# Graph-Learning-Assisted State Estimation Using Sparse Heterogeneous Measurements

Han Yue*, Wentao Zhang†, Ugur Yilmaz‡, Tuna Yildiz‡, Heqing Huang†, Hongfu Liu*, Yuzhang Lin†, Ali Abur ‡
* Brandeis University, U.S.A.,
† New York University, U.S.A.,
‡ Northeastern University-Boston, U.S.A.

***Abstract*—Unlike transmission systems, distribution systems historically lack enough measurements, making their real-time monitoring almost impossible. Recent deployment of diverse types of devices such as phasor measurement units (PMUs), smart meters, solar inverters and weather information sensors opens up new ways of monitoring these systems, with the assistance of customized machine learning (ML) applications. The paper describes a grid-model-informed machine learning (ML) tool which integrates heterogeneous data streams and creates synchronous measurement snapshots to be used by a hybrid robust state estimator (SE) which provides not only accurate state estimates but also real-time feedback for ML model refinement. Improved monitoring performance due to the use of developed computational framework is experimentally observed by simulated scenarios on an electric utility's distribution system.**

***Index Terms*—Distribution systems, graph learning, machine learning, robust state estimation, system monitoring.**

## I. Introduction

With the proliferation of distributed energy resources (DERs), the operating conditions of distribution systems are becoming more uncertain and volatile, and the dispatch, control, and protection applications require timely and accurate monitoring. This is accomplished via state estimation (SE) which utilizes measurements acquired from various points in the system and determines the best estimate of the system state. In the absence of sufficient measurements, state estimation cannot be carried out due to the lack of network observability. A major challenge in power distribution system monitoring is the diversity of the reporting rates of measurements from phasor measurement units (PMUs), supervisory control and data acquisition (SCADA) systems, and advanced metering infrastructure (AMI). In particular, sensors with slow reporting rates, such as smart meters, cannot keep up with the execution rate of state estimation, leaving gaps of observability between the arrival of two measurement samples. A comprehensive review of the recent work recognizing and attempting to address this challenge can be found in [1]. A data-driven technique is employed to predict the consumption patterns of customers without smart meters to enhance the observability of distribution systems [2]. The authors of [3] propose using historical low-voltage side smart meters to forecast load and DERs injections via the support vector machine with optimally tuned parameters. In [4], an optimal measurement placement is proposed to obtain the accurate pseudo measurement even with limited knowledge of the profile of the injected power. A popular category of approaches is the use of machine learning (ML) techniques to predict measurements with slow reporting rates to enhance their time resolution and meet state estimation requirements [5]. There are two main limitations of the existing work. First, most of these methods completely disregard the power network models or simply learn the mapping with the underlying assumption of fixed network topology. Hence, they fail to produce fully satisfactory results under network model changes which are frequent in feeder operation. Second, most of them do not receive proper supervision from power system domain knowledge. As a result, the predicted measurements by the ML models are not compliant with the power flow model, reducing their accuracy and reliability.

Another major challenge in distribution system monitoring is the timely detection and identification of events such as faults, line switching, and substantial DER/load switching. In recent years, several interesting alternative solutions have been proposed, mainly focusing on detecting those events [6], [7], [8], [9]. These methods are well-established, with detailed examples illustrating their application. Many rely on graph search or brute force to identify power grid events, inherently complicating their implementation or limiting their scalability. Therefore, in this work a sparse estimation approach is based on the Least Absolute Shrinkage and Selection Operator (LASSO) will be employed. Event detection (ED) and SE are two closely related problems in power system monitoring. For example, refs. [10], [11] establish a generalized SE framework based on the minimum-information-loss (MIL) theory, unifying analog and digital quantities in the estimation of system states and switch statuses and proving that the weighted-least-square (WLS) and weighted-least absolute-value (WLAV) estimators are special cases under the MIL principle. In distribution systems, a major challenge in event detection is the sparsity of time-synchronized measurements. To this end, a sparse estimation-based approach is developed

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energy Technologies Office Award Number DE-EE0009356. Submitted to the 23rd Power Systems Computation Conference (PSCC 2024).

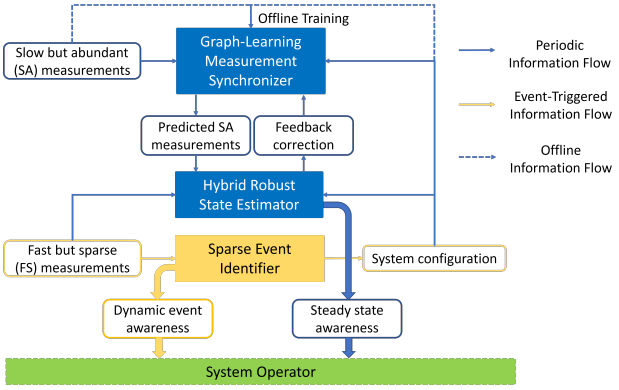



Fig. 1: Proposed distribution system monitoring framework.

in this paper to detect such events using a few synchronized measurements.

This paper describes a comprehensive paradigm for distribution systems SE and ED using sparse and heterogeneous measurements. The main objective is to make the states and events in distribution systems fully observable. This will be accomplished by developing a grid-model-informed machine learning (ML) tool that uses several different types of measurement data to create a consistent and reliable set of synchronous measurement snapshots for a robust SE. The differentiating features of the proposed framework include the use of a graph neural network to capture topology information, and the feedback of the robust SE to enhance the ML model's prediction accuracy and compliance with the physical power flow model. The proposed work also features a highly efficient implementation of the SE in large systems, as well as the detection and location of events based on sparse PMU measurements. The proposed work is validated using data and measurements from a utility Microgrid.

## II. Objective and Problem Statement

The developed SE and ED paradigm can be best described with the block diagram given in Fig. 1. Measurements in distribution systems are categorized into two sets: real-time measurements with fast scan rates and little time skews, yet being insufficient to make the system observable, constituting the *fast and sparse*" (**FS**) set; and others received at much slower scan rates with significant time skews, yet being widely populated, constituting the *slow but abundant* (**SA**) set. Typical examples of FS measurements are PMUs and SCADA measurements. Typical examples of SA measurements are AMI measurements. This assumes that the limited number of FS measurements is insufficient to deliver an observable system. There are also measurements provided by smart meters, but received at a slower rate that cannot satisfy SE requirements. Such measurements are designated as "slow but abundant" (SA) measurements. The objective is to provide comprehensive situational awareness (both steady-state operation and during dynamic events) to grid operators by exploiting the above-described set of diverse data and measurements. *Situational awareness* of distribution systems has two aspects: *steady state awareness* achieved by SE, and *dynamic event awareness* achieved by ED.

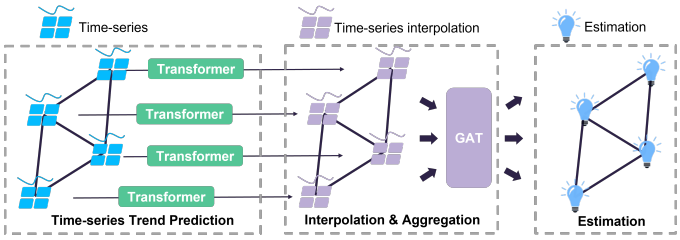



Fig. 2: Graph-learning-based measurement synchronizer.

In order to achieve these objectives, three modules shown in Fig. 1 are developed: *graph-learning measurement synchronizer*, *hybrid robust state estimator*, and *sparse event identifier*. The blue elements in the figure represent periodic information flows providing *steady state awareness*, and the yellow elements in the figure represent event-triggered information flows providing *dynamic event awareness*. These modules will be described in greater detail below.

## III. Development of the Machine Learning Based Measurement Synchronizer

The *graph-learning measurement synchronizer* aims to predict the values of the SA measurements when they are not refreshed by SE and needs to be executed, such that they are synchronized with the FS measurements forming complete snapshots. Suppose the SA measurements are refreshed with time interval $T_S$ (e.g., 30 minutes), and the SE is executed with time interval $T_F$ (e.g., 1 minute). Figure 2 shows the framework of the NG-Transformer model. First, the Transformer [12] takes time series of SA measurements including active and reactive power with time interval $T_S$ as input, and forecasts active and reactive power $T_S$ ahead. Then, Cubic Spline Interpolation is adopted to obtain preliminary estimation at time interval $T_F$. Finally, the Graph Attention Network (GAT) [13], [14] takes FS measurements, including bus voltages and line active and reactive power flows, as input to refine the preliminary estimation at time interval $T_F$ and obtain the final prediction result of SA measurements. Details of each part are as follows.

The **Transformer** [12] module is to handle the prediction time-series SA measurement data, i.e., the nodal active and reactive power injections measured by smart meters at time interval $T_S$. Time-series data can be denoted by $X_i^{(t)} = \{\mathbf{x}_i^{(t-T+1)}, \mathbf{x}_i^{(2)}, ..., \mathbf{x}_i^{(t)}\}$, where $\mathbf{x}_i^{(t)}$ denotes the features of node $i$ at the $t$-th timestamp, and $T$ denotes the time steps of Transformer. The Transformer consists of Positional Encoding and several Transformer Encoders.

Positional Encoding is designed for the model to make use of the order of the sequence without involving recurrence and convolution. To inject some information about the relative or absolute position of the time-series data, we adopt a fixed positional encoding method [15], which is sine and cosine functions of different frequencies as follows:

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d}), \quad (1)$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d}), \tag{2}$$

where $pos$ is the position, and $i$ is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid. The wavelengths form a geometric progression from $2\pi$ to $10000 \cdot 2\pi$. Since for any fixed offset $k$, $PE_{pos+k}$ can be represented as a linear function of $PE_{pos}$, it would allow the model to easily learn to attend by relative positions. Then the positional encoding is added to the input data as follows:

$$H = X + PE, \text{ and } \hat{H} = \hat{X} + PE, \tag{3}$$

where $H$ and $\hat{H}$ are embeddings for $X$ and $\hat{X}$, respectively. While this positional encoding method is fixed, there are no learnable parameters in this part. We do not add an additional learned embedding layer to convert input data of the first and second periods to the same dimension because the dimensions of the two pieces of data not only have the same length but also have the same meaning.

Each Transformer Encoder has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise, fully connected feed-forward network. There is a residual connection [16] around each of the two sub-layers, followed by layer normalization [17]. It first defines a scaled dot-product attention function, which is shown as:

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{Q \cdot K^\top}{\sqrt{d_k}})V, \tag{4}$$

where $Q$, $K$, and $V$ are embeddings generated by $H$ and $\theta_A$, denoting queries, keys, and values, respectively. Here $d_k$ is the dimension of $K$. Based on Eq. (4), multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions, which can be formulated as:

$$E = \text{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_\text{h})\theta_O, \tag{5}$$

$$\text{head}_\text{i} = \text{Attention}(H \cdot \theta_{Q_i}, H \cdot \theta_{K_i}, H \cdot \theta_{V_i}), i \in [1, h], \tag{6}$$

where $h$ denotes the number of heads, and $\theta_O$, $\theta_Q = \{\theta_{Q_1}, ..., \theta_{Q_h}\}$, $\theta_K = \{\theta_{K_1}, ..., \theta_{K_h}\}$, and $\theta_V = \{\theta_{V_1}, ..., \theta_{V_h}\}$ are learnable parameters. With a layer normalization function $\text{LayerNorm}(\cdot)$, the output of the first sub-layer in Transformer Encoder can be written by:

$$\tilde{E} = \text{LayerNorm}(H + E). \tag{7}$$

Then for the second fully connected layer together with another normalization layer, the Transformer Encoder generates a hidden representation $Z$ by the following equation:

$$Z = \text{LayerNorm}(\tilde{E} + \tilde{E} \cdot \theta_F), \tag{8}$$

where $\theta_F$ denotes the learnable parameters in the fully connected layer. Lastly, a fully connected layer for prediction is built as follows:

$$P = Z \cdot \theta_P, \tag{9}$$

where $\theta_P$ is the learnable parameters in the prediction layer.

We adopt mean squared error (MSE) in our objective function for Transformer, which is as follows:

$$\mathcal{L}_T = \sqrt{\frac{1}{n}\sum_i^n (P_i - Y_i)^2}, \tag{10}$$

where $Y_i$ denotes the ground truth values of node $i$.

**Cubic Spline Interpolation** is a form of interpolation where the interpolant is a special type of piecewise cubic polynomial called a cubic spline. That is, instead of fitting a single, high-degree polynomial to all of the values at once, cubic spline interpolation fits low-degree cubic polynomials to small subsets of the values. Specifically, we assume that the points $(x_i, y_i)$ and $(x_{i+1}, y_{i+1})$ are joined by a cubic polynomial $S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i$ that is valid for $x_i < x < x_{i+1}$ for given $i$. To find the interpolating function, we must first determine the coefficients $a_i, b_i, c_i, d_i$ for each of the cubic functions. For $n$ points, there are $n-1$ cubic functions to find, and each cubic function requires four coefficients. In our setting, we use $n = 4$ points to get SA measurement estimation at time interval $T_F$ from the SA measurement prediction at time interval $T_S$, where the first 3 points are from historical data and the last point is from the prediction of Transformer.

**Graph Attention Network** (GAT) [13] is an attention-based architecture for graph-structured data. It is used to refine the SA measurement estimation at time interval $T_F$ based on the FS measurements and the topological information of the distribution grid. The grid is represented as a graph, where nodal active and reactive power injections are taken as nodal features, and line active and reactive power flows are taken as edge features. GAT computes the hidden representations of each node in a graph by attending over its neighbors, following a self-attention strategy. In the attention mechanism, normalized coefficients are calculated by the following equation:

$$\alpha_{i,j} = \frac{\exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{\Theta}\mathbf{h}_i \,\|\, \mathbf{\Theta}\mathbf{h}_j]\right)\right)}{\sum_{k\in\mathcal{N}(i)\cup\{i\}} \exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{\Theta}\mathbf{h}_i \,\|\, \mathbf{\Theta}\mathbf{h}_k]\right)\right)}, \tag{11}$$

where $\alpha_{i,j}$ denotes the normalized coefficient of edge $(i, j)$ for node $i$, $\mathcal{N}(i)$ is the set of all neighbor nodes of node $i$, $\mathbf{h}_i$ denotes features of node $i$, $\mathbf{\Theta}$ is a learnable matrix applied to every node, $\sigma$ is an activation function using LeakyReLU, and $a$ is a shared single-layer feedforward neural network as the attentional mechanism. Here $\|$ is the concatenation operation. In order to handle edge features as well, we follow the idea of Edge-Featured Graph Attention Network (EGAT) [14], which considers edge features in the attention mechanism as follows:

$$\alpha_{i,j} = \frac{\exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{\Theta}\mathbf{h}_i \,\|\, \mathbf{\Theta}\mathbf{h}_j \,\|\, \mathbf{\Theta}_e\mathbf{e}_{i,j}]\right)\right)}{\sum_{k\in\mathcal{N}(i)\cup\{i\}} \exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{\Theta}\mathbf{h}_i \,\|\, \mathbf{\Theta}\mathbf{h}_k \,\|\, \mathbf{\Theta}_e\mathbf{e}_{i,k}]\right)\right)}, \tag{12}$$

where $\mathbf{\Theta}_e$ is a learnable matrix applied to every edge, and $\mathbf{e}_{i,j}$ denotes features of edge $(i, j)$. With the attention mechanism, the hidden representation of an aggregation can be written as:

$$\mathbf{h}_i = \sigma\Big(\sum_{j\in\mathcal{N}(i)} \alpha_{i,j}\mathbf{\Theta}\mathbf{h}_j\Big), \tag{13}$$

where $\mathbf{h}_i$ is initialized as input features of node $i$.

We adopt mean squared error (MSE) in our objective function for GAT, which is as follows:

$$\mathcal{L}_G = \sqrt{\frac{1}{n}\sum_{i}^{n}(h_i - g_i)^2}, \tag{14}$$

where $g_i$ denotes the gap between the ground truth and the interpolation value of node $i$.

The parameters of the deep learning model are optimized by the Adam optimizer, and the hyperparameters are chosen based on a validation set.

## IV. Robust Scalable State Estimator

Having a robust and scalable state estimator is crucial for estimating the system states in the presence of biased measurements within an acceptable computational time. A Weighted Least Absolute Values (WLAV) based estimator is developed for this purpose. It uses the virtual synchronous measurement snapshot provided by the ML-based measurement synchronizer as well as any available real-time measurements. WLAV SE has the implicit capability of selecting a minimum observable set of measurements that yield the minimum weighted sum of the absolute differences between estimated and "virtually" measured values. It automatically rejects gross errors and also ignores lower accuracy measurements. This is particularly important for the integration of the virtual measurements provided by the ML model, as the prediction based on spatio-temporal correlation is not always reliable and compliant with the physical power flow model. However, the WLAV SE is generally more computationally expensive than the conventional WLS SE, hence as efficient solution must be developed for large-size distribution systems. In this paper, we propose a Massively Parallel Distributed (MPD) SE framework which makes the computation time almost independent of system size.

Implementation of the MPD SE framework involves dividing the network into several zones. This enables the solution of each zone to be executed independently in parallel. However, this partitioning procedure inadvertently excludes those measurements incident to boundary buses. Consequently, measurement redundancy drops at the zone boundaries, which could potentially impact the robustness of the estimator. It is noted that while a boundary bus may be connected to one or more buses from other zones, an internal bus is only connected to buses in the same zone. Hence, those measurements incident to internal buses are never removed by network partitioning. This observation is exploited and multiple copies of the system are generated, ensuring that each bus serves as an internal bus in at least one copy of the system. Different partitioning algorithms are employed depending on the system structure to optimize the partitioning of the power system network for the MPD algorithm. Once the original copy of the system is partitioned into $n_0$ zones, an appropriate algorithm is used to generate the required system copies. The reasoning for using two different partitioning algorithms and details of the algorithms can be found in [18].

Since there are multiple estimates of the states in different copies, the resultant estimate is obtained by taking the mean value of **only** the estimates of **internal buses** in each copy as shown in (15). This yields a robust estimate of the system state, free of gross errors.

$$\hat{x_i} = (1/n^*_{zones}) \cdot \sum_{c=1}^{n^*_{zones}} x^c_{i,internal}, \tag{15}$$

where $x^c_{i,internal}$ is the estimation of the state of an internal bus in copy c, and $n^*_{zones}$ is the number of zones that $x_i$ is the state of an internal bus.

## V. Detection and Location of Outages

Timely detection of events such as line outages is a necessity for forming the correct network model of a distribution system, which is essential for the usability of the graph-learning-based measurement synchronizer and the robust state estimator. Thus, this paper presents a methodology for timely line outage detection utilizing a limited number of PMUs.

To make a model for the line outage detection, the DC power flow model is used since it provides a reasonably accurate linear approximation between real power injections and voltage phase angles [19]. The following linear formulation can be used to relate the changes in bus phase angles ($\Delta\theta$) to the real power injections ($\Delta P$).

$$B * \Delta\theta = \Delta P, \tag{16}$$

where

- $B = \begin{cases} if k \neq j \rightarrow B_{kj} = -\frac{1}{x_{kj}} \\ if k = j \rightarrow B_{kk} = \sum_{j=1, j\neq k}^{N} \frac{1}{x_{kj}} \\ 0 \quad otherwise \end{cases}$
- $x_{kj}$ is the branch reactance value between bus k and bus j,
- $\Delta\theta$ is the angle difference of buses (post outage theta – pre outage theta),
- $\Delta P$ is the injections at corresponding buses connected to disconnected line.

When one of the lines goes out of service, $B$ matrix and the bus angles ($\Delta\theta$) will change accordingly. However, rather than modifying the matrix $B$, a line outage can be represented by virtual power injections ($\Delta P_s$ and $\Delta P_t$) at the terminal buses of the disconnected branch. These injections have identical magnitudes but opposite signs, and their absolute value is equal to the real power flow of the line. This makes the net flow effectively zero while ensuring the terminal bus phase angles align with the post-outage solution [20].

Then, in order to solve (16) a well-known sparse solution method, namely LASSO (Least Absolute Shrinkage and Selection Operator) is used [21]. Considering the DC model given in (16), matrix $B^{-1}$ is typically badly conditioned, i.e. it is nearly singular. Therefore, before applying LASSO, the following QR decomposition strategy is applied to (16). The justification for employing the QR decomposition is comprehensively discussed in [22].

$$\begin{bmatrix} Q_{ee} & Q_{ei} \\ Q_{ie} & Q_{ii} \end{bmatrix} * \begin{bmatrix} R_{ee} & R_{ei} \\ 0 & R_{ii} \end{bmatrix} * \begin{bmatrix} \Delta\theta_{ee} \\ \Delta\theta ii \end{bmatrix} = \begin{bmatrix} \Delta P_{ee} \\ \Delta P_{ii} \end{bmatrix}, \quad (17)$$

$$R_{ii} * \Delta\theta_{ii} = [Q_{ie}^T Q_{ii}^T] * \begin{bmatrix} \Delta P_{ee} \\ \Delta P_{ii} \end{bmatrix}, \quad (18)$$

- subscript $i$ refers to buses with PMUs and,
- subscript $e$ refers to buses without PMUs.

In addition to that, to further reduce the number of non-zeros found by LASSO solution connectivity matrix $A$ is introduced to (18). This will yield the following:

$$R_{ii} * \Delta\theta_{ii} = \begin{bmatrix} Q^T{}_{ie} & Q^T{}_{ii} \end{bmatrix} * A^T * \begin{bmatrix} \Delta\widetilde{P}_{ee} \\ \Delta\widetilde{P}_{ii} \end{bmatrix}, \quad (19)$$

where

- $A$ is the connectivity matrix $b \times n$.
- $n$ is the number of buses and,
- $b$ is the number of branches in the system.

Then well-known LASSO approach is performed as:

$$\Delta\tilde{P} := \min_{\Delta\widetilde{P}} \frac{1}{2} ||\Delta\theta' - M\Delta\widetilde{P}||_2^2 + \lambda ||\Delta\widetilde{P}||_1, \quad (20)$$

where

- $\Delta P = A^T \Delta\widetilde{P}$,
- $M = \begin{bmatrix} Q^T{}_{ei} & Q^T{}_{ii} \end{bmatrix} A^T$,
- $\Delta\theta' = R_{ii}\Delta\theta_{ii}$.

Upon calculation of $\Delta\widetilde{P}$, the location of the line outage can be ascertained by identifying the position of the maximum value within $\Delta\widetilde{P}$ vector. Then, based on the detected line outages, the network model of the distribution grid can be updated and fed to the graph-learning-based measurement synchronizer and the robust state estimator.

## VI. Closed loop Estimation and Training Framework

The graph-learning measurement synchronizer and the robust state estimator function in a closed loop aid each other. The way the graph-learning measurement synchronizer assists the robust state estimator is by making the system observable and enhancing measurement redundancy. However, these measurements are predicted based on the spatio-temporal correlations and there is no guarantee that they will exactly follow the physical laws described by the power flow model. Furthermore, it does not have the capability to suppress the impact of bad data. The robust state estimator can provide the required compliance check and help to fine-tune the graph-learning measurement synchronizer. Once the SE converges, the residuals of the predicted SA measurements will be fed back to the graph-learning measurement synchronizer, which will be re-trained by adding the residuals into the loss function to be minimized. As such, the prediction errors of the SA measurements will be used to back-propagate the parameters in the deep layers.

TABLE I: MAEs of SA measurement prediction results by the proposed and baseline methods.

| Model | Active Power | Reactive Power |
|---|---|---|
| Transformer | 24.6426 | 5.9463 |
| NG-Transformer | 1.0875 | 1.4501 |

## VII. Simulation Results

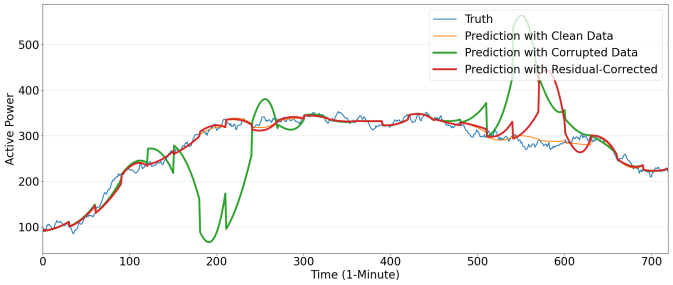
The proposed distribution system monitoring paradigm is validated on a real-world utility Microgrid. 3-month AMI data with 30-minute time resolution recorded from the field is used for validating the methods. In order to test the adaptiveness of the ML model to topology changes, power flows under 100 different topologies with randomly selected switch statuses are generated for training, validation, and testing purposes. It is assumed that $T_S$ = 30 minutes, and $T_F$ = 1 minute.

Scalability of the estimator is illustrated using two very large-scale networks (VLSN) with meshed and strictly radial topologies containing 12589 buses. The strictly radial network with the same number of buses is synthetically generated by removing the loops in the meshed network. Hence, meshed VLSN has 17529 branches, there are only 12588 branches in the strictly radial VLSN. In this section, the unit used for active power is denoted in kilowatts (kW), and the unit used for reactive power is denoted in kilovolt-amperes reactive (kVAR).

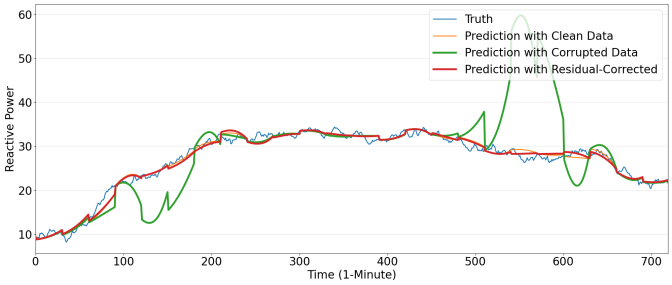
### A. Validation of Graph-Learning Measurement Synchronizer under Topology Changes

One of the main features of the proposed graph-learning method is the adaptiveness to system topology changes, as it can take topological information as input. In order to validate this feature, we compare the SA measurement prediction error of the proposed NG-Transformer model and that of a standard Transformer model, a state-of-the-art deep learning model that cannot take graph input. In this experiment, it is assumed that the system is already observable by FS measurements. The performance under an unobservable system will be presented in Section VII-C. The accuracy of the prediction results is evaluated by mean absolute error (MAE).

Assuming all 1-minute interval line active and reactive power injection data are given, Table I shows the prediction results of the proposed model and the baseline model under changing topologies. From the results of Table I, it is clear that topology information has a significant effect on model prediction accuracy. The NG-Transformer model achieves much higher prediction accuracy than the standard Transformer model. The reason is that the standard Transformer model is unaware of the topology changes and develops the same mapping between FS measurements and SA measurements, which actually changes with the topology. The NG-Transformer model, on the other hand, embeds the topology information with the graph convolution.




(a) Active power prediction for Node 93




(b) Reactive power prediction for Node 93

Fig. 3: Prediction results using different kinds of data for Node 93.

TABLE II: The performance of MPD SE in Meshed VLSN

| Number of Zones in the Original Copy | 1 | 4 | 8 | 16 | 32 | 48 |
|---|---|---|---|---|---|---|
| Computational Time (seconds) | 194.92 | 31.20 | 12.11 | 2.59 | 1.26 | 0.70 |
| Number of Required Cores | 1 | 8 | 20 | 36 | 75 | 104 |

TABLE III: The performance of MPD SE in Radial VLSN

| Number of Zones in the Original Copy | 1 | 4 | 8 | 16 | 32 |
|---|---|---|---|---|---|
| Computational Time (seconds) | 17.70 | 4.45 | 2.27 | 1.08 | 0.61 |
| Number of Required Cores | 1 | 5 | 11 | 21 | 42 |

### *B. Validation of Physical Computational Performance of MPD Robust Estimator*

As described in Section IV, to make the computational time of the estimator almost independent of the system size, a parallel computational framework is presented.

Since the size of the tested micro grid is relatively small in the order of hundreds of nodes, to demonstrate the scalability of the MPD method, the testing is performed on two VLSN with meshed and strictly radial topologies. Tests are conducted in MATLAB environment using the Massachusetts Green High Performance Computing Center facilities which houses a 128 Core, 2.4 GHz Intel CPUs, with 8 GB RAM.

In Table II, cpu times for different partitioning schemes are given for the meshed VLSN. The cpu time is reduced below 1 second when the original copy is partitioned into 48 zones which requires a total of 104 CPU cores. The additional 56 zones are created to prevent loss of any measurement redundancy as explained in methodology. Similarly, the cpu time of the SE in the radial VLSN is reduced to less than 1 second when the original copy is partitioned into 32 zones as shown in Table III. The number of additional system copies are significantly less than those of meshed VLSN since the number of boundary buses are quite small in radial systems.

### *C. Validation of Closed Loop Operation of Measurement Predictor and State Estimator*

In order to validate the effectiveness of ML retraining based on the SE's feedback, we down-sampled the node active and reactive power injection data to 30-minute intervals as the SA measurement input of the model from AMI and designed three scenarios for comparison: clean data, corrupted data, and residual-corrected data. In the clean data scenario, the measurements do not contain any gross errors. In the corrupted data scenario, the measurements contain gross errors with random magnitudes and durations, and the robust SE does not provide any feedback to the ML model. In the residual-corrected data scenario, the measurements contain the same gross errors as the corrupted data scenario, but the ML model is retrained based on the measurement residuals provided by the robust WLAV SE, which forms the proposed closed-loop operation framework.

Table IV shows the prediction results with clean, corrupted, and residual-corrected data. For model training, we use the same NG-Transformer model with different SA and FS measurement data but the same topology information. The topology information varies 10 times throughout the whole period. For the test, we use different kinds of data as input, but use ground truth from clean data for all of the trained models.

From the results of Table IV, we can see that the corruption of data has a significant influence on model training. Compared with the clean data scenario, the prediction performance is much degraded when data is corrupted. When the ML model is retrained based on the measurement residuals provided by the WLAV SE, the performance improves a lot compared to corrupted data. In fact, the performance gets very close to that of the clean data scenario, implying that the impact of the data corruption is largely suppressed. Figure 3 is an example of the prediction results of a node, from which it is clearly observed that the deviation of prediction due to the presence of gross errors in certain time sections is largely eliminated. The prediction results of the residual-corrected data scenario are very similar to those of the clean data scenario. This validates the concept of mutual assistance between graph-learning-based measurement synchronizer and robust state estimator.

TABLE IV: FS measurement prediction results with different data over the first group of bus voltage control and distribution system topology information in terms of MAE.

| Data | Active Power | Reactive Power |
|---|---|---|
| Clean | 11.9233 | 5.7904 |
| Corrupted | 29.1641 | 9.7516 |
| Residual-Corrected | 13.9248 | 6.2632 |

TABLE V: The results of the line outage detection algorithm.

| Disconnected Branch | Detected Line Outage Branch / $\Delta\widetilde{P}$ | |
|---|---|---|
| **42-49** / -4.4153 | 42-49 | -4.5193 |
| **59-61** / -0.6758 | 58-61 | -0.7177 |
| **82-83** / -4.6718 | 82-83 | -4.8802 |
| **64-61** / 1.2346 | 64-61 | 1.1854 |
| **8-5** / 34.3989 | 8-5 | 36.9010 |
| **94-100** / 0.1305 | 94-100 | 0.1084 |
| **94-95** / 1.4166 | 94-95 | 1.1456 |
| **89-92** / 13.5493 | 89-92 | 10.4212 |
| **101-102** / -1.3665 | 101-102 | -1.3464 |

### *D. Validation of Line Outage Detection Method*

The proposed line outage detection method is tested in IEEE 118 Bus system to validate its effectiveness [23]. To test the approach, first limited number of PMU devices are placed at buses 2, 13, 22, 39, 49, 53, 58, 63, 81, 84, 103, 105, 106, 114. Additionally, each of these PMUs is presumed to possess a channel for measuring one current phasor along an incident branch. This capability to measure current phasors facilitates the computation of voltage phasors at the distant end of the corresponding branches. Consequently, utilizing the available voltage and current phasors, the phase angles of buses 2, 12, 13, 15, 22, 23, 39, 40, 53, 54, 58, 56, 63, 59, 81, 80, 84, 85, 103, 110, 105, 108, 106, 107, 114, 115 are determined for the IEEE-118 bus system.

Once the locations of PMUs are determined, 9 different sample outage scenarios are created and tested to demonstrate the utilization of the proposed line outage detection method. The results given in Table V show successful identification of line outages even when they do not occur in the vicinity of PMUs.

## VIII. Conclusion

In this paper, a new framework for monitoring distribution systems is described. It features a graph-learning model for synchronizing fast and slow measurements, a robust and scalable state estimator based on the physical power flow model, and the closed-loop operation paradigm between the two for further enhancing their performance. The paper also introduces an ED method that can pinpoint line outages in distribution systems based on sparse PMU measurements. Test results from a real-world system demonstrate that the proposed framework can adapt to topology changes, integrate information from spatio-temporal patterns and power flow model derived from physical laws, and achieve both steady state awareness and dynamic event awareness for distribution systems with diverse measurements.

## References


[1] G. Cheng, Y. Lin, A. Abur, A. Gómez-Expósito, and W. Wu, "A survey of power system state estimation using multiple data sources: Pmus, scada, ami, and beyond," *IEEE Transactions on Smart Grid*, pp. 1–1, 2023.

[2] Y. Yuan, K. Dehghanpour, F. Bu, and Z. Wang, "A multi-timescale data-driven approach to enhance distribution system observability," *IEEE Transactions on Power Systems*, vol. 34, no. 4, pp. 3168–3177, 2019.

[3] J. Zhao, C. Huang, L. Mili, Y. Zhang, and L. Min, "Robust medium-voltage distribution system state estimation using multi-source data," in *2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, 2020.

[4] J. Liu, F. Ponci, A. Monti, C. Muscas, P. A. Pegoraro, and S. Sulis, "Optimal meter placement for robust measurement systems in active distribution grids," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 5, pp. 1096–1105, 2014.

[5] 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, 2019.

[6] T. Banerjee, Y. C. Chen, A. D. Dominguez-Garcia, and V. V. Veeravalli, "Power system line outage detection and identification a quickest change detection approach," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3450–3454, IEEE, 2014.

[7] S. Azizi, M. R. Jegarluei, A. M. Kettner, and A. S. Dobakhshari, "Wide-area line outage monitoring by sparse phasor measurements," *IEEE Transactions on Power Systems*, pp. 1–11, 2022 (Early Access).

[8] Y. Liao, Y. Weng, C.-W. Tan, and R. Rajagopal, "Quick line outage identification in urban distribution grids via smart meters," *CSEE Journal of Power and Energy Systems*, vol. 8, no. 4, pp. 1074–1086, 2022.

[9] J. E. Tate and T. J. Overbye, "Line outage detection using phasor angle measurements," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1644–1652, 2008.

[10] H. Sun, F. Gao, K. Strunz, and B. Zhang, "Analog-digital power system state estimation based on information theory—part i: Theory," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1640–1646, 2013.

[11] H. Sun, F. Gao, K. Strunz, B. Zhang, and Q. Li, "Analog-digital power system state estimation based on information theory—part ii: Implementation and application," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1647–1655, 2013.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[13] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, *et al.*, "Graph attention networks," *Stat*, vol. 1050, no. 20, pp. 10–48550, 2017.

[14] Z. Wang, J. Chen, and H. Chen, "Egat: Edge-featured graph attention network," in *International Conference on Artificial Neural Networks, 2021*, pp. 253–264, Springer, 2021.

[15] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *International Conference on Machine Learning*, 2017.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[17] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[18] U. C. Yilmaz and A. Abur, "A robust parallel distributed state estimation for large scale distribution systems," *IEEE Transactions on Power Systems*, pp. 1–9, 2023.

[19] K. Purchala, L. Meeus, D. Van Dommelen, and R. Belmans, "Usefulness of DC power flow for active power flow analysis," in *IEEE Power Engineering Society General Meeting, 2005*, pp. 454–459 Vol. 1, 2005.

[20] R. Emami and A. Abur, "External system line outage identification using phasor measurement units," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1035–1040, 2013.

[21] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[22] T. Yildiz and A. Abur, "Improved line outage detection in transmission systems with few PMUs," in *2022 North American Power Symposium (NAPS)*, pp. 1–5, 2022.
[23] A. R. Al-Roomi, "Power flow test systems repository," MathWorks. [Online]. Available: https://al-roomi.org/power-flow/118-bus-system.