
Conformalized Prediction of Post-Fault Voltage Trajectories Using Pre-trained and Finetuned Attention-Driven Neural Operators

Amirhossein Mollaali

School of Mechanical Engineering
Purdue University
West Lafayette, IN, USA, 47906
amollaal@purdue.edu

Gabriel Zufferey

Facultad de Ingeniería Eléctrica y Electrónica
Escuela Politécnica Nacional
Quito, Ecuador, 170525
gabriel.zufferey@epn.edu.ec

Gonzalo Constante-Flores

Davidson School of Chemical Engineering
Purdue University
West Lafayette, IN, USA, 47906
geconsta@purdue.edu

Christian Moya

Department of Mathematics
Purdue University
West Lafayette, IN, USA, 47906
cmoyacal@purdue.edu

Can Li

Davidson School of Chemical Engineering
Purdue University
West Lafayette, IN, USA, 47906
canli@purdue.edu

Meng Yue

Interdisciplinary Science Department
Brookhaven National Laboratory
Upton, NY, USA, 11973
yuemeng@bnl.gov

Guang Lin

Department of Mathematics and School of Mechanical Engineering
Purdue University
West Lafayette, IN, USA, 47906
guanglin@purdue.edu

Abstract

This paper proposes a new data-driven methodology for predicting intervals of post-fault voltage trajectories in power systems. We begin by introducing the Quantile Attention-Fourier Deep Operator Network (QAF-DeepONet), designed to capture the complex dynamics of voltage trajectories and reliably estimate quantiles of the target trajectory without any distributional assumptions. The proposed operator regression model maps the observed portion of the voltage trajectory to its unobserved post-fault trajectory. Our methodology employs a pre-training and fine-tuning process to address the challenge of limited data availability. To ensure data privacy in learning the pre-trained model, we use merging via federated learning with data from neighboring buses, enabling the model to learn the underlying voltage dynamics from such buses without directly sharing their data. After pre-training, we fine-tune the model with data from the target bus, allowing it to adapt to unique dynamics and operating conditions. Finally, we integrate conformal prediction into the fine-tuned model to ensure coverage guarantees for the predicted intervals. We evaluated the performance of the proposed methodology using the New England 39-bus test system considering detailed models of voltage and frequency controllers. Two metrics, Prediction Interval Coverage

Probability (PICP) and Prediction Interval Normalized Average Width (PINAW), are used to numerically assess the model’s performance in predicting intervals. The results show that the proposed approach offers practical and reliable uncertainty quantification in predicting the interval of post-fault voltage trajectories.

1 Introduction

Stability in power systems refers to the system’s inherent ability to return to an operational equilibrium following a physical disturbance, such as a contingency [9, 35]. Assessing this ability is essential for ensuring the continuous and reliable operation of power systems. To maintain stability after potential contingencies like faults or equipment failures, system operators conduct offline dynamic studies in a preventive manner [42]. These studies evaluate the system’s dynamic response to faults and disconnections of components by using model-based simulators, which numerically solve a set of nonlinear differential-algebraic equations (DAEs) to track system trajectories after each contingency [44, 19, 30]. Although model-based approaches provide highly accurate evaluations, their significant computational costs make them impractical for real-time applications, limiting their usefulness in situations that require rapid decision-making [25, 6, 47].

To overcome these challenges, modern measurement systems—such as phasor measurement units (PMUs) and advanced communication technologies—now provide operators with real-time data that can enhance stability analysis and help secure the electrical energy supply [40, 31, 51]. In this context, data-driven methodologies have emerged as a promising alternative to traditional physics-based methodologies. These approaches use historical or synthetic data to predict power system stability with significantly faster computation times. Our work advances this class of data-driven methods to improve the effectiveness and efficiency of stability analysis in power systems.

In recent years, machine learning (ML) techniques have been employed to assess various metrics of power system stability by leveraging real-time data from PMUs, enabling the development of real-time stability assessment tools. Some of the most prominent ML models include artificial neural networks [1], convolutional neural networks [37], ensemble learning [18], stacked autoencoders [46], and long short-term memory networks [36]. However, traditional deep learning methods operate in finite-dimensional spaces, limiting their ability to handle the complexity of power system dynamics. Voltage trajectories are inherently infinite-dimensional as continuous functions of time and other parameters. By accounting for these infinite dimensions, function spaces offer a more accurate representation of these trajectories compared to finite-dimensional vector spaces.

Operator learning has emerged as a powerful framework for mapping function spaces [20, 41, 14, 22, 52]. A prominent method of operator learning is the Deep Operator Network (DeepONet) [20, 13, 11, 48, 17, 27]. DeepONet architecture efficiently addresses the complexities of function space mappings, substantially impacting scientific machine learning by providing robust tools for solving high-dimensional nonlinear problems. DeepONet has shown notable success in advanced mathematical applications, such as approximating the solution operator of parametric partial differential equations (PDEs) [12, 49, 45, 16, 8] and multi-scale problems [49, 15]. Beyond theoretical achievements, the practical utility of DeepONet is demonstrated in real-world applications, such as optimizing the rib profile [33], predicting the solution of disk–planet interactions in protoplanetary disks [23], forecasting stresses in elastoplastic structures [10], and predicting the crack path [7].

DeepONet has also shown significant potential in power systems, particularly for predicting post-fault trajectories. In this context, in [29] authors proposed a method utilizing DeepONet-based models to estimate predictive intervals for these trajectories. They first employed B-DeepONet, which uses stochastic gradient Hamiltonian Monte Carlo for uncertainty quantification but requires numerous hyperparameters, making the training process expensive. To address this, the authors introduced Prob-DeepONet, which offers automated uncertainty quantification with minimal computational overhead through a probabilistic training strategy. However, this approach assumes a normal distribution for the target solution, which is not necessarily accurate and can negatively affect the prediction performance. Additionally, their methodology is unable to guarantee the coverage of the predicted intervals. On the other hand, their method only uses data from the target bus for training the model and cannot utilize data from neighboring buses, making it prone to challenges of low-data regimes in power systems. To overcome the limitations of low-data regimes, the authors of [38] proposed DeepGraphNet to predict post-fault trajectories. Their architecture incorporates graph neural networks, enabling the framework

to learn from data across neighbor buses in the power system, thus mitigating low-data regime issues. However, they used a centralized learning approach to train the model, which violates the privacy of the buses, making its practical implementation challenging. Additionally, DeepGraphNet is deterministic, lacking the capability to provide uncertainty quantification for its predictions. Moreover, the models proposed in these studies lack the ability to handle heterogeneous input data across different time domains, which reduces their practicality for power systems, considering the randomness of fault occurrences in these systems.

Existing approaches do not offer a comprehensive solution for predicting post-fault trajectory intervals without relying on distributional assumptions. Additionally, these models are unable to guarantee the coverage probability of the predicted intervals. Furthermore, they fail to address the challenges of low-data regimes while maintaining the privacy of individual buses.

To bridge this gap, we propose a comprehensive approach for predicting the intervals of post-fault voltage trajectories in power systems. Our methodology focuses on the development of a quantile-based operator learning model that estimates predictive intervals without relying on distributional assumptions, making it more flexible and robust across different scenarios. To overcome issues related to limited data, we implement a pre-training and fine-tuning strategy. In the pre-training phase, we use data from neighboring buses to capture the underlying dynamics of voltage trajectories. This phase incorporates federated learning to maintain data privacy while enhancing the model's robustness. After pre-training, the model is fine-tuned with data from the target bus, which further improves the accuracy and reliability of predictions for that specific bus. Finally, to ensure that the predicted intervals have the desired coverage guarantees, we integrate conformal prediction methods into the fine-tuning process.

The main contributions of this work are as follows:

- *QAF-DeepONet*: We introduce QAF-DeepONet, a novel data-driven framework for predicting post-fault voltage trajectory intervals. It employs an attention mechanism and a Fourier feature network to capture dynamic behaviors and fluctuations in voltage trajectories while handling heterogeneous input data. QAF-DeepONet estimates lower and upper quantiles of the target solution to capture the range of possible outcomes.
- *Federated Learning Pre-Training*: We employ federated learning to pre-train QAF-DeepONet using distributed data from neighboring buses. This approach leverages the available data to address the issue of low-data regime and to enhance the model's generalizability and robustness via merging while preserving data privacy.
- *Fine-Tuning for Targeted Bus*: We fine-tune the pre-trained model with data from the target bus, improving the reliability and precision of the estimated predictive intervals.
- *Conformal Prediction Integration*: We integrate conformal prediction techniques into the fine-tuned QAF-DeepONet predictions, ensuring the predicted intervals meet the desired coverage guarantees. This adjustment provides statistically reliable intervals, supporting effective decision-making in power systems.

The remainder of the paper is organized as follows: Section 2 introduces the problem statement and the main challenges of predicting post-fault trajectories. Section 3 presents the proposed methodology and the training strategy. Section 4 shows the numerical experiments to evaluate the performance of the proposed methodology. Finally, Sections 5 and 6 describe our future work and conclude the paper.

2 Preliminaries

In this section, we outline the problem statement and the primary challenges associated with predicting post-fault trajectories.

2.1 Problem Statement

Assessing the short-term stability of a power system is imperative to ensuring the continuous and reliable operation of power systems. This type of stability refers to the power system's ability to find a stable equilibrium point after being subject to disturbances, such as faults or equipment failures. The dynamic behavior of the system's trajectories is governed by: (i) the dynamics of mechanical

(e.g., turbines), electrical (e.g., generators, lines, transformers, loads), and control systems (e.g., governor, and automatic voltage regulators) and (ii) the algebraic physical laws determining the power flows throughout the power grid. Such behavior can be modeled as a system of DAEs across three different stages: pre-fault, fault-on, and post-fault. Each stage has its own unique set of dynamics that determine the stability and transient behavior of the system's trajectories.

The behavior of the system's trajectories, illustrated in Figure 1, during these stages, can be analyzed by examining them as follows:

1. *Pre-Fault Dynamics*: Prior to any disturbance, the power grid is assumed to be in an equilibrium state at time t_0 . This equilibrium persists as long as the system remains undisturbed, up until a fault occurs at time t_f . During this pre-fault stage, the dynamics of the system are governed by the following set of DAEs:

$$\dot{x}(t) = f_P(x(t), z(t)), \quad \text{for } t \in [t_0, t_f], \quad (1a)$$

$$0 = g_P(x(t), z(t)), \quad (1b)$$

where $x(t)$ represents the state vector of the system, and $z(t)$ is the set of algebraic variables. These equations describe the stable operation of the system prior to any fault, assuming the system is initially in a steady state [34].

2. *Fault-on Dynamics*: When a fault occurs at time t_f , the system transitions into a faulted condition, which leads to a temporary topological change in the system. This faulted state lasts until the fault is cleared at time t_{cl} by a protective relay, which isolates the faulted component. The time interval from the occurrence of the fault to its isolation is referred to as clearing time. During this period, the dynamics of the system are altered and are described by the following set of DAEs:

$$\dot{x}(t) = f_F(x(t), z(t)), \quad \text{for } t \in [t_f, t_{cl}], \quad (2a)$$

$$0 = g_F(x(t), z(t)). \quad (2b)$$

The fault-on dynamics capture the system's immediate response to the disturbance, reflecting the changes in system behavior caused by the fault.

3. *Post-Fault Dynamics*: After the fault is cleared at time t_{cl} , the system enters a transient response as it attempts to return to a stable equilibrium. This phase continues until a specified time horizon, T , and the dynamics during this stage are governed by:

$$\dot{x}(t) = f_{PF}(x(t), z(t)), \quad \text{for } t \in [t_{cl}, T], \quad (3a)$$

$$0 = g_{PF}(x(t), z(t)). \quad (3b)$$

The post-fault dynamics are key for understanding the system's ability to regain stability after a disturbance and to identify potential instabilities that may arise after the fault occurs.

2.2 Problem Setup

This paper focuses on the real-time prediction of post-fault voltage trajectory intervals using local measurements. The approach is structured into two parts: offline training and online inference. In the offline training stage, the model is trained using historical data from the target bus and its neighboring buses. During online inference, the trained model uses local measurements from the target bus to predict voltage trajectory intervals in real-time.

The real-time prediction of voltage trajectory intervals presents several significant challenges:

1. *Infinite-Dimensional Problem*: Voltage trajectories are continuous functions, making their prediction inherently an infinite-dimensional problem. Such a problem can be addressed by operator regression tools where both the input and output are variable continuous functions.
2. *Heterogeneous Input Data*: The random nature of fault clearing time, i.e., t_{cl} varies due to delays or sensing/acting equipment failures, resulting in highly variable input domains. This variability challenges the standardization of the problem and requires models that can adapt to different domains.
3. *Predictive Interval Estimation*: The method should be capable of estimating predictive intervals for practical uncertainty quantification. This ensures that the model provides reliable and informative predictions for real-time decision-making tools.

4. *Low Data Regime*: There is often a limited dataset available for predicting post-fault voltage at a target bus. This scarcity of data can hinder the training process of the models, necessitating a method that effectively enables the model to learn from other sources, such as neighboring buses.
5. *Privacy Concerns*: When using voltage information from neighbor buses to mitigate the low data regime, it is crucial to protect the privacy of each bus's data. Ensuring data privacy while still using distributed information for model training is a significant challenge that must be addressed.
6. *Robustness Requirements*: Reliable predictions are crucial for maintaining the stability of the system. Post-fault voltage prediction methods must deliver robust predictions under various conditions. In addition, providing coverage guarantees for predictive intervals is highly valuable, yet challenging. Achieving this combination of robustness and coverage is essential for effective power system management, although it is difficult to achieve.

3 Methodology

In this section, we present our comprehensive approach to addressing the challenges of predicting post-fault voltage trajectories in power systems. To handle the infinite-dimensional nature of voltage prediction, our methodology leverages deep operator learning-based models. Considering the dynamics of the problem, we propose a modified DeepONet-based model, adapting the vanilla DeepONet to make it suitable for predicting voltage trajectory intervals. This modification allows the model to manage heterogeneous input data and ensures effective adaptation to varying input domains. The proposed model integrates uncertainty quantification capabilities by estimating predictive intervals and providing reliable and informative predictions. To address low data regimes, we utilize a pre-training and fine-tuning process, which can maintain privacy by enabling the model to learn from distributed data sources while maintaining data confidentiality. Finally, we employ conformal prediction techniques to ensure robust and reliable predictions by providing coverage guarantees. The following sections provide detailed descriptions of each of these techniques and strategies.

3.1 Neural Operators

To achieve our goal of reliably predicting post-fault voltage trajectories, we need to develop a model that can handle the infinite-dimensional nature of this problem. To this end, we employ neural operators, in particular, deep operator learning techniques to handle this complexity effectively. Accordingly, in the next sections, we first formalize the problem as an operator regression task. Next, we outline the vanilla deep operator network structure and then provide a step-by-step explanation of how we modify this structure to develop our model, which is designed for predicting post-fault voltage trajectory prediction intervals.

3.1.1 Operator Regression Formulation

We aim to approximate an operator $\mathcal{G}$ that maps an input function $u(t) \in \mathcal{U}$ to its corresponding post-fault trajectory function $v(t) \in \mathcal{V}$. Formally, the operator $\mathcal{G}$ is defined by the following mapping:

$$\mathcal{G} : \mathcal{U} \rightarrow \mathcal{V} \quad (4)$$

where $\mathcal{U}$ represents the vector space of the input functions, and $\mathcal{V}$ represents the vector space of the output functions, specifically the post-fault voltage trajectories.

In the dynamics of the power grid, the dynamic states x and the algebraic states z , described in Section 2, are not fully observable. Consequently, we used the system's observable data to define the input function space in the mapping $\mathcal{G}$. Specifically, we define the observable portion of voltage trajectories as the input function $u \in \mathcal{U}$ in this mapping. This allows us to utilize the voltage's past behavior to predict its future behavior, particularly its corresponding post-fault trajectory. Conceptually, this approach aligns with the Mori-Zwanzig formalism which addresses the dynamics of complex systems by focusing on observable quantities [5]. In this context, $\mathcal{G}$ uses the observed data to infer the dynamic response of the system to the fault, without direct access to the internal states.

To define the input function $u \in \mathcal{U}$ and the target solution $v \in \mathcal{V}$, we split the voltage trajectory, denoted as $\mathcal{R}(t)$, into two segments. We assume that the first segment is observable and consider it as the input function of the mapping $\mathcal{G}$, while the second segment, which is unobservable, is our target solution to predict.

Since the pre-fault and fault-on conditions of the system influence the transient response in the post-fault stage, it is intuitive that the time domain of the input function, denoted as $\mathcal{T}_u$, should encompass both of these stages of the voltage's trajectory. Additionally, to accurately map and understand the system response to a fault, we assume that a small portion of the post-fault stage is observable, referred to as the observable post-fault duration, denoted as Δt_{obs} .

Consequently, we define the input function time domain $\mathcal{T}_u$ as follows:

$$\mathcal{T}_u = [t_0, t_f) \cup [t_f, t_{cl}) \cup [t_{cl}, t_{cl} + \Delta t_{\text{obs}}] \quad (5)$$

where $\mathcal{T}_u$ includes the pre-fault stage $[t_0, t_f)$, the fault-on stage $[t_f, t_{cl})$, and the observable part of the post-fault trajectory $[t_{cl}, t_{cl} + \Delta t_{\text{obs}}]$. Notably, Δt_{obs} is an adjustable hyperparameter that directly affects the performance of the mapping by the operator $\mathcal{G}$. Carefully selecting Δt_{obs} is crucial for optimizing the model's prediction performance and usability. While a smaller Δt_{obs} is advantageous for practical use, as the model will require less information as input; however, making it too small may negatively impact the model's performance.

The output function time domain, denoted as $\mathcal{T}_v$, is defined as the complement of $\mathcal{T}_u$, representing the time domain of the unobservable part of the post-fault trajectory:

$$\mathcal{T}_v = (t_{cl} + \Delta t_{\text{obs}}, T] \quad (6)$$

where T is the maximum time of prediction. Based on this definition, for a sample trajectory $\mathcal{R}(t)$, the input function is $u = \mathcal{R}[\mathcal{T}_u]$ and the output function is $v = \mathcal{R}[\mathcal{T}_v]$. Figure 1 illustrates the temporal domains $\mathcal{T}_u$ and $\mathcal{T}_v$ for a sample voltage trajectory.

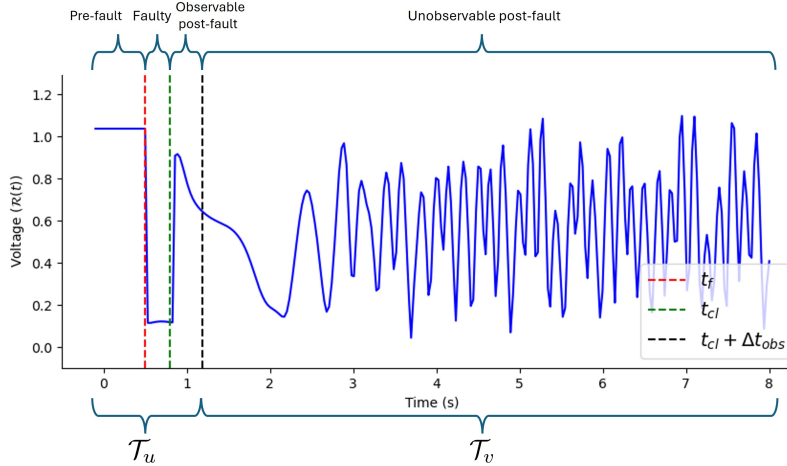


Figure 1: Temporal domains $\mathcal{T}_u$ and $\mathcal{T}_v$ for a sample voltage trajectory. The input function u is defined over $\mathcal{T}_u$, and the output function v is defined over $\mathcal{T}_v$. Δt_{obs} is an adjustable hyperparameter that determines the duration of observable part of the post-fault stage, directly influencing the performance of the mapping $\mathcal{G}$.

3.1.2 Deep Operator Network

DeepONet is an advanced neural network architecture designed to approximate operators by extending the universal approximation theorem to functions [4, 21]. DeepONet approximates the target operator $\mathcal{G}$ using two multilayer perceptron (MLP) sub-networks: the branch network, which encodes the input function, and the trunk network, which processes the evaluation points at the output domain. Subsequently, the outputs of these networks are combined through a linear product to approximate the operator. Given $u \in \mathcal{U}$ as the input function and $t \in \mathcal{T}_v$ as the evaluation point in the time domain of output, DeepONet approximates the unobserved post-fault trajectory v using the following linear

221 representation:

$$v(u, t) \approx \mathcal{G}^\theta(u)(t) = \sum_{i=1}^p \phi_i(u) \psi_i(t) \quad (7)$$

222 Here, ϕ_i and ψ_i represent the basis functions generated by the branch and trunk networks, respectively,
 223 while p denotes the number of basis functions. Figure 2 illustrates the components of DeepONet
 and their integration. However, the current configuration of the DeepONet architecture utilizes MLP

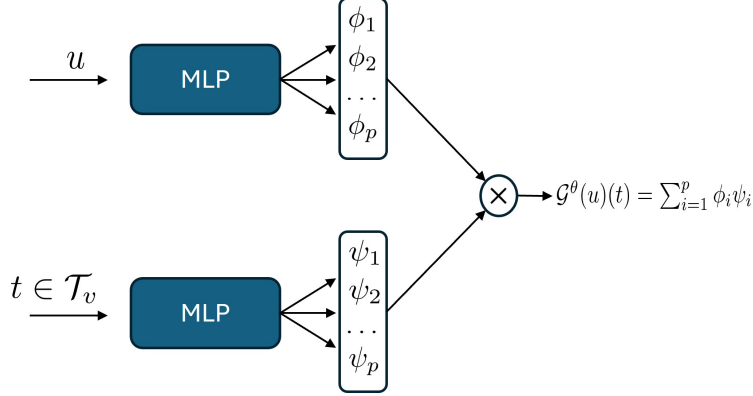


Figure 2: A schematic representation of the DeepONet architecture. The figure illustrates the two main components: the branch network, which encodes the input function $u \in \mathcal{U}$, and the trunk network, which processes the evaluation points $t \in \mathcal{T}_v$. The outputs of these networks are merged through an inner product to approximate the target operator $\mathcal{G} \approx \mathcal{G}^\theta$.

224 networks, which may not be suitable for processing input functions that contain temporal sequences.
 225 Moreover, the post-fault voltage trajectories contain fluctuation patterns that are highly challenging
 226 for an MLP to capture accurately.
 227

228 3.1.3 Attention-Fourier Deep Operator Network (AF-DeepONet)

229 To address the limitations of DeepONet in predicting post-fault voltage trajectories, we introduce
 230 AF-DeepONet, an enhanced version of DeepONet that more effectively processes input functions
 231 and comprehends fluctuations in voltage trajectories. Figure 3 illustrates the architecture of AF-
 232 DeepONet, which consists of two primary components: the Attention-Based Branch Network and
 233 the Fourier-Featured Trunk Network.

Attention-Based Branch Network: The branch network integrates a self-attention mechanism followed by a multi-layer perceptron (MLP) to process the input function u . In particular, given a projected padded input $\hat{u} \in \mathbb{R}^{T \times d}$, the self-attention uses trainable linear projections to compute the queries, keys, and values, i.e., $Q, K, V \in \mathbb{R}^{T \times d}$. Then, we compute the self-attention output as follows:

$$O = \text{softmax}(QK^\top) \cdot V \in \mathbb{R}^{T \times d}$$

234 . By employing self-attention, it selectively focuses on the most relevant parts of the input function,
 235 thereby enhancing its ability to capture dependencies and temporal dynamics. This selective weight-
 236 ing of input function components ensures an effective approximation of the target operator. The
 237 subsequent MLP refines these features to produce the basis functions (ϕ_i).

238 *Fourier-Featured Trunk Network:* The trunk network encodes the output location $t \in \mathcal{T}_v$ using a
 239 Fourier features network as proposed by [39]. This network helps DeepONet to capture fluctuations
 240 and oscillatory behaviors, which is particularly useful for the post-fault voltage stage, which often
 241 contains high-frequency features [24]. The Fourier features network employs a random Fourier
 242 mapping:

$$\gamma(x) = [\sin(Bx), \cos(Bx)] \quad (8)$$

243 where $B \in \mathbb{R}^{m \times d}$ is a matrix with entries sampled from a Gaussian distribution $N(0, \sigma^2)$. The
 244 encoded output location is then processed by an MLP to produce the basis functions ψ_i .

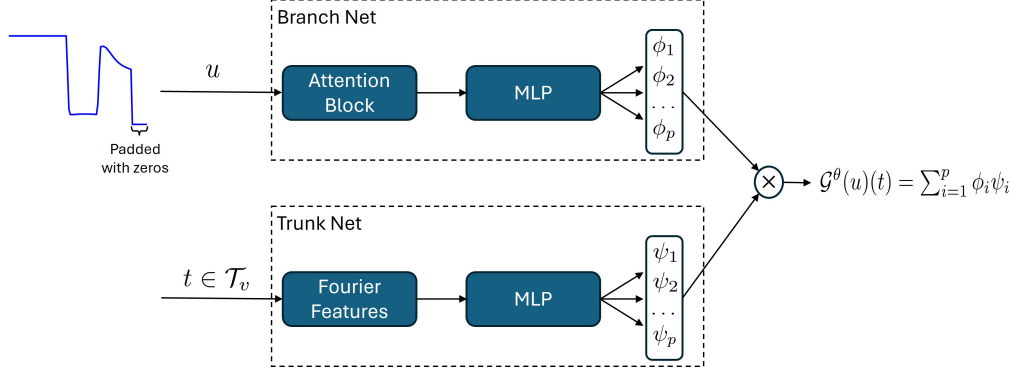


Figure 3: Schematic representation of the Attention-Fourier Deep Operator Network (AF-DeepONet).

245 Ultimately, the AF-DeepONet approximates the target operator $\mathcal{G}$ using the same linear trainable
 246 representation as described in Equation 7.

247 *Handling Variable Input Lengths:* Given that the fault clearing time, i.e., t_{cl} is a random variable,
 248 the length of $\mathcal{T}_u$ is also random (see equation 5), with the pre-fault stage having a variable duration
 249 across different voltage trajectories. This variability conflicts with the network architecture, which
 250 requires a fixed domain for its input. To address this, we employ a padding technique. We consider a
 251 maximum possible input function time T_{max} , based on the maximum time for fault clearance. In
 252 this approach, we pad each voltage trajectory data to this maximum length with zero-masked signals,
 253 where the actual input function occupies the initial part and the remainder is filled with zeros, i.e.,
 254 $\mathcal{T}_u \subseteq [t_0, T_{max}]$. This approach ensures that the network can handle variable input lengths without
 255 compromising its structure.

256 3.1.4 Quantile Attention-Fourier Deep Operator Network (QAF-DeepONet)

257 Power systems require robust predictive intervals to ensure stability and reliability under dynamic
 258 conditions. These intervals aid operators in managing instability and improving decision-making
 259 under uncertainty. This necessity drives our approach, underscoring the critical role of estimating
 260 uncertainty for stable and reliable forecasting. Consequently, it is essential to modify the AF-
 261 DeepONet framework to enable practical uncertainty quantification. To this end, we introduce
 262 the Quantile Attention-Fourier Deep Operator Network (QAF-DeepONet), which enhances AF-
 263 DeepONet to empirically approximate uncertainty in post-fault trajectory predictions.

264 Quantiles are particularly effective for uncertainty quantification because they do not assume any
 265 specific distribution, allowing them to handle distribution-free uncertainty. This inherent flexibility
 266 makes quantiles robust against skewness, heavy tails, and outliers, ensuring their applicability across
 267 various data types and scenarios. The conditional quantile function, q_τ , defines the τ -th percentile in
 268 the distribution of Y , conditional on $X = x$. Mathematically, this function is expressed as:

$$q_\tau(x) = \inf\{y \in \mathbb{R} : \mathbb{P}(Y \leq y \mid X = x) \geq \tau\} \quad (9)$$

269 In essence, given $X = x$, q_τ provides the value below which a fraction τ of the conditional distribution
 270 of Y lies.

271 Building on the AF-DeepONet framework, the QAF-DeepONet is specifically designed to approxi-
 272 mate the conditional quantiles of the target solution. For a specified level of miscoverage $\alpha \in (0, 1)$,
 273 the QAF-DeepONet estimates the $\alpha/2$ -th and $(1 - \alpha/2)$ -th conditional quantiles, denoted as $q_{\alpha/2}$
 274 and $q_{1-\alpha/2}$, respectively.

275 Figure 4 illustrates the QAF-DeepONet architecture. This architecture introduces additional branches
 276 and trunks to AF-DeepONet architecture specifically designed to approximate the $q_{\alpha/2}$ and $q_{1-\alpha/2}$
 277 quantiles. In this configuration, the basis functions ϕ and ψ , generated by the original AF-DeepONet,
 278 flow into these new shared branches and trunks. These new sub-networks then process the existing
 279 basis functions to construct new ones, capable of estimating the specified quantiles through their

280 following inner products:

$$q_{\alpha/2}(u, t) \approx \mathcal{G}_{\alpha}^{\theta}(u)(t) = \sum_{i=1}^s \phi_i^b \psi_i^t \quad (10)$$

281

$$q_{1-\alpha/2}(u, t) \approx \mathcal{G}_{1-\alpha/2}^{\theta}(u)(t) = \sum_{i=1}^s \phi_i^t \psi_i^b \quad (11)$$

282 Here, ϕ_i^b and ψ_i^t serve as the basis functions for estimating $q_{\alpha/2}$, while ϕ_i^t and ψ_i^b are used for
283 estimating $q_{1-\alpha/2}$. The number of these basis functions is denoted by s .

284 *Dataset Generation:* To generate input and output functions, we first choose the observable post-fault
285 duration Δt_{obs} . For each voltage trajectory $\mathcal{R}_j$, considering the clearing time t_{cl} and chosen Δt_{obs} ,
286 we split $\mathcal{R}_j$ into the observable trajectory $u_j = \mathcal{R}_j[\mathcal{T}_u]$ and the unobservable trajectory $v_j = \mathcal{R}_j[\mathcal{T}_v]$.
287 We then pad u_j with zeros to a predefined length to create the input function. For convenience in
288 this paper, we refer to this padded version of u_j same as u_j . The output function v_j represents the
289 unobserved post-fault trajectory.

290 Furthermore, for both the training and the inference phases in QAF-DeepONet, it is essential to
291 structure the datasets to be compatible with the DeepONet configurations. To achieve this, the
292 continuous function u_j is discretized at the m locations, that is, $u_j \in \mathbb{R}^m$, which serves as input to
293 the branch net. Moreover, for each u_j , we randomly select n_{loc} number of output locations $t \in \mathcal{T}_v$
294 and sample the output function at these points, that is, $\{G^{(i)}\}_{i=1}^{n_{loc}} = \{v_j(t^{(i)})\}_{i=1}^{n_{loc}}$. This approach
295 is consistently applied across all voltage trajectories to assemble a dataset of triplets, denoted as
296 $\mathcal{D} = \{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^N$.

297 *Loss Function:* To optimize the QAF-DeepONet, we minimize the following loss function across a
298 dataset of N triplets, denoted as $\mathcal{D} = \{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^N$:

$$\mathcal{L}(\theta, \mathcal{D}) = \frac{1}{N} \sum_{i=1}^N \rho_{\tau} \left(G^{(i)}, \mathcal{G}_{\tau} \left(u^{(i)} \right) (t^{(i)}) \right), \quad (12)$$

299 where ρ_{τ} represents the pinball loss function for quantiles $\tau \in \{\alpha/2, 1 - \alpha/2\}$. The function is
300 specifically defined as follows:

$$\rho_{\tau}(y, \hat{y}) = \begin{cases} \tau(y - \hat{y}) & \text{if } y - \hat{y} > 0, \\ (1 - \tau)(\hat{y} - y) & \text{otherwise.} \end{cases} \quad (13)$$

301 This loss function allows the evaluation of model performance based on quantile-specific discrepancies
302 between the predicted intervals and actual values.

303 After training, the QAF-DeepONet estimates predictive intervals for any new test sample as follows:

$$\widehat{CL}(\hat{u}_{\text{test}}, x_{\text{test}}) = \left[\mathcal{G}_{\alpha/2}^{\theta}(u_{\text{test}})(t_{\text{test}}), \mathcal{G}_{1-\alpha/2}^{\theta}(u_{\text{test}})(t_{\text{test}}) \right], \quad (14)$$

304 where $\mathcal{G}_{\alpha/2}^{\theta}(u_{\text{test}})(t_{\text{test}})$ and $\mathcal{G}_{1-\alpha/2}^{\theta}(u_{\text{test}})(t_{\text{test}})$ denote the predicted lower and upper quantiles,
305 respectively, for the interval with miscoverage level α .

306 3.2 Training Strategy

307 In power systems, faults are rare events, which makes it impractical to obtain extensive fault data for
308 various scenarios. Furthermore, data from the power system are often private and not easily accessible,
309 leading to limited data availability. It is particularly challenging to obtain voltage trajectory data for a
310 specific bus. However, data-driven approaches, such as deep operator learning, often require large
311 datasets for high performance.

312 To address this issue, our methodology utilizes a pre-training and fine-tuning strategy, making
313 maximum use of the available data in the system. In this strategy, we first pre-train the model on a
314 broader dataset from neighboring buses to learn the underlying voltage dynamics. We then fine-tune
315 the model using data from the target bus, aiming for reliable post-fault trajectory predictions. Figure 5
316 shows a schematic of a power system, highlighting the target bus and its neighboring buses.

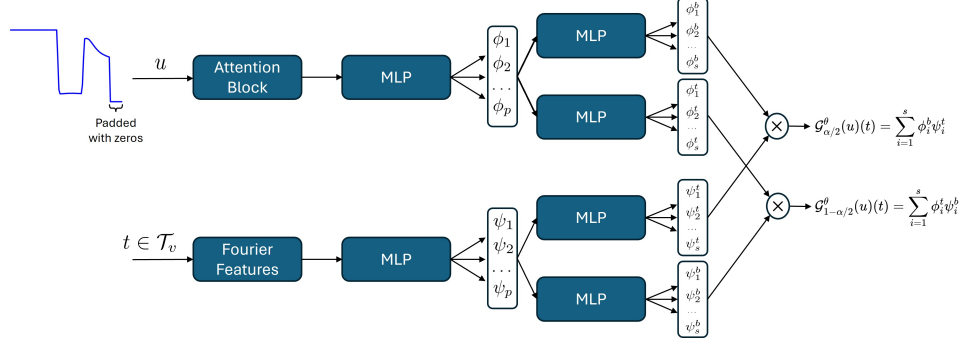


Figure 4: Schematic representation of the Quantile Attention-Fourier Deep Operator Network (QAF-DeepONet).

3.2.1 Pre-Training Model

The pre-training phase is crucial in our methodology, providing the foundation for developing a robust and accurate model. We utilize data from neighboring buses of the target bus to pre-train the model. However, due to privacy concerns, costs, and logistical constraints, direct data sharing is not feasible. To overcome this, we employ federated learning, enabling collaborative training of local models to obtain a *merged* model without direct data sharing.

Federated Deep Operator Network Learning: Federated learning effectively addresses privacy and related issues by decentralizing the training process. In this strategy, each bus's model is trained locally, sharing only its parameters θ with other buses, thus preserving data privacy. To implement this strategy, we have adapted the approach outlined in [26] and improved in [50].

Consider a target bus with $\mathcal{N}_c$ neighboring buses. Let θ_k^c represent the parameters of the model for the c -th bus in iteration k . The parameters γ_k^c are the updated values derived from θ_k^c through a local Adam optimization step at the same iteration. With K denoting the number of local round updates, the model's parameters are updated using the following procedure:

$$\theta_{k+1}^c = \begin{cases} \gamma_{k+1}^c & \text{if } (k+1) \bmod K \neq 0 \\ \sum_{c=1}^{\mathcal{N}_c} \frac{1}{\mathcal{N}_c} \gamma_{k+1}^c & \text{if } (k+1) \bmod K = 0 \end{cases} \quad (15)$$

In this procedure, bus c independently updates its parameters using local Adam optimization. For most iterations, the updated parameters γ_{k+1}^c are directly adopted as the new parameters θ_{k+1}^c for the local model of the bus c . However, after every K local update, the parameters are averaged across all neighboring buses to form a new global parameter set. Then, this averaged model becomes the updated local model for all buses. After training, the averaged model serves as the pre-trained model for the target bus, benefiting from the collective knowledge of neighboring buses and enhancing its robustness and generalization.

Pre-Training Dataset: The federated learning dataset comprises voltage trajectories from the neighboring buses of the target bus. Each neighboring bus dataset contains its own voltage trajectory data, used locally to train its corresponding local model. The pre-training dataset for $\mathcal{N}_c$ neighboring buses is represented as $\mathcal{D}_P = \{\mathcal{D}_c\}_{c=1}^{\mathcal{N}_c}$, where $\mathcal{D}_c = \{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^{n_c}$ denotes the dataset for bus c , with n_c being the number of training triples for that bus. The size of the pre-training dataset is $n_p = \sum_{c=1}^{\mathcal{N}_c} n_c$.

3.2.2 Finetuning the Pre-Trained Model

After pre-training the model, we proceed to fine-tune it using data from the target bus. Since the pre-trained model has already captured the underlying dynamics of the voltage trajectories, fine-tuning allows it to adapt to the unique attributes of the target bus using a relatively small dataset, effectively overcoming the challenge of limited data availability. Through this process, the model learns the specific dynamics and operational characteristics of the target bus, thereby enhancing its precision and reliability.

351 *Fine-Tuning Dataset:* The fine-tuning dataset consists of voltage trajectories from the target bus.
 352 We transform each of these trajectories into QAF-DeepONet-compatible formats and assemble the
 353 fine-tuning dataset $\mathcal{D}_{\mathcal{F}} = \{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^{n_f}$, where n_f is the number of training triples for the
 354 fine-tuning dataset.

355 Let $\bar{\theta}$ represent the parameters of the pre-trained QAF-DeepONet. To fine-tune the model for the
 356 target bus, a local update is performed on $\bar{\theta}$ using the target bus dataset $\mathcal{D}_{\mathcal{F}}$, applying local Adam
 357 optimization to update the parameters to the fine-tuned values, denoted as θ^f .

358 It is important to note that the pre-trained model contains extensive information about the local
 359 voltage dynamics; therefore, the fine-tuning process must be proceed carefully to prevent catastrophic
 360 forgetting. To this end, an early stopping technique is implemented, which helps the model adapt to
 361 the dynamics of the target bus without forgetting the core features learned during the pre-training
 362 process.

363 3.3 Conformal Prediction with QAF-DeepONet

364 The predictive interval estimated in Equation 14 often fails to meet the desired coverage probability
 365 criterion. To address this issue, our proposed methodology integrates the Split Conformal Quantile-
 366 DeepONet Prediction methods introduced in [28]. The conformal prediction method ensures that the
 367 prediction intervals are calibrated to achieve the desired coverage probability, thereby improving the
 368 reliability and robustness of the predictions [32].

369 This is particularly important in power systems, where accurate and reliable post-fault trajectory pre-
 370 dictions are crucial for system stability and operational planning. Guaranteeing coverage probability
 371 of the predicted intervals enhances the assessment of risks and uncertainties associated with fault
 372 events, leading to more informed decision-making and proactive measures. This strengthens the
 373 overall resilience of the power system, enabling operators to anticipate and mitigate potential issues
 374 more effectively.

375 To ensure the coverage guarantee, the conformal approach employs a calibration dataset $\mathcal{D}_{cal} =$
 376 $\{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^{n_{cal}}$ for calibrating model predictions. Utilizing a trained QAF-DeepONet, which
 377 approximate quantile operators $\mathcal{G}_{\alpha/2}^{\theta}$ and $\mathcal{G}_{1-\alpha/2}^{\theta}$ for a specified miscoverage level $\alpha \in (0, 1)$, we
 378 first define the score function for calibration data triplets $(u^{(i)}, t^{(i)}, G^{(i)})$ as follows:

$$s(u, t, G) = \max \left\{ \mathcal{G}_{\alpha/2}^{\theta}(u)(t) - G, G - \mathcal{G}_{1-\alpha/2}^{\theta}(u)(t) \right\} \quad (16)$$

379 This score function measures the error incurred by QAF-DeepONet when the operator target G falls
 380 outside the estimated quantiles. Using this function, we compute the calibration scores for the entire
 381 calibration dataset $\{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^n$: $s_1 = s(u^{(1)}, t^{(1)}, G^{(1)}), \dots, s_n = s(u^{(n)}, t^{(n)}, G^{(n)})$.

382 Next, we calculate the calibration factor, $\hat{q}$, as the $\frac{[(n+1)(1-\alpha)]}{n}$ -th quantile of the calibration scores
 383 $s_1, \dots, s_n$. Finally, for any new test data (u_{test}, t_{test}) , we modify the original estimated predictive
 384 interval via Equation 14 to the calibrated predictive interval:

$$\widehat{CL}_c(u_{test}, t_{test}) = \left[\mathcal{G}_{\alpha/2}^{\theta}(u_{test})(t_{test}) - \hat{q}, \mathcal{G}_{1-\alpha/2}^{\theta}(u_{test})(t_{test}) + \hat{q} \right]. \quad (17)$$

385 This predicted interval is adaptive, distribution-free, and guarantees coverage probability of predicted
 386 intervals, ensuring reliable performance. The summary of the described approach is detailed in
 387 Algorithm 1.

388 4 Numerical Experiments

389 In this section, we assess the performance of our proposed methodology in estimating predictive
 390 intervals for post-fault trajectories across various scenarios.

391 4.1 Data

392 We validate the proposed methodology using the New England 39-bus test system, shown in Figure 5.
 393 Notably, we use a realistic setup where each generating unit includes detailed models for voltage and

Algorithm 1: Split Conformal QAF-DeepONet Prediction

Input: Trained QAF-DeepONet model, Calibration dataset $\mathcal{D}_{\text{cal}} = \{u^{(i)}, t^{(i)}, G^{(i)}\}_{i=1}^{n_{\text{cal}}}$,
Miscoverage level α

Output: Calibrated prediction interval for test data $(u_{\text{test}}, t_{\text{test}})$

Step 1: Calculate calibration scores

for $i = 1, \dots, n$ do

$$s_i = \max \left\{ \mathcal{G}_{\alpha/2}^\theta(u^{(i)}, t^{(i)}) - G^{(i)}, G^{(i)} - \mathcal{G}_{1-\alpha/2}^\theta(u^{(i)}, t^{(i)}) \right\}$$

Step 2: Calculate calibration factor

Sort $s_1, s_2, \dots, s_n$ in ascending order, then calculate the calibration quantile:

$$\hat{q} \leftarrow \left[\frac{(n+1)(1-\alpha)}{n} \right]\text{-th quantile of the calibration scores } s_1, \dots, s_n$$

Step 3: Calibrate prediction interval

for new test data $(u_{\text{test}}, t_{\text{test}})$ compute:

$$\widehat{CL}_c(u_{\text{test}}, t_{\text{test}}) = [\mathcal{G}_{\alpha/2}^\theta(u_{\text{test}}, t_{\text{test}}) - \hat{q}, \mathcal{G}_{1-\alpha/2}^\theta(u_{\text{test}}, t_{\text{test}}) + \hat{q}]$$

return $\widehat{CL}_c(u_{\text{test}}, t_{\text{test}})$

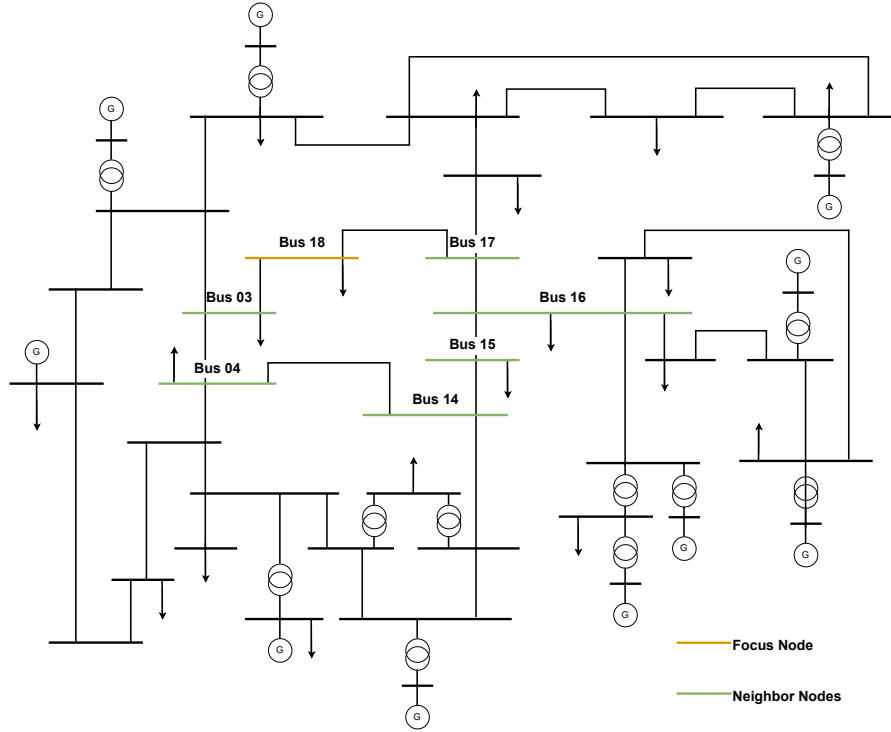


Figure 5: New England 39-bus test system.

394 frequency controllers, specifically an automatic voltage regulator and governor. The parameters for
395 the network, generators, and controllers are sourced from the DIGSILENT PowerFactory library.

396 To generate realistic datasets, voltage trajectories are generated for each bus under the following
397 simulation conditions:

- 398 • *Demand variation:* Active and reactive power demands are randomly varied by $\pm 30\%$ of
399 their nominal values.
- 400 • *Network contingencies:* In each simulation, (i) a three-phase short circuit occurs in one of
401 the system's transmission lines, representing an N-1 contingency, (ii) after the fault, the line

is isolated following the clearing time, and (iii) the fault location is randomly chosen along the line, ranging from 1% to 99% of its total length.

- *Clearing time*: The fault clearing time is uniformly sampled between 100 ms and 333 ms.

Furthermore, throughout this paper, we denote the number of voltage trajectories used for pre-training, fine-tuning, and calibration by $N_{\mathcal{P}}$, $N_{\mathcal{F}}$, and N_{cal} , respectively.

4.2 The Model

We employ the proposed QAF-DeepONet, configured as outlined in Section 3.1.4. The hyperparameters of the model are selected through a hyperparameter tuning process.

4.3 Training Protocol

Our goal is to predict post-fault voltage trajectory intervals of a target bus in the power system. To this end, we consider the neighboring buses shown in Figure 5. In this neighborhood, there are 7 buses; we select Bus 18 as the target bus and the remaining six buses as its neighboring ones.

We aim to predict the 95% predictive interval (with a miscoverage level $\alpha = 0.05$) for the post-fault voltage trajectories of the target bus. Specifically, we designed the QAF-DeepONet to predict the 0.025-th and 0.975-th quantiles, which represent the lower and upper bounds of the interval. The 95% predictive interval is then constructed using these two quantiles.

4.4 Evaluation Metrics

To evaluate QAF-DeepONet’s predicted intervals for post-fault trajectories, we use two metrics: Prediction Interval Coverage Probability (PICP) and Prediction Interval Normalized Width Average (PINAW).

Prediction Interval Coverage Probability (PICP): PICP measures the proportion of actual values within the predicted intervals:

$$\text{PICP} = \frac{1}{N} \sum_{i=1}^N I(y_i \in [\hat{y}_{i,\text{lower}}, \hat{y}_{i,\text{upper}}]), \quad (18)$$

where N is the number of data points, y_i is the actual value, $\hat{y}_{i,\text{lower}}$ and $\hat{y}_{i,\text{upper}}$ are the bounds of the predicted interval, and I is an indicator function. A higher PICP indicates better reliability.

Prediction Interval Normalized Width Average (PINAW): PINAW assesses the average width of prediction intervals relative to data variability:

$$\text{PINAW} = \frac{1}{N} \sum_{i=1}^N \frac{\hat{y}_{i,\text{upper}} - \hat{y}_{i,\text{lower}}}{y_{\max} - y_{\min}}, \quad (19)$$

where $y_{\max}$ and $y_{\min}$ are the maximum and minimum observed values. Generally, lower PINAW values suggest more precise predictive intervals.

Together, these metrics evaluate the reliability and precision of the predictive performance of the model.

It should be noted that to evaluate the models, we consider 100 test trajectories, which are unseen during the training process, and report the average PICP and PINAW values across all test trajectories in the following analyses.

4.5 Model Evaluation

Figure 6 illustrates the prediction results of the fine-tuned model alongside the corresponding conformal predictions for four sample voltage trajectories. For these results, the model was initially pre-trained using $N_{\mathcal{P}} = 6000$ trajectories from neighboring buses, subsequently fine-tuned with $N_{\mathcal{F}} = 3000$ trajectories from the target bus, and finally calibrated using $N_{cal} = 100$ trajectories from the same bus to calibrate the predictions within the conformal prediction framework. These

441 predictions consider the scenario with a relatively short observable post-fault duration, $\Delta t_{obs} = 400$
 442 ms. Despite this, the fine-tuned model provides precise and reliable predictive intervals for post-fault
 443 trajectories. However, the fine-tuned prediction does not cover some parts of the true trajectory. To
 444 address this, conformal prediction adjusts the predicted interval, ensuring that most parts are covered
 445 and guaranteeing the required coverage probability, leading to more reliable predictions.

446 In the following sections, we analyze both pre-trained and fine-tuned models to evaluate the ef-
 447 fectiveness of pre-training in capturing targeted mappings and the improvements achieved through
 448 fine-tuning. In addition, we evaluate the results of conformal prediction, discussing its effectiveness,
 449 advantages, and limitations. For each model, we will perform a sensitivity analysis to understand
 450 the impact of observable post-fault duration Δt_{obs} and the number of trajectories used during the
 451 training process on the quality of the predicted intervals. Finally, we will compare these results to
 452 highlight key insights and discern notable differences.

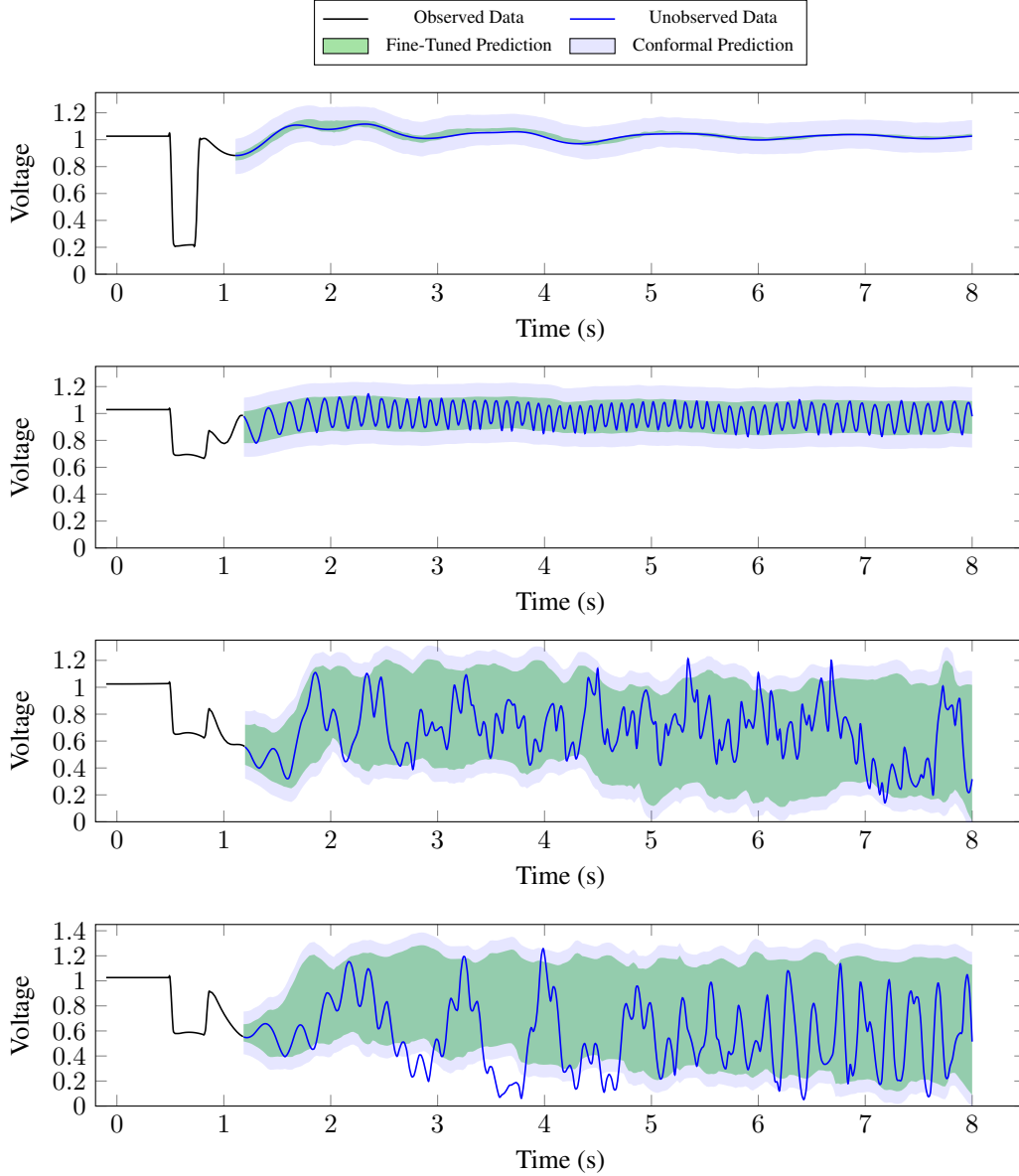


Figure 6: Predicted 95% predictive interval of post-fault trajectories using the fine-tuned model ($N_{\mathcal{P}} = 6000$, $N_{\mathcal{F}} = 3000$) and corresponding conformal predictions ($N_{\text{cal}} = 100$) for four sample voltage trajectories with an observable post-fault duration of $\Delta t_{obs} = 400$ ms.

4.5.1 Evaluating the Pre-trained Model

We evaluated the zero-shot prediction capabilities of the pre-trained model, which means its ability to make accurate predictions without any additional fine-tuning. This is done by testing the model on data from the target bus whose dynamics was not part of the training dataset. This evaluation is crucial because it provides insight into the adaptability and potential of the model in scenarios without fine-tuning.

Table 1 shows the average metric values for the estimated predictive intervals of the pre-trained models for post-fault trajectories on 100 test trajectories. For these predictions, we consider a scenario where the observable post-fault duration is $\Delta t_{obs} = 500$ ms and the pre-trained model is trained on $N_P = 6000$ voltage trajectories from neighboring buses. The results show relatively high PICP values and relatively low PINAW values, highlighting the models' high performance in zero-shot prediction scenarios. This combination underscores the effectiveness of the models in providing reliable and precise predictions even without fine-tuning.

We also performed sensitivity experiments to evaluate how different parameters affect its predictive capabilities. That is, we assess the model's performance by varying observable post-fault durations Δt_{obs} and training dataset sizes. Figure 7 illustrates the zero-shot prediction performance across these variables, showing that increasing both Δt_{obs} and dataset size improves the quality of the model's predicted intervals. In the following, we discuss the effects of these factors in detail.

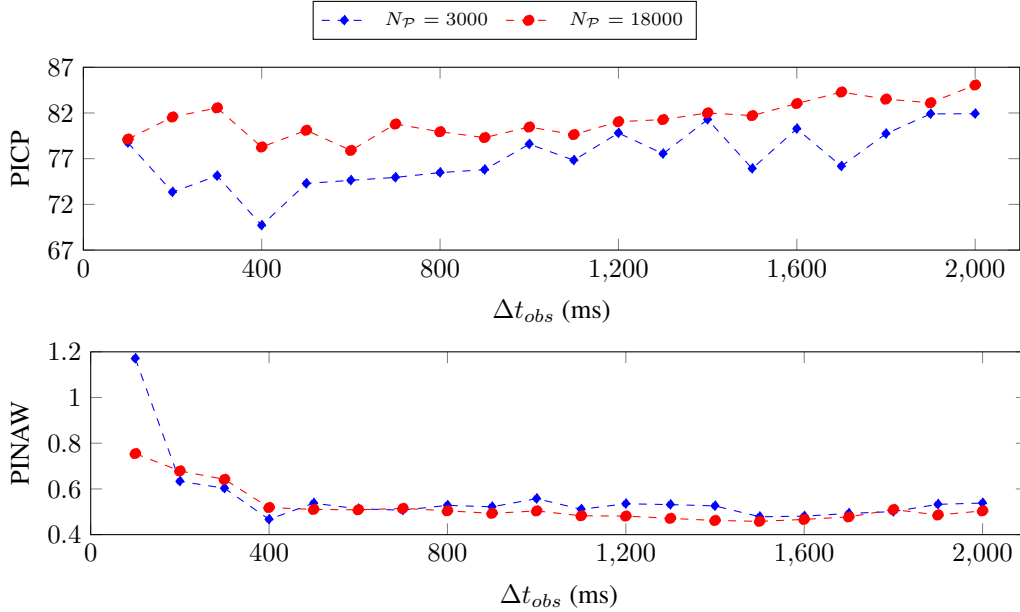


Figure 7: Average value of PICP and PINAW metrics for predicted intervals from pre-trained models, designed for various observable post-fault durations Δt_{obs} and trained on different dataset sizes, evaluated across 100 test trajectories.

Observable Post-Fault Duration Effect: Figure 7 shows that increasing Δt_{obs} maintains the mean PICP within a comparable range. However, increasing Δt_{obs} up to approximately 600 ms leads to a decrease in the value of the PINAW metric, resulting in narrower and more precise predicted intervals. Specifically, we find that PINAW decreases with respect to Δt_{obs} for $\Delta t_{obs} < 600$ ms, following a power law. For instance, for the model trained on the pre-training dataset containing $N_P = 3000$ voltage trajectories, we have:

$$\text{PINAW} \propto \Delta t_{obs}^{-0.53} \quad (20)$$

Beyond $\Delta t_{obs} > 600$ ms, PINAW converges to a constant value, indicating that further increasing Δt_{obs} does not yield additional gains. This is likely due to data saturation, where the model has already captured the most significant patterns and variations in the post-fault trajectories by the time Δt_{obs} reaches 600 ms.

481 *Dataset Size Effect:* Increasing the dataset size from 3000 to 18000 significantly improves the PICP
 482 metric while keeping the PINAW metric stable. This enhancement indicates that a larger dataset
 483 enables the model to capture a more comprehensive range of patterns and variations in post-fault
 484 trajectories, leading to more reliable predictions.

485 4.5.2 Evaluating the Fine-tuned Model

486 After pre-training the QAF-DeepONet using data from neighboring buses, we further fine-tune it
 487 with data from the target bus to enhance predictions for that specific bus.

488 Table 1 presents the average metric values for the fine-tuned model’s estimated predictive intervals of
 489 post-fault trajectories across 100 test cases, with the observable post-fault duration set to $\Delta t_{obs} = 500$
 490 ms. The corresponding pre-trained model is trained on $N_{\mathcal{P}} = 6000$ voltage trajectories from
 491 neighboring buses, while the fine-tuning is conducted on $N_{\mathcal{F}} = 3000$ voltage trajectories from
 492 the target bus. The results indicate that the fine-tuned model achieves higher PICP values while
 493 maintaining comparable PINAW values, demonstrating that, compared to the pre-trained model,
 494 fine-tuning enhances model performance and reliability in post-fault prediction.

	PICP%	PINAW
Pre-trained Model	76.18	0.53
Fine tuned Model	81.47	0.51

Table 1: Average value of PICP and PINAW metrics for estimated predictive intervals of post-fault trajectories across 100 test cases, with the observable post-fault duration set to $\Delta t_{obs} = 500$ ms. The pre-trained model was trained on $N_{\mathcal{P}} = 6000$ voltage trajectories from neighboring buses, while the fine-tuned model was refined using $N_{\mathcal{F}} = 3000$ voltage trajectories from the target bus. The results demonstrate the fine-tuned model’s improved performance, reflected in higher PICP values and comparable PINAW values, indicating enhanced reliability in post-fault prediction.

495 Figure 8 illustrates the average metrics values for the fine-tuned model predictions across various
 496 Δt_{obs} and training dataset sizes for two different fine-tuning dataset sizes. As observed, similar to the
 497 pre-trained model, increasing the duration of observation time and the size of the training dataset
 498 enhances the model’s prediction performance.

499 *Observable Post-Fault Duration Effect:* The Figure 8 shows that increasing Δt_{obs} enhances both
 500 metrics. Specifically, for $\Delta t_{obs} < 1000$ ms, the value of PICP increases with increasing observable
 501 duration, and beyond this point, it almost stabilizes. On the other hand, for $\Delta t_{obs} < 400$ ms, the value
 502 of PINAW decreases with increasing observable duration and then stabilizes. For $\Delta t_{obs} > 1000$ ms,
 503 both the PICP and PINAW values stabilize, indicating that further increases in Δt_{obs} do not provide
 504 additional gain. This behavior is consistent with what we observed in the zero-shot prediction.

505 *Dataset Size Effect:* For $\Delta t_{obs} < 1000$ ms, similar to zero-shot prediction, increasing the dataset size
 506 from $N_{\mathcal{F}} = 500$ to 3000 in fine-tuning results in an improvement of the model’s predicted intervals,
 507 reflected by a higher PICP, while maintaining the PINAW metric relatively stable. On the other hand,
 508 for $\Delta t_{obs} > 1000$ ms, the PICP metric for these datasets will almost converge. As the observation
 509 time increases, the model acquires more information about the system, reducing the amount of data
 510 required to achieve similar levels of accuracy and precision in its predictions. Consequently, for
 511 higher values of Δt_{obs} , the PICP values converge for both dataset sizes.

512 4.5.3 Evaluating Conformal Prediction

513 As illustrated in Figure 8, the average PICP values for the predicted intervals from the fine-tuned
 514 model are below the desired training goal of 95%. To address this, our proposed methodology
 515 integrates conformal prediction with the fine-tuned model results, modifying the predicted intervals
 516 to ensure that they achieve the desired coverage using calibration data. To this end, we calibrate the
 517 fine-tuned predictions using the calibration dataset $\mathcal{D}_{cal}$, consisting of data from $N_{cal} = 100$ unseen
 518 voltage trajectories of the target bus.

519 Figure 9 shows the average values of the metrics for the conformal prediction results. As illustrated,
 520 the predicted intervals for almost all observation times Δt_{obs} and both dataset sizes converge to the
 521 target value of 95%. However, for models trained on smaller datasets and with shorter observation

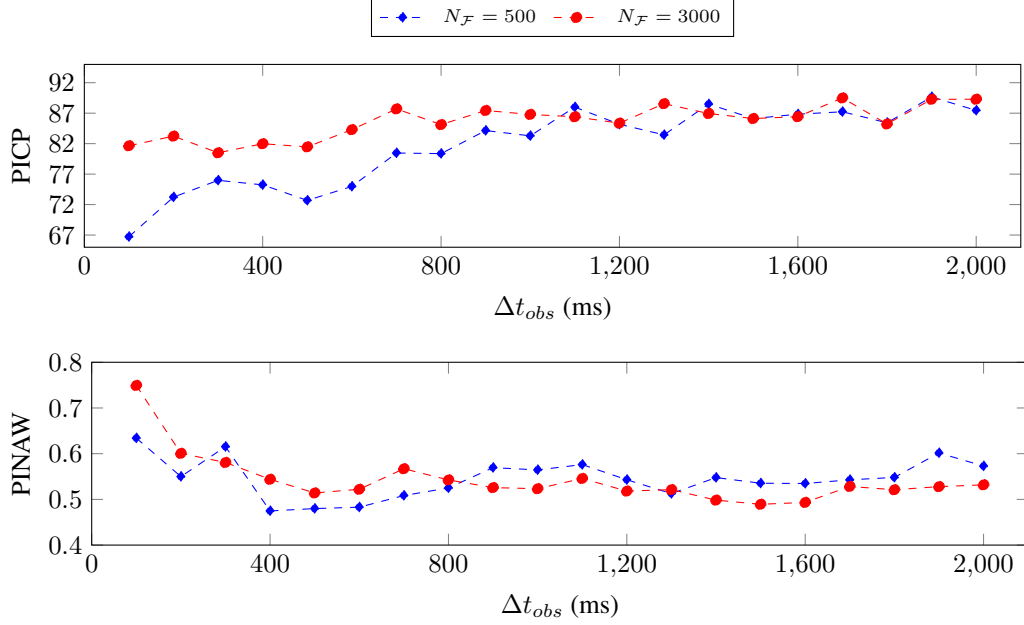


Figure 8: Average value of PICP and PINAW metrics for predicted intervals from fine-tuned models, designed for various observable post-fault durations Δt_{obs} and trained on different dataset sizes, evaluated across 100 test trajectories. The corresponding pre-training for these models was conducted using $N_{\mathcal{P}} = 6000$ voltage trajectories from neighboring buses.

times, the base estimator (trained model) exhibits higher uncertainty. In such conditions, the conformal prediction PICP does not get as close to 95% due to insufficient variability and information for accurate calibration.

While conformal prediction adjusts the estimated predictive intervals to achieve the desired coverage, it results in increased PINAW values, leading to wider intervals. Figure 9 shows that for the observable parts with $\Delta t_{obs} < 1000$, the larger dataset results in lower PINAW values compared to the smaller dataset. This can be attributed to the larger dataset providing more comprehensive coverage of various post-fault scenarios, allowing the model to make more precise predictions. However, after $\Delta t_{obs} > 1000$, the predictive performance of both models converges, resulting in comparable PINAW values. This convergence suggests that beyond a certain point, the benefit of having a larger dataset diminishes, and both models face similar levels of uncertainty, leading to comparable interval widths to ensure the desired coverage.

As illustrated in Figure 6, although conformal prediction uniformly extends the width of the predicted interval for both stable and unstable trajectories, the relative impact varies between the two. For stable trajectories, which generally have less variation and narrower intervals, this fixed increase results in a relatively larger expansion. In contrast, for unstable trajectories with greater variation and wider intervals, the same fixed increase results in a relatively smaller expansion. Thus, while the absolute increase in interval width is constant, its relative effect is more significant for stable trajectories and less pronounced for unstable ones. Despite this, conformal prediction still guarantees the desired coverage and enhances practical usability. Therefore, in power systems where the prediction of unstable post-fault trajectories is of high priority, incorporating conformal prediction into the approach is highly beneficial.

4.5.4 Comparative Analysis of Results from Pre-trained Model, Fine-Tuned Model, and Conformal Prediction

Figure 10 compares the pre-trained model results with those of the fine-tuned model and its corresponding conformal predictions. In this comparison, the QAF-DeepONet is first pre-trained with a dataset of $N_{\mathcal{P}} = 6000$ voltage trajectories from neighboring buses and then fine-tuned using

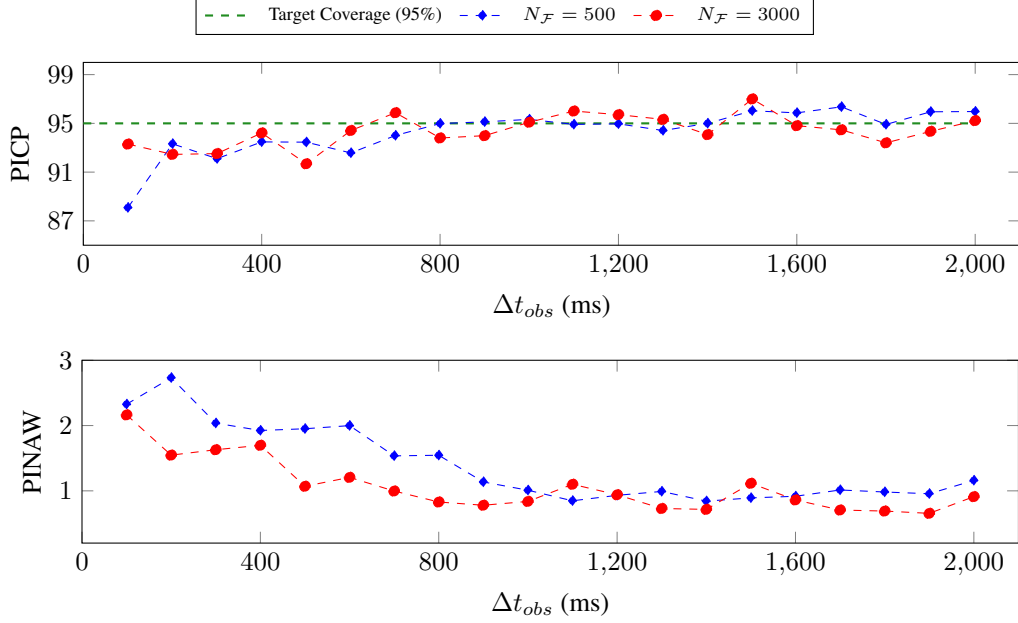


Figure 9: Average value of PICP and PINAW metrics for predicted intervals using the conformal prediction technique integrated with fine-tuned model results, presented across different observation post-fault durations Δt_{obs} and training dataset sizes. The pre-training for the fine-tuned models was conducted using $N_P = 6000$ voltage trajectories from neighboring buses, and $N_{cal} = 100$ trajectories from the target bus were used to calibrate the predicted intervals of fine-tuned models within the conformal framework.

549 $N_F = 3000$ trajectories from the target bus. Subsequently, conformal prediction is then applied to
550 the fine-tuned results, using $N_{cal} = 100$ trajectories from the target bus for calibration.

551 As shown in Figure 10, the predicted intervals by the fine-tuned model primarily differ from the
552 pre-trained model in their corresponding PICP values, while their PINAW values remain mostly
553 comparable. The PICP for the fine-tuned model is considerably higher compared to zero-shot
554 prediction, demonstrating the effectiveness of fine-tuning in improving prediction performance.
555 This improvement indicates that fine-tuning enables the model to better capture the specific voltage
556 dynamics of the target bus data, thereby enhancing its predictive performance.

557 Comparing the results of the fine-tuned model with its conformal prediction, we observe that for
558 almost all Δt_{obs} , the PICP converges to the desired training goal of 95%, which the pre-trained and
559 fine-tuned models do not achieve. However, conformal prediction leads to higher PINAW values
560 compared to the fine-tuned and pre-trained models, which presents practical challenges. There is a
561 trade-off between the guaranteed coverage probability from the conformal prediction and the width
562 of the predicted intervals. Although conformal prediction ensures coverage and reliability, larger
563 intervals can lead to overly conservative decisions with higher costs. Therefore, this balance should
564 be carefully considered based on the system's objectives.

565 5 Future Work

566 The conformal prediction method guarantees that the PICP converges to the desired value. However,
567 this method does not address the PINAW metric, resulting in wider prediction intervals during the
568 calibration process. PINAW is a crucial metric for effective decision-making as it directly impacts
569 the practical usability of the prediction intervals.

570 To address this limitation, our future work will incorporate conformal risk control[2]. This enhanced
571 approach will not only ensure the convergence of PICP to the desired goal but will also consider and
572 optimize PINAW. By doing so, we aim to develop a more comprehensive methodology for estimating

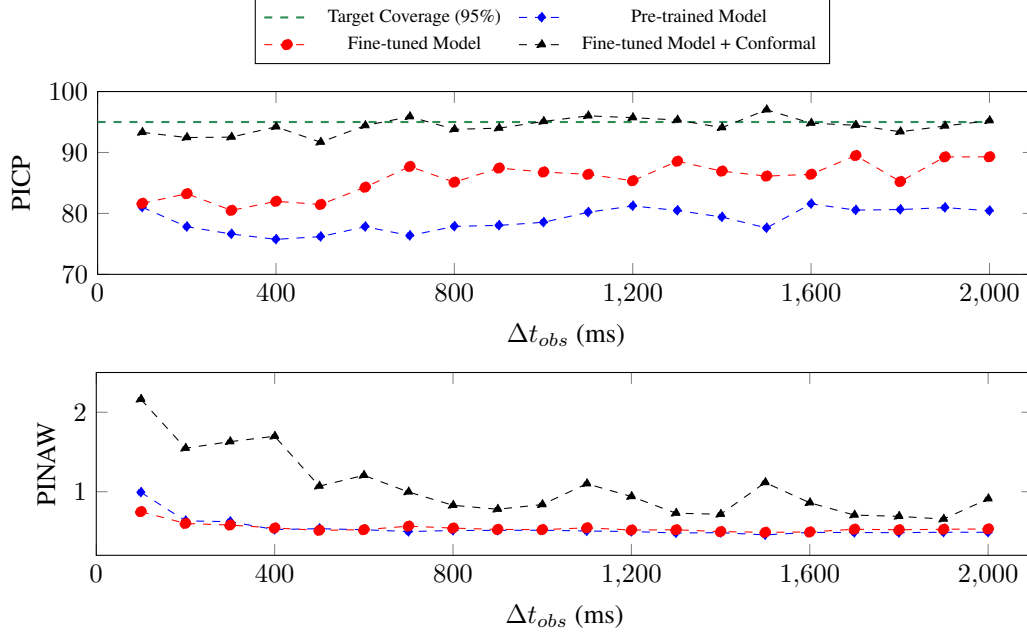


Figure 10: Comparative analysis of predicted intervals by the pre-trained model, fine-tuned model, and conformal prediction, showing differences in PICP and PINAW average values across various Δt_{obs} . The QAF-DeepONet was pre-trained with $N_{\mathcal{P}} = 6000$ voltage trajectories from neighboring buses and fine-tuned with $N_{\mathcal{F}} = 3000$ trajectories from the target bus. Conformal prediction was then applied to the fine-tuned results, using $N_{cal} = 100$ trajectories from the target bus for calibration

573 predictive intervals of post-fault trajectories that balances coverage probability and interval width,
574 thereby improving its practical applicability in decision-making.

575 In addition, this work represents a key step in developing foundation models [3] for transient stability.
576 Moving forward, we aim to design merging and federated pre-training strategies for transformers [43]
577 that enable creating larger, more capable foundation models that can effectively forecast trajectories
578 of complex power system responses.

579 6 Conclusion

580 This paper presents a novel methodology for estimating predictive intervals for post-fault voltage
581 trajectories in power systems using deep operator learning techniques. We introduce the Quantile
582 Attention-Fourier Deep Operator Network (QAF-DeepONet), which combines deep operator net-
583 works with attention mechanisms and Fourier features to handle complex voltage trajectory patterns.
584 QAF-DeepONet maps the observed segments of the voltage trajectory to predictive intervals for the
585 post-fault voltage trajectory. To address the low-regime data issue in power systems, the methodology
586 incorporates a two-stage training approach. We initially pre-train the QAF-DeepONet with data
587 from neighboring buses to learn the underlying dynamics within voltage trajectories. This approach
588 makes maximum use of available data and ensures each bus's data privacy through a federated
589 learning method that avoids direct data sharing. Subsequently, we fine-tune the model using data
590 from the target bus to learn the specific characteristics of the dynamics of its voltage trajectories.
591 Next, we integrate conformal prediction with the fine-tuned model's predictions to provide coverage
592 guarantees. We evaluated our approach using data from the IEEE-39 bus test system through a series
593 of comprehensive analyzes.

594 The results underscore the efficacy of the proposed methodology, demonstrating precise and reli-
595 able performance in zero-shot predictions for estimating the predictive interval. Furthermore, the
596 results demonstrate that fine-tuning improves the reliability and robustness of the model predictions.
597 Additionally, the integration of conformal prediction techniques enhances the model's reliability by
598 ensuring that the predicted intervals achieve the target coverage probability. However, this integration

also tends to result in wider intervals, illustrating the inherent trade-off between reliability and precision.

Acknowledgement

GL, AM and CM would like to thank the support of National Science Foundation (DMS-2053746, DMS-2134209, ECCS-2328241, CBET-2347401 and OAC-2311848), and U.S. Department of Energy (DOE) Office of Science Advanced Scientific Computing Research program DE-SC0023161, the Uncertainty Quantification for Multifidelity Operator Learning (MOLUcQ) project (Project No. 81739), and DOE–Fusion Energy Science, under grant number: DE-SC0024583.

References

- [1] Nima Amjady and Seyed Farough Majedi. Transient stability prediction by a hybrid intelligent system. *IEEE Transactions on Power Systems*, 22(3):1275–1283, 2007.
- [2] Anastasios N Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. *arXiv preprint arXiv:2208.02814*, 2022.
- [3] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.
- [4] Tianping Chen and Hong Chen. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*, 6(4):911–917, 1995. <https://doi.org/10.1109/72.392253>.
- [5] Alexandre J. Chorin, Ole H. Hald, and Raz Kupferman. Optimal prediction and the mori–zwanzig representation of irreversible processes. *Proceedings of the National Academy of Sciences*, 97:2968–2973, 3 2000.
- [6] Zhao Yang Dong, Yan Xu, Pei Zhang, and Kit Po Wong. Using is to assess an electric power system’s real-time stability. *IEEE Intelligent Systems*, 28(4):60–66, 2013.
- [7] Somdatta Goswami, Minglang Yin, Yue Yu, and George Em Karniadakis. A physics-informed variational deepoNet for predicting crack path in quasi-brittle materials. *Computer Methods in Applied Mechanics and Engineering*, 391:114587, 2022.
- [8] Zhongkai Hao, Jiachen Yao, Chang Su, Hang Su, Ziao Wang, Fanzhi Lu, Zeyu Xia, Yichi Zhang, Songming Liu, Lu Lu, et al. Pinnacle: A comprehensive benchmark of physics-informed neural networks for solving pdes. *arXiv preprint arXiv:2306.08827*, 2023.
- [9] Nikos Hatziaargyriou, Jovica Milanovic, Claudia Rahmann, Venkataramana Ajjarapu, Claudio Canizares, Istvan Erlich, David Hill, Ian Hiskens, Innocent Kamwa, Bikash Pal, Pouyan Pourbeik, Juan Sanchez-Gasca, Aleksandar Stankovic, Thierry Van Cutsem, Vijay Vittal, and Costas Vournas. Definition and classification of power system stability – revisited & extended. *IEEE Transactions on Power Systems*, 36:3271–3281, 7 2021.
- [10] Junyan He, Seid Koric, Shashank Kushwaha, Jaewan Park, Diab Abueidda, and Iwona Jasiuk. Novel deepoNet architecture to predict stresses in elastoplastic structures with variable complex geometries and loads. *Computer Methods in Applied Mechanics and Engineering*, 415:116277, 2023.
- [11] Amanda A. Howard, Mauro Perego, George Em Karniadakis, and Panos Stinis. Multifidelity deep operator networks for data-driven and physics-informed problems. *Journal of Computational Physics*, 493:112462, 2023.
- [12] Anran Jiao, Qile Yan, Jhn Harlim, and Lu Lu. Solving forward and inverse pde problems on unknown manifolds via physics-informed neural operators. *arXiv preprint arXiv:2407.05477*, 2024.

- [13] Pengzhan Jin, Shuai Meng, and Lu Lu. Mionet: Learning multiple-input operators via tensor product. *SIAM Journal on Scientific Computing*, 44(6):A3490–A3514, 2022.
- [14] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.
- [15] Chensen Lin, Zhen Li, Lu Lu, Shengze Cai, Martin Maxey, and George Em Karniadakis. Operator learning for predicting multiscale bubble growth dynamics. *The Journal of Chemical Physics*, 154(10), 2021.
- [16] Guang Lin, Christian Moya, and Zecheng Zhang. B-deeponet: An enhanced bayesian deeponet for solving noisy parametric pdes using accelerated replica exchange sgld. *Journal of Computational Physics*, 473:111713, 2023.
- [17] Guang Lin, Christian Moya, and Zecheng Zhang. Learning the dynamical response of nonlinear non-autonomous dynamical systems with deep operator neural networks. *Engineering Applications of Artificial Intelligence*, 125:106689, 2023.
- [18] Xianzhuang Liu, Xiaohua Zhang, Lei Chen, Fei Xu, and Changyou Feng. Data-driven transient stability assessment model considering network topology changes via mahalanobis kernel regression and ensemble learning. *Journal of Modern Power Systems and Clean Energy*, 8(6):1080–1091, 2020.
- [19] Yu Liu, Abhinav Kumar Singh, Junbo Zhao, A. P. Sakis Meliopoulos, Bikash Pal, Mohd Aifaa bin Mohd Ariff, Thierry Van Cutsem, Mevludin Glavic, Zhenyu Huang, Innocent Kamwa, Lamine Mili, Abdul Saleem Mir, Ahmad Taha, Vladimir Terzija, and Shenglong Yu. Dynamic state estimation for power system control and protection. *IEEE Transactions on Power Systems*, 36(6):5909–5921, 2021.
- [20] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.
- [21] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. <https://doi.org/10.1038/s42256-021-00302-5>.
- [22] Lu Lu, Xuhui Meng, Shengze Cai, Zhiping Mao, Somdatta Goswami, Zhongqiang Zhang, and George Em Karniadakis. A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data. *Computer Methods in Applied Mechanics and Engineering*, 393:114778, 2022.
- [23] Shunyu Mao, Ruobing Dong, Lu Lu, Kwang Moo Yi, Sifan Wang, and Paris Perdikaris. Ppdonet: Deep operator networks for fast prediction of steady-state solutions in disk–planet systems. *The Astrophysical Journal Letters*, 950(2):L12, jun 2023.
- [24] Amirhossein Mollaali, Izzet Sahin, Iqar Raza, Christian Moya, Guillermo Paniagua, and Guang Lin. A physics-guided bi-fidelity fourier-featured operator learning framework for predicting time evolution of drag and lift coefficients. *Fluids*, 8(12), 2023.
- [25] K. Morison, Lei Wang, and P. Kundur. Power system security assessment. *IEEE Power and Energy Magazine*, 2(5):30–39, 2004.
- [26] Christian Moya and Guang Lin. Fed-deeponet: Stochastic gradient-based federated training of deep operator networks. *Algorithms*, 15(9), 2022.
- [27] Christian Moya, Guang Lin, Tianqiao Zhao, and Meng Yue. On approximating the dynamic response of synchronous generators via operator learning: A step towards building deep operator-based power grid simulators. *arXiv preprint arXiv:2301.12538*, 2023.
- [28] Christian Moya, Amirhossein Mollaali, Zecheng Zhang, Lu Lu, and Guang Lin. Conformalized-deeponet: A distribution-free framework for uncertainty quantification in deep operator networks. *arXiv preprint arXiv:2402.15406*, 2024.

- [29] Christian Moya, Shiqi Zhang, Guang Lin, and Meng Yue. Deeponet-grid-uq: A trustworthy deep operator framework for predicting the power grid's post-fault trajectories. *Neurocomputing*, 535:166–182, 2023.
- [30] Muhammad Nadeem, Sebastian A. Nugroho, and Ahmad F. Taha. Dynamic state estimation of nonlinear differential algebraic equation models of power networks. *IEEE Transactions on Power Systems*, 38(3):2539–2552, 2023.
- [31] Avishek Paul, Innocent Kamwa, and Geza J6os. Centralized dynamic state estimation using a federation of extended kalman filters with intermittent PMU data from generator terminals. *IEEE Transactions on Power Systems*, 33(6):6109–6119, 2018.
- [32] Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. *Advances in neural information processing systems*, 32, 2019.
- [33] Izzet Sahin, Christian Moya, Amirhossein Mollaali, Guang Lin, and Guillermo Paniagua. Deep operator learning-based surrogate models with uncertainty quantification for optimizing internal cooling channel rib profiles. *International Journal of Heat and Mass Transfer*, 219:124813, 2024.
- [34] Peter W. Sauer, M. A. Pai, and J. H. Chow. *Power system dynamics and stability : with synchrophasor measurement and power system toolbox*. IEEE Press : Wiley, Hoboken, NJ, USA, second edition edition, 2017.
- [35] Jan Shair, Haozhi Li, Jiabing Hu, and Xiaorong Xie. Power system stability issues, classifications and research prospects in the context of high-penetration of renewables and power electronics. *Renewable and Sustainable Energy Reviews*, 145:111111, 7 2021.
- [36] Zonghe Shao, Qichao Wang, Yuzhe Cao, Defu Cai, Yang You, and Renzhi Lu. A novel data-driven lstm-saf model for power systems transient stability assessment. *IEEE Transactions on Industrial Informatics*, 20(7):9083–9097, 2024.
- [37] Zhongtuo Shi, Wei Yao, Lingkang Zeng, Jianfeng Wen, Jiakun Fang, Xiaomeng Ai, and Jinyu Wen. Convolutional neural network-based power system transient stability assessment and instability mode prediction. *Applied Energy*, 263:114586, 4 2020.
- [38] Yixuan Sun, Christian Moya, Guang Lin, and Meng Yue. Deepgraphonet: A deep graph operator network to learn and zero-shot transfer the dynamic response of networked systems. *IEEE Systems Journal*, 17(3):4360–4370, 2023.
- [39] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.
- [40] Vladimir Terzija, Gustavo Valverde, Deyu Cai, Pawel Regulski, Vahid Madani, John Fitch, Srdjan Skok, Miroslav M. Begovic, and Arun Phadke. Wide-area monitoring, protection, and control of future electric power networks. *Proceedings of the IEEE*, 99(1):80–93, 2011.
- [41] Tapas Tripura and Souvik Chakraborty. Wavelet neural operator for solving parametric partial differential equations in computational mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 404:115783, 2023.
- [42] T. Van Cutsem. Voltage instability: phenomena, countermeasures, and analysis methods. *Proceedings of the IEEE*, 88(2):208–227, 2000.
- [43] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- [44] Vijay Vittal, James D. McCalley, Paul M. Anderson, and A. A. Fouad. *Power System Control and Stability*. John Wiley & Sons, 2019. Google-Books-ID: obCuDwAAQBAJ.
- [45] Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. *Science advances*, 7(40):eabi8605, 2021.

- 742 [46] Xueyan Yin and Yutian Liu. Deep learning based feature reduction for power system transient
743 stability assessment. In *TENCON 2018 - 2018 IEEE Region 10 Conference*, pages 2308–2312,
744 2018.
- 745 [47] Reza Yousefian and Sukumar Kamalasadan. A review of neural network based machine learning
746 approaches for rotor angle stability control.
- 747 [48] Zecheng Zhang, Wing Tat Leung, and Hayden Schaeffer. Belnet: Basis enhanced learning, a
748 mesh-free neural operator. *arXiv preprint arXiv:2212.07336*, 2022.
- 749 [49] Zecheng Zhang, Christian Moya, Wing Tat Leung, Guang Lin, and Hayden Schaeffer. Bayesian
750 deep operator learning for homogenized to fine-scale maps for multiscale pde. *Multiscale
751 Modeling & Simulation*, 22(3):956–972, 2024.
- 752 [50] Zecheng Zhang, Christian Moya, Lu Lu, Guang Lin, and Hayden Schaeffer. D2no: Efficient han-
753 dling of heterogeneous input function spaces with distributed deep neural operators. *Computer
754 Methods in Applied Mechanics and Engineering*, 428:117084, 2024.
- 755 [51] Junbo Zhao, Antonio Gómez-Expósito, Marcos Netto, Lamine Mili, Ali Abur, Vladimir Terzija,
756 Innocent Kamwa, Bikash Pal, Abhinav Kumar Singh, Junjian Qi, Zhenyu Huang, and A. P. Sakis
757 Meliopoulos. Power system dynamic state estimation: Motivations, definitions, methodologies,
758 and future work. *IEEE Transactions on Power Systems*, 34(4):3188–3198, 2019.
- 759 [52] Min Zhu, Shihang Feng, Youzuo Lin, and Lu Lu. Fourier-deeponet: Fourier-enhanced deep
760 operator networks for full waveform inversion with improved accuracy, generalizability, and
761 robustness. *Computer Methods in Applied Mechanics and Engineering*, 416:116300, 2023.