower intervals, and consequently improved efficiency, when comparing to the global quantile approach of the Vanilla method. The k-NN normalised variant, which presented coverage very similar to the k-NN method, also presents very similar efficiency, not revealing any advantage in Case Study 1.

The ACI method maintains efficiency levels comparable to Vanilla in terms of average interval width, though it suffers from occasional width spikes during periods of consecutive prediction failures. These spikes, while mathematically justified by the update mechanism, can cause the prediction intervals to exceed the 0.95 line threshold. More detailed information about these spikes can be found in Appendix A. We observe an increase in

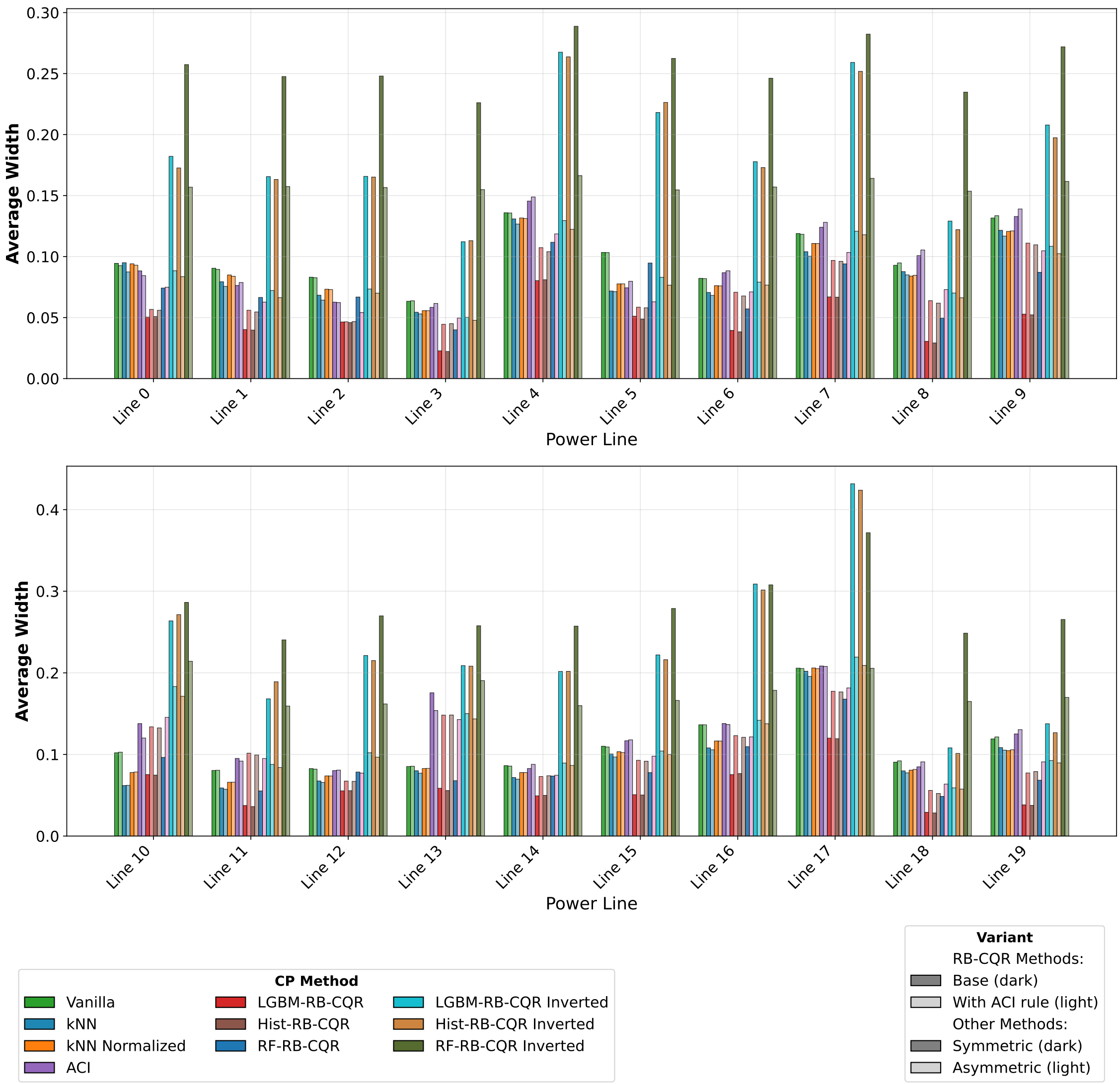

Figure 5.3: Interval width comparison across all conformal prediction methods in Case Study 1, illustrating the efficiency trade-offs associated with different uncertainty quantification approaches.

average width in lines 10 and 13, when comparing to the Vanilla and k-NN methods. This is due to the ACI's update rule, that can increase the size of the intervals when recent performance is not adequate, in order to attain the nominal coverage level. Once again, we reinforce that this width increase is positive, since although the other models have narrower intervals for this line, they also fail to achieve proper coverage.

The RB-CQR variants enhanced with the ACI update rule achieve coverage levels comparable to ACI methods while demonstrating superior efficiency in most lines. The Inverted RB-CQR variants exhibit wider intervals as expected from their conservative behaviour, with this trade-off being acceptable when aditional robustness is required. When we apply the ACI update rule to the inverted models, the average width decreases significantly, while maintaining excelent coverage levels.

5.1.3 Temporal Coverage and Interval Width Analysis

Beyond average comparisons of coverage and interval width, it is important to understand how predictive reliability evolves as the forecast horizon increases. When performing a forecast, uncertainty naturally compounds over time [53], which can prejudice coverage guarantees and increase interval widths. This subsection examines coverage and interval widths (efficiency) across the 12 forecasting horizons.

In Figure 5.4 we present the performance and efficiency of the implemented methods by forecast horizon, confirming our observations from Subsections 5.1.1 and 5.1.2.

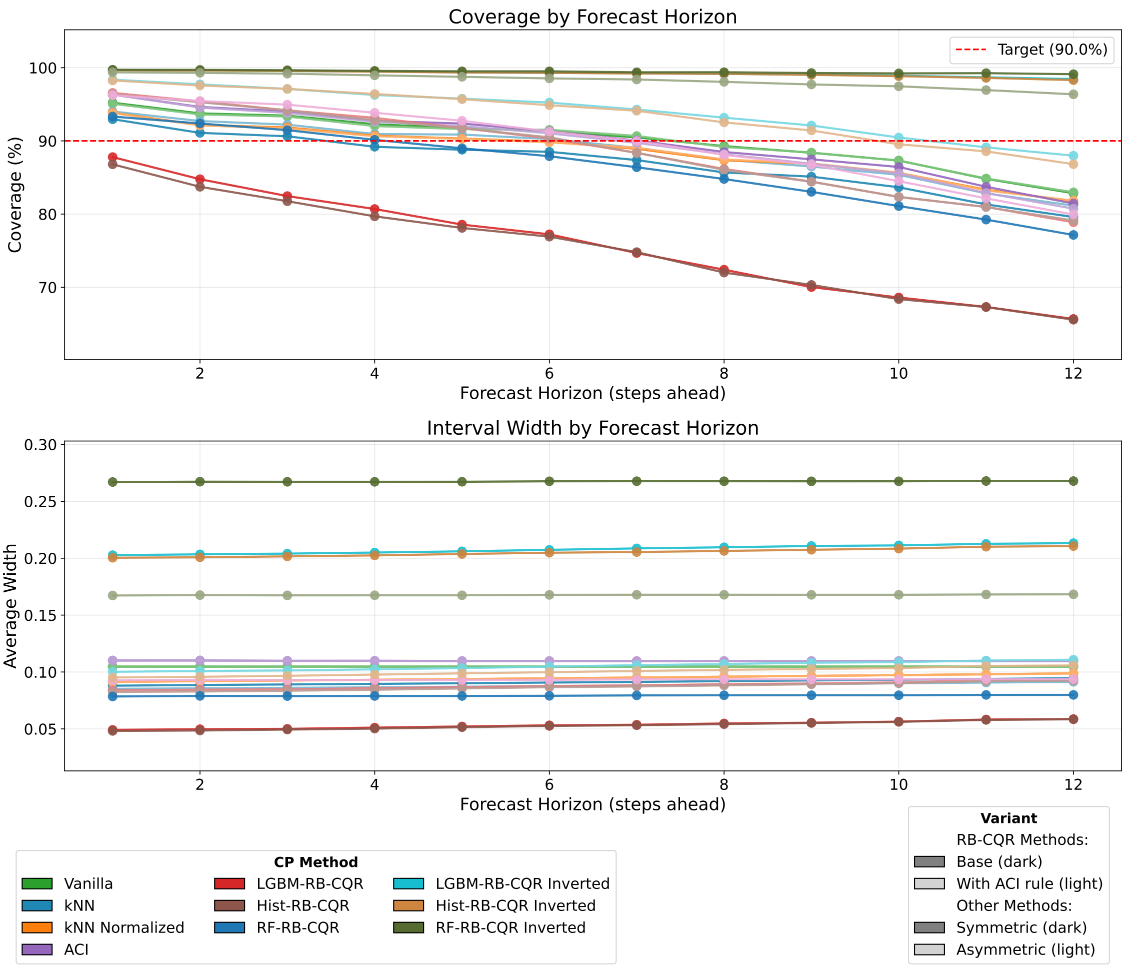

Figure 5.4: Coverage and Width by Forecast Horizon: Showing the performance of the methods, as well as their efficiency, for each forecast horizon.

As we can observe, the coverage gets worse by forecast horizon, as is to be expected, and the interval width tends to increase, although the increase in much less pronounced. The worst performing methods, the LightGBM and HistGradientBoosting backends of the RB-CQR method, have on average, for every forecast horizon, coverage below the 90% mark, failing even in the benign conditions of *Case Study 1*. Of the other methods, the coverage tends to be consistently above the 90% mark in the first 7 time-steps, starting to degrade at time-step 8. The Inverted RB-CQR methods (all the backends) are the exception, presenting coverage consistently above 90% across all forecast horizons. This validates the results from earlier sections. It is also evident that the Inverted RB-CQR methods have reduced efficiency (larger interval width), a tradeoff necessary for the increase in coverage. We conclude that the best performing Inverted RB-CQR method for *Case Study 1* is the Inverted RB-CQR with HistGradientBoosting backend, because it provides adequate coverage, while sacrificing the least in efficiency. A more detailed explanation is present in Appendix A in Figure A.18. Overall, the best performing method was the standard RB-CQR, when using the Random Forests backend, and the ACI update rule, although the Inverted RB-CQR method with HistGradientBoosting and the ACI update rule achieved very similar performance, while not displaying existence of spikes.

5.2. Case Study 2: Introduction of adversarial attacks

Case Study 2 subjects the CP methods to adversarial stress by introducing adversarial attacks on lines 3, 4, 12, 13, 14 and 15, during both calibration and testing phases. This scenario evaluates how each method adapts to more hostile operational environments where most episodes terminate prematurely due to blackouts, requiring more frequent agent interventions to try to prevent the grid from collapsing.

5.2.1 Line-level Coverage Analysis

Similarly to scenario 1, we first examine overall coverage across all transmission lines and then proceed to analysing action-influenced periods, efficiency and finally horizon specific coverage and width.

5.2.1.1 Global Coverage across lines

Coverage results under adversarial stress are shown in Figure 5.5. Each panel displays the empirical coverage achieved by all methods across the 20 transmission lines under adversarial attack. The dashed line marks the nominal 90% coverage target and we use the same colouring stategy employed in *Case Study 1*.

The Vanilla method still presents acceptable coverage, except in line 10, with most other lines close to the 90% mark.

The k-NN follows the same pattern, and its performance is similar to the Vanilla's. This could be because of the lack of similar scenarios in the calibration dataset. With only

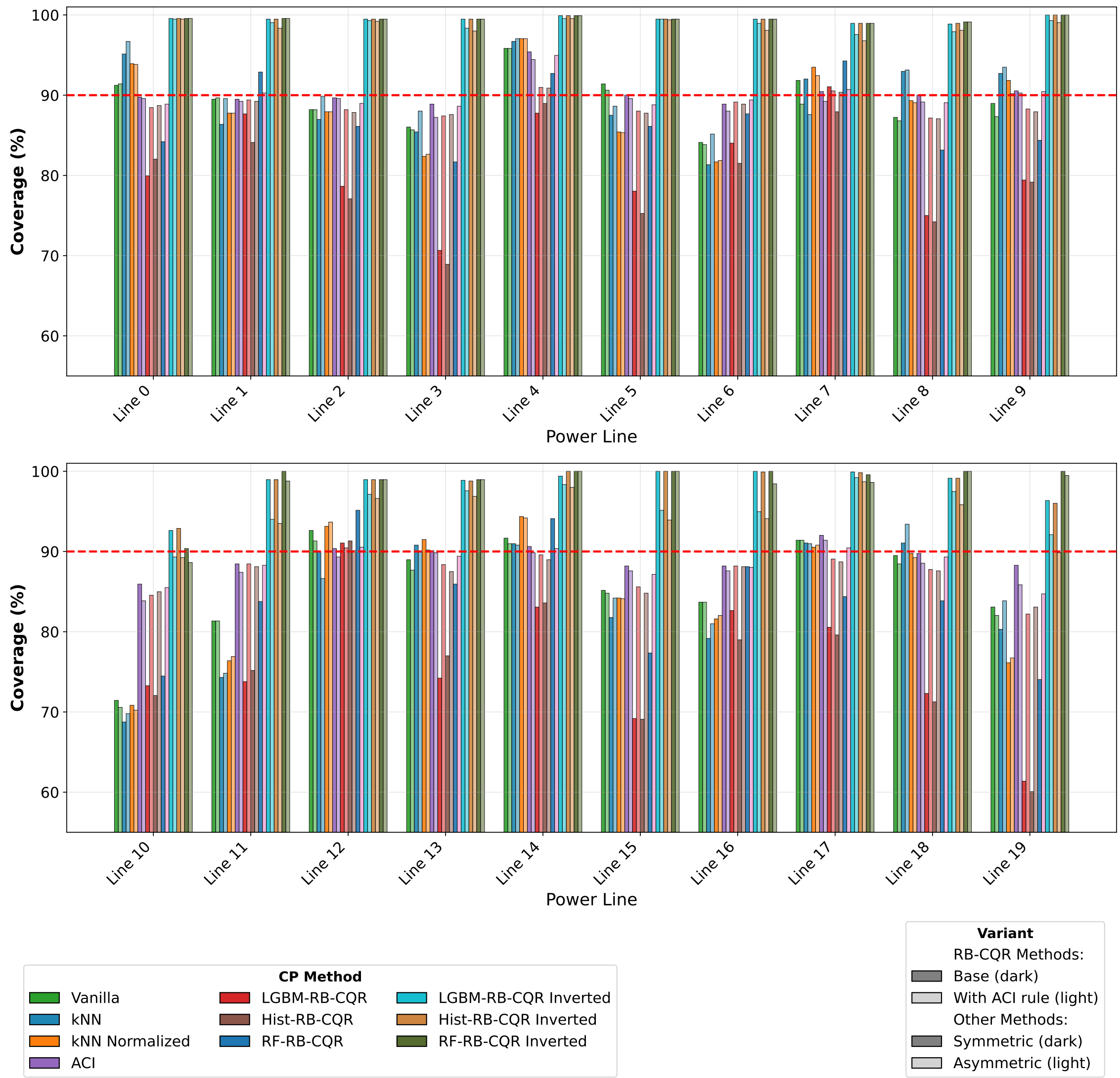

Figure 5.5: Coverage comparison across all conformal prediction methods in Case Study 2, showing empirical coverage achieved on each of the 20 transmission lines under adversarial attack conditions.

30 calibration episodes, most terminating prematurely due to attacks, the method cannot identify enough similarity patterns for many possible grid configurations. We note that in the lines that were attacked during the calibration and test phases, both the Vanilla and k-NN methods present good coverage, which reinforces the idea that these tests are being run on a complex environment, where the behaviour of each individual line affects the state of the whole grid.

The ACI method also has excelent performance when adversarial stress is introduced, showing similar performance to Case Study 1, with all transmission lines achieving coverage levels very close to the 90% target. This robustness proves that online adaptivity can effectively compensate for distribution shifts that happen at test time when the lines are attacked.

Standard RB-CQR present the same patterns seen in Case Study 1, with coverage failures comparable to the poor baseline performance. The ACI update rule provides these methods with enough adaptivity to reach the nominal coverage level, and once again, the Random Forests backend seems to be the better choice.

As expected, all Inverted RB-CQR variants show coverage levels above the required 90% target, due to their conservative behaviour that might be useful in order to provide some aditional resilience against the uncertainty generated by adversarial attacks.

5.2.1.2 Action-influenced coverage across lines

The action-influenced analysis has more significance in the adversarial context, because unlike Case Study 1, the grid suffers from more instability cause by the adversarial attacks, and therefore the CurriculumAgent is forced to act more often to try to prevent the grid from blackouts.

As we can observe in Figure 5.6, the more abundant action-influenced data provides statistical validation of method performance during the most critical operational periods, in which we observe the majority of the models performing well.

5.2.2 Interval Width Analysis

The efficiency under adversarial stress is presented in Figure 5.7, showing the effects of introducing adversarial attacks on the average interval width of the prediction intervals.

Contrary to intuition, the interval widths presented in Figure 5.7 are very close to those presented in Figure 5.3 of Case Study 1. This is a result of the exchangeability requirement for the CP methods based on split conformal prediction, which states that only the calibration and test data must be exchangeable. Although we cannot assume exchangeability due to the data being time series, since we attacked the same lines when collecting the calibration data and during testing, the results make sense.

The Vanilla method maintains similar interval width characteristics to Case Study 1. The k-NN method, which outperformed the Vanilla method in Case Study 1, now presents more mixed results, having worse efficiency in some lines, although both methods present very similar efficiency.

The ACI method maintains efficiency comparable to the Vanilla method while achieving better coverage performance, establishing it as a good strategy for both Cases 1 and 2.

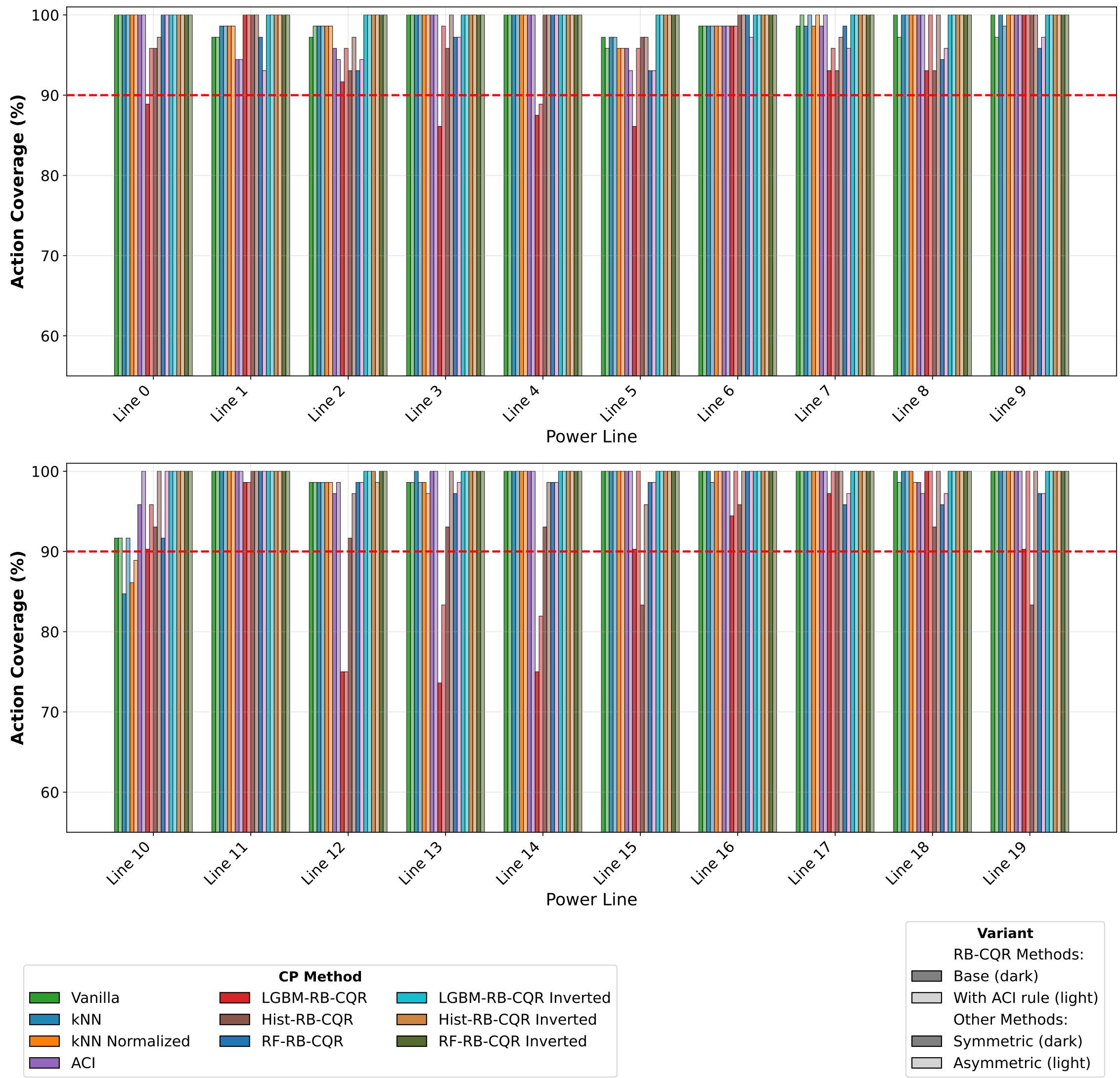

Figure 5.6: Coverage performance during action-influenced periods for selected conformal prediction methods in Case Study 2, showing method behaviour during frequent CurriculumAgent interventions under adversarial stress.

The RB-CQR methods with the ACI update rule present better efficiency than the ACI method, while achiving similar coverage levels and as expected, the Inverted RB-CQR variants exhibit the widest intervals, but this trade-off can become valuable when reliability is necessary under attack conditions.

5.2.3 Temporal Coverage and Interval Width Analysis

In Figure 5.8 we present the performance and efficiency of the implemented methods by forecast horizon under adversarial stress, revealing how the adversarial attacks affect the temporal evolution of uncertainty quantification.

In adversarial conditions, we notice earlier coverage degradation compared to Case Study 1. Most methods follow the same pattern and fall below 90% around time-step 5-6 instead

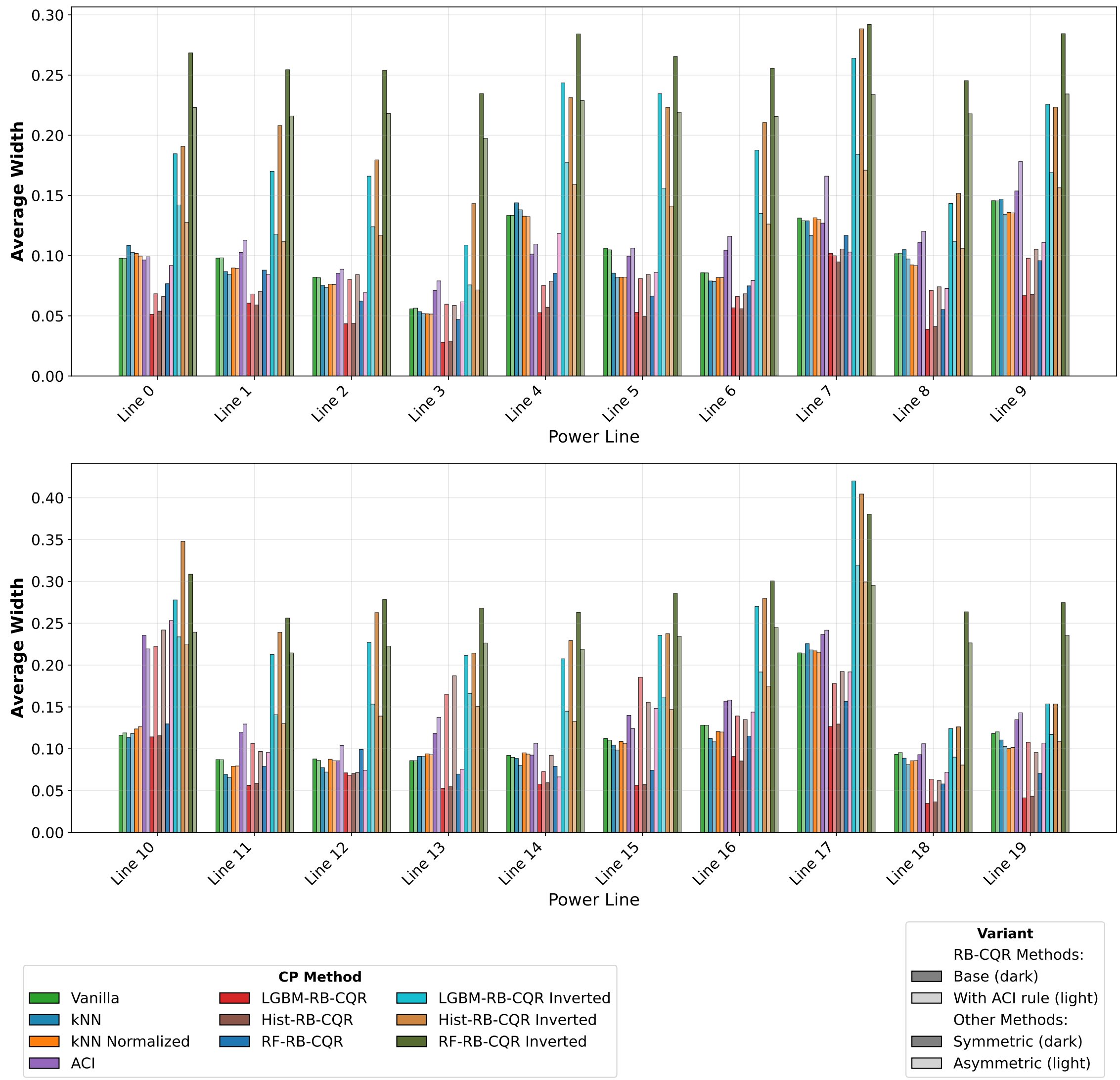

Figure 5.7: Interval width comparison across all conformal prediction methods in Case Study 2, illustrating the efficiency impacts of the adversarial stress. The efficiency does not change a lot compared to Case Study 1.

of time-step 8.

The ACI method demonstrates remarkable resilience to adversarial stress, maintaining coverage levels closest to the 90% target across all forecast horizons, only dropping below the 90% mark in timestep 8, presenting similar behaviour to Case Study 1.

The base RB-CQR methods that use gradient boosting (the Hist-RB-CQR and LGBM-RB-CQR) show the worst coverage by forecast horizon. The Random Forests backend shows better results, bringing the model closer to the overall patterns. When using the ACI update rule, the RB-CQR methods show performance comparable to the ACI method, while providing better interval width by forecast horizon.

In the case of the Inverted methods, we observe the over-performing coverage, closer

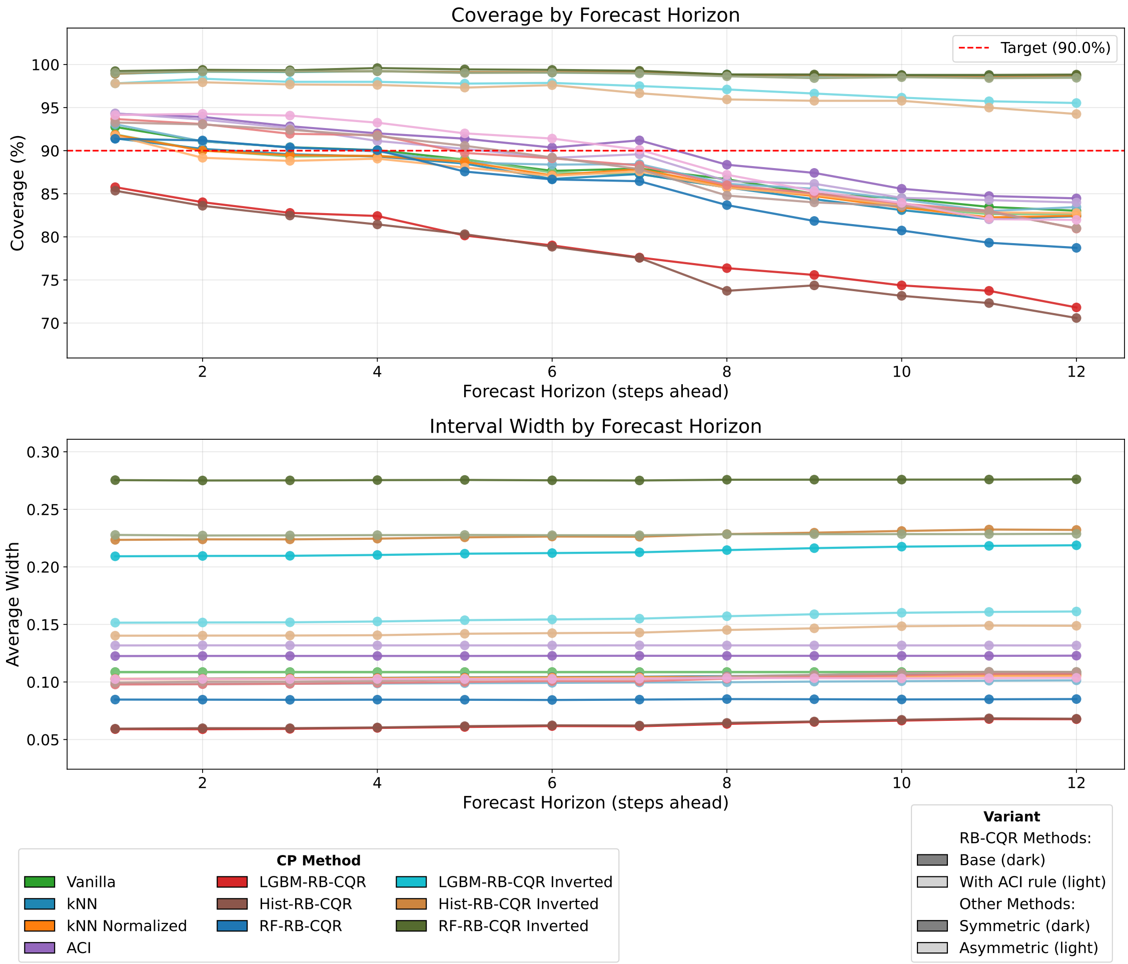

Figure 5.8: Coverage and Width by Forecast Horizon Under Adversarial Stress: Demonstrating method performance and efficiency degradation with adversarial attacks across the 12 prediction horizons.

to the 100% mark. We note that, in CP, using $\alpha = 0.1$, we aim to obtain coverage of at least 90%, meaning that an over-performing coverage does not mean that the method is performing better. Regardless, the increase in coverage can be beneficial to ensure extra robustness. Using the ACI update rule with the Inverted methods, brings the coverage by forecast horizon closer to the other methods, this time reducing the size of the intervals, providing good performance. The decrease in interval width by using the update rule, makes these methods worth considering. Notice that the HistGradientBoosting backend with the update rule has average width very close to the ACI method, while providing the natural heteroskedasticity of RB-CQR.

We conclude that the best performing method for Case 2 is the RB-CQR method with Random Forests and the ACI update rule, which provides very good results while obtaining excellent efficiency. The ACI method also presents excelent coverage, and has the advantage of being less computationally expensive in comparison to the method above, and must be considered when computational performance is required.

5.3. Case Study 3: The Ensemble Approach

Case Study 3 introduces the ensemble defense strategy, which presents a different approach to adversarial resilience. The idea is to anticipate potential attack scenarios through training diversity by using 20 specialized models, each trained in a scenario where only one line is attacked (one for each line), with the objective of creating a diverse knowledge base for adversarial scenarios. During testing, the same coordinated multi-line attacks from Case Study 2 target lines 3, 4, 12, 13, 14, and 15, but the ensemble defense operates by selecting the worst-case prediction from all 20 specialized models, creating inherently conservative intervals that prioritize coverage guarantees over interval efficiency.

5.3.1 Line-level Coverage Analysis

Similarly to scenarios 1 and 2, we first examine overall coverage across all transmission lines and then we examine periods influenced by the CurriculumAgent's actions (action-influenced periods). We then analyse efficiency and finally talk about the horizon-specific results.

5.3.1.1 Global coverage across lines

The coverage results for the ensemble strategy are shown in Figure 5.9.

Both the Vanilla and k-NN methods perform well in line 10, which had the worst performance in Case Study 2, demonstrating that adversarial preparation can enable even simple global calibration methods to handle attacks effectively, and we see most lines very close to or above the 90% coverage requirement. Both the k-NN method and the normalised k-NN variant show substantial improvement over Case Study 2. The ACI method continues delivering excelent performance, maintaining coverage across all transmission lines.

Standard RB-CQR, which failed failed in Cases 1 and 2, now achieve universal coverage across all backends, both with and without ACI enhancement. This difference confirms the utility of the Ensemble approach: even models that failed in the previous cases, can achieve good coverage. With the use of the ACI update rule, the RB-CQR methods keep displaying the good performance obtained in the previous cases.

The Inverted RB-CQR variants reveal a limitation within the ensemble context: the combination of worst-case aggregation with the conservative nature of the inverted method

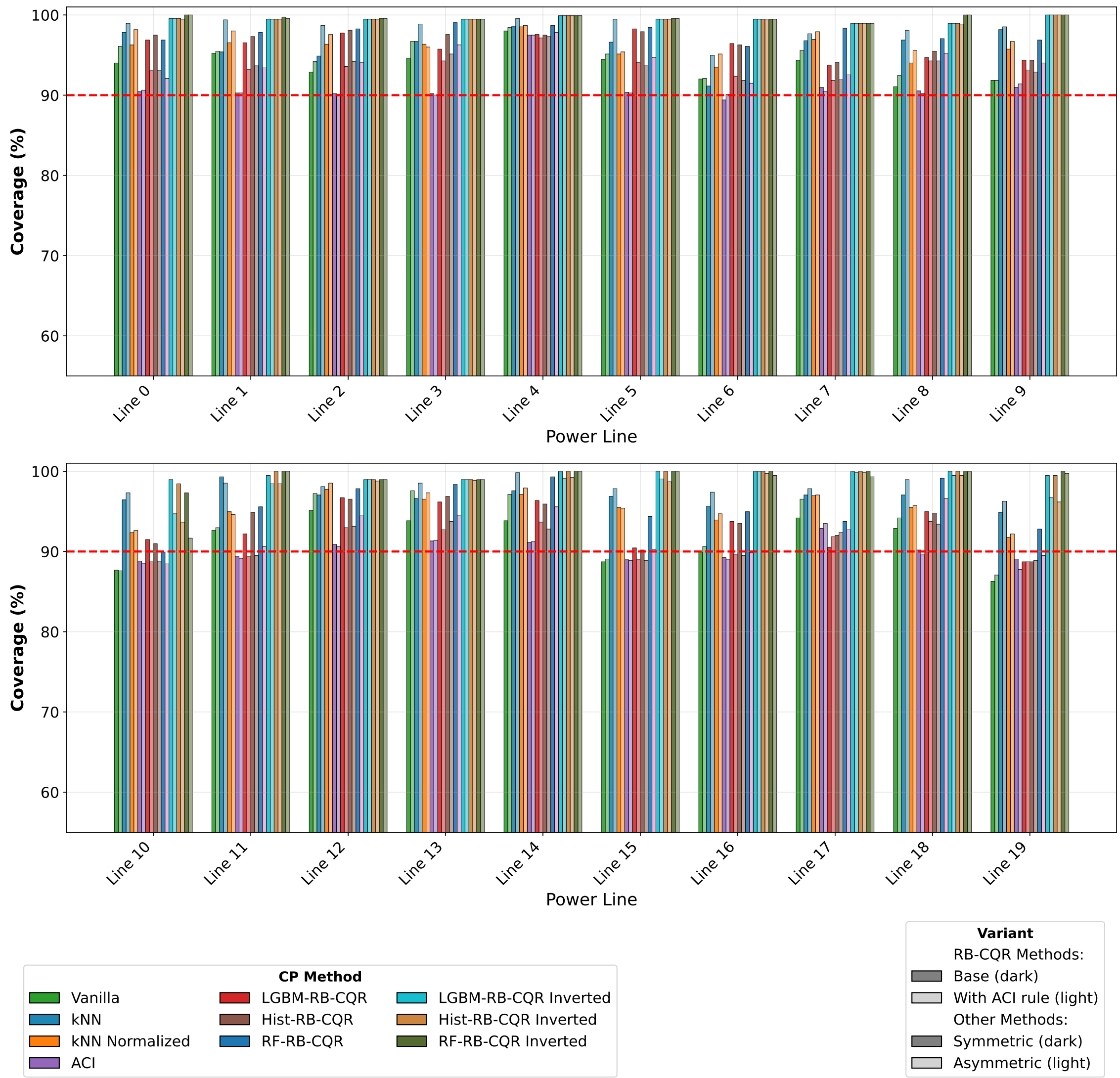

Figure 5.9: Coverage comparison across all conformal prediction methods in Case Study 3, showing empirical coverage achieved by the ensemble defense strategy with the 20 line-specific models.

creates too much conservatism that sacrifices efficiency without providing meaningful additional protection. Since ensemble preparation already provides adversarial protection through training, the additional conservatism becomes redundant and counterproductive.

5.3.1.2 Action-influenced coverage across lines

The action-influenced analysis under ensemble defense is shown in Figure 5.10. The diverse training enables consistent protective behaviour when the agent must act. All methods demonstrate consistent behaviour patterns that reflect the ensemble's preparation for adversarial conditions.

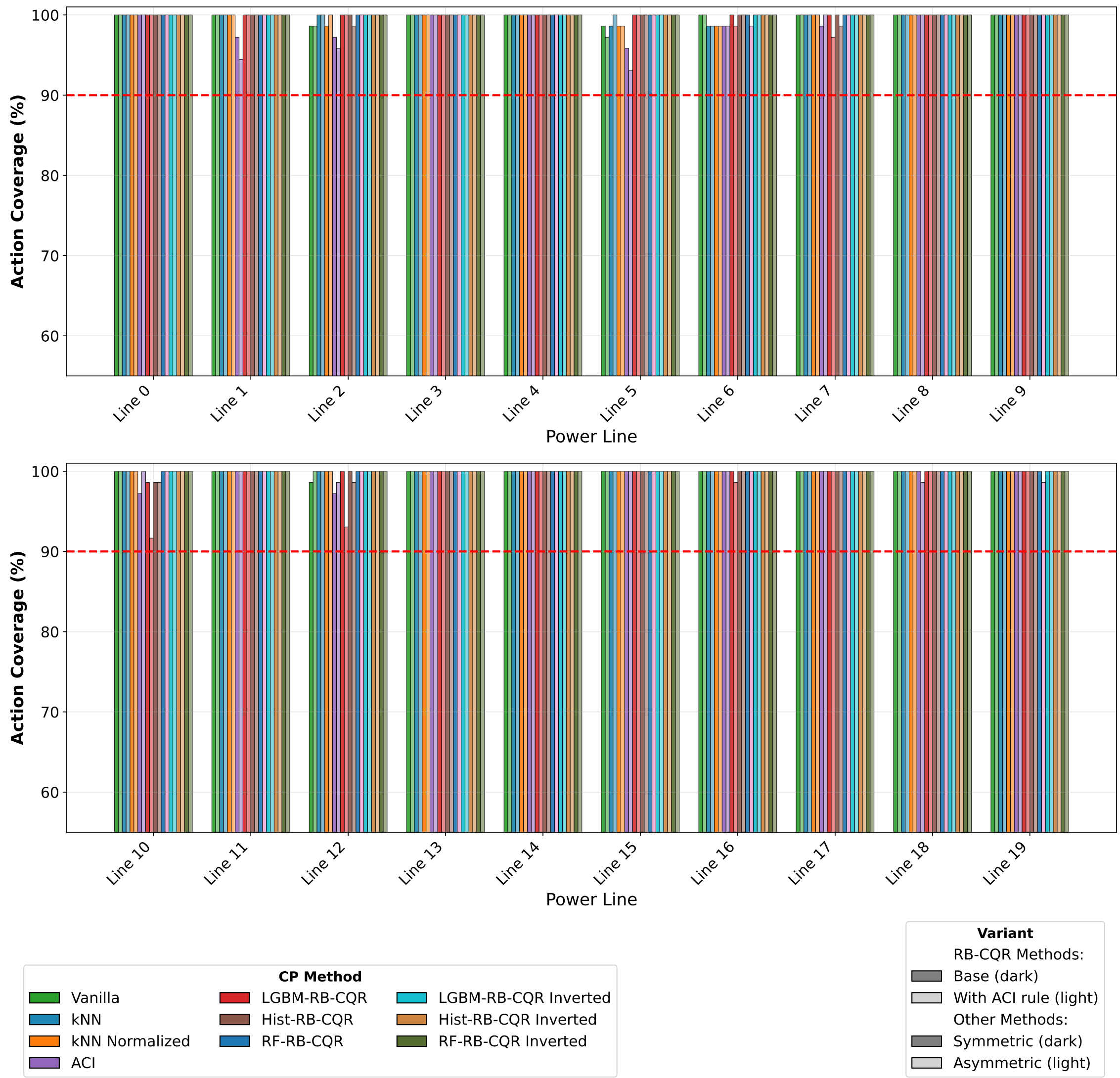

Figure 5.10: Coverage performance during action-influenced periods in Case Study 3, showing ensemble defense behaviour during more frequent CurriculumAgent interventions with adversarial preparation.

5.3.2 Interval Width Analysis

The efficiency under ensemble defense is presented in Figure 5.11 and reveals the trade-off between coverage protection and prediction precision that characterizes this worst-case aggregation strategy: the ensemble approach imposes efficiency penalties across most methods as the cost of adversarial protection through worst-case aggregation.

The Vanilla method shows notably wider intervals compared to Case Study 2. The k-NN method suffers from a much bigger increase in width, providing reduced efficiency. Interestingly, the k-NN normalised variant presents efficiency that is comparable to the Vanilla method, making this variant prefered over the base k-NN method in Case Study 3. The asymmetric variants, which in the other Case Studies were not relevant to the

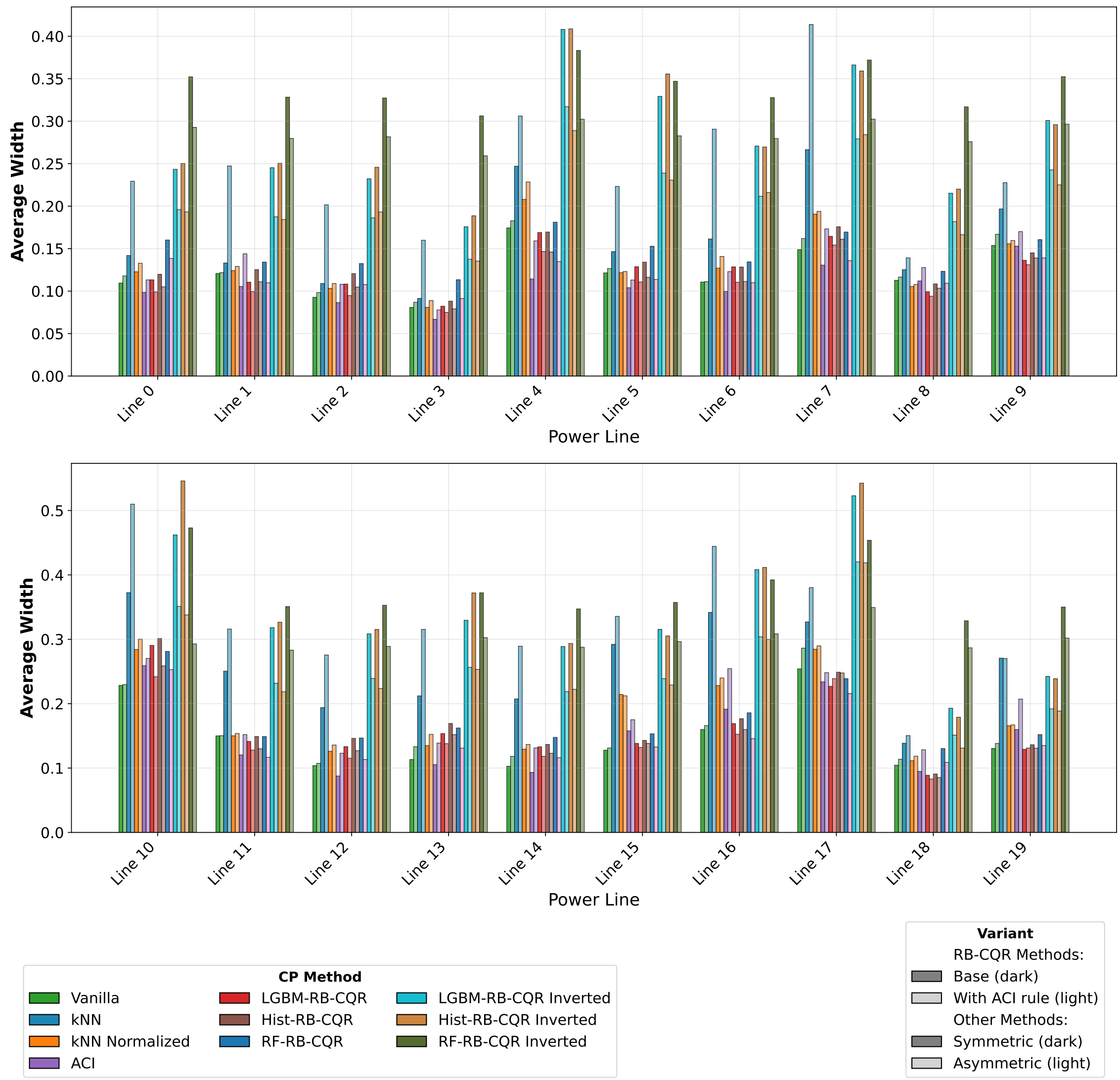

Figure 5.11: Interval width comparison across all conformal prediction methods in Case Study 3, illustrating the efficiency-coverage trade-offs inherent in ensemble defense strategies with worst-case aggregation.

analysis, now present worse performance, specially in the base k-NN model.

The ACI method demonstrates very good results in the ensemble context, getting the desired coverage with reduced interval width, therefore showing improved efficiency. This is due to the ACI update rule's ability to provide online interval adjustment, which only decreases the interval width when the model's recent coverage is good, and therefore works well in the ensemble approach, showing that it can optimise the wider intervals resulting of using worst-case aggregation.

All RB-CQR methods achieve coverage on all lines and are also positively affected by the ACI update rule, maintaining coverage while reducing interval width, therefore improving overall efficiency.

The Inverted RB-CQR variants exhibit much higher efficiency penalties, with wider intervals than the worst-performing standard ensemble method, confirming their unsuitability for the ensemble-based strategy, even when using the adaptivity from the ACI update rule.

5.3.3 Temporal Coverage and Interval Width Analysis

The worst-case aggregation of the Ensemble approach prioritizes coverage over efficiency, which changes the average coverage and width by forecast horizon compared to the previous Case Studies.

In Figure 5.12 we present the performance and efficiency of each method method across the 12 forecasting horizons, showing that the ensemble approach demonstrates exceptional coverage stability across all forecast horizons, with most methods achieving performance above the 90% nominal level for all 12 time steps, which represents a significant improvement over both Cases 1 and 2, and validates the use of this approach: the worst-case aggregation strategy successfully provides coverage guarantees. Unfortunately, this comes at the cost of efficiency, which is evidenced by the increased interval widths across all methods compared to single-model approaches.

We observe a notable difference in the k-NN method, that can be found in more detail in Appendix A, where normalised variants demonstrate superior efficiency compared to non-normalised approaches, contrasting with the findings from Case Studies 1 and 2. The symmetric normalised k-NN variation achieves the best performance out of all the k-NN methods.

The ACI method maintains its adaptation advantages, achieving coverage levels comparable to other methods while providing superior efficiency, due to its ability to optimise even the conservatism of worst-case aggregation.

The RB-CQR methods demonstrate universal coverage, representing the biggest improvement amongst all methods, when comparing to previous case studies. The RB-CQR with Random Forests and the ACI enhancement achieves the best efficiency amongst backends while maintaining robust coverage guarantees, followed closely by the RB-CQR method using the HistGradientBoosting backend and ACI update rule.

The Inverted RB-CQR methods exhibit excessive conservatism within the ensemble context, as aformentioned.

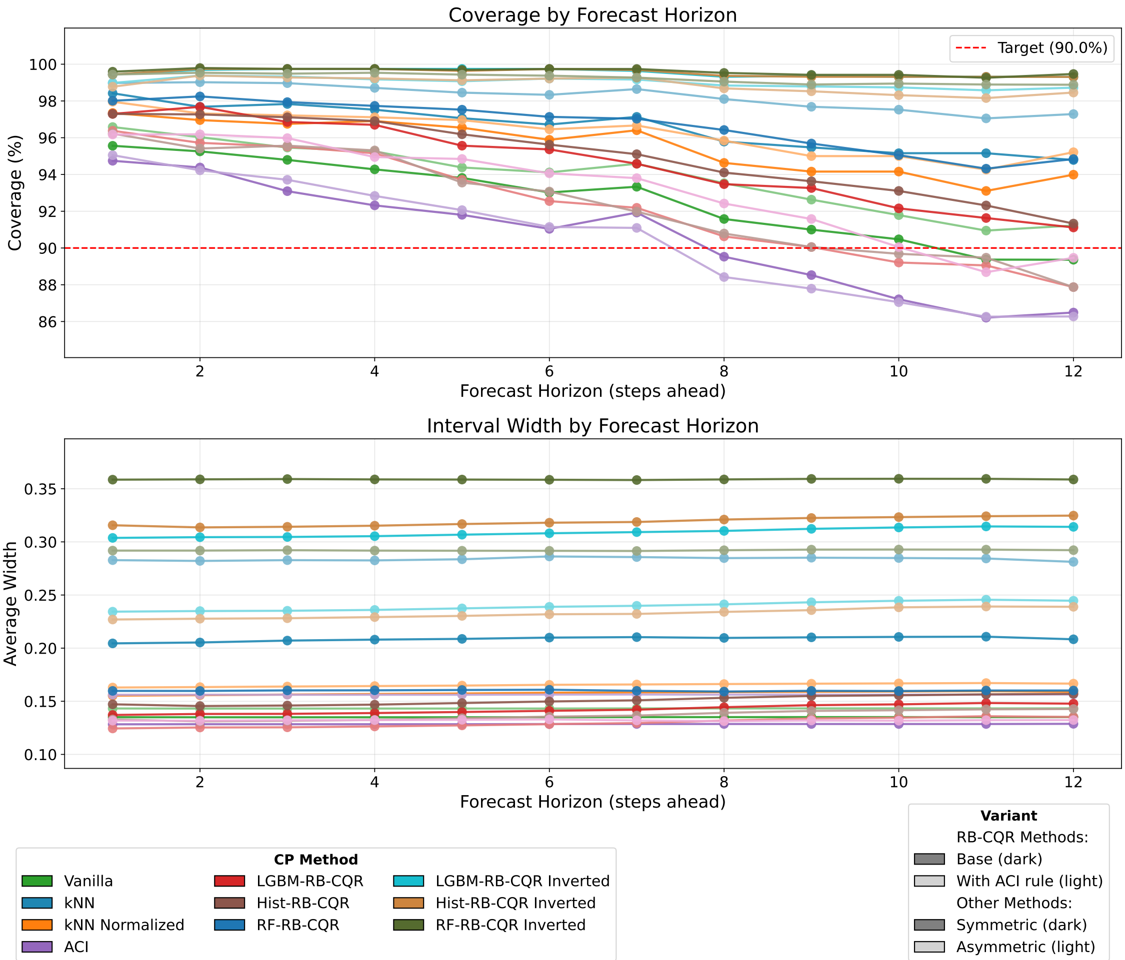

Figure 5.12: Coverage and Width by Forecast Horizon in Case Study 3, illustrating how adversarial preparation and worst-case aggregation affect temporal prediction reliability and efficiency trade-offs.

We conclude that in Case Study 3, the RB-CQR method with Random Forests and the ACI update rule is the best implemented method. We choose this method because of its inherent heteroskedasticity adaptation, and because it achieves very good coverage and efficiency results. The simpler ACI method, also produces excelent coverage and efficiency, but is not adaptive to heteroskedasticity. When computational complexity is an issue, the ACI method should be considered.

6. Conclusions and Future Work

In this Chapter, we present the conclusions obtained with the development of this thesis, and give directions for the future work to be performed.

6.1. Conclusions

In this thesis we developed a framework for applying CP methods to power system forecasting within the Grid2Op simulation environment, with the objective of providing uncertainty quantification across three operational scenarios. We also provided the implementation and evaluation of four distinct CP methods with multiple variations adapted to work in Grid2Op.

Our experimental progression from benign baseline conditions through adversarial stress testing to the ensemble defense demonstrates that conformal prediction method selection cannot be divorced from operational context. There is no universal solution, and optimal approaches must align with the operational requirements.

In what concerns the studied methods, there is one that stands out: the ACI. This method, although having some disadvantages, such as not naturally handling heteroskedasticity, proves consistent in all three Case Studies. The RB-CQR methods, an idea adapted from the CQR method [22], when enhanced with the ACI update rule, show significant promise. We also explored the Inverted RB-CQR method, which resulted in often exagerated coverages of 100%, which can be beneficial when guaranteeing coverage is the top priority. In what concerns the symmetric vs asymmetric variants of the proposed methods, we conclude that they performed about the same, which indicates that the ρ parameter distribution is not asymmetric. In most cases, the symmetric variant slightly outperforms the asymmetric variant. Table 6.1 presents a simplified overview with some advantages and disadvantages of each method.

The use of three case studies had the objective of answering different questions. *Case Study 1* operates under stable conditions, and evaluates model performance when the calibration and test data come from the same distribuition as the data used to train the load and generation forecaster. *Case Study 2* tries to understand what happens to the power grid when some lines, which might be more prone to attacks, are attacked during testing time. The calibration data uses the same attacked lines, but the same pre-trained forecaster with no knowledge of attacks was used. *Case Study 3* takes an alternate approach, which consists of diverse training. If there is an adversarial attack, there is a possibility that it could happen in any particular line. By training 20 specialized models, each exposed to single-line attack scenarios, and aggregating their predictions through worst-case selection, the ensemble approach addresses the question: ”What would happen to each transmission line in the worst-case scenario if any line were under attack?”. The ensemble strategy contributes to achieving robustness and provides a foundation for reliable uncertainty quantification in the energy sector.

Table 6.1: Comparison of Conformal Prediction Methods

Method	**Advantages**	**Disadvantages**	**Observations**	**Symmetric vs Asymmetric**	**Introduction of the ACI Update Rule**	**Conditional Coverage**
Vanilla	Good baseline model.	Fixed interval width for each line, not adapting to different regions of uncertainty.		No significant variation	N/A	No
kNN	Good performance in general, adapting to heteroskedasticity.	Lack of similar patterns in calibration can lead some lines to provide worse performance. Efficiency is compromised in Case Study 3.	Should always be chosen over the Vanilla method.	No significant variation.	N/A	Approximates
kNN normalised	Good performance in general, adapting to heteroskedasticity.	Lack of similar patterns in calibration can lead some lines to provide worse performance. Increased computational complexity.	Outperforms the standard kNN method in Case Study 3, and has similar performance on the other cases. Should be preferred over the standard kNN method.	No significant variation.	N/A	Approximates
ACI	Excellent coverage and width in all three Case Studies, adapting to recent performance.	Lacks natural adaptation to heteroskedasticity. Regions with high uncertainty can present spikes.	Consistent behaviour and marginal coverage, naturally adapted to time-series.	No significant variation.	N/A	No
RB-CQR	Naturally adaptive to heteroskedasticity.	Bad performance on Case Studies 1 and 2.	The Random Forests backend outperforms the other backends.	N/A	The coverage improves across all backends, making its performance comparable to the ACI method. Spike behaviour is introduced in some lines.	Approximates
Inverted RB-CQR	Conservative coverage, giving some safety margins.	Significantly increased interval width.	Not recommended in Case Study 3.	N/A	The coverage improves efficiency. HistGradientBoosting backend should be chosen over other backends.	Approximates

This research demonstrates that conformal prediction is viable for power system uncertainty quantification, with the need to select an appropriate method for each scenario.

Moreover, we establish that CP can maintain coverage guarantees, even under adversarial attacks. Not surprisingly, since we are working with time-series, the adaptive methods relying on the ACI update rule are the most promising candidates. This result is supported by the literature. For example, Zaffran et. al, argue that ACI is a good procedure for time series with general dependency [32].

6.2. Future Work

In future research we intend to introduce other more advanced methods, such as AgACI [32] and DtACI [33], which can provide an advantage in the performance/efficiency tradeoff, by trying to solve the spikes created naturally by the ACI update rule. These spikes are explained in Appendix A. We also intend to introduce Signal Temporal Logic (STL) [54, 55], in order to introduce system specifications, and issue binary alerts when these specifications are violated.

References

[1] A. N. Angelopoulos, S. Bates *et al.*, "Conformal prediction: A gentle introduction," *Foundations and Trends® in Machine Learning*, vol. 16, no. 4, pp. 494–591, 2023. [Cited on pages 1 and 7.]

[2] B. Donnot, "Grid2op - a testbed platform to model sequential decision making in power systems," 2020. [Online]. Available: https://GitHub.com/rte-france/grid2op [Cited on pages 4 and 15.]

[3] M. Lehna, J. Viebahn, A. Marot, S. Tomforde, and C. Scholz, "Managing power grids through topology actions: A comparative study between advanced rule-based and reinforcement learning agents," *Energy and AI*, vol. 14, p. 100276, 2023. [Cited on pages 4, 17, 22, and 76.]

[4] scikit-learn developers, "sklearn.ensemble.histgradientboostingregressor," https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.HistGradientBoostingRegressor.html, 2024, accessed: January 15, 2025. [Cited on pages 4, 22, and 39.]

[5] AI4RealNet Consortium, "Ai4realnet project," https://ai4realnet.eu/project/, 2023. [Cited on page 4.]

[6] Eurostat, "Glossary: Point forecast," https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Point_forecast, European Commission, 2024, accessed: 2025-07-04. [Cited on page 4.]

[7] T. Gneiting, "Making and evaluating point forecasts," *Journal of the American Statistical Association*, vol. 106, no. 494, pp. 746–762, 2011. [Cited on page 5.]

[8] P. Pinson, H. Madsen, H. A. Nielsen, G. Papaefthymiou, and B. Klöckl, "From probabilistic forecasts to statistical scenarios of short-term wind power production," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, vol. 12, no. 1, pp. 51–62, 2009. [Cited on page 5.]

[9] P. Pinson and R. Girard, "Evaluating the quality of scenarios of short-term wind power generation," *Applied Energy*, vol. 96, pp. 12–20, 2012. [Cited on page 6.]

[10] G. Shafer and V. Vovk, “A tutorial on conformal prediction.” *Journal of Machine Learning Research*, vol. 9, no. 3, 2008. [Cited on pages 7, 8, 9, and 11.]

[11] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005, vol. 29. [Cited on pages 7, 9, and 30.]

[12] J. Lei, M. G’Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, “Distribution-free predictive inference for regression,” *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1094–1111, 2018. [Cited on pages 8, 10, 13, and 30.]

[13] M. Van Oijen, “Bayesian methods for quantifying and reducing uncertainty and error in forest models,” *Current Forestry Reports*, vol. 3, no. 4, pp. 269–280, 2017. [Cited on page 8.]

[14] A. C. Davison and D. V. Hinkley, *Bootstrap methods and their application*. Cambridge university press, 1997, no. 1. [Cited on page 8.]

[15] A. M. Zoubir and D. R. Iskandler, “Bootstrap methods and applications,” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 10–19, 2007. [Cited on page 8.]

[16] M. Sesia and E. J. Candès, “A comparison of some conformal quantile regression methods,” *Stat*, vol. 9, no. 1, p. e261, 2020. [Cited on page 9.]

[17] R. Foygel Barber, E. J. Candes, A. Ramdas, and R. J. Tibshirani, “The limits of distribution-free conditional predictive inference,” *Information and Inference: A Journal of the IMA*, vol. 10, no. 2, pp. 455–482, 2021. [Cited on page 9.]

[18] J. A. Martinez, U. Bhatt, A. Weller, and G. Cherubin, “Approximating full conformal prediction at scale via influence functions,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 6, 2023, pp. 6631–6639. [Cited on page 9.]

[19] L. Wasserman, A. Ramdas, and S. Balakrishnan, “Universal inference,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 29, pp. 16 880–16 890, 2020. [Cited on page 10.]

[20] L. Lindemann, Y. Zhao, X. Yu, G. J. Pappas, and J. V. Deshmukh, “Formal verification and control with conformal prediction,” *arXiv preprint arXiv:2409.00536*, 2024. [Cited on pages 11 and 13.]

[21] A. N. Angelopoulos and S. Bates, “A gentle introduction to conformal prediction and distribution-free uncertainty quantification,” *arXiv preprint arXiv:2107.07511*, 2021. [Cited on pages 11 and 13.]

[22] Y. Romano, E. Patterson, and E. Candes, “Conformalized quantile regression,” *Advances in neural information processing systems*, vol. 32, 2019. [Cited on pages 13, 14, 37, 41, 66, 80, and 107.]

[23] K. Q. Zhou and S. L. Portnoy, “Statistical inference on heteroscedastic models based on regression quantiles,” *Journal of Nonparametric Statistics*, vol. 9, no. 3, pp. 239–260, 1998. [Cited on page 13.]

[24] Y. Renkema, N. Brinkel, and T. Alskaif, “Conformal prediction for stochastic decision-making of pv power in electricity markets,” *Electric Power Systems Research*, vol. 234, p. 110750, 2024. [Cited on pages 13, 23, 33, and 34.]

[25] Y. Renkema, L. Visser, and T. AlSkaif, “Enhancing the reliability of probabilistic pv power forecasts using conformal prediction,” *Solar Energy Advances*, vol. 4, p. 100059, 2024. [Cited on pages 13 and 23.]

[26] R. J. Tibshirani, R. Foygel Barber, E. Candes, and A. Ramdas, “Conformal prediction under covariate shift,” *Advances in neural information processing systems*, vol. 32, 2019. [Cited on page 13.]

[27] Y. Romano, M. Sesia, and E. Candes, “Classification with valid and adaptive coverage,” *Advances in neural information processing systems*, vol. 33, pp. 3581–3591, 2020. [Cited on page 14.]

[28] L. Guan, “Conformal prediction with localization,” *arXiv preprint arXiv:1908.08558*, 2019. [Cited on page 14.]

[29] ——, “Localized conformal prediction: A generalized inference framework for conformal prediction,” *Biometrika*, vol. 110, no. 1, pp. 33–50, 2023. [Cited on page 14.]

[30] C. Xu and Y. Xie, “Conformal prediction interval for dynamic time-series,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 11 559–11 569. [Cited on pages 14, 15, and 23.]

[31] I. Gibbs and E. Candes, “Adaptive conformal inference under distribution shift,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 1660–1672, 2021. [Cited on pages 15, 34, and 36.]

[32] M. Zaffran, O. Féron, Y. Goude, J. Josse, and A. Dieuleveut, “Adaptive conformal predictions for time series,” in *International Conference on Machine Learning*. PMLR, 2022, pp. 25 834–25 866. [Cited on pages 68 and 79.]

[33] I. Gibbs and E. J. Candès, “Conformal inference for online prediction with arbitrary distribution shifts,” *Journal of Machine Learning Research*, vol. 25, no. 162, pp. 1–36, 2024. [Cited on pages 15, 68, and 79.]

[34] C. Xu and Y. Xie, “Sequential predictive conformal inference for time series,” in *International Conference on Machine Learning*. PMLR, 2023, pp. 38 707–38 727. [Cited on page 15.]

[35] “Welcome to grid2op’s documentation,” 2025. [Online]. Available: https://grid2op.readthedocs.io/en/latest/ [Cited on pages 15 and 18.]

[36] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014. [Cited on page 15.]

[37] R. Bellman, “A markovian decision process,” *Journal of mathematics and mechanics*, pp. 679–684, 1957. [Cited on page 15.]

[38] Electric Power Research Institute, “AI.EPRI L2RPN Challenge Portal,” https://www.epri.com/l2rpn, accessed: 3 August 2025. [Cited on page 15.]

[39] A. Izadian, *Fundamentals of modern electric circuit analysis and filter synthesis*. Springer, 2019. [Cited on page 16.]

[40] R. S. Sutton, A. G. Barto *et al.*, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1. [Cited on page 17.]

[41] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996. [Cited on page 17.]

[42] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2020. [Online]. Available: http://incompleteideas.net/book/RLbook2020.pdf [Cited on page 17.]

[43] C. Kath and F. Ziel, “Conformal prediction interval estimation and applications to day-ahead and intraday power markets,” *International Journal of Forecasting*, vol. 37, no. 2, pp. 777–799, 2021. [Cited on page 22.]

[44] C. O’Connor, M. Bahloul, R. Rossi, S. Prestwich, and A. Visentin, “Conformal prediction for electricity price forecasting in the day-ahead and real-time balancing market,” *Energy and AI*, vol. 21, p. 100571, 2025. [Cited on page 23.]

[45] J. De la Torre, L. R. Rodriguez, F. E. Monteagudo, L. R. Arredondo, and J. B. Enriquez, "Electricity price forecast in wholesale markets using conformal prediction: Case study in mexico," *Energy Science & Engineering*, vol. 12, no. 3, pp. 524–540, 2024. [Cited on page 23.]

[46] W. Wang, B. Feng, G. Huang, C. Guo, W. Liao, and Z. Chen, "Conformal asymmetric multi-quantile generative transformer for day-ahead wind power interval prediction," *Applied Energy*, vol. 333, p. 120634, 2023. [Cited on page 23.]

[47] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016. [Cited on page 28.]

[48] H. Papadopoulos, K. Proedrou, V. Vovk, and A. Gammerman, "Inductive confidence machines for regression," in *European conference on machine learning*. Springer, 2002, pp. 345–356. [Cited on page 30.]

[49] H. Papadopoulos, V. Vovk, and A. Gammerman, "Regression conformal prediction with nearest neighbours," *Journal of Artificial Intelligence Research*, vol. 40, pp. 815–840, 2011. [Cited on page 31.]

[50] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017. [Cited on page 39.]

[51] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001. [Cited on page 39.]

[52] M. Folk, G. Heber, Q. Koziol, E. Pourmal, and D. Robinson, "An overview of the hdf5 technology suite and its applications," in *Proceedings of the EDBT/ICDT 2011 workshop on array databases*, 2011, pp. 36–47. [Cited on page 44.]

[53] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018. [Cited on page 53.]

[54] L. Lindemann, X. Qin, J. V. Deshmukh, and G. J. Pappas, "Conformal prediction for stl runtime verification," in *Proceedings of the ACM/IEEE 14th International Conference on Cyber-Physical Systems (with CPS-IoT Week 2023)*, 2023, pp. 142–153. [Cited on page 68.]

[55] F. Cairoli, N. Paoletti, and L. Bortolussi, "Conformal quantitative predictive monitoring of stl requirements for stochastic processes," 2023. [Cited on page 68.]

[56] V. Taquet, V. Blot, T. Morzadec, L. Lacombe, and N. Brunel, “Mapie: an open-source library for distribution-free uncertainty quantification,” *arXiv preprint arXiv:2207.12274*, 2022. [Cited on page 77.]

[57] ——, “Tutorial for time series,” https://mapie.readthedocs.io/en/stable/examples_regression/1-quickstart/plot_ts-tutorial.html, 2024, mAPIE Documentation, Tutorial for time series. [Cited on page 77.]

[58] H. Susmann, A. Chambaz, and J. Josse, “Adaptiveconformal: An r package for adaptive conformal inference,” *arXiv preprint arXiv:2312.00448*, 2023. [Cited on page 78.]

[59] Y. Romano, “cqr: Conformalized quantile regression,” https://github.com/yromano/cqr, accessed: 2025-01-26. [Cited on page 107.]

A. Complementary Analysis

In this appendix, we present some aditional information to compliment our Experimental Results of Chapter 5.

Our presentation strategy emphasizes both individual method characteristics and comparative performance across the three case studies. For each method, we examine worst-case scenarios and average performance, giving some aditional remarks when necessary. The visualisations have action-influenced period highlighting (in yellow) to mark the timesteps after an agent intervention was necessary.

A.1. Case Study 1: Operating under benign conditions

Case Study 1 establishes the baseline performance characteristics of CP methods under benign operational conditions, where no transmission lines experience adversarial attacks during neither the calibration or testing phases.

A.1.1 Vanilla Method

The Vanilla method creates prediction intervals by computing either the $1-\alpha$ quantile of all calibration residuals for each transmission line (symmetric variant) or the $\alpha/2$ and $1-\alpha/2$ quantiles (asymmetric variant).

The analysis reveals that in chronic 931, line 13 exhibited the poorest coverage performance, as demonstrated in Figure A.1.

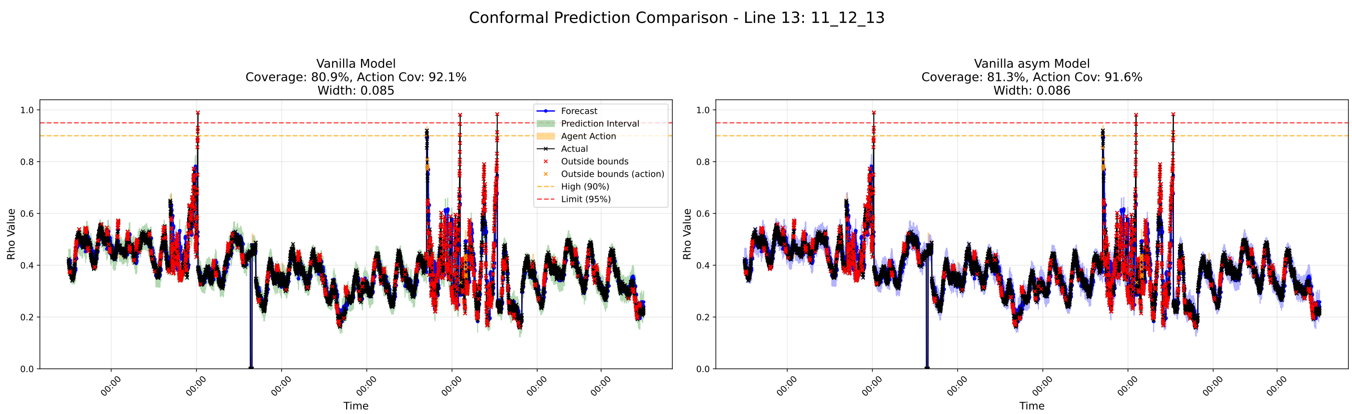

Figure A.1: Case Study 1 Vanilla method: Coverage performance for line 13, the worst performing line in the Vanilla method. The worst periods for coverage can be seen in the regions where red is more predominant, which are regions where the ρ value changes abruptly.

While lines 10 and 13 struggle to achieve perfect coverage, the method demonstrates reliable performance on all other lines. Figure A.2 illustrates the typical performance characteristics, showing line 7, an example of what happens in the rest of the lines, which meet the coverage requirements.

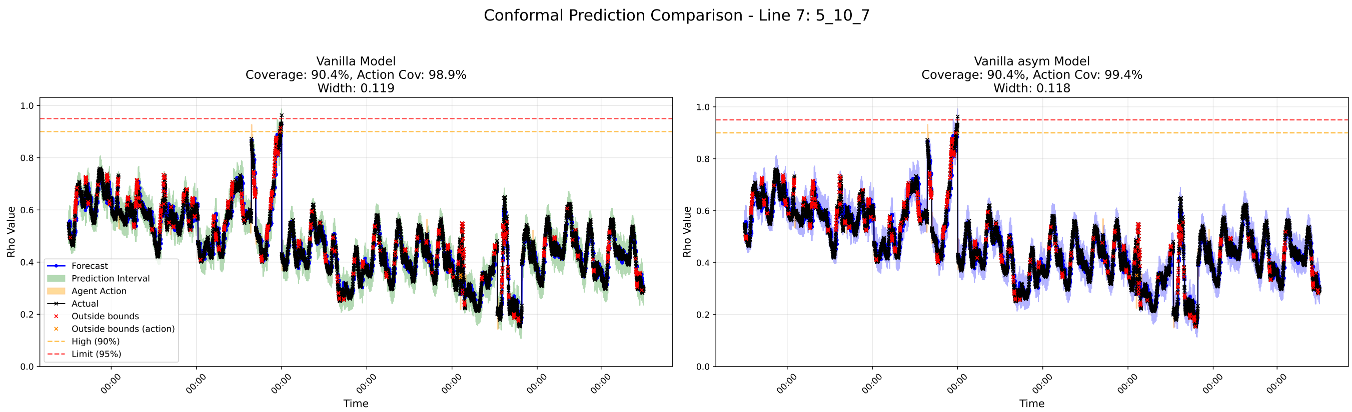

Figure A.2: Case Study 1 Vanilla method: Line 7, an example of the representative coverage performance across the remaining transmission lines. In Case Study 1, most lines present this behaviour.

A.1.2 k-Nearest Neighbors

The k-NN method introduces the concept of local uncertainty, no longer using the single quantile per line approach, but using the most similar past scenarios to construct the quantiles. In doing so, we expect the method to be more adaptive, providing an alternative to the Vanilla simple approach. As detailed in Subsection 3.4.2 and Subsubsections 3.4.2.1 and 3.4.2.2, the k-NN approach identifies the k most similar operational states from the calibration dataset and constructs intervals based on the local uncertainty characteristics of these neighbors. In Case Study 1, line 13 again represented the most challenging scenario for coverage achievement (Figure A.3).

The normalized variant, introduced in Subsection 3.4.3 and inspired by Article [3], tries to improve this idea, by scaling nonconformity scores according to predicted uncertainty levels. This normalization step aims to improve conditional coverage by adjusting residual magnitudes to their expected variability. Unfortunately, our analysis concluded that the variant does not offer any real advantages in Case Study 1 as we can see in Figures A.3 and A.4, and has the disadvantage of being even more computationally expensive.

In what concerns the symmetric vs asymmetric implementations, we also observe that none present a clear advantage over the other, achieving very similar coverage levels and efficiency.

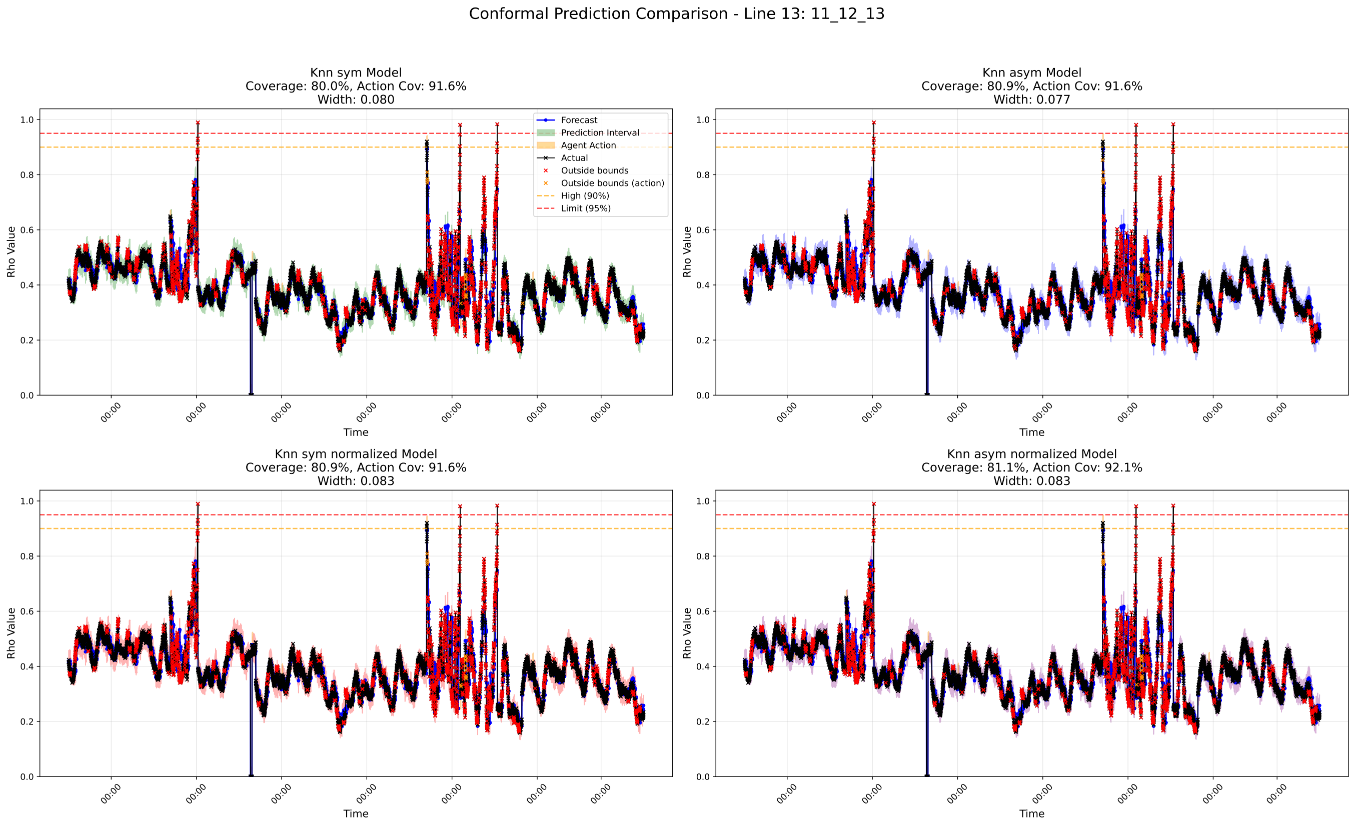

Figure A.3: Case Study 1 k-NN method: Coverage performance for line 13, the line presenting the worst coverage. The regions where ρ changes more abruptly have the worst coverage.

A.1.3 Adaptive Conformal Inference (ACI)

The ACI method introduces temporal adaptivity to CP, and enables the dynamic adjustment of interval widths based on the most recent coverage performance, being very effective for time series, which is clear when comparing Figure A.5 with the corresponding Vanilla results in Figure A.1. Under identical conditions, the ACI method manages to achieve proper coverage in every line.

However, the ACI method has a problem that must be considered. When the method experiences consecutive prediction failures, resulting in high violation rates, interval width spikes occur. These spikes, while mathematically justified by the update rule, can be a problem because they can cause predictions to exceed critical operational thresholds (such as $\rho_{limit} = 0.95$), specially because the system does not seem to often suddenly increase the ρ values above the ρ_{limit}.

This spike phenomenon is not unique to our implementation, as evidenced by similar patterns in the MAPIE library's documentation [56, 57], illustrated in Figure A.7. This

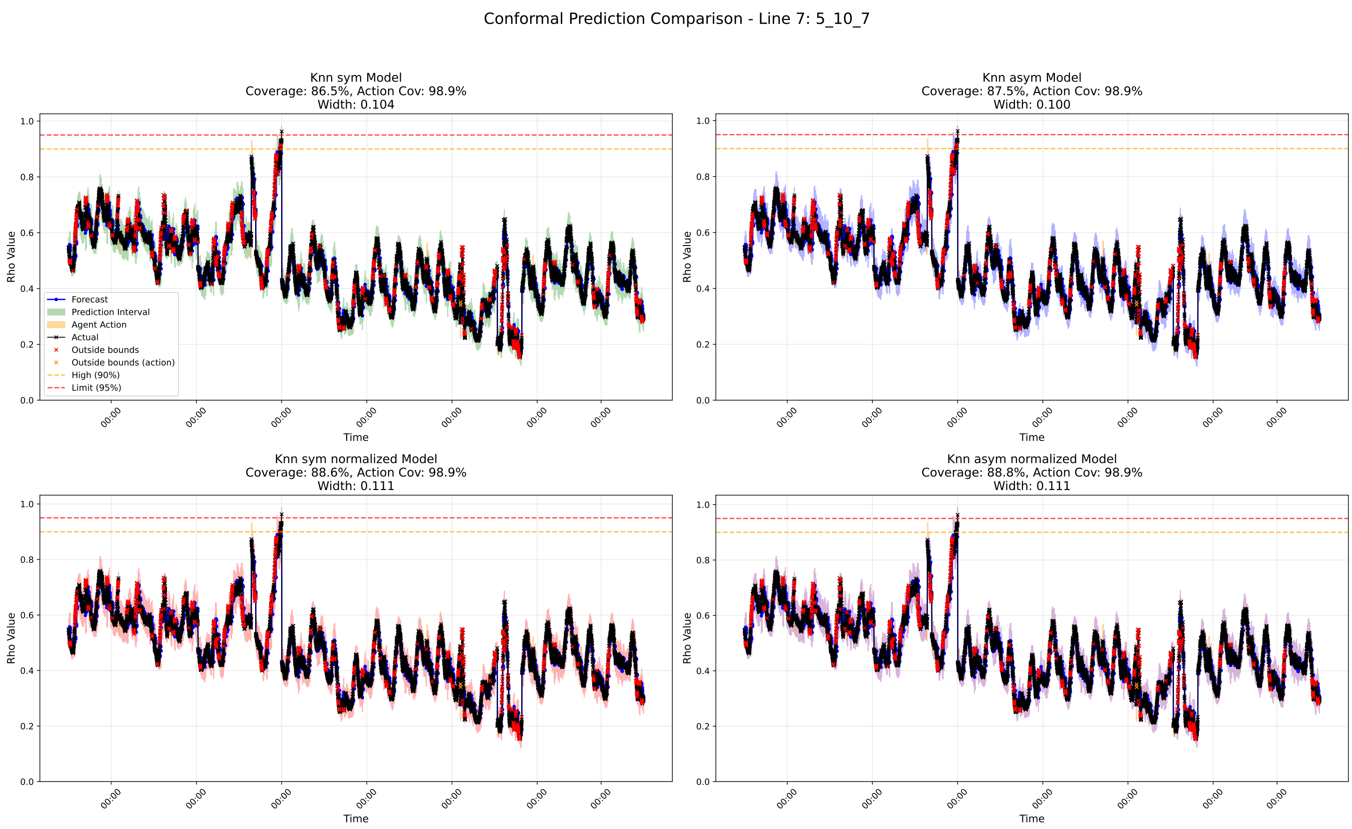

Figure A.4: Case Study 1 k-NN method: Showing the performance of the k-NN method on line 7 for symmetric and asymmetric variants (including normalized versions). We can observe some adaptivity, with the intervals increasing and decreasing in size, although this difference is more apparent in the following Case Studies.

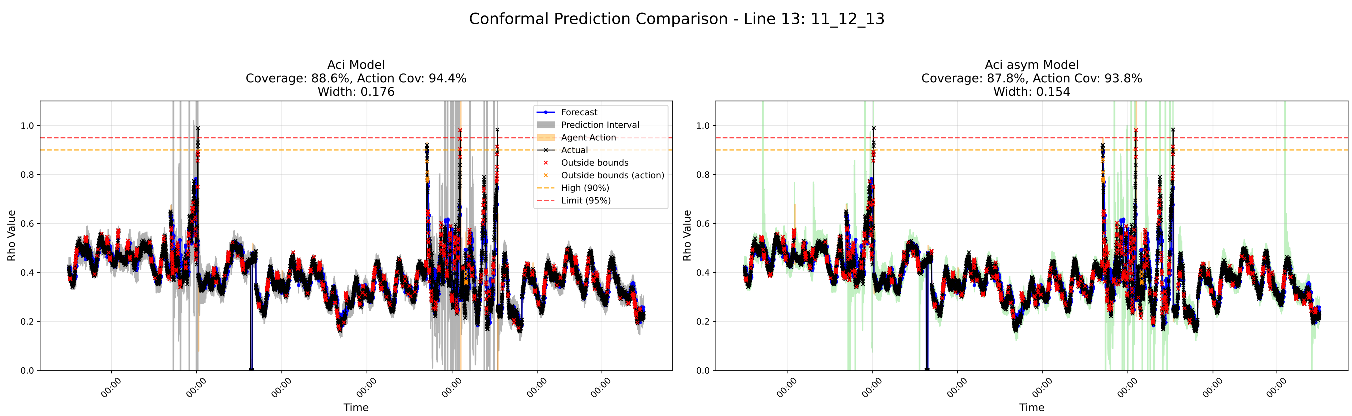

Figure A.5: Case Study 1 ACI method: Coverage performance for line 13 (almost 90% coverage), demonstrating the improvement achieved compared to the previous methods. We also see the introdution of the spike behaviour in the regions of high uncertainty.

consistency across implementations confirms that the behaviour is due to the ACI update rule rather than an error in our implementation. The spike behaviour originates from the γ value choice: larger values of γ lead to more adptive but less stable intervals, and lower values lead to more stable but less adaptive intervals. In [58] the authors comment that if the if the learning rate (γ) is "'... too large, then the intervals will oscilate widely". This is a problem because there is not a right value of γ, since this changes thoughout each episode, although there are more recents methods like the AgACI [32] and DtACI [33] that try to improve on this behaviour by trying to also learn the value of γ, but due to time constraints they were not explored in this thesis. We chose $\gamma = 0.05$, a value normally used in Literature.

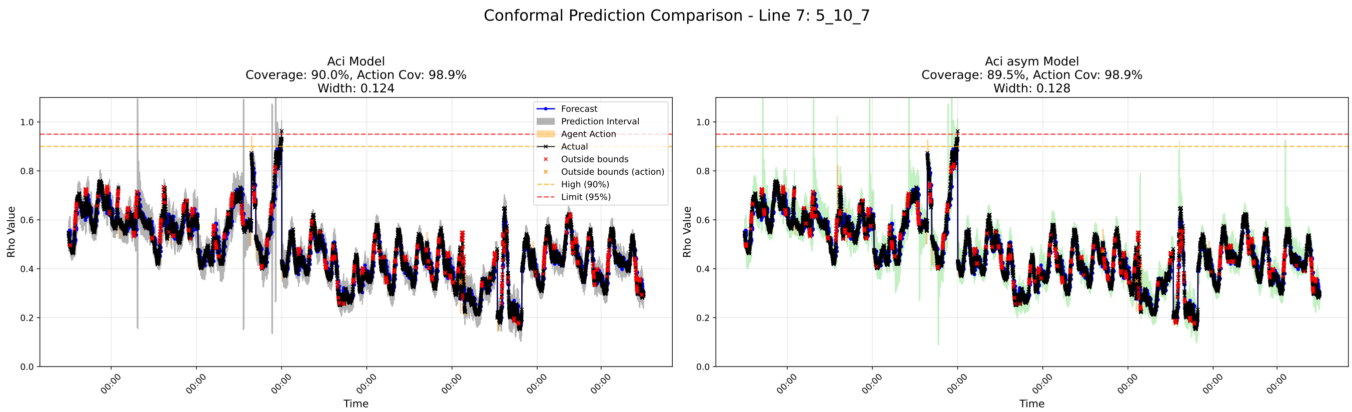

Figure A.6: Case Study 1 ACI method: Line 7 showing the typical coverage performance with visible spike occurrences.

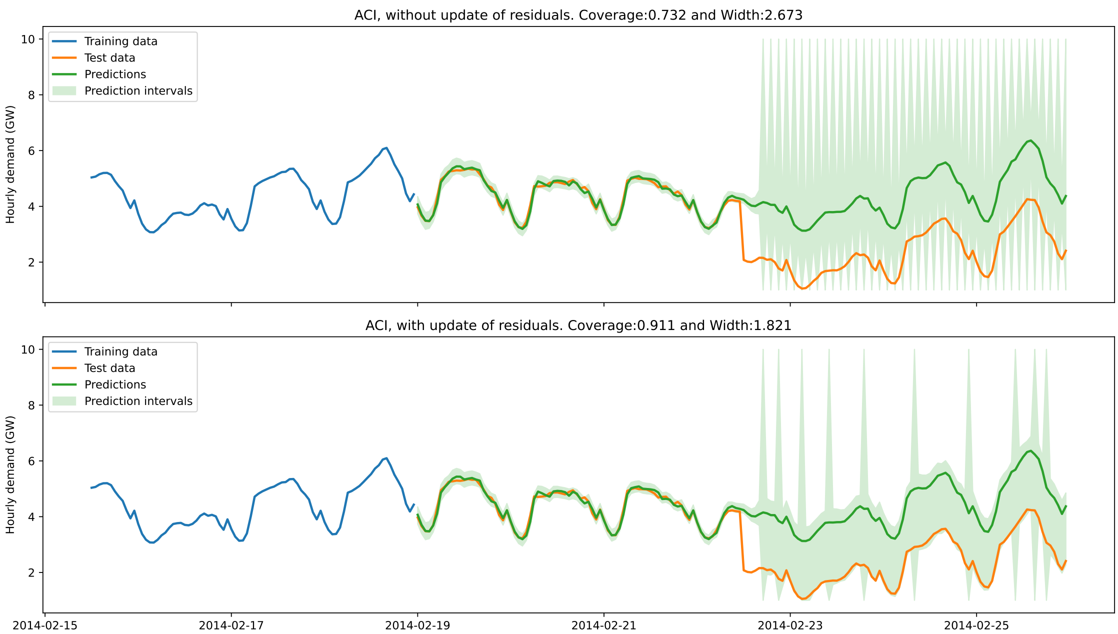

Figure A.7: External validation: MAPIE library ACI implementation example, showing similar spike phenomena, confirming that this behaviour is due to the adaptive conformal inference approach rather than implementation-specific.

Figure 5.1 demonstrates that virtually all lines achieve coverage levels very close to or exceeding the required 90% threshold, establishing the method's reliability for coverage guarantee maintenance.

The action-influenced time step analysis, presented in Figure 5.2, maintains consistency with the overall coverage patterns. The slight distributional differences reflect the limited sample size of agent interventions which is more characteristic in stable operating environments.

Figure 5.3 showed that the efficiency of the ACI method is comparable to the Vanilla method, which combined with better coverage performance, shows the importance of the adaptive intervals when we are dealing with time series.

The forecast horizon behaviour, illustrated in Figure 5.4, reveals that coverage deteriorates with increasing forecast distance, as expected, but the interval width exhibits approximately constant behaviour across horizons. This counterintuitive result is due to our specific forecasting implementation described in Section 3.1. The constant width stems from our 12-step-ahead forecasting procedure Figure 3.2 where the ACI update rule 3.10 applies to each individual step within the sequence, immediately before starting a new forecast. Then, when a new forecast is started, the interval width reflects the most recent delayed update from the previous sequence, creating consistent widths across the subsequent 12-step prediction horizon.

A.1.4 Residual-Based Conformalised Quantile Regression

The RB-CQR method uses machine learning backends to directly estimate residual quantiles rather than relying on residual-based calibration. It is an idea adapted from the CQR method [22], in which we estimate the distribution of the residuals resulting of our forecaster's predictions, instead of trying to model the target distribution directly. This was necessary because we have the requirement of using a pre-trained forecaster which makes point forecasts (future load and generation values) in order to obtain the line capacity (ρ value) by running a powerflow.

In the worst performing line, presented in Figure A.8, we observe unsatisfactory behaviour: the standard RB-CQR implementations struggle to achieve adequate coverage even under the benign conditions of Case Study 1. The upper plots show poor performance on all machine learning backends. In the lower plots, we add the ACI update rule to the models, which allows them to achieve good performance, even in the worst line. But it also introduces the same spike phenomenon presented in the previous section.

In our analysis, we identified the Random Forests as performing best among the three backends. A more detailed analysis of this variant is presented in Figure A.9, confirming

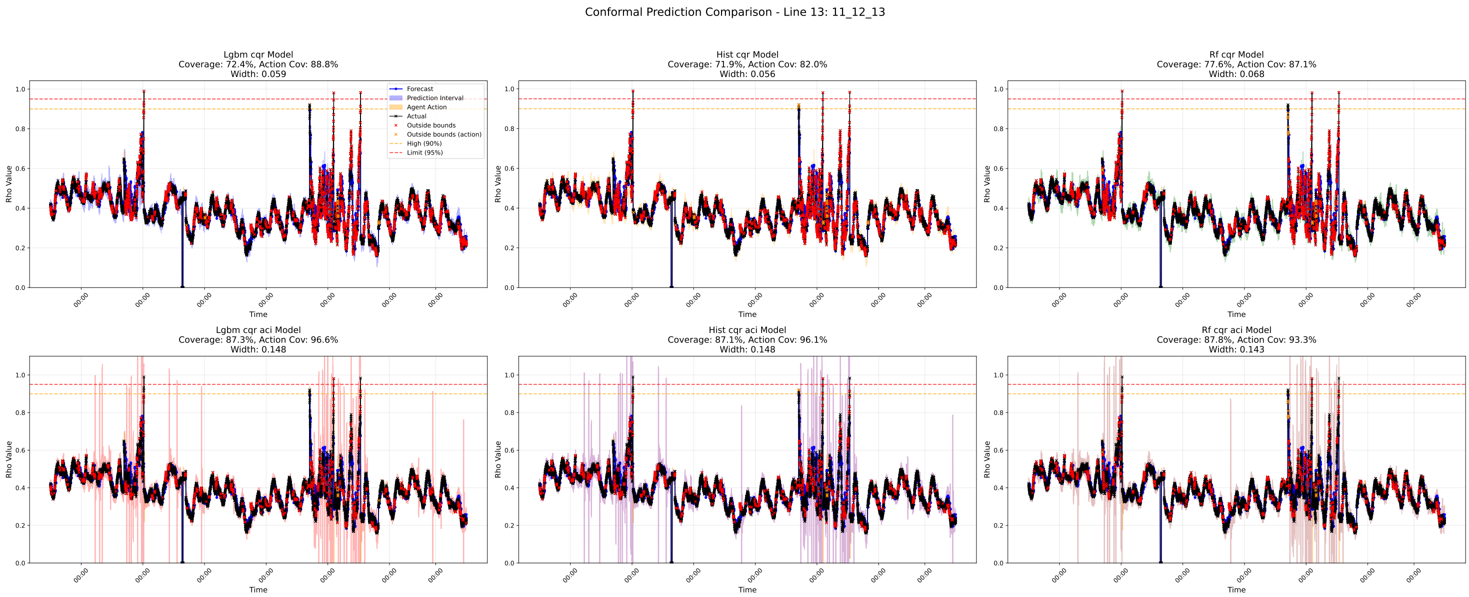

Figure A.8: Case Study 1 RB-CQR method: Showing line 13, the worst performing line. In the upper plots, we see the base RB-CQR methods and in the lower plots we introduce the ACI update rule, which shows the same spike behaviour, but improves the coverage to the required levels.

that even using Random Forests, the base RB-CQR method fails for the same lines and does not achieve coverage.

Figure A.9: Case Study 1 RB-CQR with Random Forests: Summary statistics demonstrating that even the most promising machine learning backend fails to achieve coverage in most power lines.

When introducing the ACI update rule 3.10 we solve the coverage deficiency, but once again introduce the same spike phenomenon observed in the ACI method. Figure A.10 demonstrates this using line 16 as an example. This spike behaviour is not universal across all transmission lines. Figure A.11 shows line 6 as an example of good behaviour, where the combination achieves both coverage and interval stability, because these spikes only tend to happen in regions of higher uncertainty, for example when the distribution shifts.

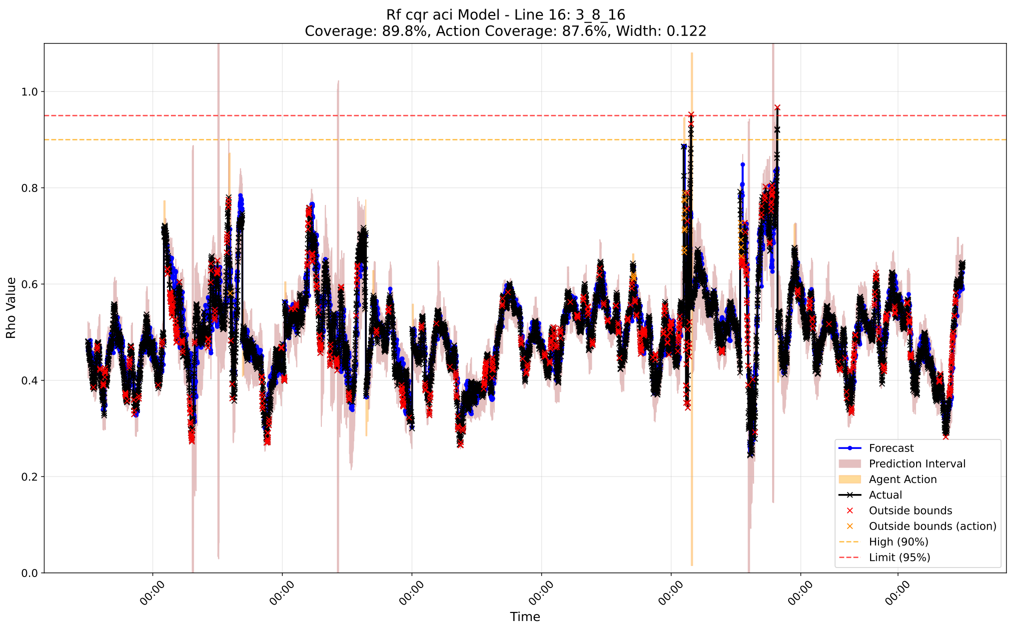

Figure A.10: Case Study 1 RB-CQR with Random Forests and ACI update rule: Coverage is attained, but some lines (like line 16), also show the spike behaviour of the ACI update rule.

Figure A.12 presents the summary statistics of using the Random Forests backend with the ACI update rule, the most effective RB-CQR implementation in Case Study 1. It achieves coverage levels comparable to the ACI method, but provides better efficiency.

A.1.5 Inverted Residual-Based Conformalised Quantile Regression

When analysing the Inverted method, we see the effect of its conservatism. Figure A.13 shows line 13, in which the base RB-CQR method failed to achieve coverage, and the inverted variant easily succeeds. All the backends achieve coverage levels above the nominal target, demonstrating conservative estimation. In this context, the ACI update rule works differently as we can see from the comparison between upper plots (standard performance) and lower plots (with ACI enhancement): rather than addressing coverage violations, it optimizes interval width, improving efficiency without introducing spikes. This

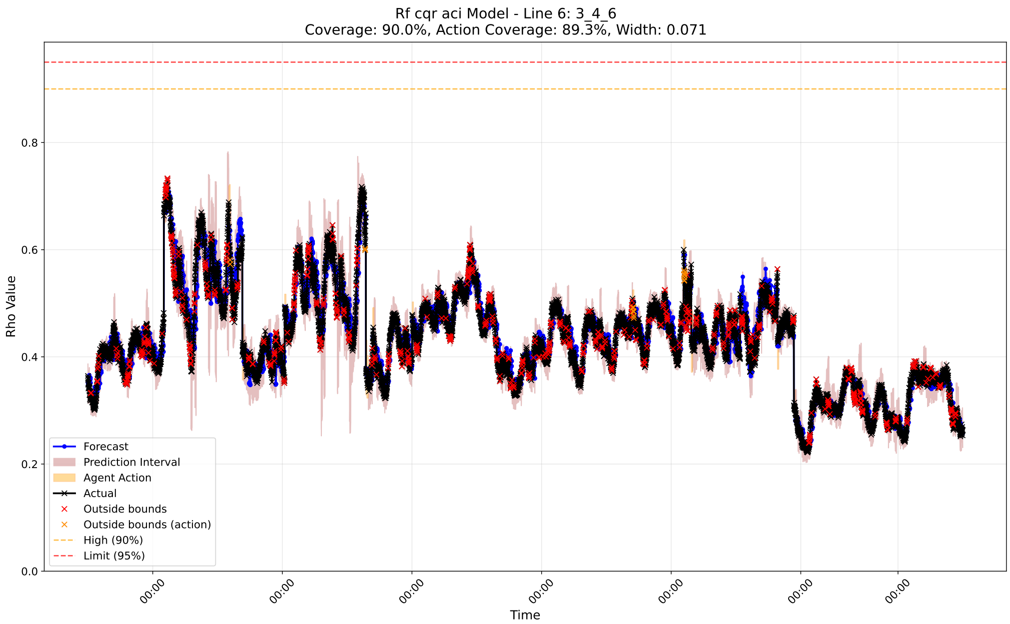

Figure A.11: Case Study 1 RB-CQR with Random Forests and the ACI update rule: Example of good performance in line 6, showing good coverage, efficiency, and little spike behaviour.

Figure A.12: Case Study 1 RB-CQR with Random Forests and the ACI update rule: This variation shows coverage comparable to the ACI method, but provides better efficiency.

is because the spikes are linked to coverage violations; when violations are rare, the update rule focuses on interval contraction rather than expansion.

The coverage analysis across all variants, presented in Figure 5.1, confirms the universal

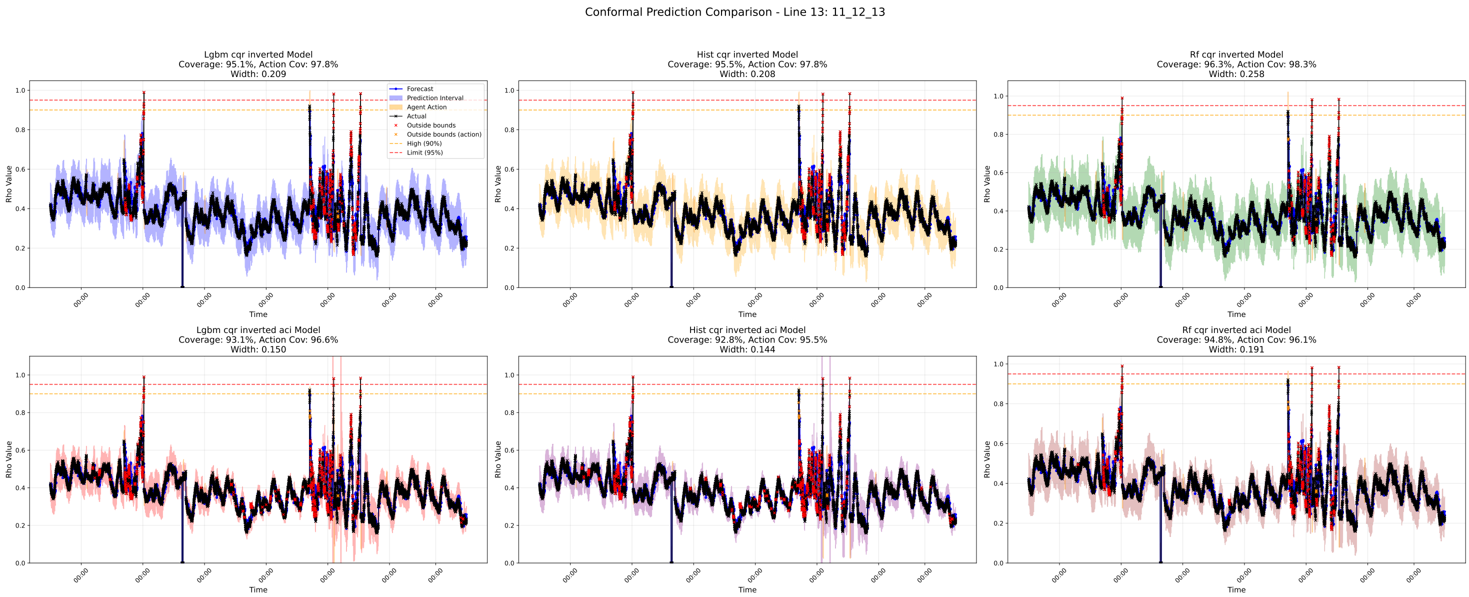

Figure A.13: Case Study 1 Inverted RB-CQR method: Demonstration of the over-performance in coverage (upper plots) and the gains in efficiency when applying the ACI update rule (lower plots), this time showing less spikes.

over-performance characteristic. All three backend implementations not only reach but also exceed the coverage threshold. This method can be considered when coverage must be guaranteed, but it must be noted that this coverage comes at the cost of the loss of efficiency. The analysis of the Random Forests backend, shown in Figure A.14, exemplifies this trade-off, and confirms our analysis. The method which previously failed to achieve coverage, now gets coverage close to 100% in all lines. But the increase in width is also significant.

Figures A.15 and A.16, show how applying the ACI update rule still maintains the good performance of the method, increasing efficiency, while avoiding the spikes seen in previous approaches.

The performance assessment of the Inverted RB-CQR method using Random Forests and the ACI update rule is presented in Figure A.17.

The forecast horizon analysis, illustrated in Figure 5.4, revealed that the Histogram Gradient Boosting backend has superior balance between coverage and efficiency compared to Random Forests. We confirm these results in Figure A.18, where the combination maintains excellent coverage while achieving better width. This analysis establishes the Inverted RB-CQR with ACI enhancement and Histogram Gradient Boosting backend as the best Inverted RB-CQR configuration. Figure A.19 shows line 16, an example of good

Figure A.14: Case Study 1 Inverted RB-CQR with Random Forests: Excellent coverage with obvious efficiency loss (interval width).

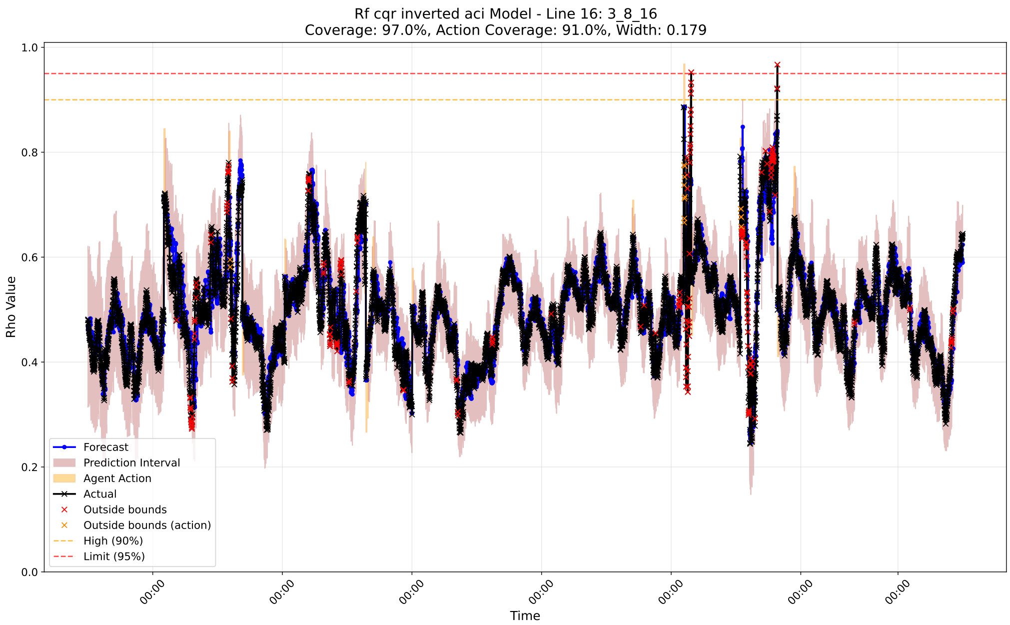

Figure A.15: Case Study 1 Inverted RB-CQR with Random Forests and ACI update rule: In this context, the ACI update rule mainly improves efficiency, therefore it does not cause frequent spikes.

performance, with excelent coverage, good interval width and absence of spikes.

The Inverted RB-CQR method can be valuable when we are interested in conservative uncertainty, although we need to again reinforce that a coverage higher than 90% is not

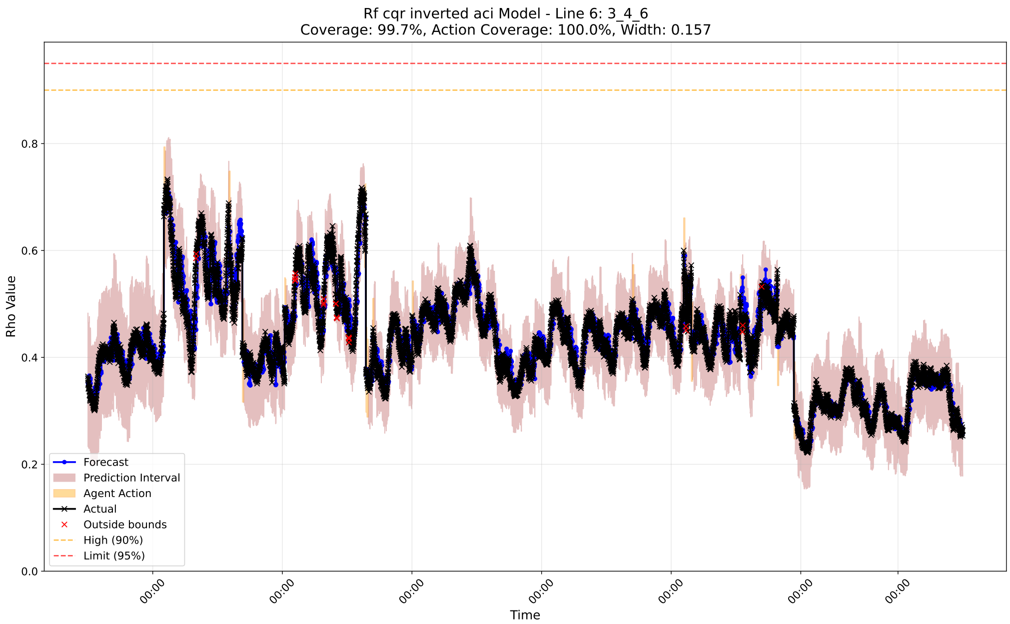

Figure A.16: Case Study 1 Inverted RB-CQR with Random Forests and ACI: Showing line 6, with high coverage and improved interval width compared to the standard inverted version, showing the good effects of the ACI update rule.

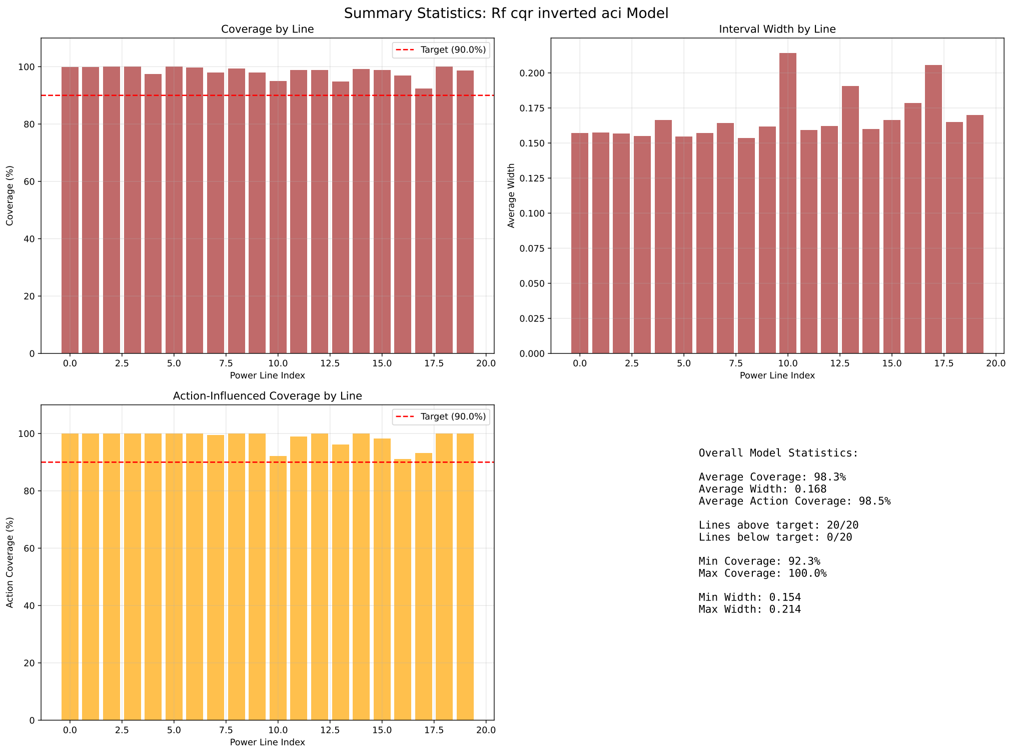

Figure A.17: Case Study 1 Inverted RB-CQR with Random Forests and ACI update rule: Excelent coverage, but sacrificing efficiency.

better than a 90% coverage. We remind the reader that the objective of CP, when using a significance level of $\alpha = 0.1$ is to obtain intervals that achieve **at least** 90% coverage. The choice between standard and inverted approaches ultimately depends on what should

Figure A.18: Case Study 1 Inverted RB-CQR with Histogram Gradient Boosting and ACI update rule: The best Inverted RB-CQR configuration, showing still excellent coverage with superior width efficiency, when compared to the other backends.

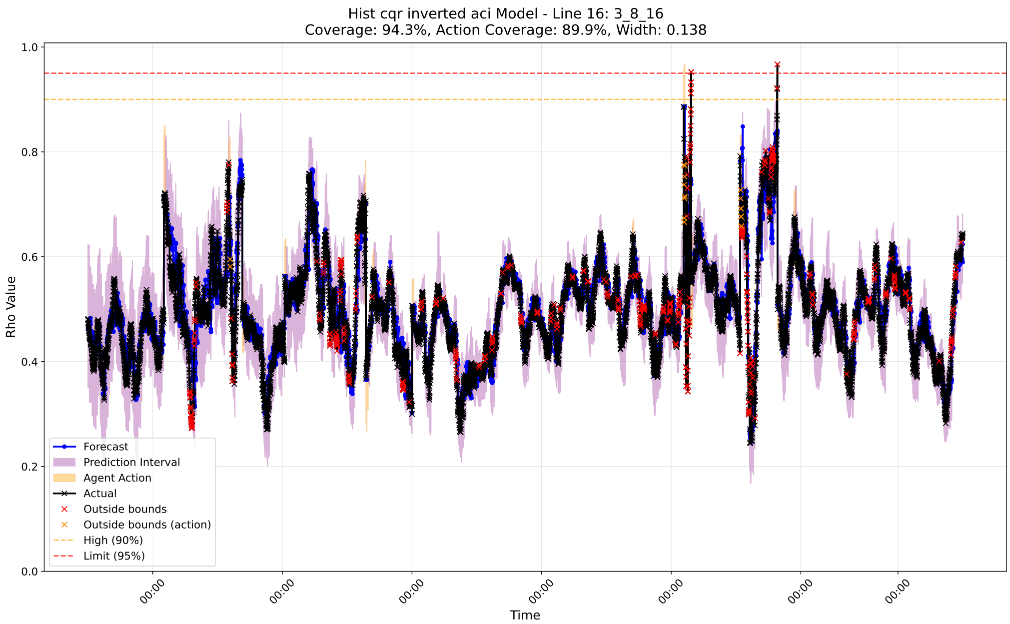

Figure A.19: Case Study 1 Inverted RB-CQR with Histogram Gradient Boosting and ACI update rule: Line 13, verifying the absence of problematic behaviour while achieving a good balance of coverage and width.

be prioritized, either over-coverage to have some additional resilience, or more efficient performance.

A.2. Case Study 2: Introduction of adversarial attacks

Case Study 2 subjects the CP methods to a stress test by introducing adversarial attacks on some lines during both calibration and testing phases: lines 3, 4, 12, 13, 14 and 15. Attacking the lines mentioned above changes the system dynamics, causing most episodes to terminate prematurely due to grid instability.

A.2.1 Vanilla

Having established the baseline Vanilla method in Case Study 1, we now examine how this approach responds to the adversarial attacks. Our analysis begins with line 10 in chronic 931, which exhibits the worst coverage performance under adversarial conditions, as shown in Figure A.20.

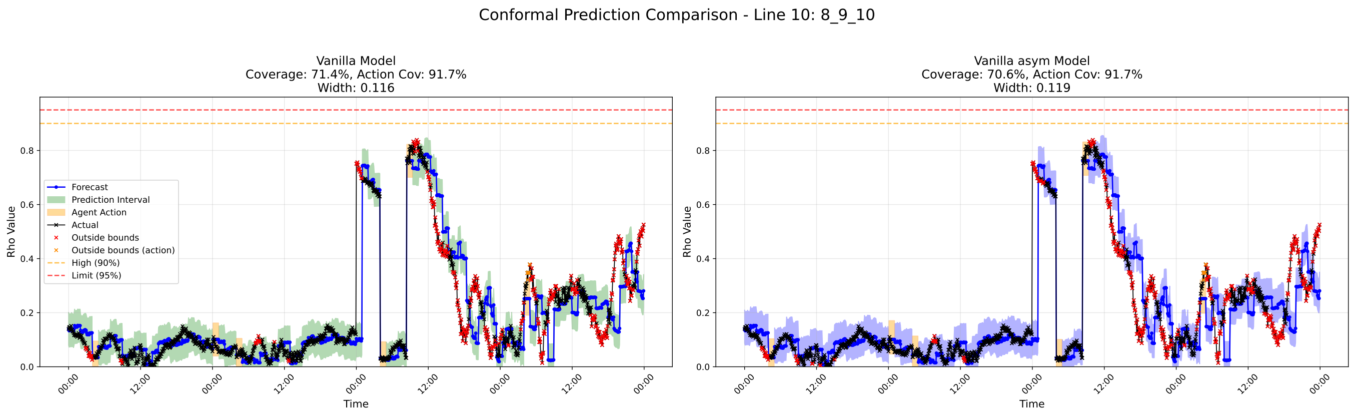

Figure A.20: Case Study 2 Vanilla method: Line 10, showing the line with the worst performance. In the regions of highest uncertainty, the forecaster fails to predict the behaviour, because it was not trained with attacks. This scenario was not representative in the calibration set, which means that Vanilla CP does not provide adequate coverage.

Despite some individual line struggles, such as line 10, the method demonstrates resilience. Figure A.21 shows a line with excelent coverage. In fact, the majority of the lines maintain acceptable performance levels.

A.2.2 k-Nearest Neighbors

The k-NN method's localized calibration strategy, which demonstrated efficiency improvements in Case Study 1's stable environment, presents worse performance under adversarial conditions. It is normal that, when adding adversarial attacks, the effects on the power grid can become even more unpredictable. So there is a scarcity of similar scenarios in the calibration dataset. Line 10 again represents the most challenging scenario, as illustrated in Figure A.22.

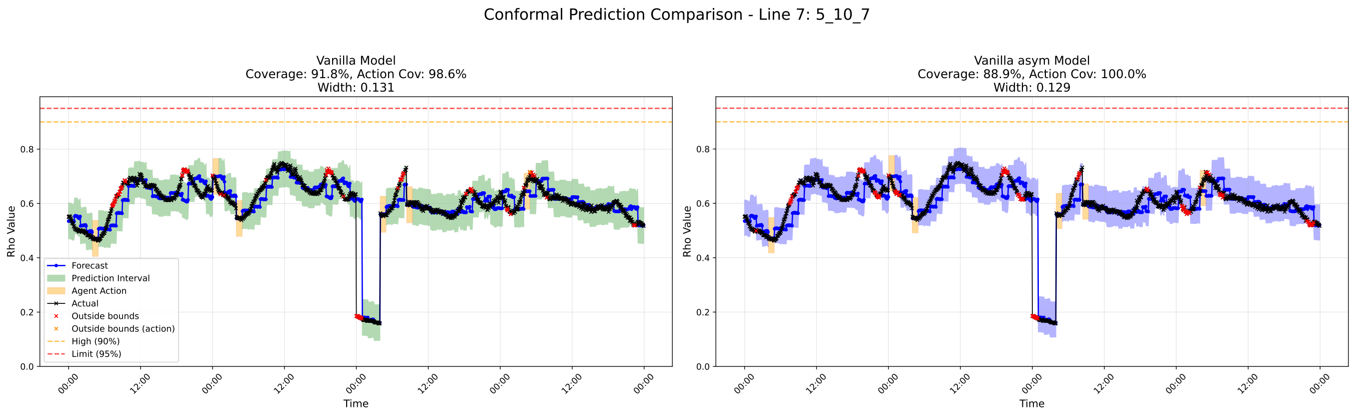

Figure A.21: Case Study 2 Vanilla method: Showing line 7, a good example of how the majority of the lines behave, in the typical scenario represented in the calibration set.

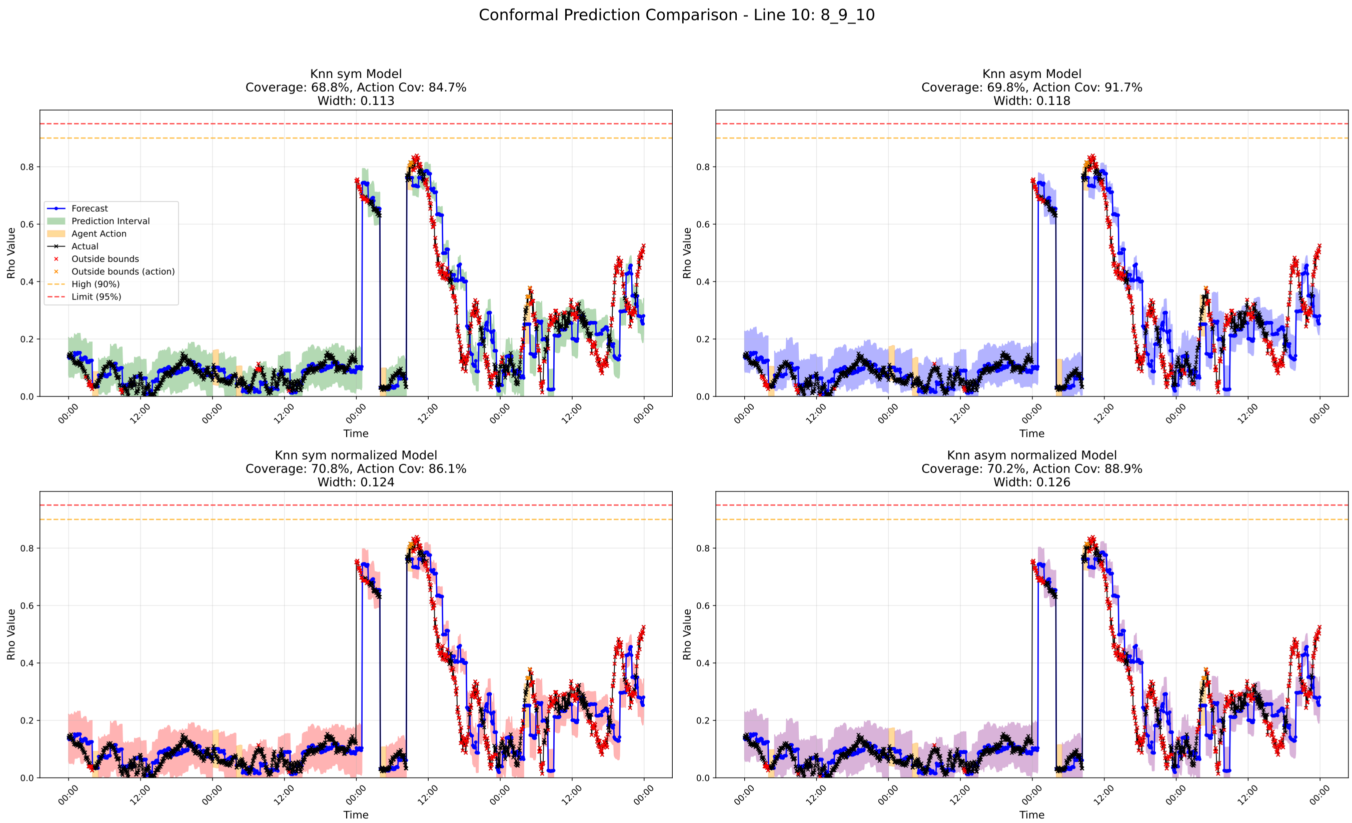

Figure A.22: Case Study 2 k-NN method: We observe 10, the worst performing line, as an example of how the method fails to achieve coverage, despite its theoretical advantages over the Vanilla method. These advantadges are not guaranteed in time-series specially under distribuition shift.

Despite these challenges, the method maintains reasonable performance in most lines, as shown in Figure A.23, achieving coverage very close to the Vanilla methdod.

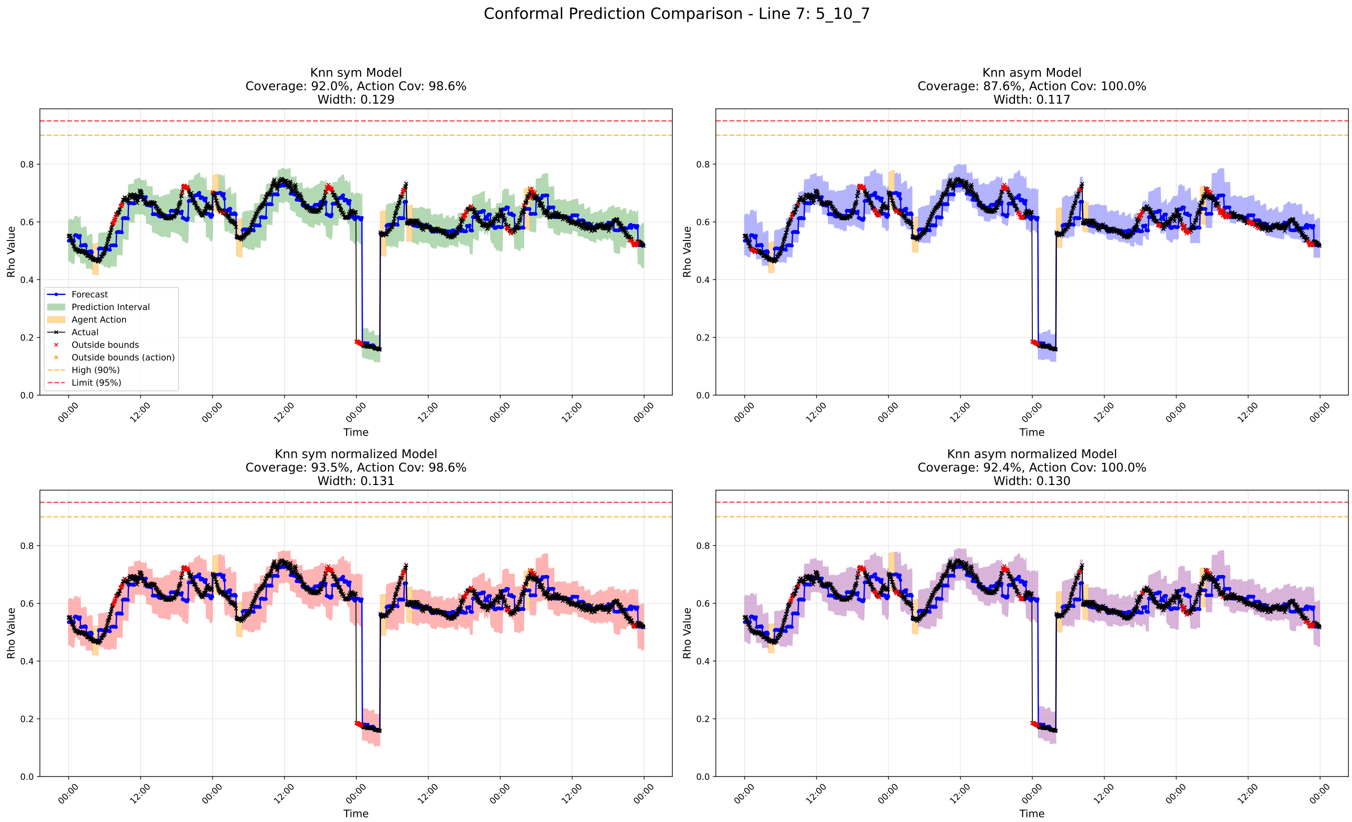

Figure A.23: Case Study 2 k-NN method: We show line 7, an example of how the k-NN method behaves in most transmission lines. We can see that it outperforms the Vanilla method in this line, and shows more adaptivity (the intervals change in size in different uncertainty regions).

A.2.3 Adaptive Conformal Inference

The ACI method ability to adjust intervals based on recent violations becomes even more important when adversarial attacks can create unpredictable changes in the grid behaviour, and therefore create higher uncertainty.

The performance advantages of this method in Case Study 1, compared to both Vanilla and k-NN implementations, translates to the adversarial scenario, which can be seen when comparing Figure A.24 with the corresponding Vanilla results. The adaptive approach responds to attack-induced violations successfully by expanding the intervals, maintaining coverage guarantees, although we still see the spikes mentioned in the previous Case Study.

In sum, the ACI approach is very well suited for adversarial scenarios, demonstrating that temporal adaptivity can effectively overcome the challenge of distribution shift between a benign forecaster and hostile operational environment.

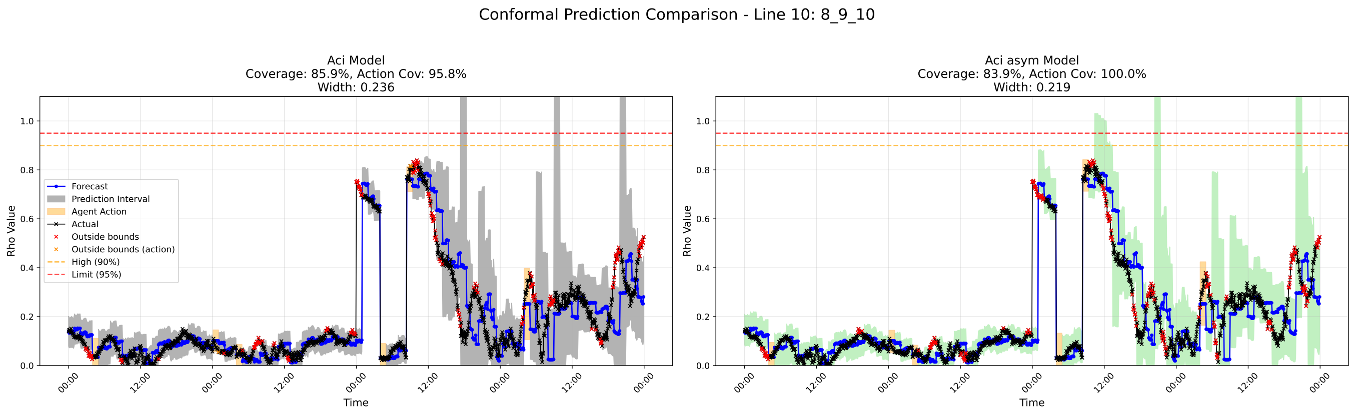

Figure A.24: Case Study 2 ACI method: Case of line 10, that demonstrates how the ACI update rule manages to increase the interval sizes when the uncertainty is high. Compared to the previous methods, the ACI is much closer to attaining coverage. We can also see the introduction of spikes in high uncertainty regions.

A.2.4 Residual-Based Conformalised Quantile Regression

The RB-CQR method's backends, which struggled to achieve adequate coverage even under Case Study 1's benign conditions, also struggle in the adversarial conditions of Case Study 2.

Figure A.25, shows the worst performing line (line 10). The upper plots reveal coverage failures that match the poor performance observed in Case Study 1, while the lower plots demonstrate, once again, the importance of the ACI enhancement for the utility from these backends in Case Study 2.

Figure A.26 shows line 8, in which we can also see the effects of using the ACI update rule. The coverage is improved, and since this line does not present high uncertainty, the interval widths only increase a little.

The backend comparison analysis, which was presented in Figure 5.5, determined that the Random Forests backend outperformed the other backends, which was also the prefered method in Case Study 1, but even this backend cannot achieve good performance independently under adversarial stress, as detailed in Figure A.27. This failure reinforces the insight from Case Study 1 that the implemented RB-CQR methods perform poorly without the adaptivity brought by the ACI update rule.

We verify that the ACI enhancement once again addresses the coverage deficiencies but it also introduces the same spike phenomenon observed in Case Study 1, as can be observed in Figure A.28.

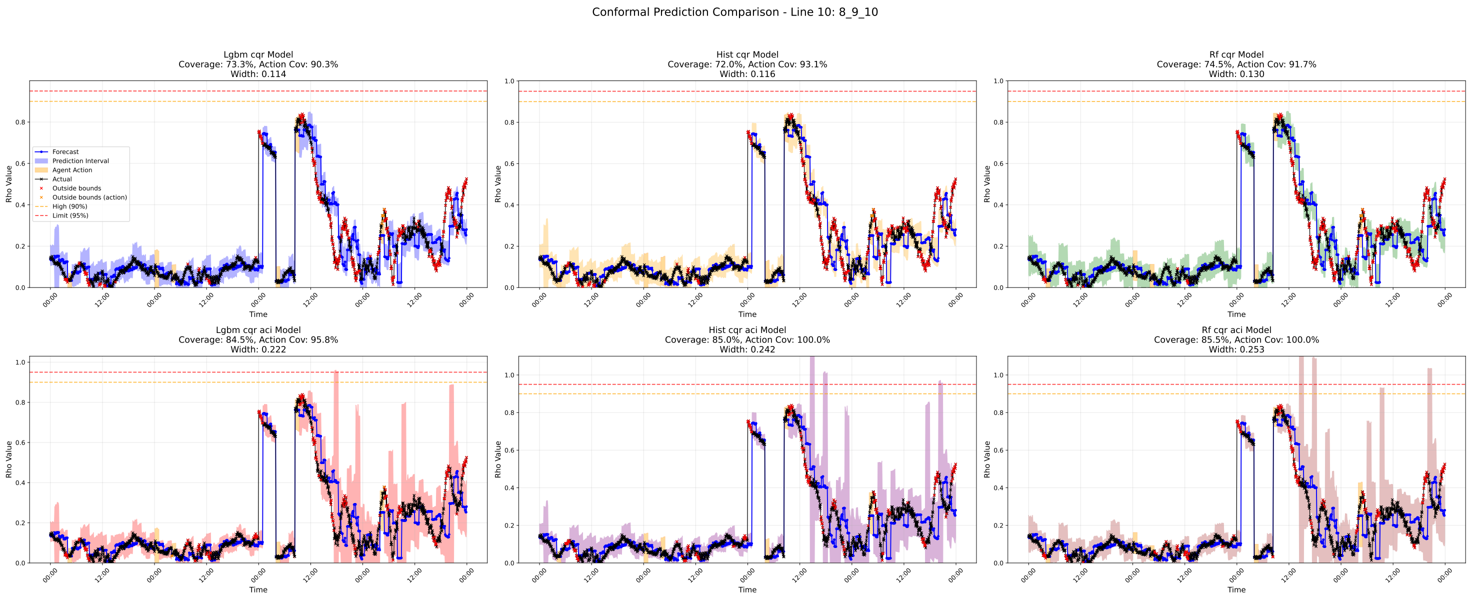

Figure A.25: Case Study 2 RB-CQR method: Machine learning backend performance, again showing the same patterns. Without the ACI update rule (upper plots), all these backends fail to achieve coverage.

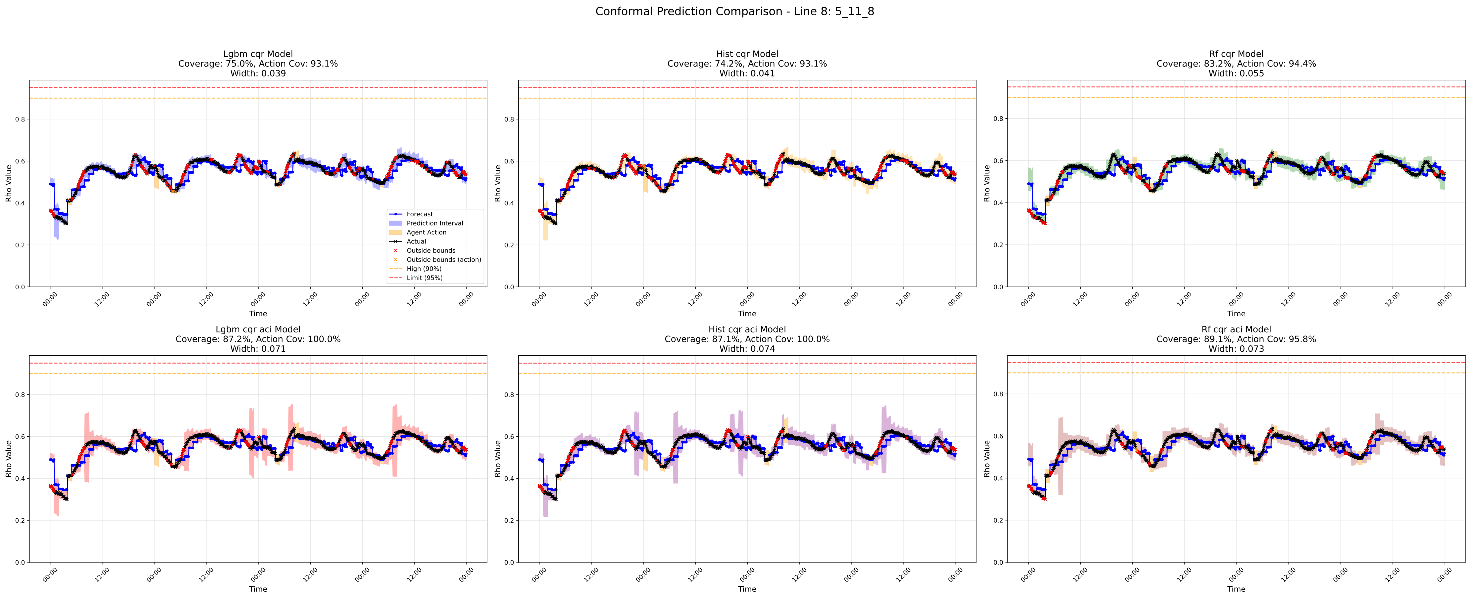

Figure A.26: Case Study 2 RB-CQR method: Example of line 8, showing how the ACI update rule (lower plots) solves the methods' coverage deficiencies.

Fortunately, we notice stable behaviour present across most transmission lines. Line 6 is an example, as shown in Figure A.29.

The overall performance assessment, presented in Figure A.30, establishes this configuration as the optimal machine learning-based approach for adversarial scenarios, although when using the ACI update rule, all backends show similar results. While matching

Figure A.27: Case Study 2 RB-CQR with Random Forests: Summary statistics of the best-performing RB-CQR backend, without the ACI update rule. There are some lines that are far from the coverage objective.

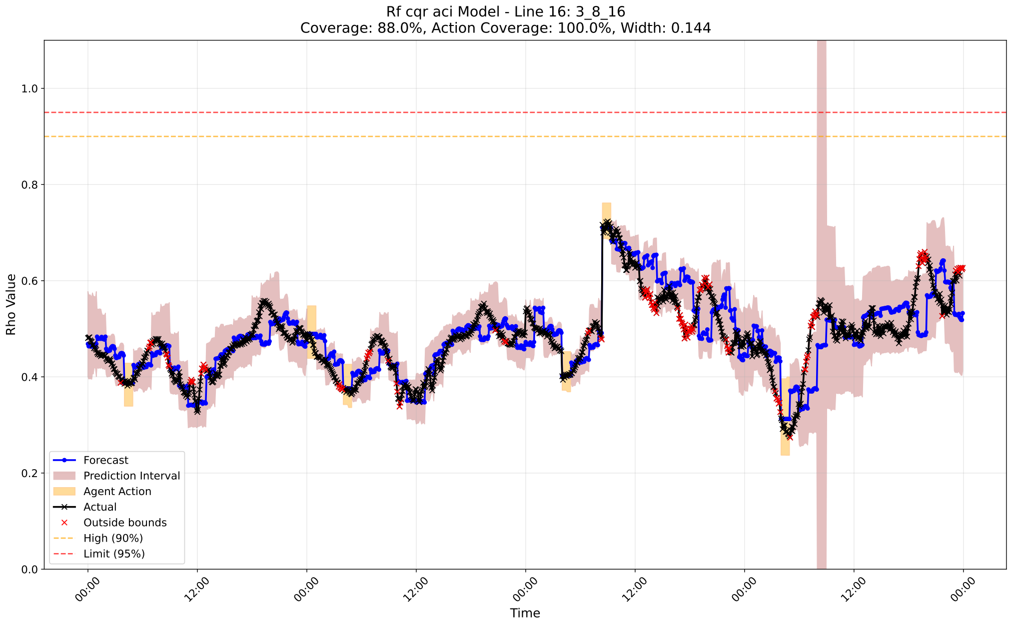

Figure A.28: Case Study 2 RB-CQR with Random Forests and ACI update rule: The ACI update rule solves the coverage issues, but it introduces some spikes in regions of higher uncertainty.

the coverage performance of the ACI method, the combination offers superior efficiency that justify the additional complexity, and also naturally handles heteroskedasticity.

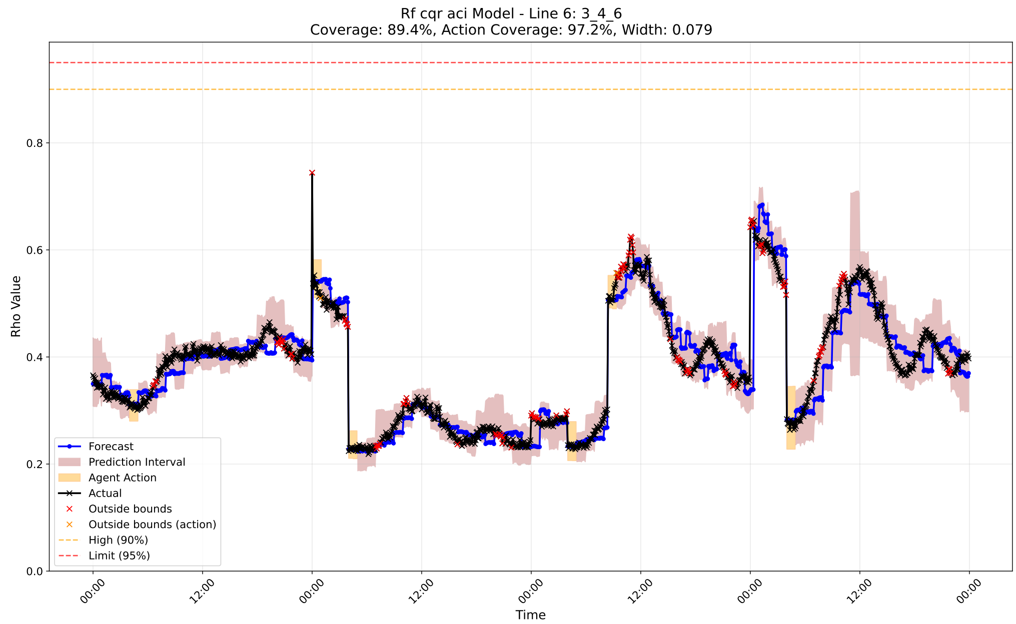

Figure A.29: Case Study 2 RB-CQR with Random Forests and ACI update rule: Line 6, as an example of this variaton's behaviour in most of the lines.

Figure A.30: Case Study 2 RB-CQR with Random Forests and ACI update rule: The best performing variant of the RB-CQR method for Case Study 2, with similar performance to the ACI method, while providing better efficiency.

A.2.5 Inverted Residual-Based Conformalised Quantile Regression

The Inverted RB-CQR method's conservative characteristics, which provided over-performance in Case Study 1's stable environment, can be even more valuable under adversarial stress, as we can see in Figure A.31. Unlike the standard RB-CQR variants that struggle under adversarial stress, all Inverted models have coverage above 90%, which provides resilience against the uncertainty increase introduced with the adversarial attacks.

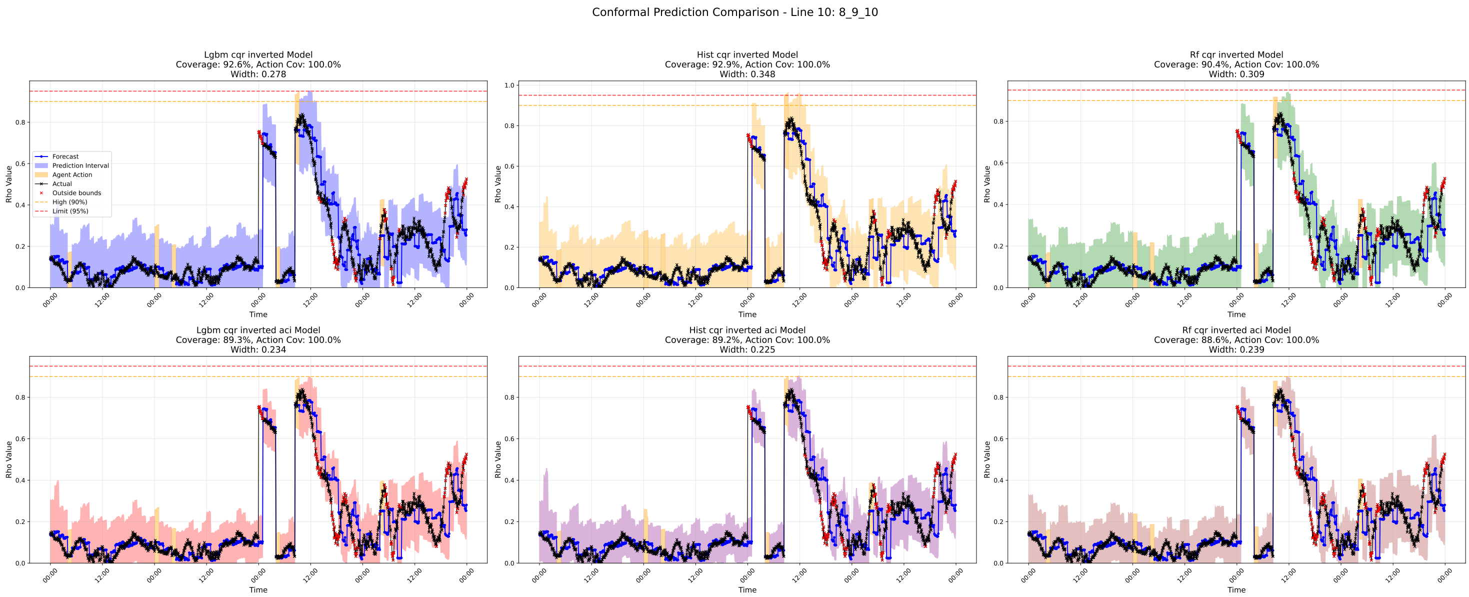

Figure A.31: Case Study 2 Inverted RB-CQR method: We present line 10, showing the method's conservative characteristics, which are alleviated by the ACI update rule (lower plots).

We detail the performance of the Random Forests variant, shown in Figure A.32, which demonstrates the method's strength in Case Study 2. The coverage achievement comes with increased interval width, but this trade-off can be valuable when coverage robustness is paramount.

In Figures A.33 and A.34 we see that the application of the ACI update rule improves the efficiency, maintains coverage, and does not introduce the spike behaviour because instead of adressing converage violations (which now are more rare), the adaptive mechanism only has to decrease interval width.

The summary statistics illustrated in Figure A.35, establish the Inverted method with the ACI update rule well-suited for adversarial scenarios. The combination achieves coverage levels that provide some safety margins although the efficiency is not ideal.

The forecast horizon analysis in Figure 5.8 showed that the Histogram Gradient Boosting variant performs better than the Random Forests (in the Inverted case), while providing similar coverage. We confirm this by presenting Figure A.36, and validate it with line 13 in Figure A.37, which confirms the absence of spikes while achieving a better balance of

Figure A.32: Case Study 2 Inverted RB-CQR with Random Forests: Conservative over-performance in almost all lines, with clear efficiency disadvantages.

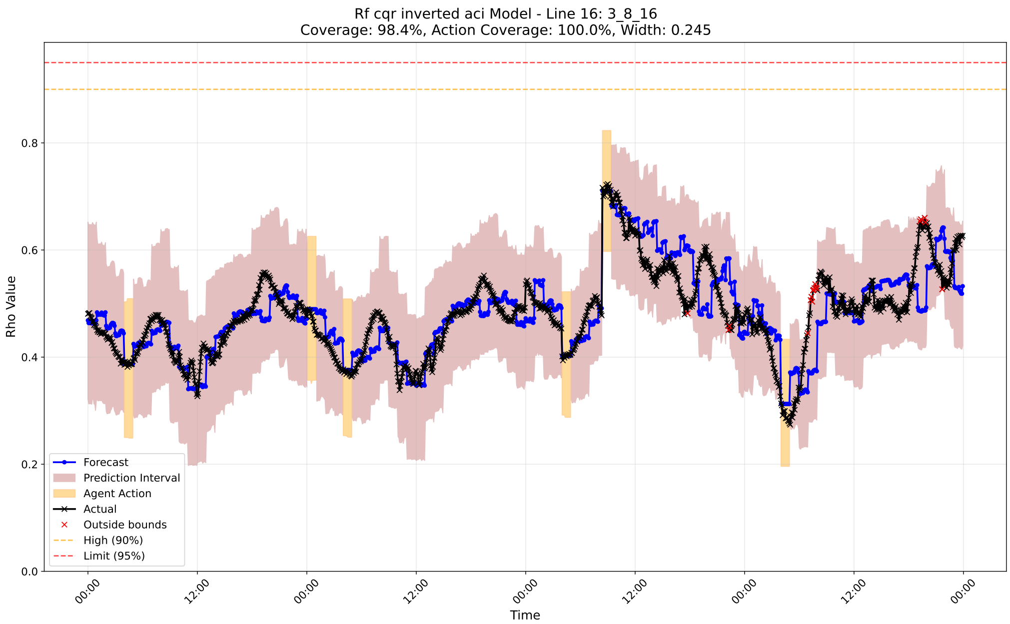

Figure A.33: Case Study 2 Inverted RB-CQR with Random Forests and ACI update rule (line 16): We see the efficiency increase provided by the ACI update rule, while still achieving excellent coverage. We note that even with the update rule, the efficiency is still reduced compared to the non-inverted methods.

conservative coverage with enhanced efficiency. This evaluation establishes the Inverted RB-CQR method with Histogram Gradient Boosting backend and ACI update rule, as a

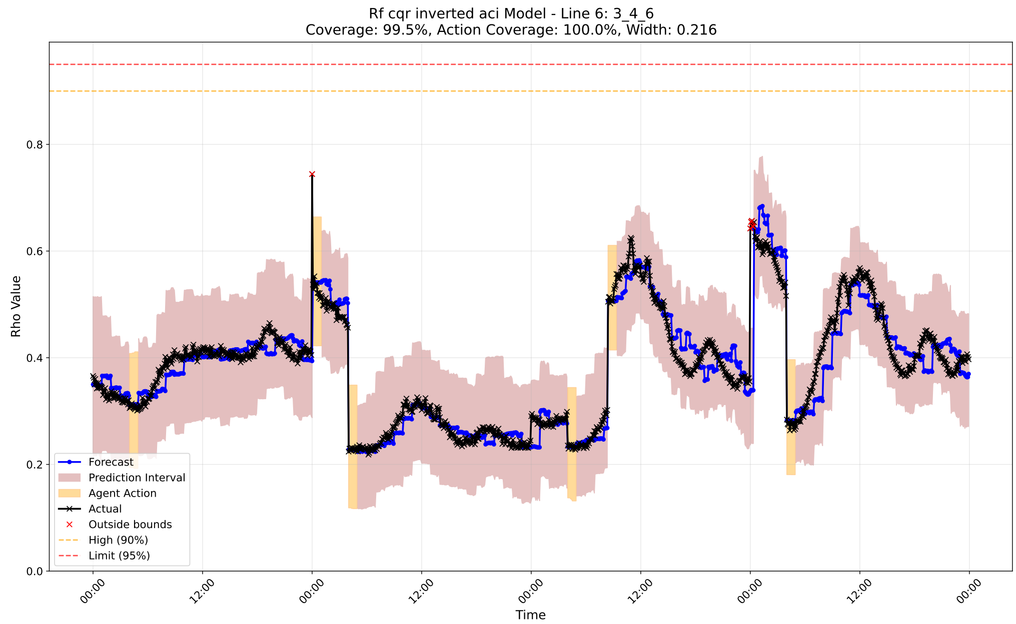

Figure A.34: Case Study 2 Inverted RB-CQR with Random Forests and ACI update rule: Line 6, another example of the ACI update rule advantages.

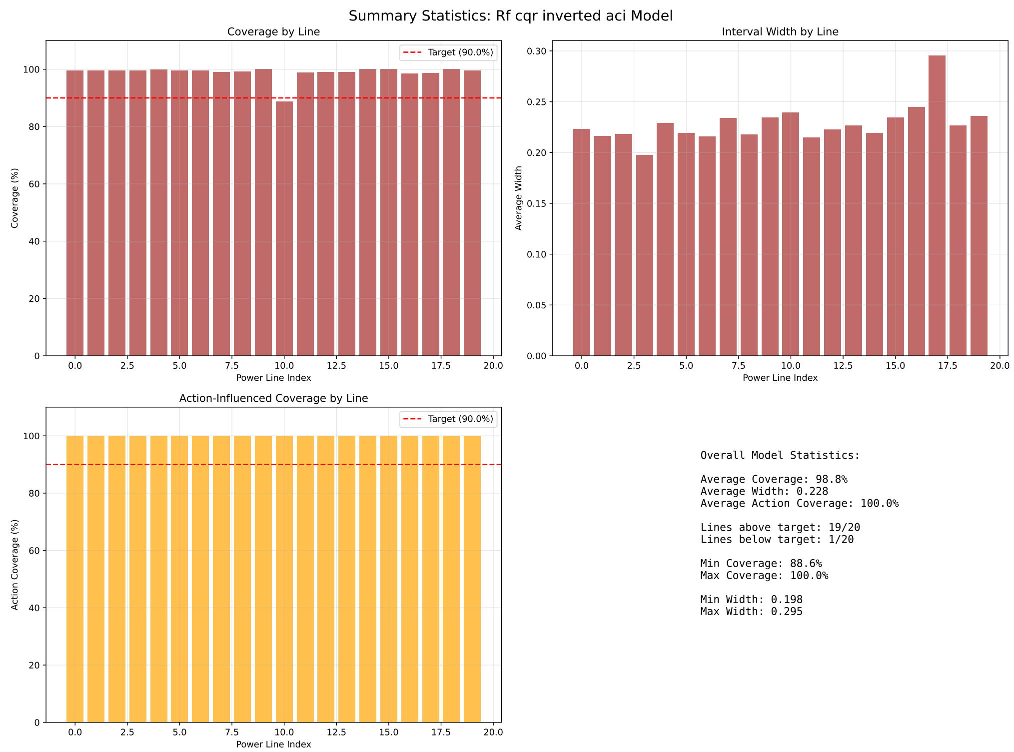

Figure A.35: Case Study 2 Inverted RB-CQR with Random Forests and ACI update rule: Excellent coverage with the tradeoff being the lack of efficiency

candidate to best performing method, since it provides excelent coverage, does not suffer from the spike behaviour we saw in other methods, and has acceptable efficiency.

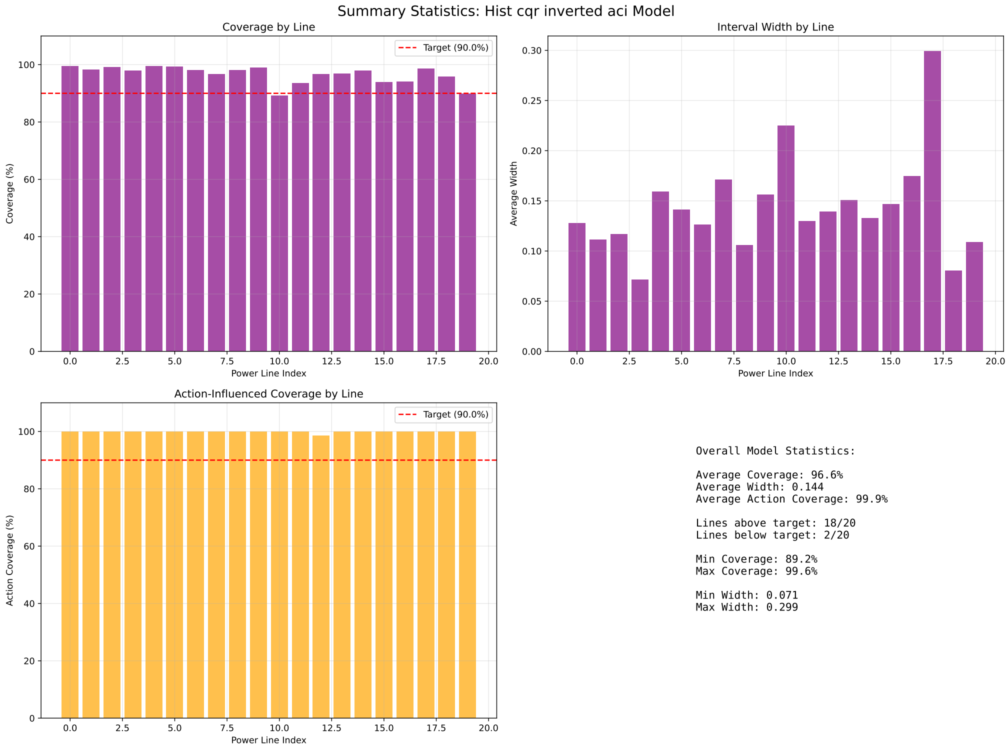

Figure A.36: Case Study 2 Inverted RB-CQR with Histogram Gradient Boosting and ACI update rule: This variant also shows great coverage, but with superior efficiency when compared to the Random Forests backend.

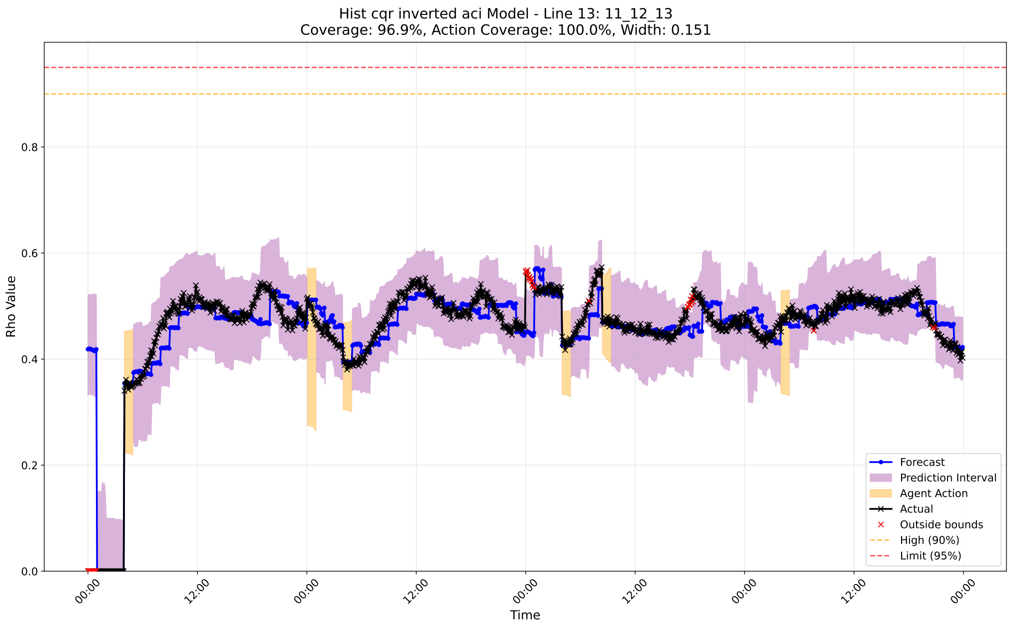

Figure A.37: Case Study 2 Inverted RB-CQR with Histogram Gradient Boosting and ACI update rule: An example on line 13, showing the method's conservative performance with excellent coverage, and acceptable efficiency.

A.3. Case Study 3: The Ensemble Approach

Case Study 3 introduced the ensemble strategy that consists of 20 specialized models, each trained under a single-line attack scenario. Model 0 experiences adversarial attacks exclusively on line 0 while maintaining normal operations on all other transmission lines, Model 1 faces attacks solely on line 1, and this pattern continues for the whole network.

During the testing phase, the same coordinated multi-line attacks from Case Study 2 are executed, targeting lines 3, 4, 12, 13, 14, and 15, and use the worst-case out of all the 20 models, as explained in Section 5.3. The idea stems from the fact that there is no prior knowledge of when or in which lines adversial attacks are bound to happen, so we must prepare for the worst.

The visualisation methodology remains consistent with the previous Case Studies, where yellow intervals denote agent intervention periods with subsequent 12-step action-influenced phases.

A.3.1 Vanilla

The Vanilla method's global quantile approach, which worked as a baseline in Case Studies 1 and 2, but failed to achieve coverage in some lines, presents good performance in Case Study 3, as we can see when we analyse line 10, the worse performing line in Case Study 2, illustrated in Figure A.38. Exactly the same episode is being ran in Cases 2 and 3, with the same lines being deterministically attacked at the same timesteps, so the difference in results is purely due to the differences between the approaches. One has a single model, in which the lines attacked during testing are also attacked during calibration, and the other uses the aforementioned ensemble strategy during calibration.

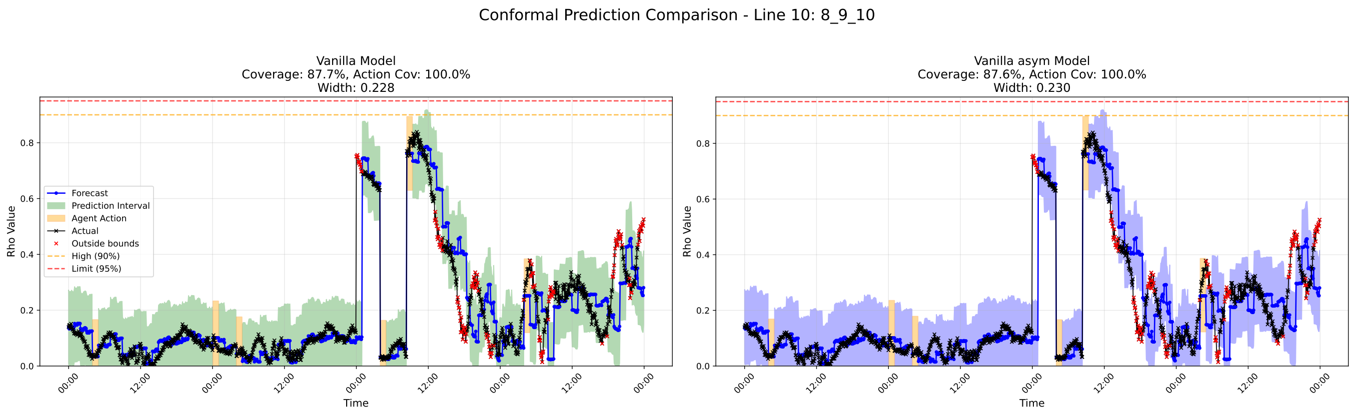

Figure A.38: Case Study 3 Vanilla method: The effects of the Ensemble strategy. Line 10, which in Case 2 was the worst performing line, now shows complete coverage.

Line 7, illustrated by Figure A.39, provides another example of the Ensemble good coverage.

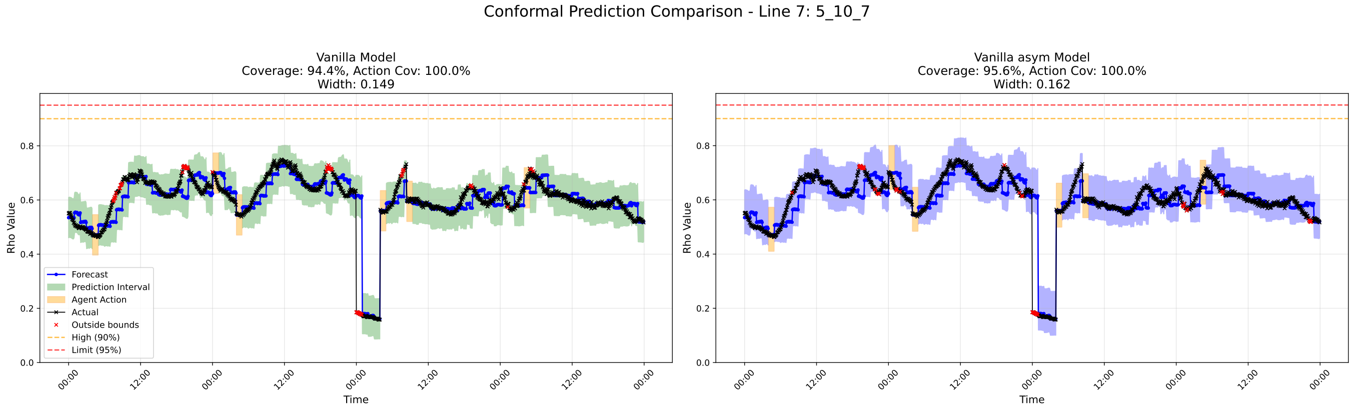

Figure A.39: Case Study 3 Vanilla method: Line 7, as another example of how the Ensemble strategy works well with the Vanilla method.

As we saw in Figure 5.9, the Ensemble approach provides coverage in all transmission lines. During action-influenced periods, we see the same pattern.

A.3.2 k-Nearest Neighbors

The k-NN method also shows superior performance in the Ensemble approach.

The most challenging scenario from Case Study 2: line 10's poor coverage performance, achieves the intended coverage in the ensemble mode, as illustrated in Figure A.40. Figure A.41 presents line 7, in which we also observe adequate coverage.

Unlike Case Studies 1 and 2, where we observed no real difference between all k-NN variants, this is not verified in Case 3. Observing Figures A.40 and A.41, we see that the normalised variants have much better efficiency, and very similar coverage.

A.3.3 Adaptive Conformal Inference

The ACI method's temporal adaptivity, which demonstrated exceptional performance in both baseline and stress-test scenarios, keeps showing the same good results in the ensemble strategy: the results in line 10, shown in Figure A.42, are very comparable to the results for the same line in Case Study 2 (Figure A.24).

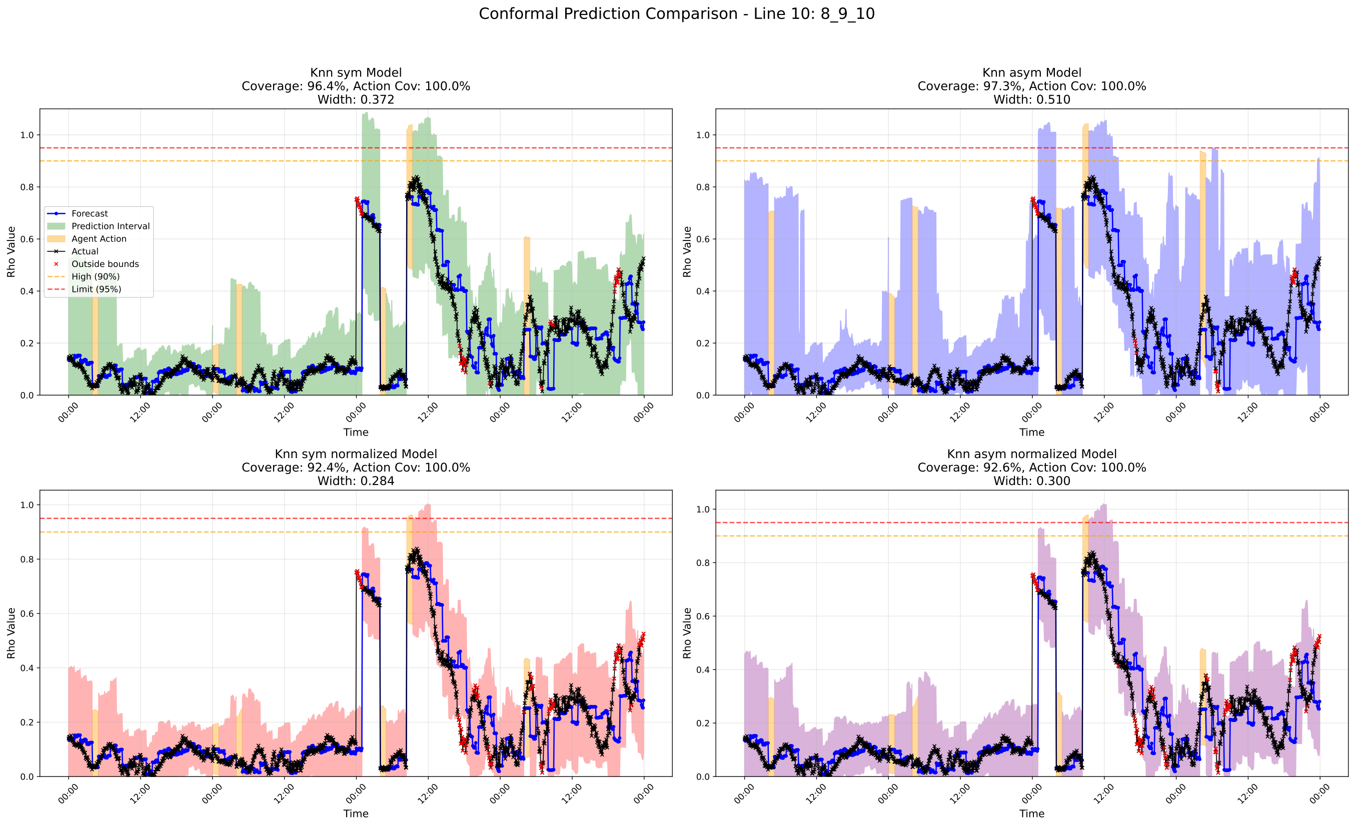

Figure A.40: Case Study 3 k-NN method: Line 10, which in Case Study 2 was the worst performing line, now shows complete coverage. The normalised variants (lower plots) outperform the non-normalised versions (upper plots).

A.3.4 Residual-Based Conformalised Quantile Regression

The RB-CQR method's machine learning backends, which required the ACI update rule to achieve coverage in Case Studies 1 and 2, independently achieve coverage in Case Study 3. The improvement is shown in Figure A.43.

The introduction of the ACI update rule, increases the efficiency of the models, without sacrificing coverage, particularly when we consider the Random Forests variant, which has minimal spikes compared to the other backends. We see the same behaviour in Figure A.44. Based on performance evaluation, this configuration has the best balance of coverage and efficiency, that can be observed in Figure A.45.

A.3.5 Inverted Residual-Based Conformalised Quantile Regression

The Inverted RB-CQR method's conservative characteristics, which provided valuable robustness margins in Case Studies 1 and 2, become counterproductive within the ensemble framework. Since the worst-case aggregation strategy already produces conservative intervals, the additional conservatism that is introduced by the Inverted RB-CQR method

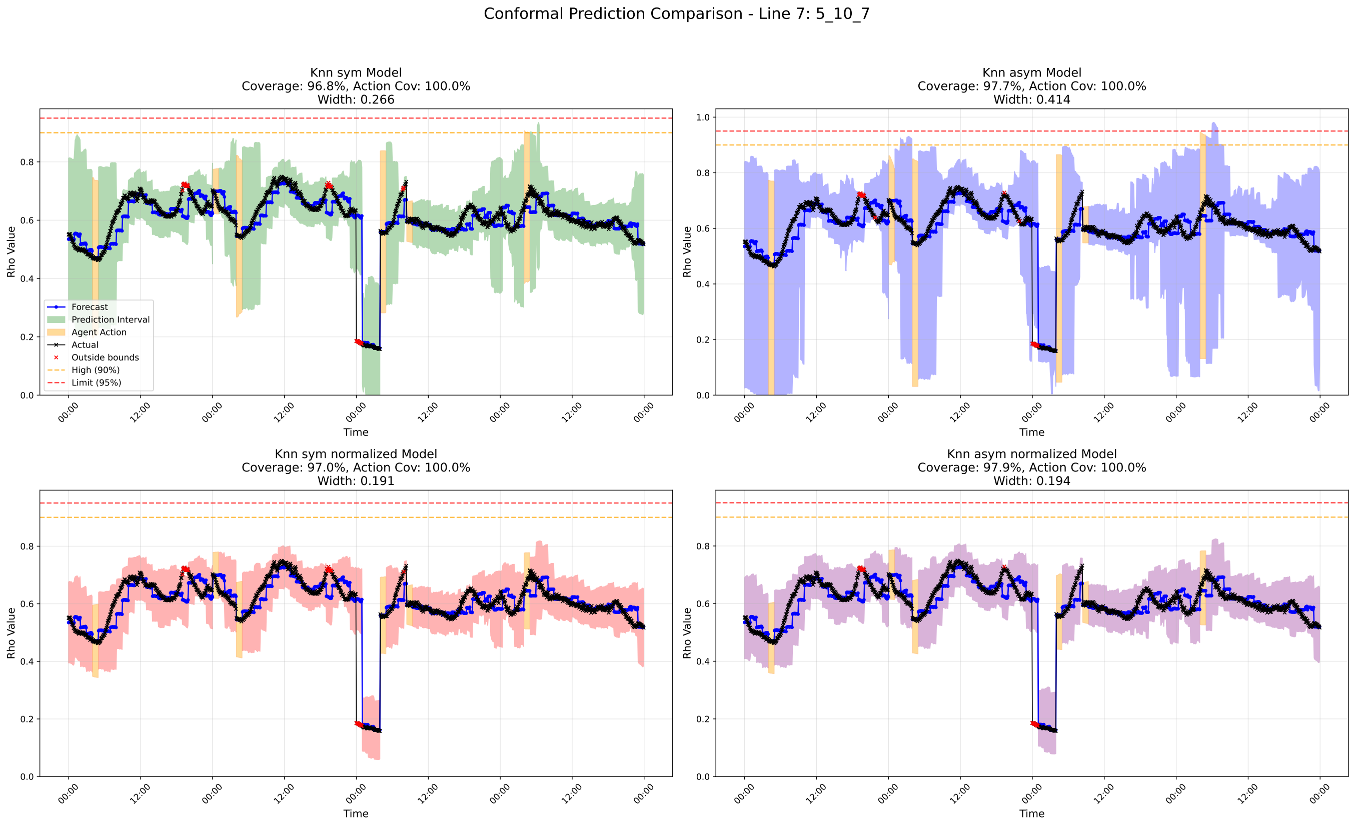

Figure A.41: Case Study 3 k-NN method: Line 7, serving as an example of how the k-NN method behaves in Case 3 across the lines. The normalised version outperforms the base version with more efficient intervals.

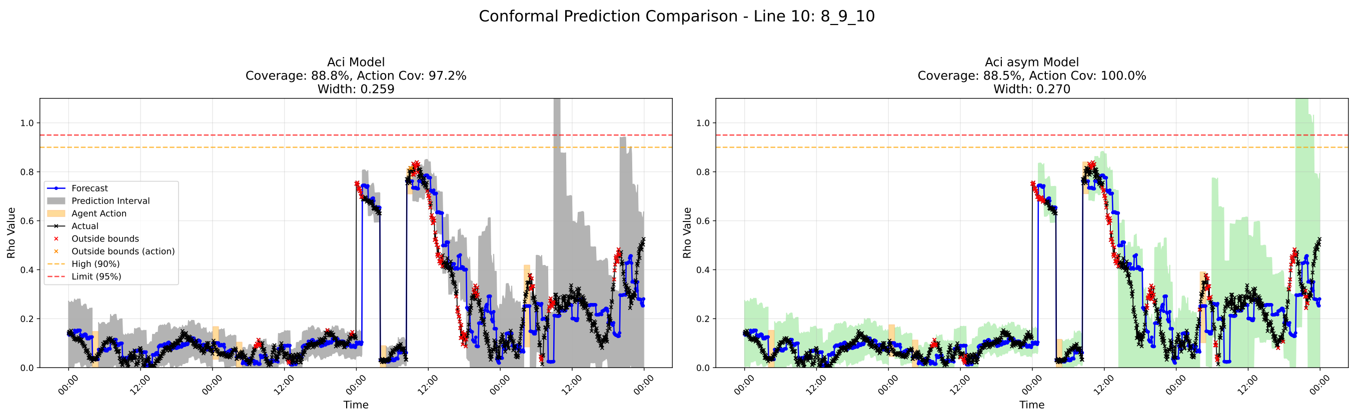

Figure A.42: Case Study 3 ACI method: Line 10, showing that good performance is maintained by the ACI update rule. We still observe the presence of the spikes.

is not beneficial. We can see this over-conservative behaviour in Figure A.46, where the intervals are too wide. Unlike in Case Study 2, where the additional robustness provided good protection against adversarial uncertainty, in the ensemble context, which already provides some robustness, the added conservatism becomes redundant and wasteful.

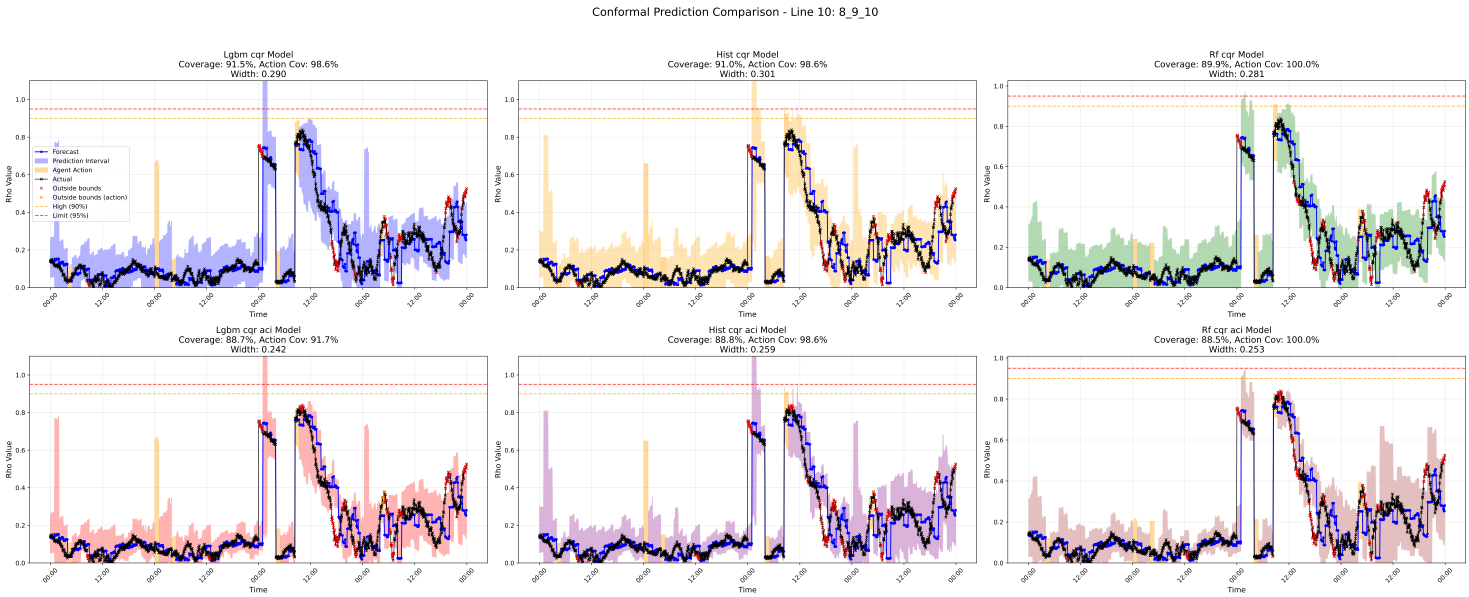

Figure A.43: Case Study 3 RB-CQR method: These methods, that failed in other case studies, now show excelent coverage.

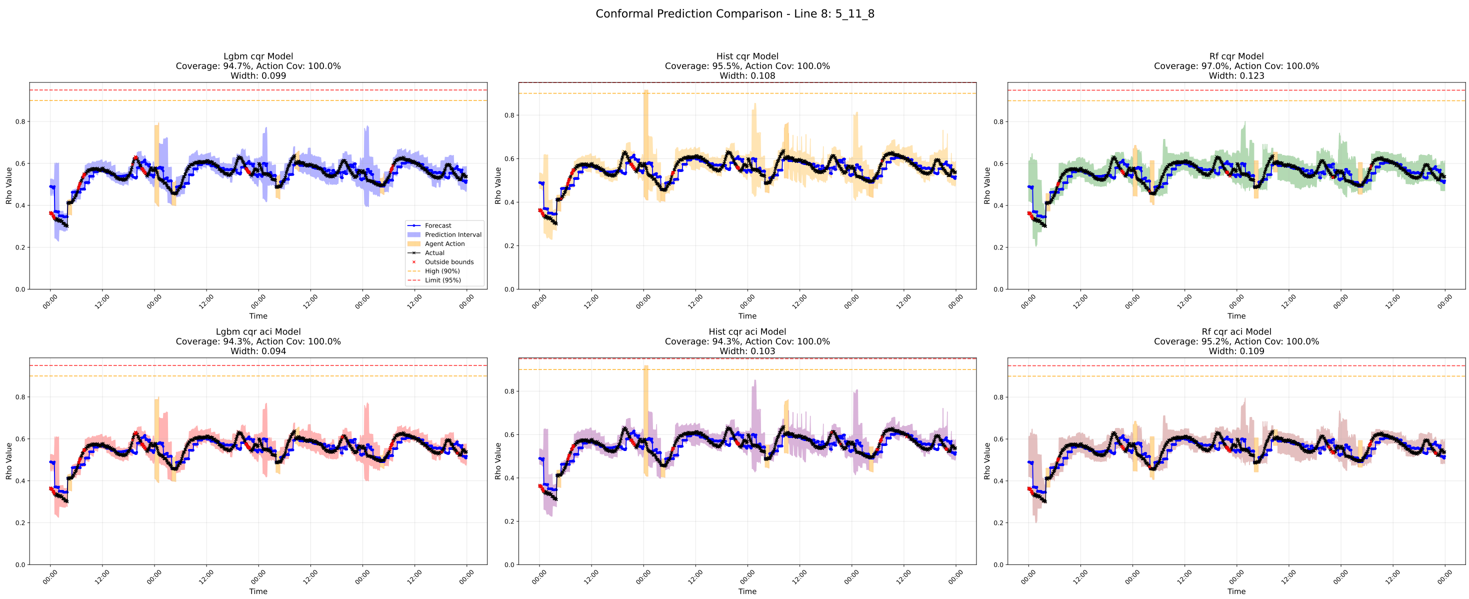

Figure A.44: Case Study 3 RB-CQR method: An example in line 8 that shows the patterns observed in most lines. The Random Forests variation shows the most consistency. All the backends benefit from the ACI update rule introduction.

We present the summary statistics of the Random Forests method in Figure A.47, and observe the efficiency penalty coming from the excessive conservatism. Comparing it with the standard Random Forests performance in Figure A.45 reveals that the interval width increases provide no beneficial trade-off since the ensemble strategy already ensures adequate coverage and adding more conservatism only decreases efficiency. Even with the application of ACI update rule providing some width reduction, the resulting performance

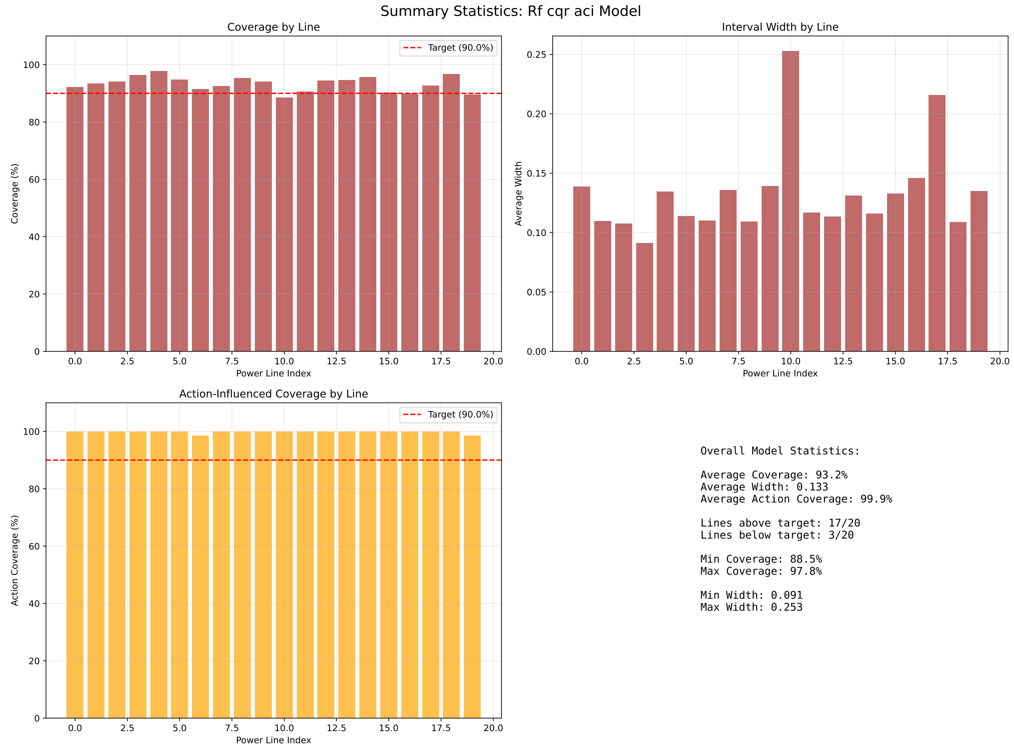

Figure A.45: Case Study 3 RB-CQR with Random Forests: Best machine learning performance in the ensemble approach, with the ACI update rule as an adaptive mechanism.

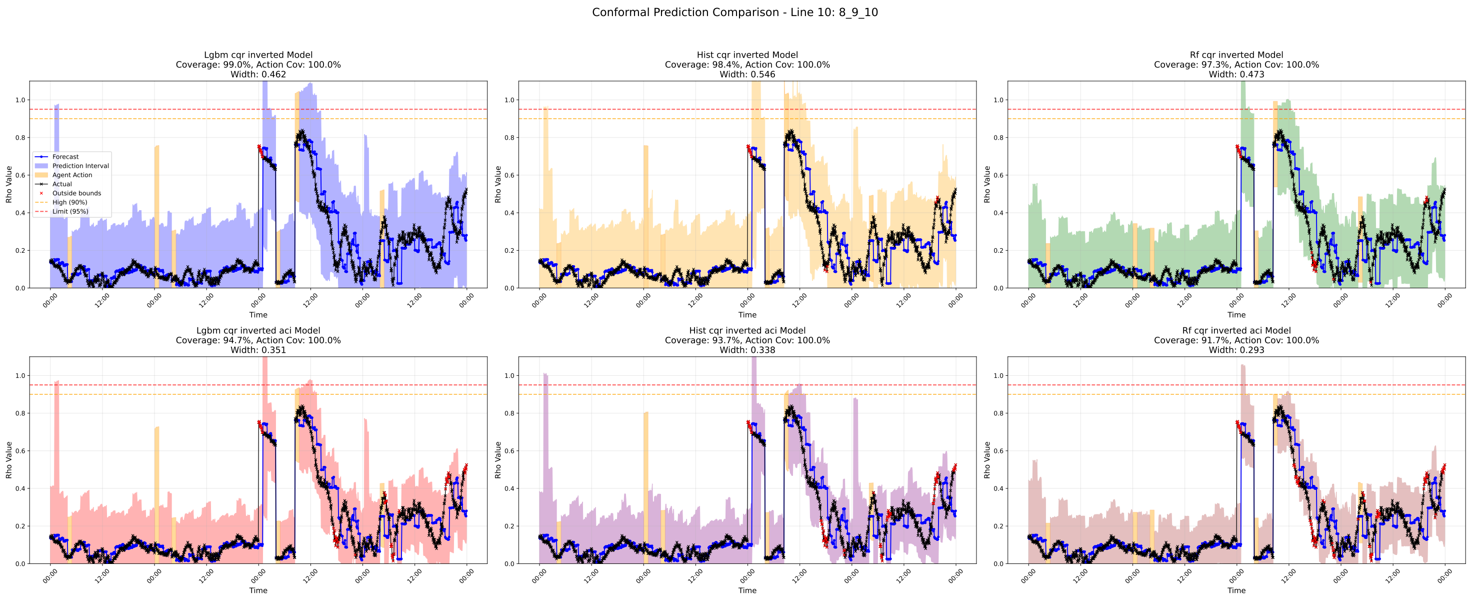

Figure A.46: Case Study 3 Inverted RB-CQR method: We see that in the Ensemble approach, the inverted method is overly conservative in all backends. The introduction of the ACI update rule helps on improving efficiency, but compared to previous approaches, this over-conservatism is redundant.

seems to be inferior to the standard ensemble approaches: introducing the ACI update rule improves the efficiency, as shown in Figure A.48, but the improvement is not significant, specially because we see previous approaches outperforming this one, so the cost in efficiency loss is not worth it.

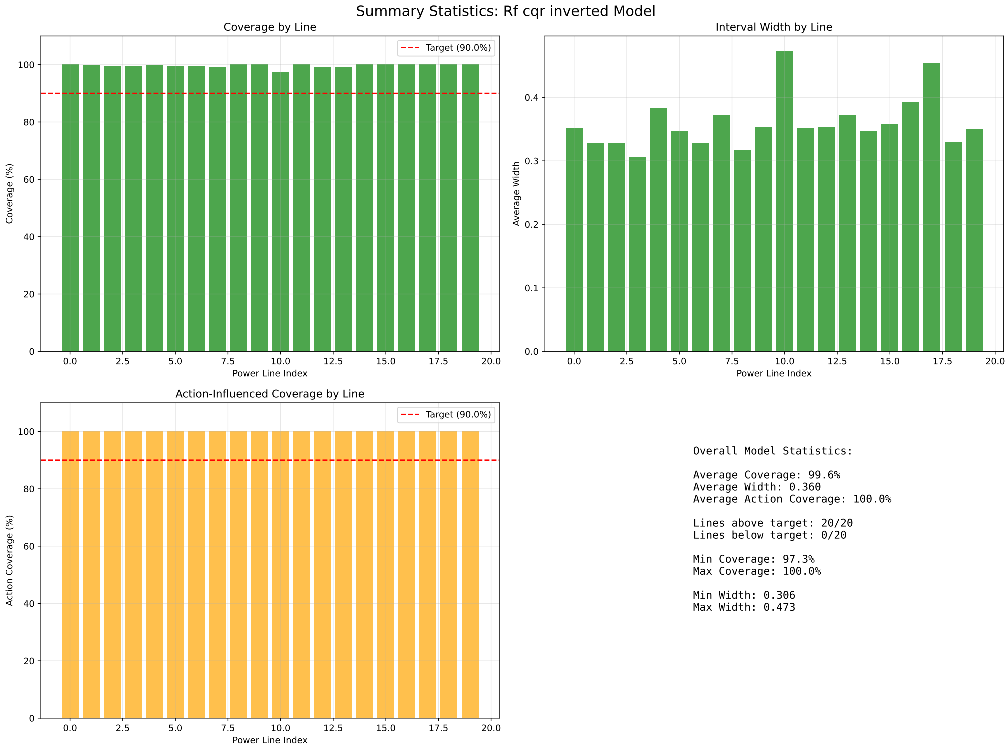

Figure A.47: Case Study 3 Inverted RB-CQR with Random Forests: Excessive conservatism demonstration, showing that although the coverage is met, the efficiency is too reduced.

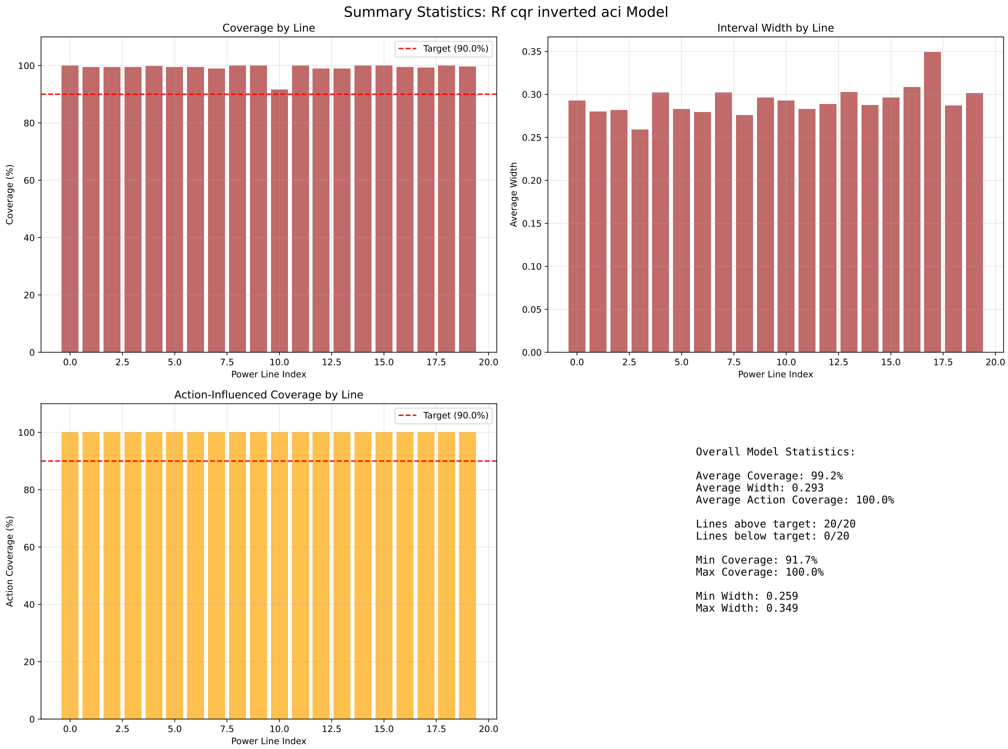

Figure A.48: Case Study 3 Inverted RB-CQR with Random Forests and ACI update rule: Even with the ACI update rule, the interval width is too wide on most transmission lines, which indicates the Inverted method has no benefits over the non-inverted method in Case Study 3.

B. Source code change

In this appendix, we present the source code change that was applied to the experiment done in the CQR paper [22]:

```
class QuantileRegErrFunc(RegressionErrFunc):

    ...

    def apply(self, prediction, y):
        y_lower = prediction[:,0]
        y_upper = prediction[:,-1]
        # Changed from
        # error_low = y_lower - y
        # error_high = y - y_upper
        # To:
        error_low = y - y_lower
        error_high = y_upper - y
        err = np.maximum(error_high,error_low)
        return err
```

The original experiment can be found in the CQR GitHub repository [59].