Online Learning-based Mutually-aided State Estimation and Forecasting for PV

Hanshan Qing, *Student Member*, *IEEE*, Wangyuan Ding, *Student Member*, *IEEE*, Abhinav Kumar Singh, *Member*, *IEEE*, and Efstratios Batzelis, *Senior Member*, *IEEE*

***Abstract*—Traditional photovoltaic (PV) forecasting algorithms rely heavily on historical generation data or physical models, both of which are frequently unavailable or unreliable in practice. This paper addresses this challenge by proposing an online learning-based mutually-aided state estimation and forecasting (MASEF) algorithm, which integrates state estimation (SE) and PV forecasting into a coupled stochastic system and operates effectively without site-specific historical PV generation data or physical models. Inspired by the Kalman filter (KF), a mutually-aided online learning loop is established in the MASEF algorithm: a Bayesian neural network (BNN) approximates PV generation (prediction stage), which allows SE to refine the system state (update stage). These estimates are then fed back to the forecaster for online learning. The BNN further enhances reliability by providing probabilistic outputs and quantifying uncertainty. The forecasted values serve as pseudo-measurements, which are critical in scenarios with limited observability or noisy data. Case studies demonstrate that the MASEF algorithm achieves performance competitive with state-of-the-art algorithms even in the complete absence of PV generation data or physical models. Furthermore, the results confirm the robustness of the MASEF algorithm to measurement noise and parameter inaccuracies, highlighting its adaptability and practical applicability in diverse power grid environments.**

***Index Terms*—Bayesian neural network, online learning, state estimation, PV forecasting.**

I. INTRODUCTION

POWER systems depend on accurate state estimation (SE) and forecasting to maintain secure, reliable, and economically optimized operations, including security-constrained economic dispatch (SCED) [1]. While various algorithms (e.g., statistical , physical , and hybrid time-series) have been developed for photovoltaic (PV) forecasting [2]-[4], real-world systems are often constrained by limited measurement infrastructure. This scarcity complicates both SE and forecasting , particularly when site-specific PV generation data or models are inadequate. For instance, in the Pennsylvania-New Jersey-Maryland Interconnection (PJM) system, real-time PV generation data are reported exclusively by large-scale solar parks [5], and the UK grid similarly suffers from frequent inconsistencies in the operational data submitted by solar installations [6]. Consequently, small or newly installed PV systems often lack the requisite telemetry or models, limiting the applicability of traditional SE and forecasting algorithms.

Forecast-aided state estimation (FASE) partially mitigates these constraints by utilizing forecasted outputs as pseudo-measurements for SE [7], [8]. Neural network (NN)-based FASE such as those using long short-term memory (LSTM) networks in AC/DC distribution systems is a common algorithm for replacing missing or bad data [9]-[11]. However, standard NN-FASE typically employs a unidirectional data flow where pre-trained estimates feed into the SE. This creates a strong dependence on historical data and model accuracy. In scenarios characterized by sparse or poor-quality PV generation data, this algorithm struggles to produce reliable estimates, thereby degrading overall SE performance.

Kalman filter (KF) algorithms offer an alternative, featuring a bidirectional data flow with prediction and update stages that iteratively refine the SE solution [12]. Various KF variants have been developed, including the PV-assisted interleaved extended Kalman filter (PV-IEKF) [13], data-driven unscented Kalman filter (UKF) [14], and ensemble Kalman filter (EnKF) [15]. However, this bidirectional data flow does not constitute a true mutually-aided online learning loop. Traditional KF algorithms are fundamentally model-based; their prediction stage relies on a predefined state transition model to propagate the state. This model is often fixed or requires pre-training on historical data [14], [15]. Furthermore, even hybrid NN-KF algorithms, which utilize NNs to generate pseudo-measurements [16], typically do not use online SE results to retrain the core dynamic forecasting model . Consequently, these algorithms remain vulnerable when underlying system dynamics are unknown, non-linear, or time-varying.

To bridge this gap, this paper proposes a new algorithm built on a mutually-aided online learning loop , with Bayesian neural networks (BNNs) serving as the enabling technology. The key advantage of BNNs over deterministic NNs

---

Manuscript received: September 5, 2025; revised: December 2, 2025; accepted: March 3, 2026. Date of CrossCheck: March 3, 2026. Date of online publication: XX XX, XXXX.

The code and data used to generate the findings in this study are available from the first author upon request.

H. Qing, A. K. Singh (corresponding author), and E. Batzelis are with School of Electronics and Computer Science, University of Southampton, Southampton, U.K. (e-mail: H.Qing@soton.ac.uk ; A.K.Singh@soton.ac.uk; E.Batzelis@soton.ac.uk).

W. Ding is with Informatics Institute, University of Amsterdam, Amsterdam, Netherlands (e-mail: W.Ding@uva.nl).

DOI: 10.35833/MPCE.2025.000837

lies in their inherent ability to model uncertainty in both model parameters and outputs [17], [18]. This property is critical: the predictive uncertainty of BNNs provides a dynamic, uncertainty-aware weight for its own forecast, thereby enabling a stable feedback loop with the state estimator. While BNNs and related methods like Bayesian physics-informed neural networks (BPINNs) have been applied to SE [19]-[21], existing applications are typically unidirectional (i. e., forecast then estimate) , which retains a dependency on the initial training data.

TABLE I
COMPARISON OF MASEF AND EXISTING FASE ALGORITHMS

Category	Reference	Requirement of physical model or historical data	Providing pseudo measurement forecasting (PV generation and/or load)	Providing state forecasting (in addition to SE)
Unidirectional, data-driven	[9]	√	×	√
	[10], [11]	√	√	×
Bidirectional, model-based	[13], [14], [22]-[24]	√	×	√
	[15]	√	√	√
Hybrid	[16]	√	√	×
MASEF		×	√	×

As shown in Table I, existing FASE algorithms generally fall into two categories: ① unidirectional, data-driven algorithms that rely on historical data ; and ② bidirectional, model-based algorithms that require predefined system models. However, even hybrid algorithms [15], [16] do not achieve a truly symbiotic relationship between the state estimator and the forecaster, as they still cannot co-evolve online without historical pre-training. To address this gap, this paper proposes an online learning-based mutually-aided state estimation and forecasting (MASEF) algorithm, which integrates SE and PV forecasting into a coupled stochastic system and operates effectively without site-specific historical PV generation data or models. This algorithm fundamentally addresses the challenges of data and model unavailability. The main contributions of this paper are as follows.

1) SE and forecasting can mutually aid each other in a mutually-aided online learning loop within the coupled stochastic system, which completely eliminates the dependency on site-specific historical PV data or physical models.

2) The MASEF algorithm leverages generic, geographically-agnostic clear-sky data for pre-training. Online SE estimates then serve as as the primary learning signal , where the predictive variance of BNN is exploited to adaptively weight pseudo-measurements , enabling robust online adaptation.

3) By utilizing data-free PV forecasts as pseudo-measurements, the proposed algorithm restores observability at unmonitored PV injection buses. This increases measurement redundancy, thereby enhancing SE resilience against measurement noise and bad data.

4) The error dynamics of the coupled stochastic system are derived, and the conditions required for stable convergence are provided.

The remainder of this paper is organized as follows . Section II outlines the power system SE formulation . Section III details the BNN formulation. The MASEF algorithm and the convergence analysis are presented in Section IV. Case studies presented in Section V evaluate the performance of the proposed algorithm regarding robustness, convergence, and scalability. Finally, Section VI concludes the paper .

II. Power system SE Formulation

The objective of SE is to estimate the true state of the system under quasi-steady-state conditions. This is achieved using the following measurement model [25]-[28]:

$$\boldsymbol{z}_t=\boldsymbol{H}\bar{\boldsymbol{x}}_t+\boldsymbol{v}_t \tag{1}$$

where $z_t \in \mathbb{R}^{N_m}$ is the measurement vector at time t, comprising bus voltages, power injections, line flows, and pseudo-measurements for unmonitored loads or generation, and N_m is the number of measurements, comprising bus voltages and power injections; $\bar{x}_t \in \mathbb{R}^{N_n}$ is the true state vector comprising voltage magnitudes and phase angles, which is linked to the measurements by the matrix $\boldsymbol{H} \in \mathbb{R}^{N_m \times N_n}$, and N_n is the total number of true state states; and $\boldsymbol{v}_t \sim \mathcal{N}(0, \boldsymbol{R})$ is the zero-mean measurement noise vector with covariance $\boldsymbol{R} \in \mathbb{R}^{N_m \times N_m}$.

The measurement noise is assumed to be bounded such that $\left\| \boldsymbol{v}_t \right\| \le \alpha \left\| \boldsymbol{z}_t \right\|$, where α is the maximum allowable percentage error [29].

1) Weighted least squares (WLS)-based SE

The WLS-based SE estimate $\hat{\boldsymbol{x}}_t$ is obtained by minimizing the weighted sum of squared residuals:

$$\hat{\boldsymbol{x}}_t=(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z}_t \tag{2}$$

Using this estimated state, the reconstructed measurement vector $\hat{z}_t \in \mathbb{R}^m$ is given by $\hat{z}_t=\boldsymbol{H}\hat{\boldsymbol{x}}_t$. The discrepancy between the estimated and true measurements is quantified by the error vector $\boldsymbol{e}_{\mathrm{SE},t}$:

$$\boldsymbol{e}_{\mathrm{SE},t}=\hat{\boldsymbol{z}}_t-\bar{\boldsymbol{z}}_t=\boldsymbol{H}(\hat{\boldsymbol{x}}_t-\bar{\boldsymbol{x}}_t)-\boldsymbol{v}_t \tag{3}$$

where $\bar{\boldsymbol{z}}_t$ is the true state vector.

2) Pseudo measurements

In real-world systems, available telemetry is often insufficient to ensure full system observability. To address this, pseudo-measurements derived from historical load or generation profiles are augmented to the measurement vector. Alternatively, these pseudo-measurements are generated by FASE algorithms using forecasted values . In this study, we employ the latter approach, generating pseudo-measurements dynamically to restore observability and aid the SE process.

III. BNN FORMULATION

A. Probabilistic Forecasting with Uncertainty Quantification

A BNN offers principled uncertainty quantification, which is essential for the MASEF algorithm. Unlike standard neural networks that learn a single weight vector $\boldsymbol{w}$, a BNN learns the posterior distribution $p(\boldsymbol{w}|\mathcal{D})$ over weights given training data $\mathcal{D}$.

The predictive distribution for a new input $\boldsymbol{x}^*$ is obtained by marginalizing over the posterior (4) with Gaussian likelihood $p(y^*|\boldsymbol{x}^*, \boldsymbol{w}) = \mathcal{N}(y^*|f_{\boldsymbol{w}}(\boldsymbol{x}^*), \sigma_\epsilon^2)$, where y^* is the new output; $f_{\boldsymbol{w}}(\boldsymbol{x}^*)$ is the network output; and σ_ϵ^2 is the aleatoric uncertainty.

$$p(y^*|\boldsymbol{x}^*, \mathcal{D}) = \int p(y^*|\boldsymbol{x}^*, \boldsymbol{w}) p(\boldsymbol{w}|\mathcal{D}) \mathrm{d}\boldsymbol{w} \tag{4}$$

where $p(y^*|\boldsymbol{x}^*, \mathcal{D})$ is the posterior predictive distribution of a new output y^* given a new input $\boldsymbol{x}^*$ and the training data $\mathcal{D}$.

Since the true posterior is intractable, we use variational inference (VI) with a parameterized approximation $q_{\boldsymbol{\phi}}(\boldsymbol{w})$, which is optimized via the evidence lower bound (ELBO):

$$\mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{w})}(\ln p(\mathcal{D}|\boldsymbol{w})) - KL(q_{\boldsymbol{\phi}}(\boldsymbol{w}) \| p(\boldsymbol{w})) \tag{5}$$

where $\mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi})$ is the ELBO objective; $\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{w})}$ is the expectation taken under the variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{w})$; $p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is the prior distribution, and $\boldsymbol{I}$ is the identity matrix ; and $KL(\cdot)$ is the Kullback-Leibler divergence.

Using Monte Carlo sampling with T weight samples $\{\boldsymbol{w}_t\}_{t=1}^T \sim q_{\boldsymbol{\phi}}(\boldsymbol{w})$, the predictive mean μ_{pred} and variance σ_{pred} are expressed as:

$$\mu_{\text{pred}} = \frac{1}{T}\sum_{t=1}^{T} f_{\boldsymbol{w}_t}(\boldsymbol{x}^*) \tag{6}$$

$$\sigma_{\text{pred}}^2 = \underbrace{\frac{1}{T}\sum_{t=1}^{T}(f_{\boldsymbol{w}_t}(\boldsymbol{x}^*))^2 - \mu_{\text{pred}}^2}_{\text{Epistemic}} + \underbrace{\sigma_\epsilon^2}_{\text{Aleatoric}} \tag{7}$$

where $f_{\boldsymbol{w}_t}(\boldsymbol{x}^*)$ is the network output at test point $\boldsymbol{x}^*$ under the t^{th} weight sample $\boldsymbol{w}_t \sim q_{\boldsymbol{\phi}}(\boldsymbol{w})$ and T is the total number of Monte Carlo samples.

For each pseudo-measurement i at time t, we obtain: ① forecast mean (pseudo-measurement value) $\tilde{z}_{\text{BNN},i,t} \triangleq \mu_{\text{pred}}$; and ② total uncertainty (used as $R_{\text{pseudo},i,t}$ in WLS) $\sigma_{\text{BNN},i,t}^2 \triangleq \sigma_{\text{pred}}^2$.

B. BNN Architecture and Training

The BNN takes historical SE estimates $\hat{\boldsymbol{z}}_{i,t-1} = [\hat{z}_{i,t-1}, \hat{z}_{i,t-2}, \ldots, \hat{z}_{i,t-k}]^{\mathrm{T}} \in \mathbb{R}^k$ as input and outputs the predictive distribution:

$$p(\tilde{z}_t|\hat{\boldsymbol{z}}_{i,t-1}) = f_{\boldsymbol{\phi}_{i,t-1}}(\hat{\boldsymbol{z}}_{i,t-1}) \tag{8}$$

where $p(\tilde{z}_t|\hat{\boldsymbol{z}}_{i,t-1})$ is the predictive distribution over forecast output $\tilde{z}_t$ conditioned on this history; and $f_{\boldsymbol{\phi}_{i,t-1}}(\hat{\boldsymbol{z}}_{i,t-1})$ is the BNN mapping parameterized by variational parameters $\boldsymbol{\phi}_{i,t-1}$, which are updated online at each time step using the latest SE output. The BNN is updated using the forecasting error computed against the SE estimate:

$$e_{\text{BNN},i,t} = \tilde{z}_{\text{BNN},i,t} - \hat{z}_{i,t} \tag{9}$$

where $\hat{z}_{i,t}$ is the i^{th} component of SE estimates $\hat{\boldsymbol{z}}_t \in \mathbb{R}^m$. The actual forecasting error with respect to the true (unknown) value $\bar{z}_{i,t}$ is:

$$e_{\text{actual},i,t} = \tilde{z}_{\text{BNN},i,t} - \bar{z}_{i,t} = e_{\text{BNN},i,t} + e_{\text{SE},i,t} \tag{10}$$

where $e_{\text{SE},i,t} \triangleq \hat{z}_{i,t} - \bar{z}_{i,t}$ is the SE error. This reveals that SE errors propagates through the learning process of BNN.

IV. MASEF ALGORITHM AND CONVERGENCE ANALYSIS

A. Algorithm Description

The flowchart of the MASEF algorithm is shown in Fig. 1, where $t_{\max}$ is the total number of time steps.

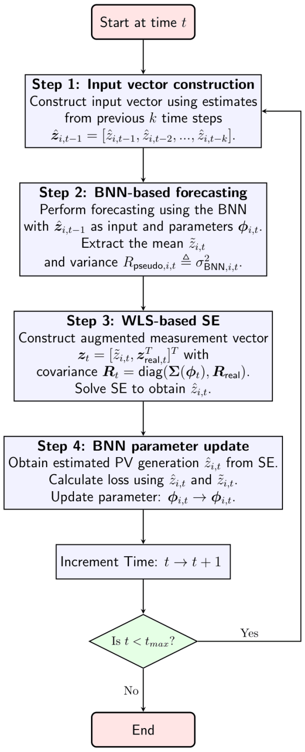

Fig. 1. Flowchart of MASEF algorithm.

B. Convergence Analysis

1) Dynamics of Coupled Stochastic System

BNN parameters $\boldsymbol{\phi}_t \in \mathbb{R}^d$ and SE estimates $\hat{\boldsymbol{z}}_t \in \mathbb{R}^m$ evolve jointly in the coupled stochastic system as:

$$\boldsymbol{\phi}_{t+1} = \boldsymbol{\phi}_t - \eta_t \hat{\nabla}_{\boldsymbol{\phi}} \mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi}_t, \hat{\boldsymbol{z}}_t) \tag{11}$$

$$\hat{\boldsymbol{z}}_t = \boldsymbol{H}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}_t^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}_t^{-1}\boldsymbol{z}_t \tag{12}$$

where $\hat{\nabla}_{\boldsymbol{\phi}}\mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi}_t, \hat{\boldsymbol{z}}_t)$ is the stochastic gradient estimator of ELBO loss function $\mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi}_t, \hat{\boldsymbol{z}}_t)$ using parameter $\boldsymbol{\phi}_t$ and $\hat{\boldsymbol{z}}_t$ as input at time step t; η_t is the learning rate at time step t; $\boldsymbol{z}_t = [(\boldsymbol{\mu}(\boldsymbol{\phi}_t))^{\mathrm{T}}, \boldsymbol{z}_{\text{real},t}^{\mathrm{T}}]^{\mathrm{T}}$ contains BNN-based forecasting vector $\boldsymbol{\mu}(\boldsymbol{\phi}_t) \in \mathbb{R}^p$ (with p pseudo-measurements) and physical sensor measurement vector $\boldsymbol{z}_{\text{real},t} \in \mathbb{R}^q$; and $\boldsymbol{R}_t = \text{diag}(R_{pseudo,i,t}, \boldsymbol{R}_{real}) = \text{diag}\left(\Sigma\left(\boldsymbol{\phi}_t\right), \boldsymbol{R}_{real}\right)$, $\boldsymbol{\Sigma}(\boldsymbol{\phi}_{t-1})$ is the predictive covariance matrix of BNN, and $\boldsymbol{R}_{\text{real}}$ is the covariance matrix of real measurements.

The SE error $\boldsymbol{e}_{\text{SE},t}$ and BNN-based forecasting error $\boldsymbol{e}_{\text{actual},t}$

are defined as: $\boldsymbol{e}_{\text{SE},t} \triangleq \hat{\boldsymbol{z}}_t - \bar{\boldsymbol{z}}_t$ and $\boldsymbol{e}_{\text{actual},t} \triangleq \boldsymbol{\mu}(\boldsymbol{\phi}_t) - \bar{\boldsymbol{z}}_{\text{pseudo},t}$. $\bar{\boldsymbol{z}}_{\text{pseudo},t}$ is the true value of forecasted pseudo measurements.

2) Self-stabilization Mechanism

Convergence is enabled by adaptive error coupling through uncertainty-weighted WLS integration.

Lemma 1 (WLS error coupling) For a single pseudo-measurement with BNN variance σ^2, the SE error e_{SE} satisfies:

$$e_{\text{SE}} = C(\sigma^2) e_{\text{actual}} + \delta(\boldsymbol{v}_{\text{real}}) \tag{13}$$

where e_{actual} is the BNN forecast error; $\delta(\boldsymbol{v}_{\text{real}})$ is the bounded physical sensor noise with $\mathbb{E}[\delta^2] \leq K_\delta^2 K_v^2$, and K_δ and K_v are the bound on the sensor noise sensitivity and bound on the magnitude of the physical sensor measurements $\boldsymbol{v}_{\text{real}}$, respectively; and $C(\sigma^2) = \gamma/(\sigma^2 + \gamma)$ is the coupling coefficient that satisfies ① $0 < C(\sigma^2) < 1$ for all $\sigma^2 > 0$; ② $\partial C/\partial \sigma^2 < 0$ (monotonically decreasing); ③ $\lim_{\sigma^2 \to \infty} C(\sigma^2) = 0$ and $\gamma = \boldsymbol{h}_1^{\text{T}} (\boldsymbol{H}_{\text{real}}^{\text{T}} \boldsymbol{R}_{\text{real}}^{-1} \boldsymbol{H}_{\text{real}})^{-1} \boldsymbol{h}_1 > 0$, where $\boldsymbol{h}_1$ is the Jacobian row vector associated with the BNN pseudo-measurement, and $\boldsymbol{H}_{\text{real}}$ is the Jacobian sub-matrix for physical sensor measurements.

Proof The detailed proof of Lemma 1 is given by Theorem 1 in Supplementary Material A.

Interpretation: when BNN variance σ^2 increases , the coupling coefficient $C(\sigma^2)$ decreases, automatically down-weighting erroneous estimates in SE. This self-stabilization prevents catastrophic error propagation during BNN training.

3) Convergence Guarantee

Under standard regularity conditions (i.e., Lipschitz continuity, strong convexity, and bounded noise, as detailed in Assumptions A1-A5 in Supplementary Material A), we establish convergence to a bounded region.

Theorem 2 (convergence to bounded region) Let $\boldsymbol{\phi}^* = \arg\min\limits_{\boldsymbol{\phi}} \mathbb{E}[\mathcal{L}(\boldsymbol{\phi}, \bar{\boldsymbol{z}})]$ denote the optimal parameters and define the Lyapunov function $V_t \triangleq \mathbb{E}[\left\| \boldsymbol{\phi}_t - \boldsymbol{\phi}^* \right\|^2]$,which measures the expected squared distance of the current parameters $\boldsymbol{\phi}_t$ from the optimum $\boldsymbol{\phi}^*$, where $L(\boldsymbol{\phi}, \bar{\boldsymbol{z}})$ is the expected negative ELBO loss of the BNN with respect to the true state $\bar{\boldsymbol{z}}$, serving as the training objective whose minimization drives the variational parameters $\boldsymbol{\phi}$ toward their optimal values $\boldsymbol{\phi}^*$. Under Assumptions A1-A5 in Supplementary Material A, the following convergence guarantees hold for the coupled MASEF system.

1) Constant learning rate

For $\eta_t = \eta$ with $0 < \eta < 1/m$, there exists:

$$\limsup_{t \to \infty} V_t \leq \frac{K_b^2}{m^2} + \frac{\eta (K_g^2 + K_b^2 + L_\phi^2)}{m} \tag{14}$$

where $K_b = L_z K_e$ is the gradient bias bound, and L_z is the Lipschitz constant of the loss gradient, controlling smoothness, K_e is the bound of the SE error defined in Lemma 1; K_g is the stochastic gradient noise bound; L_ϕ is the Lipschitz constant of the loss with regard to the true state, linking SE errors to gradient bias; and m is the strong convexity parameter.

2) Diminishing learning rate

For $\eta_t = c/(t+1)$ with $c > 1/(2m)$, there exists:

$$\limsup_{t \to \infty} V_t \leq \frac{K_b^2}{m^2} = \frac{(L_z K_e)^2}{m^2} \tag{15}$$

3) Almost sure convergence

Under diminishing rates, $\boldsymbol{\phi}_t$ converges almost surely to closed ball $\mathcal{B}(\boldsymbol{\phi}^*, r)$ with radius $r = L_z K_e / m$.

Proof The proof proceeds in three steps:

1) Decompose the stochastic gradient $\hat{\nabla}_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}_t, \hat{\boldsymbol{z}}_t)$ into ideal gradient $\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}_t, \bar{\boldsymbol{z}}_t)$, deterministic bias $\boldsymbol{B}_t$ (from SE error), and zero-mean Monte Carlo noise $\boldsymbol{M}_t$.

2) Show via Lemma 2 that the WLS coupling coefficient $C(\sigma^2)$, which governs how strongly the BNN forecast influences the SE estimate, decreases automatically as the BNN predictive variance σ^2 increases, thereby limiting the gradient bias to $\left\| \boldsymbol{B}_t \right\|^2 \leq K_b^2$ and preventing divergence even when forecast quality is poor.

3) Apply Lyapunov-based analysis to prove convergence with residual error proportional to physical sensor noise K_v^2.

The complete derivation and proof of Theorem 2 are given by Theorem A2 in Supplementary Material A.

Practical implications: Theorem 2 guarantees convergence even when training on SE estimates instead of true values. The irreducible error $K_b^2/m^2 \propto K_v^2$ reflects a fundamental limitation of learning from noisy coupled measurements. The self-stabilization mechanism (Lemma 1) ensures that errors remain bounded even during transient inaccuracies in the BNN.

V. CASE STUDY

Case studies were conducted in this section to analyze and assess the performance of the MASEF algorithm.

A. Case Study Design

1) Synthesis of PV Generation Data and Model

The update rate of the static SE in this study was set to once per minute. Consequently, an artificial PV dataset with one-minute resolution was generated using the Python package pvlib to simulate PV generation. To this end, the tenth entries in the California Energy Commission (CEC) PV and inverter databases were selected [30], [31]. Meteorological data for PV generation were sourced from a station at the University of Oregon, USA, provided by the Measurement and Instrumentation Data Center (MIDC) of the U.S. National Renewable Energy Laboratory (NREL) [32]. The synthesized dataset spans from May 1, 2020 to May 9, 2020, and is used in the subsequent case studies. In both the IEEE 14-bus and IEEE 30-bus systems, the PV system is placed at Bus 8. Additional details regarding the PV generation data synthesis process are listed in Table II.

2) IEEE SE Test Systems

Two standard systems, namely the IEEE 14-bus and IEEE 30-bus systems, were used for simulation. In these systems, state vectors consist of voltage angles and magnitudes [33], [34]. True state and measurement values for these systems were obtained by solving power flow using pandapower, a Python package for power system analysis [35]. The measurement set used for performing SE is identical to that in [36].

The MASEF algorithm operates under the assumption that PV generation data and models are unavailable. Hence, when performing SE using the MASEF algorithm on the IEEE 14-bus system, the real power injection measurement at Bus 8 was replaced by a pseudo-measurement generated by the BNN. Meanwhile, the reactive power injection at the same bus was retained as a virtual measurement (set to be zero). Removing both real and reactive power injection measurements at Bus 8 would render this bus unobservable, meaning that no SE solution would exist for the system. Thus, using the BNN-based forecasting value of PV generation as a pseudo-measurement restores the observability of the system. For the IEEE 30-bus system, a pseudo-measurement was introduced to Bus 8 in a similar manner.

TABLE II
DETAILS REGARDING PV GENERATION DATA SYNTHESIS PROCESS

Step	Detail
Define temperature model parameters	Use sapm for open_rack_glass_glass
Configure PV system	Select PV and inverter modules with surface_tilt = 30, surface_azimuth = 180
Set up model chain	Set aoi_model “no_loss”, spectral_model = “no_loss”
Fetch weather data	Station ID: UOSMRL, dates: 2020-01-01 to 2020-12-31
Preprocess weather data	Keep ghi, dhi, dni, temp_air, wind_speed
Run model	Use ModelChain.run_model() with the preprocessed data
Extract results	Output mc.results.dc

3) Random Noise

Gaussian-distributed random noise was introduced into both the measurements and system parameters. Specifically, 1% noise was added to the slack bus measurements and 3% to the other measurements, while 5% noise was added to the meteorological weather data [37], [38]. Note that weather data are required for all state-of-the-art SE algorithms against which the MASEF algorithm is compared, but not for the MASEF algorithm itself. The noise was generated using np.random.normal with zero mean and standard deviation. A random seed of 42 was set to ensure reproducibility. In Section V-F, the power system measurement noise is varied to 5% and 10% to compare the robustness of the MASEF algorithm against existing algorithms.

4) Error Metrics

Three error metrics were used to evaluate estimation and forecasting performance: mean absolute error (MAE), root mean square error (RMSE), and total vector error (TVE). For scalar values x_i (true) and y_i (estimated), MAE and RMSE are defined as:

$$\begin{cases} MAE=\dfrac{1}{N}\sum\limits_{i=1}^{N}\left| y_i-x_i \right| \\ RMSE=\sqrt{\dfrac{1}{N}\sum\limits_{i=1}^{N}(y_i-x_i)^2} \end{cases} \tag{16}$$

where N is the number of elements. For vectors $\boldsymbol{x}_i$ (true) and $\boldsymbol{y}_i$ (estimated), TVE is defined as:

$$TVE=\frac{\left\| \boldsymbol{y}_i-\boldsymbol{x}_i \right\|}{\left\| \boldsymbol{x}_i \right\|} \tag{17}$$

The forecasted time series of PV generation obtained using different methods were normalized by the capacity of PV plant C_{PV}:

$$S_{\text{normalized}}=\frac{S}{C_{PV}}\times 100\% \tag{18}$$

where S is the original time series; $S_{\text{normalized}}$ is the normalized time series; and C_{PV} is the capacity of the PV plant. The power output under the standard test condition (STC) $P_{STC}=180$ MW was used as C_{PV} in the normalization process.

5) BNN Settings

A BNN based on the Temporal Fusion Transformer (TFT) from the Darts Python library was used as the forecaster [39], [40]. TFT is known for its strong performance in time series forecasting and its ability to provide forecast confidence, as discussed in [39]. The TFT model was configured with an input sequence length of 60 steps to predict a single step ahead. It utilized 128-dimensional hidden layers, 16 LSTM layers, and an 8-head attention mechanism with a dropout rate of 0.1. Probabilistic forecasting was enabled via quantile regression (implemented as QuantileRegression in Darts), estimating uncertainty at quantiles ranging from 0.1 to 0.9 in increments of 0.05. Layer normalization and static covariates were incorporated to further enhance stability and predictive accuracy. The optimizer, torch.optim.Adam with a learning rate of 0.001, was executed via PyTorch Lightning on a GPU for computational efficiency. A random seed of 48 was set to ensure reproducibility. No scaling was applied to the data, except in simulations where meteorological data were used. In those cases, meteorological data were normalized by the peak values and constrained to the range [0, 1]. Additionally, the BNN outputs were clipped to the interval $[0, C_{PV}\times 1.96]$, reflecting the 99% Gaussian confidence interval. This constrains $e_{\text{BNN},i,t}$ to improve convergence, as explained in Section IV-B. For consistency, this constraint was also applied to the comparison algorithms in Section V-B for consistency. However, if the constraint resulted in a non-positive definite covariance matrix that prevented an SE solution, it was removed.

6) Pretraining

A pre-training phase of 100 epochs with a batch size of 32 was carried out using a generic one-month synthesized dataset. This dataset was generated through the pvlib.location.Location and location.get_clearsky functions. The clear-sky data were computed using approximate surface azimuth and tilt angles. By default, this computation employed the Ineichen model [41], thereby speeding up model convergence and improving performance in low-observability scenarios. It is worth noting that the only information required to generate this pre-training dataset is the geographic location of the PV plant. Plane of array (POA) irradiance was then derived using the Klucher model from global horizontal irradiance (GHI), diffuse horizontal irradiance (DHI), and direct normal irradiance (DNI) obtained from the synthesized clear sky data. Finally, the PV generation series is calculated as [42]:

$$P_{PV}=\frac{G_{POA}}{G_{STC}}P_{STC} \tag{19}$$

where G_{POA} and G_{STC} is the POA and STC irradiance (1000 W/m^2), respectively; and P_{STC} is the power output under STC.

The primary purpose of this pre-training is to enable the network to learn low-frequency patterns from generic clear-sky data, while reserving the online learning for capturing high-frequency patterns. The generic clear-sky data are geographically agnostic and are not intended to be accurate for the specific site; its sole purpose is to provide the model with a robust initial understanding of a typical diurnal PV generation profile. This, in turn, significantly accelerates the convergence of the subsequent online learning phase. The weights obtained from this 100-epoch pre-training serve as the initial weights ($\theta_{i,0}$) for the BNN at the start of the online learning phase. No parameters were frozen, and all weights were updated during the online learning phase.

In the MASEF algorithm, a TFT model performing self-regression (i.e., using the same time series for input and output) was pre-trained and then applied online. To validate the impact of data availability, a comparative variant named MASEF-W was also implemented, in which the TFT model takes scaled weather data as an additional input. This variant was used to verify that the enhanced tracking observed in KF-based algorithms is attributable to the additional information provided by real-time weather data.

7) Scalability Test Setup

To evaluate the scalability of the MASEF algorithm, we extended the simulation to the larger IEEE 30-bus system. In this setup, three independent PV generation units were installed at Buses 8, 16, and 19, representing a more distributed generation scenario. The corresponding real power injection measurements were replaced by pseudo-measurements.

Synthetic PV generation profiles for these three locations were generated using the Python package pvlib, utilizing real-world high-resolution weather data from the MIDC of NREL. The simulation parameters for each site were configured as follows.

1) Bus 8: a PV system located at the University of Arizona, USA (station ID: UAT, latitude: 32.112°, longitude: −110.793°, altitude: 893 m), representing a hot desert climate.

2) Bus 16: a PV system located at the Solar Technology Acceleration Center, USA (station ID: STAC, latitude: 39.74°, longitude: −104.881°, altitude: 1614 m), representing a high-altitude, semi-arid environment.

3) Bus 19: a PV system located at the University of Oregon, USA (station ID: UOSMRL, latitude: 44.0583°, longitude: −123.076°, altitude: 128 m), representing a temperate marine climate.

For each location, a consistent PV system model was applied for the year 2020 using the same settings introduced in Table II. These diverse geographical and climatic profiles ensure that the MASEF algorithm was tested against a variety of solar irradiance patterns, including different cloud cover dynamics and seasonal variations, thereby rigorously assessing its performance in a multi-site, scalable context. Additionally, to evaluate the trade-off between accuracy and computational speed, the number of fitting epochs per time step was increased from 1 to 50.

8) Scenario Design for Robustness Analysis

To rigorously evaluate the engineering robustness of the MASEF algorithm, a simulation framework was developed using the IEEE 14-bus system. The simulation spans a duration of 10 days, comprising 14400 one-minute time steps ($T=14400$), and is driven by time-series PV generation data. Beyond the base scenario, four distinct disturbance scenarios were designed to stress-test the stability and convergence properties of the proposed algorithm under realistic grid contingencies and adversarial conditions.

1) Scenario A: base scenario. This scenario represents normal grid operation with standard measurement noise and intact topology. Its performance serves as the benchmarking performance.

2) Scenario B: topology change ($N-1$ contingency). A persistent topology error was simulated by disconnecting Line 4 (connecting Bus 2 and Bus 4) for the entire simulation duration. This scenario tests the ability of the proposed algorithm to maintain estimation accuracy when the physical network structure deviates from the model used by the estimator, a common occurrence during circuit breaker tripping.

3) Scenario C: load step change. To verify the error convergence speed and the validity of the quasi-steady-state assumption, a sudden step increase in demand was applied. The active and reactive loads at Buses 2, 3, and 5 were randomly and simultaneously scaled up by a factor of 1.2 (a 20% increase) for the entire simulation duration.

4) Scenario D: false data injection (FDI) attack. A targeted cyber-physical attack was simulated on the measurement infrastructure. The measurement at index 17 was subjected to a continuous FDI attack where its value was multiplied by a factor of $\lambda=2.0$ at every time step. This scenario tests the resilience of the feedback loop against bad data propagation.

5) Scenario E: system fluctuation. To simulate a highly volatile environment or sensor degradation, random Gaussian noise was injected into all load and generation values (except at the PV bus) at every time step. The noise level was set to be 5% ($\sigma=0.05$), which is significantly higher than the standard measurement uncertainty, testing the filtering capabilities of the MASEF algorithm.

B. Algorithms Used for Comparison

Given that the MASEF algorithm draws inspiration from the KF, we employ a UKF and an extended Kalman filter (EKF) as comparative algorithms to estimate the power system state vector and forecast PV generation [13], [22], [23]. UKF and EKF variants utilizing a power flow model to obtain estimated power generation for updating the forecaster were also tested; however, these exhibited error accumulation over time, leading to eventual divergence of the estimators. For the UKF and EKF that do not use power flow models, the PV generation model was used to convert weather data (i.e., GHI and air temperature measurements) into PV generation for updating the PV forecasting using (20) and (21) [43], [44], since real-time PV generation data were assumed to be unavailable. The forecasted PV generation data were then used as pseudo-measurements in the update steps of the UKF and EKF .

$$P_{PV} = P_{STC} \frac{G}{G_{STC}} \left[1 + \gamma_r \left(T_{cell} - T_{STC}\right)\right] \tag{20}$$

$$T_{cell} = T_{air} + \frac{G}{800.0}\left(T_{NOCT} - 20\right) \tag{21}$$

where T_{cell} is the temperature of the PV cell ; T_{air} is the ambient air temperature; G is the solar irradiance incident on the PV panel; T_{NOCT} is the nominal operating cell temperature, which is typically provided by the manufacturer; γ_r is the temperature coefficient of power, which is typically a negative value; and T_{STC} is the STC temperature, which is typically 25°C.

Both UKF and EKF were implemented using filterpy, a Python library for KFs and other estimation filters [45]. Holt's exponential smoothing, which extends standard exponential smoothing by incorporating a trend component to update both level and trend estimates, was implemented as a forecaster for both UKF and EKF [46]. Holt's exponential smoothing is defined as:

$$l_t = \alpha_s y_t + (1 - \alpha_s)(l_{t-1} + b_{t-1}) \tag{22}$$

$$b_t = \beta_s (l_t - l_{t-1}) + (1 - \beta_s) b_{t-1} \tag{23}$$

$$\hat{y}_{t+h} = l_t + h b_t \tag{24}$$

where y_t is the actual value at time t; l_t is the estimated level at time t; b_t is the estimated trend at time t; α_s and β_s are the smoothing parameters (with $0 < \alpha_s < 1, 0 < \beta_s < 1$); and h is the number of periods ahead to be forecasted. Both EKF and UKF integrate Holt's exponential smoothing with $\alpha_s = 0.99$ and $\beta_s = 0.01$.

The EKF is designed to estimate power system states and PV generation by relying on a non-linear measurement function and its Jacobian matrix to linearize the relationships between the state variables and measurements [47]. Given the high frequency of power system SE, the change in the state vector between adjacent time steps is typically small. As will be demonstrated in the results, EKF and UKF achieve accurate forecasting and estimation with a small beta value. In contrast, the UKF uses Sigma points to handle non-linearities, thereby avoiding explicit linearization [48]. This design enables the UKF to better capture the system dynamics by incorporating weather-dependent inputs such as solar irradiance and temperature into its state transition function. Both algorithms use initial covariance matrices for state, and constant covariance matrices for process noise and measurement noise. Specifically, all diagonal entries are set to be 10^{-2} for the initial covariance matrices for state and 10^{-4} for the constant covariance matrices for process noise, and adequate values for the constant covariance matrices for measurement noise are specified in Section V-A-3). For the UKF, Sigma points are initialized using the Merwe scaling method (implemented as MerweScaledSigmaPoints from filterpy.kalman library), with parameters controlling spread ($\alpha = 0.1$), distribution shape ($\beta = 2.0$), and scaling ($\kappa = 0$) [49]. Another comparison algorithm is a UKF-based FASE enhanced by support vector machine (SVM), which will be called SVM-UKF [16]. The SVM is pre-trained using one-month synthesized PV generation data.

Several additional algorithms were included for comparison. It should be noted that MASEF algorithm, NN-based FASE, and MASEF-W are collectively referred to as WLS-based algorithms since they all employ WLS to perform SE. In contrast, WLS baseline denotes the benchmark algorithm in which WLS has access to all 32 measurement [25], including PV generation data with a fixed measurement noise variance of 10^{-4} [26] - [28]. The final algorithm, NN-FASE, shares the same TFT architecture and WLS-based SE procedure as MASEF algorithm, but is trained and updated using the true PV generation data rather than SE estimates [7], [8], serving as an upper-performance reference for the forecasting-aided SE framework.

TABLE III
COMPARISON RESULTS OF SE AND PV FORECASTING METRICS FOR DIFFERENT ALGORITHMS

Algorithm	Data or model requirement			SE					PV forecasting (p.u.)	
	PV generation data	PV generation model	Weather data	TVE (%)	Magnitude (p.u.)		Angle (°)		MAE	RMSE
					MAE	RMSE	MAE	RMSE		
WLS baseline [25]-[28]	√	×	×	2.6544	0.0077	0.0098	0.0239	0.0456		
NN-FASE [11]	√	×	×	2.6544	0.0079	0.0098	0.0239	0.0456	0.0635	0.1242
SVM-UKF [16]	√	×	×	1.1086	0.0082	0.0107	0.0129	0.0061	0.0425	0.0460
UKF	×	√	√	2.3459	0.0085	0.0126	0.0203	0.0335	0.0638	0.0821
EKF	×	√	√	2.6715	0.0076	0.0107	0.0242	0.0388	0.0488	0.0919
MASEF-W	×	×	√	2.6781	0.0079	0.0098	0.0239	0.0456	0.0572	0.177
MASEF	×	×	×	2.6544	0.0079	0.0098	0.0239	0.0456	0.0642	0.1257

C. Comparative Analysis

The comparative analysis summarized in Table III and Fig. 2 evaluates the performance variations in SE and PV forecasting algorithms across different scenarios of input data availability. MASEF algorithm is distinguished by its ability to enable accurate PV forecasting without requiring direct PV generation data or models, thereby enhancing SE robustness through forecast-aided pseudo-measurements.

1) Algorithms Using Power System Measurements and PV

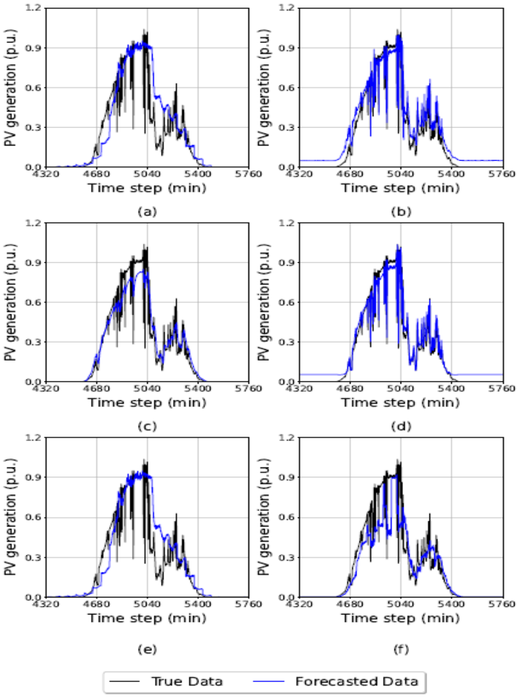

Fig. 2. Comparison results of PV tracking dynamics for different algorithms. (a) MASEF algorithm. (b) UKF. (c) EKF. (d) SVM-UKF. (e) NN-FASE. (f) MASEF-W.

Generation Data

Algorithms that leverage direct PV generation data, including WLS baseline, NN-FASE, and SVM-UKF, exhibit notable performance advantages. SVM-UKF achieves superior SE accuracy (TVE: 1.1086%, MAE of angle: 0.0061 rad) due to its advanced non-linear estimation capabilities, clearly outperforming WLSbased methods (TVE: 2.6744%, MAE of angle: 0.0239 rad). The superiority of SVM-UKF stems from its ability to capture complex non-linear relationships in the system state, which WLS baseline, relying on linearization around an operating point, may fail to capture fully. In PV forecasting, SVM-UKF similarly attains the best accuracy (MAE: 0.0425 p.u.), closely matching the ground truth. NN-FASE achieves good forecasting accuracy (MAE: 0.0635 p. u.) and moderate SE performance consistent with WLS baseline. However, all these algorithms share a critical limitation: they depend directly on PV generation data, which restricts their applicability in scenarios where such telemetry is unavailable.

2) Algorithms Using Power System Measurements, PV Generation Models, and/or Weather Data

When direct PV generation data are unavailable, algorithms utilizing weather data in combination with standard power system measurements, e. g., UKF, EKF, and MASEF-W, provide viable alternative solutions. UKF and EKF effectively utilize weather-informed PV generation models, achieving improved SE performance (MAE of angle for UKF: 0.0203 rad) compared to WLS baseline.

The inclusion of weather data in MASEF-W notably enhances forecast accuracy (MAE: 0.057 p.u.) compared to the algorithm that applies self-regression. This improvement is visually confirmed in Fig. 2, which shows a reduced forecast delay. This is because weather data (particularly irradiance) acts as a leading indicator: clouds are detected before the power drop occurs, whereas self-regressive algorithms must wait for the drop to manifest in the estimates before reacting.

3) MASEF Algorithm Using Only Power System Measurements

In severely constrained scenarios where neither PV generation data nor weather-based PV generation models are available, the MASEF algorithm demonstrates notable effectiveness. Despite this strict data limitation, the MASEF algorithm maintains SE accuracy comparable to the WLS baseline (TVE: 2.6544%, MAE of angle: 0.0239 rad).

For PV forecasting, MASEF algorithm achieves practically acceptable accuracy (MAE: 0.0642 p. u.), capturing key PV generation trends, albeit with slightly more deviations and delays compared to algorithms utilizing weather or direct PV generation data. The slight forecast delay visible in Fig. 2(a) is an expected characteristic of the MASEF algorithm: because the MASEF algorithm learns from past SE estimates rather than exogenous weather data, a minor lag is inherent to its data-free design. Importantly, the BNN acts as a low-pass filter, smoothing out noise from the estimates of WLS before feeding them back, thereby preventing error amplification. MASEF algorithm effectively restores observability at PV buses and increases measurement redundancy, improving the overall reliability of SE without external data dependencies.

D. Extended Test

To further verify the stability of the MASEF algorithm, a simulation with an extended operational period of 30 days (43200 one-minute time steps) is presented.

As shown in Table IV, the performance metrics of the MASEF algorithm over this 30-day operational period remain highly consistent with those from the 5-day test (shown in Table III). Specifically, the TVE remains at 2.5486%, and the MAE of PV forecasting actually improves slightly to 0.041 p.u., suggesting that the BNN continues to refine its learning of daily generation patterns over time. This long-term stability experimentally verifies the theoretical convergence analysis presented in Section IV-B and confirms the robustness of the "quasi-steady state assumption" over extended operational periods. It demonstrates that the mutually-aided online learning loop does not suffer from error accumulation or divergence; rather, the BNN successfully tracks the system state without drifting, even in the absence of external ground-truth data.

E. Scalability Test

To evaluate the scalability of the MASEF algorithm, simulations were conducted on a larger IEEE 30-bus system. The analysis is divided into two scenarios: ① Scenario 1: a single PV system is located at Bus 8; and ② Scenario 2: three PV systems are located at Buses 8, 16, and 19. The comparative results for the IEEE 30-bus system are summarized in Table V.

1) Analysis of Scenario 1

In Scenario 1, MASEF algorithm and WLS baseline exhibit identical SE performance (TVE: 1.5245%). This result con-

TABLE IV
COMPARISON RESULTS OF SE AND PV FORECASTING METRICS FOR DIFFERENT ALGORITHMS WITH OPERATIONAL PERIOD OF 30 DAYS

Algorithm	Data or model requirement			SE					PV forecasting (p.u.)	
	PV generation data	PV generation model	Weather data	TVE (%)	Magnitude (p.u.)		Angle (°)		MAE	RMSE
					MAE	RMSE	MAE	RMSE		
WLS baseline [25]-[28]	√	×	×	2.5486	0.0086	0.0297	0.0222	0.0407		
NN-FASE [11]	√	×	×	2.6544	0.0079	0.0098	0.0239	0.0456	0.0620	0.0760
SVM-UKF [16]	√	×	×	2.0801	0.0077	0.0116	0.0179	0.0310	0.0621	0.0762
UKF	×	√	√	2.2515	0.0075	0.0105	0.0197	0.0387	0.0616	0.0755
EKF	×	√	√	2.2072	0.0075	0.0104	0.0192	0.0321	0.0373	0.0741
MASEF-W	×	×	√	2.5480	0.0086	0.0297	0.0222	0.0407	0.0676	0.1208
MASEF	×	×	×	2.5480	0.0086	0.0297	0.0222	0.0407	0.0410	0.0920

TABLE V
COMPARATIVE RESULTS FOR IEEE 30-BUS SYSTEM

Scenario	Algorithm	TVE (%)	SE				Location	PV forecasting (p.u.)	
			Magnitude (p.u.)		Angle (°)				
			MAE	RMSE	MAE	RMSE		MAE	RMSE
1	MASEF	1.5245	0.0056	0.0075	0.0125	0.0237	Bus 8	0.0642	0.128
	WLS baseline [25]-[28]	1.5245	0.0056	0.0075	0.0125	0.0237			
2	MASEF	3.9053	0.0091	0.0490	0.0351	0.0599	Bus 8	0.0602	0.1084
							Bus 16	0.0256	0.0707
							Bus 19	0.0346	0.0841
	WLS baseline [25]-[28]	3.7000	0.0090	0.0490	0.0330	0.0589			

trasts with that observed in the sparser IEEE 14-bus system. The high measurement redundancy in the IEEE 30-bus system (78 measurements) makes the influence of a single pseudo-measurement at Bus 8 negligible, allowing the MASEF algorithm to converge to the standard solution of WLS while providing the added benefit of PV forecasting (MAE: 0.0642 p.u.).

2) *Analysis of Scenario 2*

In Scenario 2, both algorithms exhibit a degradation in SE accuracy due to the increased system stochasticity. The MASEF algorithm shows a slight relative reduction in SE accuracy compared to WLS baseline (TVE: 3.9053% versus 3.7000%), which represents the expected trade-off for operating without direct telemetry.

Forecasting accuracy at all three locations is improved compared to that in Scenario 1, which can be attributed to the increased fitting intensity (50 epochs versus 1 epoch). However, the performance varies significantly by location: Buses 16 and 19 achieve high accuracy (MAE: 0.0256 and 0.0346 p.u.), while Bus 8 lags behind (MAE: 0.0602 p.u.). This discrepancy provides critical insight into the trade-off between learning intensity and overfitting.

3) *Visual Analysis: Overfitting and Redundancy*

The time-series results in Fig. 3 reveal that while Buses 16 and 19 track the true generation closely, Bus 8 exhibits a distinct hysteresis, or lag effect, during the evening ramp-down.

It is important to note that in the IEEE 14-bus system, where fitting is limited to 1 epoch, Bus 8 does not exhibit this inertia despite having similar measurement constraints. This suggests that the topology alone is not the primary cause. Instead, the main driver is the overfitting resulting from the increased epoch count (50 epochs). By aggressively fitting to the most recent estimates of WLS, the BNN learns to replicate transient estimation errors.

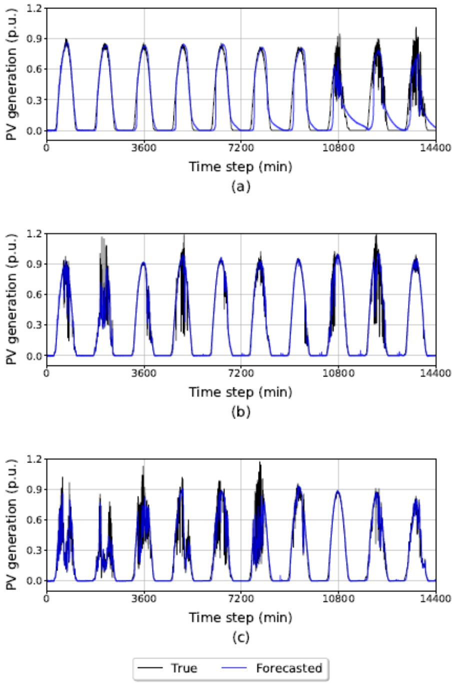

Fig. 3. Impact of measurement redundancy on forecasting accuracy in IEEE 30-bus system. (a) Bus 8. (b) Bus 16. (c) Bus 19.

This leads to a key finding regarding the stability of WLS baseline:

1) High redundancy at Buses 16 and 19: these buses are connected by two pairs of line measurements. This physical redundancy anchors the WLSestimator, preventing it from being pulled by the overfitted predictions of BNN. The WLS estimator corrects the BNN, thereby maintaining high accuracy.

2) Low redundancy at Bus 8: monitored by only a single pair of line measurements, Bus 8 lacks sufficient physical redundancy to override the BNN. Consequently, the overfitted forecast "pulls" the WLS baseline, creating a self-reinforcing lag.

Therefore, while increasing the number of fitting epochs significantly improves tracking performance by mitigating underfitting, it also introduces an overfitting risk that must be counterbalanced by sufficient measurement redundancy in the SEpart of the MASEF algorithm. This issue could be addressed in future work by introducing robust estimation methods better suited to handling pseudo-measurements.

F. Robustness Test

The SE and PV forecasting metrics with noise levels of 5% and 10% are shown in Table VI.

TABLE VI
SE AND PV FORECASTING METRICS WITH NOISE LEVELS OF 5% AND 10%

Algorithm	Noise level (%)	SE					PV forecasting (p.u.)	
		TVE (%)	Magnitude (p.u.)		Angle (rad)		MAE	RMSE
			MAE	RMSE	MAE	RMSE		
WLS baseline [25]-[28]	5	3.0920	0.0077	0.0099	0.0286	0.0505		
	10	4.3579	0.0078	0.0108	0.0416	0.0686		
NN FASE [11]	5	3.0920	0.0077	0.0099	0.0286	0.0505	0.0635	0.1242
	10	4.3579	0.0078	0.0108	0.0416	0.0686	0.0645	0.1242
SVM-UKF [16]	5	1.1922	0.0082	0.0099	0.0075	0.0147	0.0425	0.0460
	10	1.7308	0.0076	0.0108	0.0143	0.0253	0.0425	0.0460
UKF	5	2.3623	0.0085	0.0130	0.0205	0.0337	0.0635	0.0819
	10	2.3917	0.0091	0.0134	0.0206	0.0335	0.0621	0.0808
EKF	5	2.6624	0.0077	0.0107	0.0241	0.0389	0.0488	0.0919
	10	2.6523	0.0077	0.0107	0.0240	0.0386	0.0488	0.0919
MASEF-W	5	3.0966	0.0077	0.0099	0.0286	0.0504	0.0576	0.118
	10	4.3628	0.0078	0.0108	0.0416	0.0683	0.0579	0.1184
MASEF	5	3.0920	0.0077	0.0099	0.0286	0.0505	0.0642	0.1256
	10	4.3579	0.0078	0.0108	0.0416	0.0686	0.0642	0.1254

SVM-UKF consistently exhibits superior robustness, maintaining the lowest sensitivity (TVE: 1.2%-1.7%) due to its advanced non-linear modeling, which naturally filters non-Gaussian noise components more effectively than linear approximation algorithms. WLS-based algorithms, including MASEF algorithm, show increased sensitivity (TVE: 3.920% -4.3579%), as they rely on the standard least-squares objective, which is sensitive to outliers and high noise variances.

However, it is crucial to note that MASEF algorithm performs on par with other WLS-based algorithms that do use PV generation data. This indicates that the data-free nature of MASEF algorithm does not introduce additional vulnerability. Its BNN component effectively learns the noisy signal without diverging. While it does not surpass the noise floor of the underlying WLS solver, MASEF algorithm successfully maintains observability and provides forecasting capabilities (MAE: 0.064 p.u.) that remain remarkably stable even as system noise doubles from 5% to 10%.

G. Scenario Analysis

To further evaluate the resilience of MASEF algorithm and demonstrate its engineering robustness, four distinct operational scenarios are simulated: FDI attack, sudden load change (step change), topology change (such as circuit breaker tripping), and system fluctuation (3%). The performance metrics in different operational scenarios are summarized in Table VII.

1) FDI Attack and System Fluctuation

In the FDI attack and system fluctuation scenarios, the performance metrics of MASEF algorithm (TVE: 2.9095%) are consistent with those of the WLS baseline and NN-FASE. This result is significant for a mutually-aided online learning loop; it confirms that the BNN-based forecasting component does not amplify the errors introduced by bad data or high noise. While the UKF achieves a lower TVE (2.2%) due to its inherent filtering capabilities (Sigma points), MASEF algorithm demonstrates that its mutually-aided online learning loop remains stable and does not diverge even when the underlying measurements are corrupted.

2) Topology Change

In the topology change scenario (simulating a circuit breaker trip where Line 4 is disconnected), all algorithms suffer significant degradation in SE accuracy. MASEF algorithm, WLS baseline, and NN-FASE all exhibit a TVE of approximately 55%. This highlights a fundamental limitation of model-based SE, which MASEF algorithm relies on for its SE step: when the physical system topology changes but the Jacobian matrix of the estimator $\boldsymbol{H}$ is not updated, the

TABLE VII
PERFORMANCE METRICS IN DIFFERENT OPERATIONAL SCENARIOS

Scenario	Method	SE					PV forecasting (p.u.)	
		TVE (%)	Magnitude (p.u.)		Angle (rad)			
			MAE	RMSE	MAE	RMSE	MAE	RMSE
FDI attack	WLS baseline [25]-[28]	2.9144	0.0107	0.0498	0.0245	0.0455		
	NN-FASE [11]	2.9144	0.0107	0.0498	0.0245	0.0455	0.0649	0.1217
	SVM-UKF [16]	2.3631	0.0083	0.0123	0.0206	0.0337	0.0617	0.0762
	UKF	2.257	0.0084	0.0124	0.0191	0.0312	0.0620	0.0772
	EKF	2.5392	0.0081	0.0114	0.0225	0.0363	0.0441	0.0834
	MASEF	2.9144	0.0107	0.0498	0.0245	0.0455	0.0657	0.1235
Sudden load change	WLS baseline [25]-[28]	3.4967	0.0115	0.0500	0.0306	0.0497		
	NN-FASE [11]	3.4967	0.0115	0.0500	0.0306	0.0497	0.0649	0.1217
	SVM-UKF [16]	3.0238	0.0091	0.0129	0.0282	0.0403	0.0618	0.0764
	UKF	2.9395	0.0091	0.0129	0.0273	0.0374	0.0620	0.0774
	EKF	3.0792	0.0093	0.0125	0.0287	0.0406	0.0441	0.0834
	MASEF	3.4967	0.0115	0.0500	0.0306	0.0497	0.0657	0.1235
Topology change	WLS baseline [25]-[28]	54.6243	0.2835	0.4465	0.3870	0.5472		
	NN-FASE [11]	54.6243	0.2835	0.4465	0.3870	0.5472	0.0649	0.1217
	SVM-UKF [16]	53.8918	0.2719	0.4221	0.3897	0.5433	0.0620	0.0766
	UKF	53.4770	0.2703	0.4205	0.3866	0.5397	0.0623	0.0774
	EKF	NaN	NaN	NaN	NaN	NaN	NaN	NaN
	MASEF	54.6243	0.2835	0.4465	0.3870	0.5472	0.0657	0.1235
System fluctuation	WLS baseline [25]-[28]	2.9095	0.0101	0.0496	0.0248	0.0456		
	NN-FASE [11]	2.9095	0.0101	0.0496	0.0248	0.0456	0.0649	0.1217
	SVM-UKF [16]	2.3657	0.0082	0.0120	0.0208	0.0341	0.0604	0.0754
	UKF	2.2300	0.0081	0.0119	0.0194	0.0315	0.0606	0.0761
	EKF	2.5147	0.0076	0.0106	0.0226	0.0363	0.0441	0.0835
	MASEF	2.9095	0.0101	0.0496	0.0248	0.0456	0.0657	0.1235

SE fails. However, it is worth noting that while the EKF fails to converge entirely (resulting in NaN values), the MASEF algorithm maintains numerical stability, continuing to produce valid (albeit physically inaccurate) outputs.

3) *Sudden Load Change*

The sudden load change scenario simulates a sudden step increase in demand. Here, the MASEF algorithm achieves an MAE of PV forecasting of 0.0657 p.u.. This performance is highly comparable to the NN-FASE (MAE of PV forecasting: 0.0649 p. u.) and the UKF (MAE of PV forecasting: 0.0620 p.u.).

This result validates the adaptive capability of MASEF algorithm. Unlike NN-FASE, which relies on a model pretrained on historical data that may not include such step changes, MASEF algorithm adapts to the new system operating point via online learning. Although the WLS-based SE error increases slightly (TVE: 3.4967%) due to the transient, the BNN-based forecasting successfully tracks the generation profile without drifting. This confirms that the mutually-aided online learning loop can effectively handle sudden operating point shifts, performing as well as algorithms that require extensive historical data or physical models.

H. Computational Complexity

To evaluate the practical feasibility of the MASEF algorithm, its computational cost is analyzed. The simulation is performed on a workstation equipped with an Intel i7-12700 CPU, 32 GB of RAM, and an NVIDIA RTX 3070 (8 GB) GPU.

For a simulation duration of 14400 time steps (representing 10 days of operation at a 1-min resolution) in the IEEE 14-bus system, the average total runtime is approximately 7 hours. This corresponds to an average execution time of approximately 1.75 s per time step. Given that the target application is static SE with a typical update rate of 1 min, the execution time is well within the real-time operational requirement ($1.75\,\mathrm{s} \ll 60\,\mathrm{s}$). In the scalability test, the configuration with a single PV system requires approximately the same total runtime (also over 14400 time steps). In comparison, the configuration with three PV systems increases the per-step execution time to 6.75 s per time step—a 3.86-fold increase—while yielding improved forecasting accuracy.

We conclude that the primary computational bottleneck is the online learning of the TFT model at each step, given the significant increase in the computation time observed when introducing two additional PV systems and increasing the number of fitting epochs. The current TFT-based implementation prioritizes computational efficiency by performing a gradient update with one epoch at each step. A trade-off exists between the number of fitting epochs (which enables better learning of instantaneous trends) and computational efficiency. While the current TFT-based implementation is sufficient for real-time operation, the computational burden and forecasting accuracy could be further optimized in future work by replacing the TFT model with a different BNN architecture or by adopting an alternative update strategy.

VI. CONCLUSION

This paper proposed validated MASEF algorithm, which performs SE and PV forecasting and without relying on site-specific historical PV generation data or models. We demonstrated that the MASEF algorithm maintained SE accuracy comparable to WLS-based methods, while simultaneously generating reliable PV forecasts and restoring observability at unmonitored PV injection buses. The robustness of the MASEF algorithm to measurement noise and its scalability to larger systems were also validated. The error dynamics of the coupled stochastic system were rigorously derived, and its stability was formally proven using a Lyapunov-based analysis.

The case studies highlighted a critical trade-off inherent to the mutually-aided online learning loop. It couldbe observed that increasing the online learning intensity (fitting epochs increased from 1 to 50) significantly improves the ability of the proposed algorithm to track rapid power generation changes, effectively mitigating underfitting. However, the mutually-aided online learning loop introduced a risk of overfitting to transient estimation errors. The scalability analysis reveals that this risk was effectively neutralized in areas with high measurement redundancy (e.g., Buses 16 and 19), where physical measurements anchored the state estimator against forecast errors. Conversely, in areas with low measurement redundancy (e. g., Bus 8), the overfitted forecast could create a self-reinforcing "inertia", leading to a slight degradation in performance.

Thus, we conclude that while increasing the number of online learning epochs improves tracking performance, high measurement redundancy is a prerequisite for ensuring robustness against overfitting in the mutually-aided online learning loop. Future work will focus on addressing this limitation by integrating robust estimation techniques into the update stage. This would allow the system to automatically down-weight forecast that conflict with physical measurements, thereby breaking the inertia in scenarios with low measurement redundancy. Additionally, we plan to investigate the impact of communication latency within the feedback loop and to extend the MASEF algorithm to other renewable energy sources, broadening the applicability of the MASEF algorithm.

REFERENCES

[1] PJM. (2025, Oct.). Energy & ancillary services market operations. [Online]. Available: https://www.pjm.com/-/media/documents/manuals/m11.ashx

[2] S. Sobri, S. Koohi-Kamali, and N. A. Rahim, "Solar photovoltaic generation forecasting methods: a review," *Energy Conversion and Management*, vol. 156, pp. 459-497, Jan. 2018.

[3] R. Ahmed, V. Sreeram, Y. Mishra *et al.*, "A review and evaluation of the state-of-the-art in PV solar power forecasting: techniques and optimization," *Renewable and Sustainable Energy Reviews*, vol. 124, p. 109792, May 2020.

[4] M. Abdelsattar, A. AbdelMoety, and A. Emad-Eldeen, "Advanced machine learning techniques for predicting power generation and fault detection in solar photovoltaic systems," *Neural Computing and Applications*, vol. 37, pp. 1-20, Jan. 2025.

[5] PJM. (2023, Nov.). Generator operational requirements. [Online]. Available: https://www. pjm. com/-/media/DotCom/documents/manuals/archive/m14d/m14dv64-generator-operational-requirements-11-15-2023.pdf

[6] D. Stowell, J. Kelly, D. Tanner *et al.*, "A harmonised, high-coverage, open dataset of solar photovoltaic installations in the UK," *Scientific Data*, vol. 7, no. 1, p. 394, Nov. 2020.

[7] M. B. D. C. Filho and J. C. S. de Souza, "Forecasting-aided state estimation—part I: panorama," *IEEE Transactions on Power Systems*, vol. 24, no. 4, pp. 1667-1677, Nov. 2009.

[8] M. B. D. C. Filho, J. C. S. de Souza, and R. S. Freund, "Forecasting-aided state estimation—part II: implementation," *IEEE Transactions on Power Systems*, vol. 24, no. 4, pp. 1678-1685, Nov. 2009.

[9] M. Huang, Z. Wei, and Y. Lin, "Forecasting-aided state estimation based on deep learning for hybrid AC/DC distribution systems," *Applied Energy*, vol. 306, p. 118119, Jan. 2022.

[10] X. Ji, Z. Yin, Y. Zhang *et al.*, "Real-time robust forecasting-aided state estimation of power system based on data-driven models," *International Journal of Electrical Power & Energy Systems*, vol. 125, p. 106412, Feb. 2021.

[11] E. Manitsas, R. Singh, B. C. Pal *et al.*, "Distribution system state estimation using an artificial neural network approach for pseudo measurement modeling," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1888-1896, Nov. 2012.

[12] K. Nishiya, J. Hasegawa, and T. Koike, "Dynamic state estimation including anomaly detection and identification for power systems," *IEE Proceedings C (Generation, Transmission and Distribution)*, vol. 129, no. 5, pp. 192-198, Sept. 1982.

[13] G. Barchi and D. Macii, "A photovoltaics-aided interlaced extended Kalman filter for distribution systems state estimation," *Sustainable Energy, Grids and Networks*, vol. 26, p. 100438, Jan. 2021.

[14] M. Huang, Z. Wei, G. Sun *et al.*, "A historical data-driven unscented Kalman filter for distribution system state estimation," in *Proceedings of 2017 IEEE PES General Meeting*, Chicago, USA, Jul. 2017, pp. 1-5.

[15] C. Carquex, C. Rosenberg, and K. Bhattacharya, "State estimation in power distribution systems based on ensemble Kalman filtering," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6600-6610, Nov. 2018.

[16] D. Xu, Z. Wu, J. Xu *et al.*, "A pseudo measurement modeling based forecasting aided state estimation framework for distribution network," *International Journal of Electrical Power & Energy Systems*, vol. 160, p. 110116, Feb. 2024.

[17] R. M. Neal, *Bayesian Learning for Neural Networks*. New York: Springer Science & Business Media, 1996.

[18] L. Yang, X. Meng, and G. E. Karniadakis, "B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data," *Journal of Computational Physics*, vol. 425, p. 109913, Jan. 2021.

[19] N. E. Michael, S. Hasan, A. Al-Durra *et al.*, "Short-term solar irradiance forecasting based on a novel Bayesian optimized deep long short-term memory neural network," *Applied Energy*, vol. 324, p. 119727, Oct. 2022.

[20] K. R. Mestav, J. Luengo-Rozas, and L. Tong, "Bayesian state estimation for unobservable distribution systems via deep learning," *IEEE Transactions on Power Systems*, vol. 34, no. 6, pp. 4910-4920, Nov. 2019.

[21] S. Dahale and B. Natarajan, "Bayesian framework for multi-timescale state estimation in low-observable distribution systems," *IEEE Transactions on Power Systems*, vol. 37, no. 6, pp. 4340-4351, Nov. 2022.

[22] Y. Wang, Y. Sun, and V. Dinavahi, "Robust forecasting-aided state estimation for power system against uncertainties," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 691-702, Jan. 2020.

[23] W. Ma, J. Qiu, X. Liu *et al.*, "Unscented Kalman filter with generalized correntropy loss for robust power system forecasting-aided state estimation," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 11, pp. 6091-6100, Nov. 2019.

[24] L. Dang, B. Chen, S. Wang *et al.*, "Robust power system state estimation with minimum error entropy unscented Kalman filter," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 8797-8808, Nov. 2020.

[25] A. Abur and A. G. Exposito, *Power System State Estimation: Theory and Implementation*. New York: CRC Press, 2004.

[26] F. C. Schweppe and J. Wildes, "Power system static-state estimation, part I: exact model," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, no. 1, pp. 120-125, Jan. 1970.
[27] F. C. Schweppe and D. B. Rom, "Power system static-state estimation, part II: approximate model," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, no. 1, pp. 125-130, Jan. 1970.
[28] F. C. Schweppe, "Power system static-state estimation, part III: implementation," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, no. 1, pp. 130-135, Jan. 1970.
[29] T. H. Cormen, C. E. Leiserson, R. L. Rivest *et al.*, *Introduction to Algorithms*. Cambridge: MIT Press, 2022.
[30] AssessingSolar. (2025, Aug.). Solar resource assessment in Python. [Online]. Available: https://assessingsolar. org/notebooks/solar_power_modeling.html
[31] W. F. Holmgren, C. W. Hansen, and M. A. Mikofski, "Pvlib Python: a Python package for modeling solar energy systems," *Journal of Open Source Software*, vol. 3, no. 29, p. 884, Sept. 2018.
[32] F. Vignola and A. Andreas. (2013, May). University of Oregon: GPS-based precipitable water vapor (Data). [Online]. Available: http://dx. doi.org/10.7799/1183467
[33] University of Washington. (2025, Jul.). Power systems test case archive. [Online]. Available: https://labs.ece.uw.edu/pstca/
[34] W. Li, Y. Wang, and T. Chen, "Investigation on the Thevenin equivalent parameters for online estimation of maximum power transfer limits," *IET Generation, Transmission & Distribution*, vol. 4, no. 10, pp. 1180-1187, Oct. 2010.
[35] L. Thurner, A. Scheidler, F. Schäfer *et al.*, "Pandapower—an open-source Python tool for convenient modeling, analysis, and optimization of electric power systems," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6510-6521, Nov. 2018.
[36] J. Zhao. (2024, Jun.). MATLAB code of robust GM-estimator for power system state estimation using projection statistics. [Online]. Available: https://www. mathworks. com/matlabcentral/fileexchange/60838-matlab-code-of-robust-gm-estimator-for-power-system-state-estimation-using-projection-statistics
[37] R. Singh. (2009, Jan.). State estimation in power distribution network operation. [Online]. Available: http://hdl.handle.net/10044/1/5468
[38] IEEE. (1990, Mar.). ISO 9060: 1990(en) solar energy – specification and classification of instruments for measuring hemispherical solar and direct solar radiation. [Online]. Available: https://www.iso.org/obp/ui/es/#iso:std:iso:9060:ed-1:v1:en
[39] B. Lim, S. Ö. Arık, N. Loeff *et al.*, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748-1764, Oct. 2021.
[40] J. Herzen, F. Lässig, S. G. Piazzetta *et al.*, "Darts: user-friendly modern machine learning for time series," *Journal of Machine Learning Research*, vol. 23, no. 124, pp. 1-6, Jan. 2022.
[41] P. Ineichen and R. Perez, "A new airmass independent formulation for the Linke turbidity coefficient," *Solar Energy*, vol. 73, no. 3, pp. 151-157, Sept. 2002.
[42] T. M. Klucher, "Evaluation of models to predict insolation on tilted surfaces," *Solar Energy*, vol. 23, no. 2, pp. 111-114, Mar. 1979.
[43] D. L. King, W. E. Boyson, and J. A. Kratochvill. (2004, Feb.). Photovoltaic array performance model. [Online]. Available: https://web. archive. org/web/20170813030848id_/http://mauisolarsoftware. com/MSESC/xPerfModel2003.pdf
[44] J. A. Duffie and W. A. Beckman, *Solar Engineering of Thermal Processes*, 4th ed. Hoboken: Wiley, 2013.
[45] R. R. Labbe Jr. (2014). FilterPy: Kalman filtering and optimal estimation library in Python. [Online]. Available: https://github.com/rlabbe/filterpy
[46] E. S. Gardner Jr, "Exponential smoothing: the state of the art," *Journal of Forecasting*, vol. 4, no. 1, pp. 1-28, Jan. 1985.
[47] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Courier Corporation, 2007.
[48] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition*, Orlando, USA, Apr. 1997, pp. 182-193.
[49] R. V. D. Merwe, *Sigma-point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. Portland: Oregon Health & Science University, 2004.

Hanshan Qing received the B.S. degree from Xi'an Jiaotong-Liverpool University,Xi'an, China and the M.S. degree from The University of Manchester, in Suzhou, China and Manchester, U.K., respectively. Currently, he is a Ph.D. student at the University of Southampton, Southampton, United Kingdom. His research interest focuses on power system state estimation for grids with high penetration of distributed energy resources.

Wangyuan Ding received the B.S. degree from the University of Liverpool, in Liverpool, U.K., and the M.S. degree from University College London and The London School of Economics and Political Science, in London, U. K.. Currently, he is a Ph.D. student at the Informatics Institute of the University of Amsterdam. His research interest focuses on natural language processing

Abhinav Kumar Singh received the Bachelor of Technology degree from the Indian Institute of Technology Delhi, India, and the Ph.D. degree from Imperial College London, London, U.K., in 2010 and 2015, respectively, all in electrical engineering. Since 2019, he has been a Lecturer at the University of Southampton, Southampton, U.K. He received the IEEE PES Working Group Recognition Awards in 2016, 2022, and 2023 for his contributions to multiple IEEE Task Forces. He has served as an Associate Editor for IEEE Transactions on Power Systems and Journal of Modern Power Systems and Clean Energy. His research interests include dynamic modelling, estimation, and control of modern power systems.

Efstratios Batzelis received the Ph.D. degree from the National Technical University of Athens, Athens, Greece in 2016. He has been a Lecturer with the University of Southampton, Southampton, U.K., since April 2021, and a Research Fellow of the Royal Academy of Engineering. He is currently the Director of the Centre for Research in All-electric Future Technologies (CRAFT). From 2017 to 2019, he was an EU Marie-Curie Individual Fellow of photovoltaic control and integration with Imperial College London, London, U.K.. His research interests include renewable energy technology and distributed energy resource, with a focus on solar photovoltaics, power electronics control, and power system stability.