

Robust Power System State Estimation Using Physics-Informed Neural Networks

Solon Falas¹, Markos Asprou¹, *Member, IEEE*, Charalambos Konstantinou², *Senior Member, IEEE*, and Maria K. Michael¹, *Member, IEEE*

Abstract—Modern power systems face significant challenges in state estimation and real-time monitoring, particularly regarding response speed and accuracy under faulty conditions or cyber-attacks. This article proposes a hybrid approach using physics-informed neural networks (PINNs) to enhance the accuracy and robustness of power system state estimation. By embedding physical laws into the neural network architecture, PINNs improve estimation accuracy for transmission grid applications under both normal and faulty conditions, while also showing potential in addressing security concerns, such as data manipulation attacks. Experimental results show that the proposed approach outperforms traditional machine learning models, achieving up to ~83% higher accuracy on unseen subsets of the training dataset and ~65% better performance on entirely new, unrelated datasets. Experiments also show that during a data manipulation attack against a critical bus in a system, the PINN can be up to ~93% more accurate than an equivalent neural network.

Index Terms—Machine learning (ML), physics-informed neural networks (PINNs), power systems, state estimation.

I. INTRODUCTION

THE escalating global electricity demand, driven by rapid urbanization, transportation electrification, and digital technology proliferation, has underscored the need for robust and stable power systems. Ensuring the stability and security of critical infrastructures, particularly power transmission networks, is essential for economic stability and public safety.

Received 13 February 2025; revised 18 May 2025; accepted 10 June 2025. Date of publication 14 July 2025; date of current version 8 October 2025. This work was supported in part by the European Union's Horizon 2020 research and innovation programme under Grant 739551, in part the Republic of Cyprus through the Deputy Ministry of Research, Innovation and Digital Policy, the Cyprus Research and Innovation Foundation under Grant CODEVELOP-REPowerEU/1223/91 (GridGnosis), in part by the European Union's NextGenerationEU initiative and the Recovery and Resilience Plan of Cyprus (Cyprus-Tomorrow), in part by the University of Cyprus, and in part by the King Abdullah University of Science and Technology (KAUST) under Grant ORFS-CRG11-2022-5021. Paper no. TII-25-0903. (*Corresponding author: Maria K. Michael.*)

Solon Falas, Markos Asprou, and Maria K. Michael are with the Department of Electrical and Computer Engineering, KIOS Research and Innovation Center of Excellence, University of Cyprus, 1678 Nicosia, Cyprus (e-mail: falas.solon@ucy.ac.cy; asprou.markos@ucy.ac.cy; mmichael@ucy.ac.cy).

Charalambos Konstantinou is with the Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia (e-mail: charalambos.konstantinou@kaust.edu.sa).

Digital Object Identifier 10.1109/TII.2025.3582293

However, the growing complexity of modern grids, driven by renewable energy integration, adoption of smart grid technologies, and interconnected networks, presents significant challenges in monitoring, control, and system resilience [1], [2].

In particular, addressing challenges related to real-time data management and stability has become increasingly critical, necessitating advanced monitoring schemes to ensure system stability and integrity. It is therefore important to have advanced monitoring schemes to ensure the system stability and integrity. Traditional state estimation methods, such as weighted least squares, rely on Supervisory Control and Data Acquisition systems and iterative processes, but they are limited by insufficient real-time capabilities, high computational demands, and reliance on sparse data [3], [4]. These limitations are especially problematic with intermittent renewable energy sources, such as solar and wind, which cause rapid fluctuations and transient errors. Phasor measurement units (PMUs) enhance real-time monitoring with high-precision synchronized data, but conventional methods still fall short in addressing the complexity of modern systems [5].

Compounding these technical challenges, power grids have simultaneously become prime targets for cyber-attacks, such as denial of service and false data injection attacks [6], [7]. Such sensor manipulation attacks can compromise the integrity of grid monitoring, leading to potential outages, and severe physical damage [8], [9]. Therefore, a more cyber-physical approach to state estimation is imperative to enhance the resilience of power grids against such threats.

To address these limitations, machine learning (ML) techniques, including neural networks and support vector machines, have been explored to improve state estimation by leveraging statistical patterns and historical data [10]. While these data-driven approaches enhance accuracy and efficiency, they often neglect the physical constraints inherent to power systems [11], [12], limiting their reliability under varying operational conditions or sparse data [13].

Physics-informed neural networks (PINNs) [14] offer a transformative approach by embedding physical laws, constraints, and boundary conditions directly within the ML framework. By integrating principles, such as conservation laws and system constants, PINNs align predictions with the physical realities of power systems [15]. This not only improves model accuracy but also reduces data requirements, both in quantity and quality, compared to purely data-driven methods. PINNs' attributes make them particularly suited for applications requiring

robustness, accurate fault diagnosis, and rapid responses during disruptions [16], [17].

This work presents a hybrid framework for power system state estimation that leverages the capabilities of PINNs. The proposed approach seeks to improve both the accuracy and robustness of state estimation, particularly in the presence of disturbances, such as cyberattacks and operational anomalies. The primary contributions of this work are as follows.

- 1) Proposing a hybrid model that incorporates physical laws into neural networks to improve the accuracy of state estimation in power systems.
- 2) Conducting a comprehensive benchmark against state-of-the-art data-driven neural networks through an extensive set of experimental evaluations.
- 3) Demonstrating significant improvements in accuracy (up to $\sim 93\%$) and robustness under various challenging conditions, including three-phase faults and data manipulation attacks.

The proposed PINN-based approach highlights the potential for enhancing the stability, integrity, and security of power grids by utilizing advanced ML models that integrate physical principles to enable accurate and timely state estimation.

The rest of this article is organized as follows. Section II reviews existing research on physics-enhanced neural networks for state estimation. Section III explains the methodology and training process of the proposed model. Section IV presents experimental results. Finally, Section V concludes this article and provides future directions.

II. RELATED WORK

State estimation methods in power systems can be broadly categorized into three main categories: traditional techniques, data-driven techniques, and hybrid models [18], [19]. Traditional techniques, such as Newton–Raphson and weighted least squares, use physical laws to provide reliable results under steady-state conditions. However, these methods are computationally expensive and show limited ability to handle dynamic conditions. Data-driven techniques, particularly neural networks, offer advantages in handling complex nonlinearities in historical data, with faster execution than traditional methods. Yet, these approaches may lack physical consistency and are sensitive to overfitting, requiring large datasets for reliable performance and accuracy. Hybrid models, such as PINNs, bridge this gap by utilizing the strengths of data-driven approaches and combining them with physical properties, while maintaining the ability to handle data scarcity.

ML techniques for state estimation have advanced beyond traditional methodologies by integrating historical data and statistical patterns [11], [12]. While data-driven approaches offer compelling advantages, they frequently overlook fundamental physical constraints [13], potentially compromising estimation accuracy under data-limited or dynamically varying scenarios. Hybrid modeling frameworks, synthesizing data analytics with governing physical principles [20], emerge as a promising

paradigm, enabling near real-time system state characterization with enhanced robustness in cyber-physical system applications.

PINNs augment neural network architectures by systematically incorporating physical constraints as regularization terms during model training [14]. This innovative approach fundamentally transforms ML methodologies by thoughtfully integrating domain-specific physical knowledge into the learning process. By embedding first-principles constraints, PINNs generate physically consistent predictions with reduced data requirements compared to conventional neural network approaches. Their nuanced versatility in managing data-scarce scenarios, processing multidimensional inputs, and preserving physical system integrity renders them particularly promising for sophisticated power system modeling [17], [21].

On the other hand, due to the digitization of power systems applications, many functionalities of energy management systems are increasingly susceptible to cyberattacks, such as data manipulation and false data injection attacks. Such attacks often target state estimation routines and can lead to severe operational failures, compromising grid security [22]. Purely data-driven state estimators are often vulnerable to these manipulations. In contrast, hybrid methodologies enhance resilience by embedding physical laws (e.g., Kirchhoff’s laws) into the data-driven framework [23]. This enables the validation of measurement data against underlying physical principles, offering a pathway to protect systems against attacks that violate these constraints [24].

Recent research integrating physical constraints into neural networks for power system state estimation, while promising, faces limitations. A μ PMU-based partitioning strategy is proposed in [25], and a deep neural network model focused on limited observability distribution grids in [26]. Moreover, a graph neural network using Kron reduction, which addresses transmission grids [27], demonstrated parameter estimation accuracy improvements. However, these methods share common limitations. Notably, they lack rigorous regularization parameter selection, which hinders application across diverse grid configurations and systematic model optimization. Furthermore, their evaluation primarily considers typical load profiles, neglecting critical outlier scenarios, such as attacks and faults, essential for grid resilience. These events, while infrequent, can severely impact grid stability and lead to failures.

Physics-informed models for power system state estimation have utilized unrolled iterative methods for acceleration [28], [29]. While promising and robust to input perturbations, their resilience to diverse data corruption, including faults and unexpected behaviors, is limited. In addition, time-series models may exhibit biases from time-series specific periodicities. The robustness of PINNs against critical system anomalies and diverse operating conditions remains largely unexplored in power systems [16], [30]. This work addresses these limitations by proposing a methodology designed for enhanced robustness against diverse operating conditions, such as faults and cyber threats. Our approach is time-series agnostic, offers a systematic hyperparameter optimization procedure, and embeds physics laws directly into the loss function, aiming for high estimation accuracy and consistency under challenging conditions.

III. METHODOLOGY

A. Background

The foundational concept behind PINNs [14] is to incorporate physical laws directly into the training process of neural networks by embedding these laws as constraints in the loss function. In general, PINNs can approximate a target function f by training a neural network $\hat{f}(\mathbf{z}; \theta)$, parameterized by weights θ , where \mathbf{z} represents input variables. Instead of relying purely on data, PINNs include physics-based constraints that govern the system's behavior, ensuring that the model output conforms to known physical laws.

The PINN loss function combines a data-driven term, L_{data} , which minimizes prediction error on observed data, and a physics-informed term, L_{physics} , which enforces consistency with the underlying physical laws

$$L_{\text{data}} = \sum_{i=1}^N \|\hat{f}(\mathbf{z}_i; \theta) - f_i\|^2 \quad (1)$$

where $\{(\mathbf{z}_i, f_i)\}_{i=1}^N$ represents observed data points, and

$$L_{\text{physics}} = \sum_{j=1}^M \|G(\hat{f}(\mathbf{z}_j; \theta), \mathbf{z}_j)\|^2. \quad (2)$$

Here, $G(\hat{f}, \mathbf{z}) = 0$ denotes a general physical law or system constraint that should hold at each point \mathbf{z}_j in the domain, represented by $\{\mathbf{z}_j\}_{j=1}^M$.

The combined loss is expressed as

$$L = L_{\text{data}} + \lambda L_{\text{physics}} \quad (3)$$

where λ is a weighting factor that balances the data-driven and physics-informed components of the loss.

During training, minimizing L guides the model \hat{f} to align with observed data while satisfying the physical constraints, allowing PINNs to model complex behaviors accurately and consistently with the underlying physical system. This approach enhances model generalization and reduces dependence on extensive data by leveraging embedded physics, making PINNs particularly effective for applications in scientific and engineering domains.

B. Enhancing Loss Functions With Physical Constraints

In this study, we build upon the foundational principles of PINNs by systematically introducing additional terms to the loss function that impose physics-based constraints relevant to power system state estimation. These terms are derived from established relationships and laws governing the system, such as power flow equations, Kirchhoff's laws, and the admittance matrix. By embedding these constraints directly into the loss function, we ensure that the neural network's predictions align with the physical laws governing power systems. Consequently, during the training process, the network aims to minimize not only the discrepancy between predicted and actual values but also the violation of these physics-based constraints. This approach enhances the model's accuracy and reliability, making

it particularly suitable for the complexities inherent in power system state estimation.

In the proposed PINN method, the loss function (denoted as Loss) is computed as the sum of the *mean-square error (mse)* between the actual data and the inferred values (d) and the *mse* resulting from enforcing a physics equation (p) that should ideally equal zero. Also, if certain constants, such as the admittance matrix or specific system parameters, are known and remain unchanged (c), they can be included throughout the training process. Let $\lambda_d, \lambda_p, \lambda_c \in [0, 1]$ denote the weights for data, physics, and constants, respectively. These weights can be adjusted during the training process to optimize the model's performance based on the influence of each loss function parameter. Clearly, $\lambda_d + \lambda_p + \lambda_c = 1$. The Loss function can be generally represented as

$$\text{Loss} = \lambda_d \cdot d + \lambda_p \cdot p + \lambda_c \cdot c. \quad (4)$$

By integrating physics-based constraints as regularization terms, along with adjustable weighting factors, we can achieve a balance between data-driven learning and adherence to fundamental physical laws and properties. This is also helpful in avoiding overfitting of the neural network to the training data while penalizing physically unrealistic results.

C. Power System Parameter Selection

To develop the proposed PINN model, an input–output pair is first identified, where the input is correlated to the output through a physics-defined relationship that captures the system's dynamic nature. It is particularly beneficial if this relationship includes constants or topological information about the system, as this introduces additional information and constraints to the model. Next, the loss function is formulated to incorporate data, physics-based constraints, and constants, ensuring these factors are considered during the model's training process. During parameter optimization, the weights assigned to each component of the hybrid loss function, as shown in (4), are adjusted according to the specific characteristics of the system.

The aforementioned procedure for determining the PINN model is facilitated through the consideration of a traditional power system state estimation. In particular, the process of power system state estimation involves determining the voltage magnitude (V_i) and phase angle (θ_i) at each bus within the system. In the proposed PINN model, the outputs are the estimated complex voltages, while the inputs are the active power (P_i) and reactive power (Q_i) injection measurements. These measurements are chosen as inputs because they play a critical role in revealing the system's topology through data-driven patterns. The active and reactive power injections are calculated using the following equations:

$$P_i = V_i \cdot \sum_{j \in N} V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad (5a)$$

$$Q_i = V_i \cdot \sum_{j \in N} V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \quad (5b)$$

where V_i and V_j represent the voltage magnitudes at buses i and j , respectively, θ_{ij} denotes the phase angle difference between

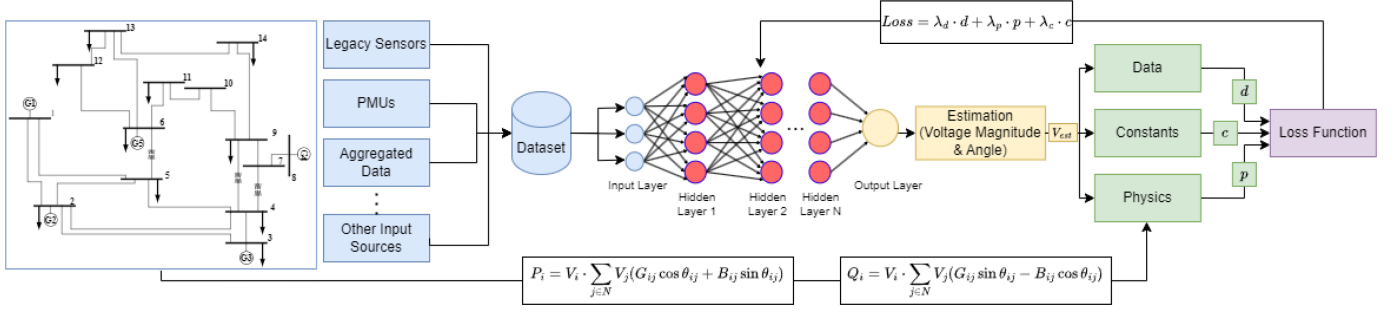


Fig. 1. Model training procedure for the power system state estimation use case.

buses i and j , G_{ij} and B_{ij} are the real and imaginary components of the admittance matrix, and N is the total number of buses in the system. As indicated by (2a) and (2b), the net power injected at a bus is determined by the complex voltage at that bus and the voltages of the buses connected to it.

The power system's topology and connectivity are represented by the admittance matrix included in these equations (G_{ij} and B_{ij}). By learning the correlations between complex powers and voltages, the neural network can uncover the underlying system topology without the need to explicitly modify the neural network architecture with system-specific parameters or special connectivity arrangements between the network's layer.

D. Loss Function Formulation

The loss function for the PINN, as formulated in (4), is designed to be minimized during the backpropagation training process, as depicted in Fig. 1. The specific parameters employed in the loss function are derived from the input–output relationships detailed in Section III-C. Minimizing this loss function enables the PINN to produce outputs that closely approximate the desired power system state. Backpropagation, a widely used technique in neural network training, iteratively updates the network's weights and biases to reduce the loss by propagating the error backward through the network. This optimization process aims to identify the parameter configuration that minimizes the error between estimated and true values, thus enhancing the performance of the power system state estimation.

Based on (4), parameter d , a very standard loss function parameter aims to drive the model toward matching the results of a known, ground-truth dataset. The model produces a pair of voltage magnitude V_{mag} and voltage angle V_{ang} for each bus at the output layer, constituting the complex voltage V_{est} . In order to ensure that the model fits the data according to a known, representative dataset, the difference between the estimated and ground-truth values V_{true} should be minimized. Thus, the d parameter is constructed as the mean-squared error between the estimated and the actual complex voltages, as shown in the following equation:

$$d = \frac{1}{n} \sum_{i=1}^n (V_{\text{est}_i} - V_{\text{true}_i})^2. \quad (6)$$

For the physics-determined part of the loss function p , the complex currents injected at each bus are used. Actually, the

complex currents relate to the system admittance matrix \mathbf{Y} (that includes information about the network connectivity) with the complex voltages as shown in the following equation:

$$\mathbf{I} = \overline{\mathbf{Y}} \cdot \mathbf{V}. \quad (7)$$

Through (7), the estimated current I_{est} is calculated using the complex voltage values generated at the output layer of the model. Furthermore, (7) is used to calculate the actual injected current using the ground-truth data from a given dataset. Having these datasets available, the mean-squared error between the actual and the estimated injected current is calculated and attributed in p , as shown in the following equation:

$$I_{\text{est}} = \overline{\mathbf{Y}} \cdot V_{\text{est}} \quad (8a)$$

$$I_{\text{true}} = \overline{\mathbf{Y}} \cdot V_{\text{true}} \quad (8b)$$

$$p = \frac{1}{n} \sum_{i=1}^n (I_{\text{est}_i} - I_{\text{true}_i})^2. \quad (8c)$$

The use of the mean-squared error of the injected currents ensures that both V_{est} and I_{est} are considered during the training procedure. As the current is correlated to the voltage through the admittance matrix, during the training procedure the PINN model is pushed toward adhering to the system's topology.

The constants of the system, represented by c , are encoded in the loss function as a mean-squared error between a known value of the system and the model's predicted value. For example, in a power system, there are buses connected to generators where a voltage regulation procedure is carried out. Therefore, we know that the voltage magnitude or angle in certain buses is going to be considered a constant, specifically during steady-state conditions. Also, in the case of steady-state conditions, it can be assumed that the slack bus is designated as a reference point and the associated voltage and angle will be constant as well. Therefore, c can be calculated as the sum of mean-squared errors between known constants and the predicted values, so that we can incentivize the model to fit these known values

$$c = \frac{1}{n} \sum_{i=1}^n (C_{\text{est}_i} - C_{\text{true}_i})^2. \quad (9)$$

The methodology and criteria outlined in this section systematically guide the selection of measurements that constitute the loss function, ensuring adaptability to various scenarios. As such, the parameters chosen can be substituted with alternative

Algorithm 1: Model Optimization Process.

Input : k : number of weights, n : step size,
 t : number of trials,
 $param_ranges$: (layers, number of
 neurons, learning rate, batch size),
 $pinn$: tuple $pinn(m, E)$ denotes model
 m with testing Mean Absolute Error
 (MAE), E , of model m

Output : $pinn_{opt}$: Outputs best model based on
 lowest MAE

```

// Initialization step
1  $N \leftarrow \binom{n+k-1}{k-1}$ ;
2  $pinn_{opt} \leftarrow (none, \infty)$ ;
// Optimization Procedure
3 for  $i \leftarrow 1$  to  $N$  do
4    $pinn_{trial} \leftarrow (none, \infty)$ ;
   // Select loss function weight
   combination for iteration  $i$ 
5    $l\_weights_i \leftarrow combination_i(n, k)$ ;
6   for  $j \leftarrow 1$  to  $t$  do
   // Select trial hyperparameters
   using the TPE meta-optimizer
7      $params_{ij} \leftarrow TPE(param\_ranges)$ ;
8      $pinn_{ij} \leftarrow model\_train(params_{ij}, l\_weights_i)$ ;
9     if  $pinn_{trial}.E > pinn_{ij}.E$  then
10      |  $pinn_{trial} \leftarrow pinn_{ij}$ ;
11   end
12 end
13 if  $pinn_{opt}.E > pinn_{trial}.E$  then
14   |  $pinn_{opt} \leftarrow pinn_{trial}$ ;
15 end
16 end
17 return  $pinn_{opt}$ 
    
```

equations and system measurements, depending on the specific application of the PINN. In our experimental validation, (4) will be employed, with the parameters d , p , and c defined as discussed in this section. These equations have been specifically tailored for the exploration of a power system state estimation use case.

E. Systematic Hyperparameter Optimization

Hyperparameter optimization is crucial for training complex neural network architectures, such as the proposed PINN model. Given the inefficiency of traditional grid and random search methods in exploring the often vast hyperparameter space, especially when balancing multiple loss function components and model parameters, a systematic approach using metaoptimizers is employed. This approach, as outlined in Algorithm 1, aims to determine the optimal configuration of adjustable loss function weights for each scenario.

The exploration process systematically investigates combinations of loss function weights $\lambda_d, \lambda_p, \lambda_c \in [0, 1]$, which represent the relative importance of the loss function terms. These weights are constrained to the range $[0, 1]$ and are sampled at intervals defined by a step size of Δ . In addition, the weights must

satisfy the condition $\lambda_d + \lambda_p + \lambda_c = 1$, introducing a constraint on their possible combinations. The process systematically evaluates all possible combinations of these weights, where the total number of combinations N is computed using the formula for combinations with repetition

$$N = \binom{n+k-1}{k-1}, \quad \text{where } n = \frac{1}{\Delta}, \text{ and } k = 3$$

where n represents the number of discrete values for each weight, and $k = 3$ corresponds to the number of weights.

For each weight combination, multiple optimization trials t are performed. During each trial, a metaoptimizer, a tree-structured Parzen estimator (TPE) in our case, selects the model's architecture and training parameters—such as the number of layers, neurons per layer, learning rate, and batch size—from predefined ranges. These trials aim to identify configurations that best align with the current weight combination, enabling a thorough exploration of the parameter space.

The optimization process systematically explores critical model parameters: layer count determines the model's representational capacity, neuron count impacts pattern recognition, learning rate controls gradient descent step size, and batch size influences computational efficiency and generalization. By employing mean absolute error (MAE) as the objective function, the approach enables a comprehensive evaluation of model performance, allowing for adaptive exploration of the parameter space.

IV. EXPERIMENTAL SETUP AND RESULTS

A. Experimental Setup

To evaluate the effectiveness of our proposed PINN model for power system state estimation, we conducted extensive simulations using the IEEE 14-bus and the IEEE 118-bus benchmark systems and generated realistic power system scenarios using the PowerWorld software [31]. The PINN model is implemented in Python with TensorFlow for efficient neural network training and trained on a variety of measurements collected under different operating conditions and disturbances. In this context, a fixed transmission grid topology with a known admittance matrix (Y) is assumed, where each bus provides real and reactive power injections. Under normal conditions, full system observability is maintained, and buses with generators are voltage-regulated. However, during transients, voltage regulation may be compromised, and no parameter is assumed constant.

The models' generalization capabilities and its ability to handle real-world uncertainties are enhanced by introducing simulated measurement noise into our datasets [32]. This noise injection methodology systematically emulates the inherent measurement inaccuracies and stochastic fluctuations characteristic of complex power system environments. By deliberately incorporating controlled noise patterns, the models' capacity to handle unpredictable signal variations and measurement imprecision is significantly improved. Data preprocessing also included standardization, min-max scaling, and replacing zero values with a small epsilon ($1 \cdot e^{-8}$). Inputs such as real (P_i) and reactive (Q_i) power, are standardized using their mean and

TABLE I
COMPUTATIONAL COSTS

Power System	Training Time (s)	Inference Time (ms)
IEEE 14	34.375	0.03946
IEEE 118	39.709	0.03479

standard deviation, and then rescaled to $[-1, 1]$ to align with the tanh activation function.

Neural network weights are initialized with the popular Glorot scheme [33], biases are set to zero, and training utilizes an early stopping strategy with a 20-epoch patience threshold, capping optimization at 100 epochs. The Adam optimizer and tanh activation function are employed, with the best-performing model (minimizing MAE) selected to prevent overfitting and preserve generalization performance. The models are optimized using the procedure shown in Algorithm 1 using the following ranges for key parameters: number of layers (2–10), neurons per layer (64–4096), learning rate ($1 \cdot e^{-5}$ to $1 \cdot e^{-1}$), and batch size (4–128). Also, the number of weights is set to $k = 3$, the step size $\Delta = 0.1$, and number of trials $t = 10$. The models' computational costs are measured during multiple optimization trials for training and during testing for inference. The average computational cost is outlined in Table I. All measurements are derived from a single NVIDIA RTX 4000 Ada Generation GPU.

B. Test Cases and Results

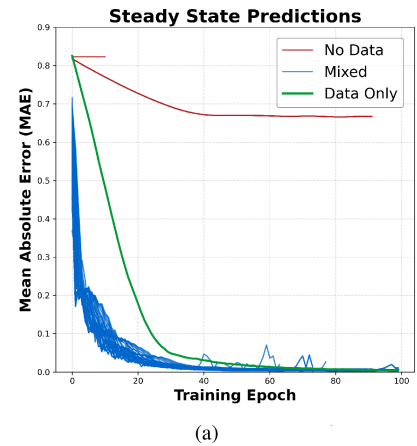
A set of datasets representing different states of the IEEE 14-bus and IEEE 118-bus system benchmark is generated to evaluate the performance of the PINN approach compared to an equivalent NN in terms of accuracy and robustness in estimating power system states. The scenarios include both normal daily load profiles and transient operating conditions, to better showcase the PINN's performance under different scenarios.

1) *IEEE 14-Bus System*: To evaluate the PINN's generalization capabilities, we employ conventional dataset splitting and test the model against entirely unseen scenarios that maintain similar system dynamics. As given in Table II, the testing scenarios for the IEEE 14-bus system encompass five distinct cases: 1) steady-state conditions with typical daily load profiles, 2) generator shutdown at bus 2, tested against system-wide load increases, 3) generation power surge at bus 2, validated against load reductions, 4) transient three-phase self-clearing faults (0.5-s duration), where the model trained on bus 2 is tested against faults at various locations (buses 5, 7, and 12), and 5) a data manipulation attack scenario, where an attacker gains control of one bus (bus 4 or 14) and tries to influence the state estimation procedure by altering sensor data feeds. This comprehensive approach assesses the model's ability to extrapolate its learned knowledge to unexpected, yet physically consistent scenarios.

In *Scenario 1*, we explored combinations of weights (λ_d, λ_p , and λ_c) [see (4)] for the data, physics, and constants terms in the loss function, using $k = 3$, step size $\Delta = 0.1$, and number

TABLE II
EXPERIMENTAL SETUP SCENARIOS (IEEE 14-BUS SYSTEM)

Scenario 1: Steady State		
Index	Training Dataset	Testing Dataset
S1.1	Daily Load Profile	Unknown subset of training set
Scenario 2: Generator Shutdown at bus 2		
Index	Training Dataset	Testing Dataset
S2.1	Generator 2 Shutdown	System Load Increase 40 MW ($\sim +9\%$)
S2.2	Generator 2 Shutdown	System Load Increase 80 MW ($\sim +18\%$)
Scenario 3: Generation Ramp-up at bus 2		
Index	Training Dataset	Testing Dataset
S3.1	Ramp-up 40 MW	System Load Decrease 40 MW ($\sim -9\%$)
S3.2	Ramp-up 40 MW	System Load Decrease 80 MW ($\sim -18\%$)
Scenario 4: 3-phase Transient Faults		
Index	Training Dataset	Testing Dataset
S4.1	Bus 2 (0.5 p.u. resistance)	Bus 5 (0.5 p.u. resistance)
S4.2	Bus 2 (0.5 p.u. resistance)	Bus 7 (0.5 p.u. resistance)
S4.3	Bus 2 (0.5 p.u. resistance)	Bus 12 (0.5 p.u. resistance)
Scenario 5: Data Manipulation Attack		
Index	Training Dataset	Testing Dataset
S5.1	Daily Load Profile	Attack on bus 4
S5.2	Daily Load Profile	Attack on bus 14



Effect of Loss Function Weights on model accuracy

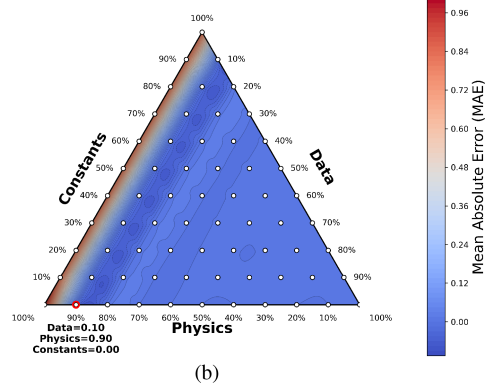


Fig. 2. *Scenario 1 (IEEE 14-bus system)*—MAE for steady-state conditions using varying λ_d, λ_p , and λ_c weights for data, physics, and constants, respectively, in the loss function. (a) MAE per training epoch. (b) λ weights heatmap.

of trials $t = 10$, while maintaining $\lambda_d + \lambda_p + \lambda_c = 1$. The impact of different weight distributions on model performance is shown in Fig. 2. The green curve represents the combination of weights that corresponds to a purely data-driven approach that is an optimized NN, but with no physics information, e.g., $(\lambda_d, \lambda_p, \lambda_c) = (1.0, 0.0, 0.0)$. Every combination below the

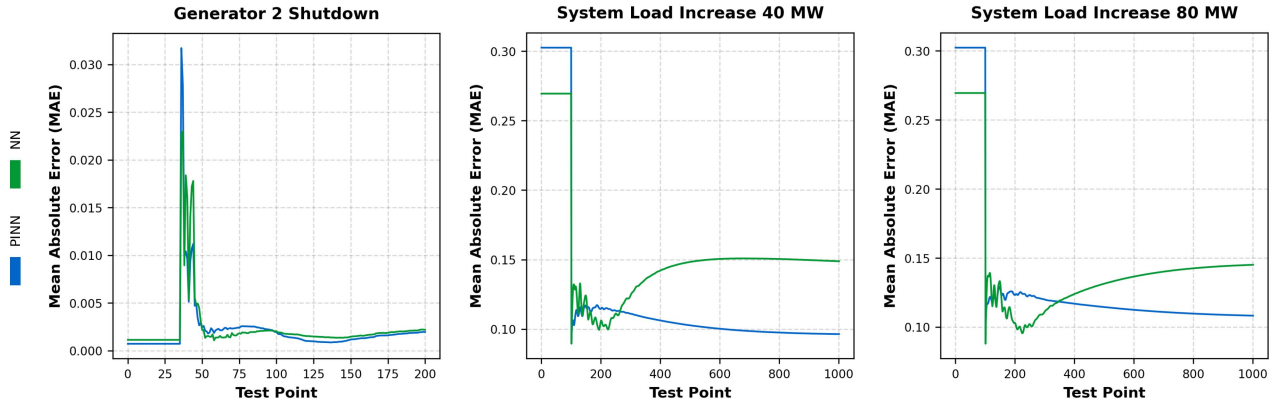


Fig. 3. Scenario 2—Testing datasets with varying generator shutdowns. Models trained on a bus 2 shutdown are tested with load spikes at equal (S2.1) and double the initial shutdown magnitude (S2.2).

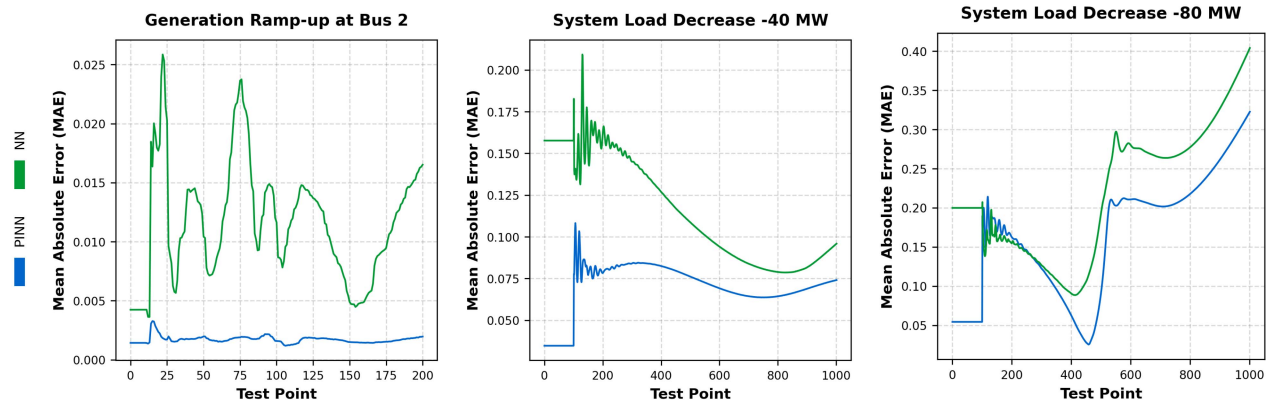


Fig. 4. Scenario 3—Testing datasets with varying generation increases. Models trained on a bus 2 ramp-up are tested with scenarios simulating load drops at equal (S3.1) and double the initial increase (S3.2).

green curve, shown in blue, represents different weight values among the data, physics, and constants terms. Our results in Fig 2(a) indicate that, in the steady-state scenario, the specific weighting for physics or constants does not significantly impact accuracy as long as the approach is not purely data-driven, each case corresponding to a different PINN with loss function as in (4). The abovementioned red curves correspond to combinations that rely heavily on constants (topmost curve) or are primarily physics-based (second set of red curves) while λ_d is zero. As shown in Fig. 2(b), a strong reliance on known system constants does not lead to satisfactory results. Overall, the PINNs exhibit faster convergence and improved model accuracy, achieving lower MAE values earlier than the NN. Notably, under steady-state conditions for the IEEE 14-bus system, the PINN achieves up to an $\sim 57.3\%$ reduction in MAE, demonstrating its superior performance.

Scenario 2 test results can be seen in Fig. 3. In the initial testing phase, the PINN does not offer substantial advantages over the NN. However, looking at Scenarios S2.1 and S2.2, we can see that the PINN offers significantly improved accuracy and stability in the predictions. While the NN's predictions seem to deteriorate over time, the PINN has a lot more stable MAE over time. In terms of accuracy, the PINN shows $\sim 23.5\%$ and $\sim 8\%$

improvements in absolute error for Scenarios S2.1 and S2.2, respectively, indicating the better generalization of the model to unknown states of the system.

Fig. 4 shows the testing results for Scenario 3. The NN's prediction accuracy seems to vary significantly in MAE based on the data points and shows significantly more fluctuations than the PINN. The PINN, on the other hand, stays consistently low in MAE across all data points while also being more accurate. Examining the unknown testing scenarios, the PINN consistently stays more accurate than the NN in all data points, again, hinting the PINN's greater generalization potential to unknown system states. Specifically, the PINN has $\sim 65.15\%$ in S3.1 and $\sim 33.23\%$ in S3.2, lower MAE than the NN.

Regarding Scenario 4, as depicted in Fig. 5, we evaluate the model's ability to extrapolate fault characteristics across varying locations. The proposed PINN demonstrates, on average, $\sim 20.53\%$ better accuracy when compared to the NN, even though both models underwent identical hyperparameter tuning. The impact of fault location on model performance is detailed in Fig. 5's multiple graphs, showing the testing accuracy along with the different scenarios (from S4.1 to S4.3, from left to right). While PINNs exhibit strong generalization capabilities

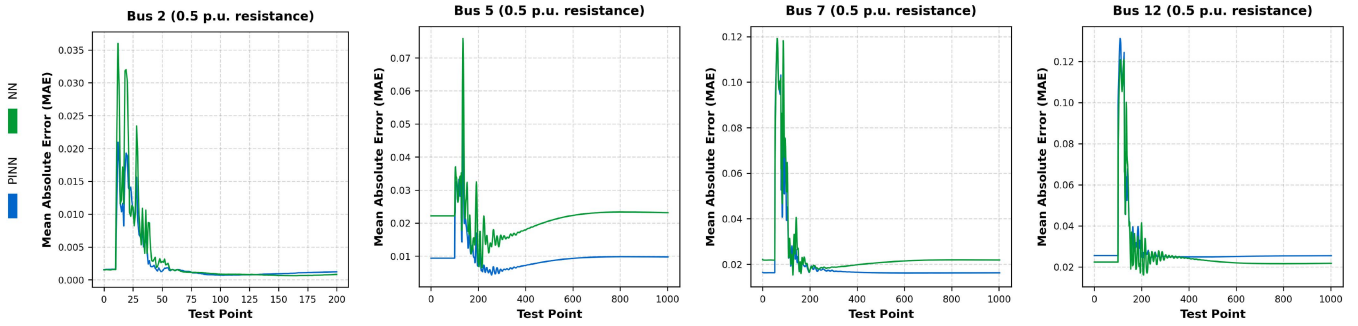


Fig. 5. *Scenario 4*—Testing datasets for three-phase fault, with different per-unit (p.u.) resistance and locations. Graphs represent the testing set, plus *Scenarios S4.1–S4.5* from left to right.

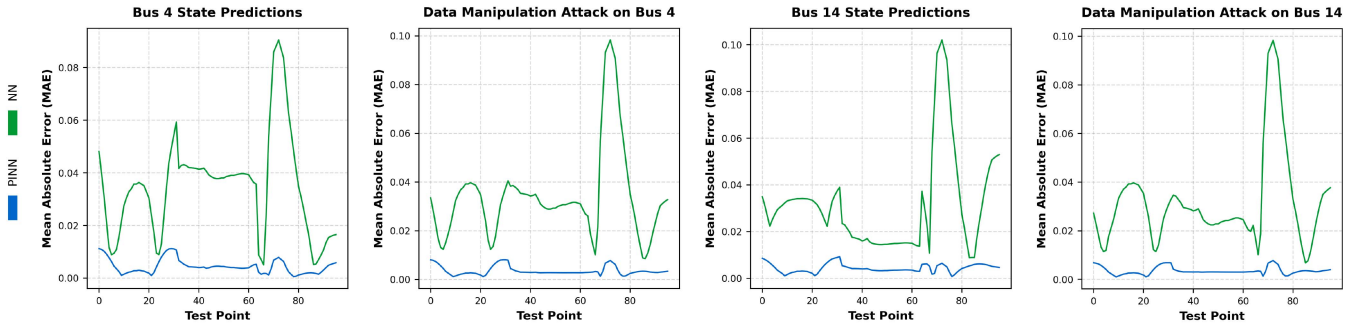


Fig. 6. *Scenario 5*—Testing datasets, showing the local and global accuracies of the models during data manipulation attacks.

for faults occurring near training locations, their accuracy diminishes as the spatial distance between training and testing locations increases, underscoring the significance of spatial correlation.

In data manipulation attacks, malicious actors strategically manipulate sensor data to undermine the integrity of real-time grid monitoring. To investigate the robustness of the proposed PINN in such attacks, we tested the PINN's accuracy, which can act as a filter against such erroneous data. Such filtering properties enhance continuous and accurate state monitoring capabilities, even under data communication disruptions. To evaluate the robustness of the IEEE 14-bus system's state estimation against data manipulation attacks, we target buses 4 and 14 in *Scenario 5* (see Table II).

Models, including both PINN and the NN, are trained using the steady-state dataset described in *S1.1*. To evaluate the PINN and NN performance, data manipulation is applied to the testing dataset. This process is employed to systematically evaluate the performance of the models under varying levels of data corruption, enabling the assessment of their tolerance to increasing measurement biases. The parameters governing this data manipulation process are defined as follows. Attack targets are chosen, namely, bus 4 and bus 14. The buses were selected based on the number of adjacent buses to the target bus, with *S5.1* using bus 4 due to having the most adjacent buses and *S5.2* using bus 14 for having the least. The attack specifically targets active and reactive power injection measurements (P and Q) at the selected bus. Across 100 consecutive time instances, the bias magnitude profile is incrementally increased relative to the

true measurement value at each test point: 10% for the first third (test points 1–33), 20% for the second third (test points 34–66), and 30% for the final third (test points 67–100). This profile is applied sequentially over the 100 test instances, modifying only the targeted P and Q measurements of the compromised bus, while all other measurements remain unaffected. This systematic application allows for evaluating the models' response and performance degradation under progressively larger measurement errors.

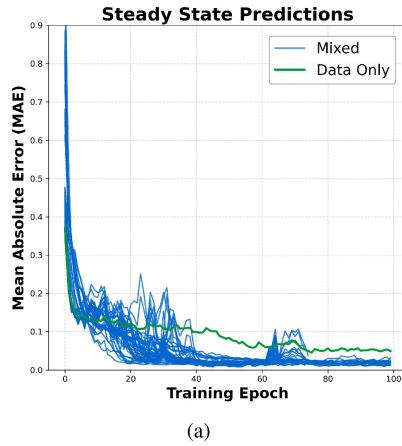
As shown in Fig. 6, the PINN significantly outperforms the equivalent NN in predicting the state of the affected bus and the overall system state, demonstrating its ability to leverage underlying physical laws to filter out the effects of the data manipulation attack. The PINN achieves up to $\sim 93\%$ improvement in average MAE for both *S5.1* and *S5.2*, highlighting its robustness against such attacks.

2) IEEE 118-Bus System: To demonstrate the scalability of the approach, experimental scenarios are applied to the IEEE 118-bus system, similar to the test cases seen in Section IV-B1. Specifically, the testing scenarios are: 1) steady-state conditions with typical daily load profiles, 2) a generator shutdown scenario at bus 26, 3) transient three-phase self-clearing fault at bus 19 (0.5 seconds duration), and 4) a data manipulation attack scenario, where an attacker gains control of bus 54 and alters the power injection measurements. The specifics of each scenario can be seen in Table III.

In *Scenario 6*, we follow the same principles as in *Scenario 1*, a steady-state daily load profile adapted to the larger system. In particular, the goal is to examine the effect of the different

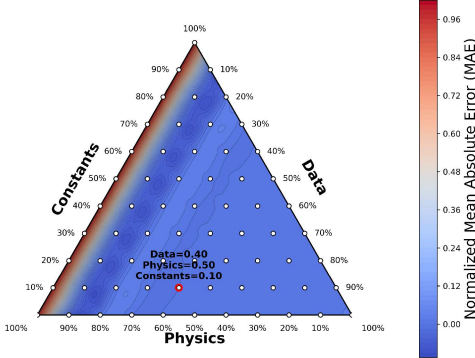
TABLE III
EXPERIMENTAL SETUP SCENARIOS (IEEE 118-BUS SYSTEM)

Scenario 6: Steady State		
Index	Training Dataset	Testing Dataset
S6.1	Daily Load Profile	Unknown subset of training set
Scenario 7: Generator Shutdown at bus 26		
Index	Training Dataset	Testing Dataset
S7.1	Generator 26 Shutdown (~314MW)	System Load Increase 314 MW
Scenario 8: 3-phase Transient Faults		
Index	Training Dataset	Testing Dataset
S8.1	Bus 19 (0.25 p.u. resistance)	Bus 33 (0.25 p.u. resistance)
Scenario 9: Data Manipulation Attack		
Index	Training Dataset	Testing Dataset
S9.1	Daily Load Profile	Attack on bus 54



(a)

Effect of Loss Function Weights on model accuracy



(b)

Fig. 7. Scenario 6 (IEEE 118-bus system)—MAE for steady-state conditions using varying λ_d , λ_p , and λ_c weights for data, physics, and constants, respectively, in the loss function. (a) MAE per training epoch. (b) λ weights heatmap.

weights of the loss function parameters on a model trained using the power system's steady state conditions. Fig. 7 shows the different PINN configurations in blue and the benchmark NN in green. The red “no data” curves, as seen in Fig. 2, are omitted here due to their unsatisfactory performance, lying outside the region of interest shown in this graph. While the PINNs show a more erratic behavior during early training epochs than in the smaller IEEE benchmark system, the end result confirms the superiority of the PINN over the NN. In particular, the final results show even greater gains in accuracy, with the best PINN

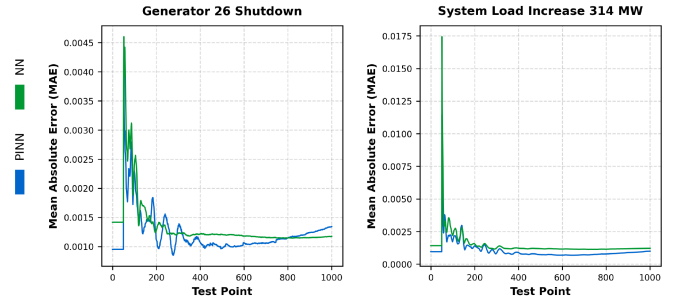


Fig. 8. Scenario 7—Testing datasets with generator shutdown. Models trained on a bus 26 shutdown are tested with load spikes at equal (S7.1) the initial shutdown magnitude.

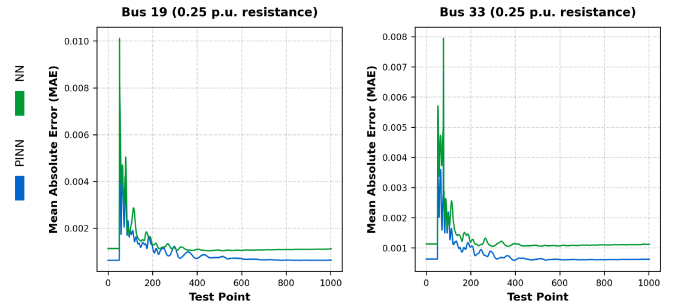


Fig. 9. Scenario 8—Testing datasets for three-phase fault. Graphs represent the testing set, plus Scenario S8.1.

configuration achieving $\sim 82.91\%$ lower MAE than the NN. The loss function weight heatmap [see Fig. 7(b)] highlights similar regions of interest as the IEEE 14-bus system.

Scenario 7 examines a generator shutdown. The model is trained using a dataset capturing the abrupt interruption of the generator at bus 26 and tested against a dataset representing an equivalent system load increase of 314 MW, distributed equally across buses 20–23 and 27. Overall, the PINN shows less extreme behavior on the initial testing dataset, while the NN seems to stabilize faster, as seen in Fig. 8. The PINN's robustness capabilities show up when testing against the equivalent load increase scenario, where it shows a clear trend, showing a 27.68% lower MAE.

Fig. 9 shows the testing results for the three-phase transient fault, namely, Scenario 8. In this scenario, the models are trained with a transient three-phase fault at bus 19 and tested against measurements representing a same-type fault in a different location, bus 33. The PINN generates more accurate results when tested against both the unknown subset of the training dataset and the completely unknown dataset, with a $\sim 37.67\%$ improvement on the testing dataset.

Finally, Scenario 9, as depicted in Fig. 10, follows the same training and testing principles outlined for Scenario 5 in Section IV-B1. The figure shows the overall average MAE of the system's predictions as well as the MAE specifically for bus 54, the attack target. Bus 54 was selected because it has a high load of 113 MW, making it one of the largest buses in the system and highly influential to the entire network. The models, both PINN and NN, are trained using the same dataset

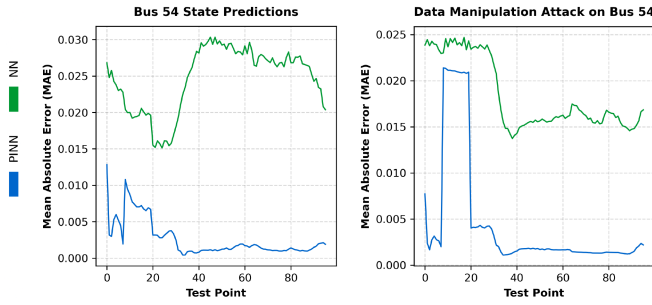


Fig. 10. Scenario 9—Testing datasets showing the local and global accuracies of the models during data manipulation attacks.

TABLE IV
STATE ESTIMATION PERFORMANCE (IEEE 118-BUS SYSTEM)

Method	Mean Absolute Error	Normalized Performance
PINN	$7.72 \cdot 10^{-3}$	1.00
GNU-GNN [28]	$9.42 \cdot 10^{-3}$	0.82
Prox-linear Net [12]	$1.26 \cdot 10^{-2}$	0.61
NN	$4.52 \cdot 10^{-2}$	0.17

as in *Scenario 6* and tested against a dataset that represents the increasing data manipulation magnitude on bus 54. The PINN shows considerable improvements of up to $\sim 76.11\%$ in accurately predicting the system-wide state.

3) *Experimental Results Discussion*: The PINN has been comprehensively tested in a large variety of datasets and has shown adequate robustness in terms of model inference. State estimation results demonstrate superior accuracy, when compared to an equivalent NN, across the 14-bus and 118-bus IEEE benchmark systems, highlighting the scalability of the approach. The associated heatmaps for the steady state scenarios show that the majority of good solutions lie in the space of $\lambda_d \simeq 10\% - 20\%$, with different combinations of physics and data showing similar results. However, this does not discount the possibility of finding the optimal solution elsewhere (see Fig. 7), but it can serve as an indication on how to reduce the hyperparameter tuning search space, if needed.

Table IV presents a comparative summary of selected methods from the literature applied to the IEEE 118-bus system, based on the accuracy reported by the respective authors. To facilitate relative evaluation, all values are normalized with respect to the performance of the proposed PINN, which is set to 1.0. The comparison draws exclusively from results reported in the original publications, without reimplementing or empirical replication. Due to differences in datasets, problem formulations, and model architectures, the table is intended as a representative summary rather than a strict benchmark. Under the reported conditions, the normalized metrics indicate that the proposed PINN achieves superior performance relative to existing methods.

V. CONCLUSION

This work contributes to the growing body of research on integrating data-driven and physics-based methodologies, demonstrating the potential of PINNs to address the dynamic

challenges of modern power systems. By embedding physical laws directly into the neural network architecture and systematically optimizing the model and hyperparameters, PINNs enhance the accuracy of power system state estimation, outperforming traditional ML models, particularly during disruptions. Importantly, our experiments show that PINNs are less prone to data manipulation attacks, a common threat to power systems, as they leverage the underlying physical constraints rather than relying solely on observed data, highlighting this approach's robustness compared to traditional techniques.

Future work could explore the adoption of a multi-PINN architecture, where each model is trained on localized data from specific regions or clusters within the power system. Clustering based on proximity or electrical characteristics could optimize the models' ability to capture local dynamics and fault behaviors. In addition, leveraging transfer learning between adjacent clusters may enhance intercluster fault prediction accuracy, thereby improving overall system reliability. Extending this framework with a physics-aware implementation opens an intriguing avenue for anomaly detection. By identifying deviations from expected system behavior, this approach could enable advanced monitoring and early detection of anomalies, providing valuable insights even in complex scenarios or under limited system observability.

REFERENCES

- [1] I. Alotaibi et al., "A comprehensive review of recent advances in smart grids: A sustainable future with renewable energy resources," *Energies*, vol. 13, no. 23, 2020, Art. no. 6269.
- [2] Y.-F. Huang, S. Werner, J. Huang, N. Kashyap, and V. Gupta, "State estimation in electric power grids: Meeting new challenges presented by the requirements of the future grid," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 33–43, Sep. 2012.
- [3] A. Sanjab et al., "Smart grid security: Threats, challenges, and solutions," 2016, *arXiv:1606.06992*.
- [4] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, "Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun.*, 2010, pp. 220–225.
- [5] N. Khanam and M. Rihan, "State estimation of smart power grid: A literature survey," in *Proc. IEEE 2nd Int. Conf. Sustain. Energy Future Electric Transp.*, 2022, pp. 1–6.
- [6] S. Vahidi, M. Ghafouri, M. Au, M. Kassouf, A. Mohammadi, and M. Debbabi, "Security of wide-area monitoring, protection, and control (WAMPAC) systems of the smart grid: A survey on challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 1294–1335, Apr.–Jun. 2023.
- [7] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1630–1638, Jul. 2017.
- [8] D. U. Case, "Analysis of the cyber attack on the Ukrainian power grid," *Electricity Inf. Sharing Anal. Center*, vol. 388, no. 1–29, 2016, Art. no. 3.
- [9] S. Soltan, P. Mittal, and H. V. Poor, "{BlackIoT} : {IoT} Botnet of high wattage devices can disrupt the power grid," in *Proc. 27th USENIX Secur. Symp.*, 2018, pp. 15–32.
- [10] Y. Weng, R. Negi, C. Faloutsos, and M. D. Ilić, "Robust data-driven state estimation for smart grid," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1956–1967, Jul. 2017.
- [11] K. R. Mestav, J. Luengo-Rozas, and L. Tong, "Bayesian state estimation for unobservable distribution systems via deep learning," *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 4910–4920, Nov. 2019.
- [12] L. Zhang, G. Wang, and G. B. Giannakis, "Real-time power system state estimation and forecasting via deep unrolled neural networks," *IEEE Trans. Signal Process.*, vol. 67, no. 15, pp. 4069–4077, Aug. 2019.

- [13] S. Markidis, "The old and the new: Can physics-informed deep-learning replace traditional linear solvers?," *Front. Big Data*, vol. 4, 2021, Art. no. 669097.
- [14] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019.
- [15] A. Angulo, M. Huerta, and F. Mancilla-David, "A neural network-aided functional model of photovoltaic arrays for a wide range of atmospheric conditions," *IEEE Trans. Ind. Inform.*, vol. 20, no. 2, pp. 2487–2496, Feb. 2024.
- [16] G. Lu and S. Bu, "Online dynamic security assessment: Using hybrid physics-guided deep learning models," *IEEE Trans. Ind. Inform.*, vol. 20, no. 11, pp. 13190–13200, Nov. 2024.
- [17] G. S. Misyris, A. Venzke, and S. Chatzivasileiadis, "Physics-informed neural networks for power systems," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2020, pp. 1–5.
- [18] K. Dehghanpour, Z. Wang, J. Wang, Y. Yuan, and F. Bu, "A survey on state estimation techniques and challenges in smart distribution systems," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2312–2322, Mar. 2019.
- [19] J. Zhao et al., "Power system dynamic state estimation: Motivations, definitions, methodologies, and future work," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3188–3198, Jul. 2019.
- [20] B. Huang and J. Wang, "Applications of physics-informed neural networks in power systems—a review," *IEEE Trans. Power Syst.*, vol. 38, no. 1, pp. 572–588, Jan. 2023.
- [21] S. Falas, M. Asprou, C. Konstantinou, and M. K. Michael, "Physics-informed neural networks for accelerating power system state estimation," in *Proc. IEEE PES Innov. Smart Grid Technol. Europe*, 2023, pp. 1–5.
- [22] N. Tatipatri and S. Arun, "A comprehensive review on cyber-attacks in power systems: Impact analysis, detection, and cyber security," *IEEE Access*, vol. 12, pp. 18147–18167, 2024.
- [23] M. Rahman, J. Yan, and E. T. Fapi, "Adversarial artificial intelligence in blind false data injection in smart grid ac state estimation," *IEEE Trans. Ind. Inform.*, vol. 20, no. 6, pp. 8873–8883, Jun. 2024.
- [24] G. B. Gaggero, P. Girdinio, and M. Marchese, "Artificial intelligence and physics-based anomaly detection in the smart grid: A survey," *IEEE Access*, vol. 13, pp. 23597–23606, 2025.
- [25] A. S. Zamzam and N. D. Sidiropoulos, "Physics-aware neural networks for distribution system state estimation," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4347–4356, Nov. 2020.
- [26] J. Ostrometzky et al., "Physics-informed deep neural network method for limited observability state estimation," 2020. [Online]. Available: <https://arxiv.org/abs/1910.06401>
- [27] L. Pagnier and M. Chertkov, "Physics-informed graphical neural network for parameter & state estimations in power systems," 2021. [Online]. Available: <https://arxiv.org/abs/2102.06349>
- [28] Q. Yang, A. Sadeghi, and G. Wang, "Data-driven priors for robust PSSE via Gauss-Newton unrolled neural networks," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 1, pp. 172–181, Mar. 2022.
- [29] L. Pagnier and M. Chertkov, "Embedding power flow into machine learning for parameter and state estimation," 2021, *arXiv:2103.14251*.
- [30] G. Andronikidis et al., "Ai-driven anomaly and intrusion detection in energy systems: Current trends and future direction," in *Proc. IEEE Int. Conf. Cyber Secur. Resilience*, 2024, pp. 777–782.
- [31] S. Peyghami, P. Davari, M. Fotuhi-Firuzabad, and F. Blaabjerg, "Standard test systems for modern power system analysis: An overview," *IEEE Ind. Electron. Mag.*, vol. 13, no. 4, pp. 86–105, Dec. 2019.
- [32] M. Asprou, E. Kyriakides, and M. Albu, "The effect of variable weights in a WLS state estimator considering instrument transformer uncertainties," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 6, pp. 1484–1495, Jun. 2014.
- [33] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist. Workshop Conf.*, 2010, pp. 249–256.