State Estimation for Power Distribution System using Graph Neural Networks

Quang-Ha Ngo
Clarkson University
hanq@clarkson.edu

Bang L. H. Nguyen
Los Alamos National Lab
bangnguyen@ieee.org

Tuyen V. Vu
Clarkson University
tvu@clarkson.edu

Tuan Ngo
Entrust Solutions Group
ngo.tuan.1985@gmail.com

2023 IEEE Electric Ship Technologies Symposium (ESTS) | 978-1-6654-6515-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/ESTS56571.2023.10220523

***Abstract*—State estimation is critical to maintaining system stability and reliability as it enables real-time monitoring of the power system operation and facilitates fault detection, minimizing the risk of power outages and improving overall system performance. This paper presents a state estimation method based on graph neural networks, aiming to improve time efficiency and extended observability. Graph neural networks can aggregate information and dependencies from voltage and power measurement at the critical buses, making them more effective for state estimation on non-grid structured data. The IEEE 123-bus system is used as a case study to evaluate comprehensively the state estimation performance. The proposed model provides a better performance for mapping measurement data with states compared to other neural networks.**

***Index Terms*—State estimation, cyber-physical system, graph neural network, deep learning, spatio-temporal.**

I. Introduction

As a highly complex cyber-physical system, a smart grid is composed of power electronic devices, a power infrastructure and a communication network. Phasor Measurement Units (PMUs) initially collect physical data before sending it to Supervisory Control and Data Acquisition Systems (SCADA). The measurement data is then sent by the communication network to the application level, where the power applications are processed and assessed [1]. The main objective is to determine the state of the system based on the measurement data, ensuring the trustworthiness of the cyber-physical pipeline.

Power system state estimation (PSSE) is highly critical in this pipeline, particularly for energy operation and central control since its functionality is directly used to process measurements and extract accurately the bus voltage magnitudes and angles of power systems [2]. Thus, the reliability of the measurement data plays an important role in ensuring the proper management of smart grids.

Data from micro phasor measuring units (PMU) is analyzed by using many state estimate schemes proposed in literature [3]–[11]. In general, these schemes can be categorized into two groups: model-based methods and data-driven methods. Model-based techniques are developed from physical equations which are relationships between power system measurements and states. Conventionally, the state-space model equations of power networks can be studied to provide better awareness for the static state estimation (SSE) or dynamic state estimation (DSE) algorithms [12]. DSE can track state values well, even in the load-change cases and identify bad data using a weighted least square or chi-square method. However, DSE is only reliant on precise dynamic system models, which may not always be observable in some situations [13]. Therefore, with the rapid advancement of state estimation techniques, large research groups are shifting to data-driven approaches.

Data-driven methods have recently emerged as an appealing research direction because they do not rely on physical equations to perform state estimation [2]. The primary trend in these methods involves leveraging artificial neural networks to describe either part of or the entire power system. This is achieved through training vast amounts of data collected from different sources. These ANN-based techniques capture functional patterns from power systems and encode them as node parameters. This method enhances the robustness of state estimation against inaccurate measurements by leveraging either a neural network or a parallel distributed processing model. In addition to ANNs, kernel-weighted neighbors and kernel function methods have also been utilized for grid current-state estimation [2].

The accuracy of state estimation relies on both the type of model and the reliability of measurements used in the estimation process. Many models that have been investigated in several papers include multi-layer perception (MLP) [8]–[10], Bayesian deep learning for static states estimation [14], [15], recurrent neural networks (RNN) with gated-based or long short term memory (LSTM) for state estimation and forecasting [16], convolutional neural networks (CNN) [17], generative adversarial networks and auto-encoders [18]. In fact, a dynamic power system, which includes nodes and edges, has a structure as same as a graph but most mentioned networks may not work well with power system information with specific connections among nodes. In other words, the ANNs are limited in their ability to capture the complex relationships and dependencies that exist in graphs [19]. Thus, it is necessary to explore a new model that be superior to ANNs in handling graph-structured data.

This paper proposes a state estimation scheme that utilizes load-change cases and voltage measurements. These data can be recorded from various sources such as μPMU (micro Phasor Measurement Unit), smart meters, and advanced metering devices. The proposed approach aims to enhance time efficiency and improve robustness and provide a comprehensive analysis of the results when we compare the capability of three networks including Multi-layer Perceptron (MLP), Convolutional

Neural Network (CNN), and Graph Convolutional Network (GCN) for state estimation.

The contributions of this paper are twofold: the proposed scheme can track well changes of state values including voltage magnitude and phase angle in load-change cases; comprehensive studies and comparisons among different neural networks such as Multi-layer Perception, Convolutional Neural Network, and Graph Convolutional Network are also provided.

Particularly, the proposed network could incorporate traditional model-based methods for state observability improvement [20]. However, this procedure is not explored in this paper. The remaining sections are structured as follows. Section II introduces Graph Neural Network and other models considering spatial-temporal data. Section III describes the IEEE 123-bus distribution system, data collection, pre-processing, and a series of case studies to demonstrate the effectiveness of the proposed scheme. The results are discussed in Section IV and Section V concludes the paper.

II. Neural Network Structure

In this section, we describe the overall architecture and information flow of neural network-based estimators. Also discussed are the loss functions and back-propagation.

A. Multiple Layer Perceptron

Multiple layer perceptron is an artificial neural network that consists of multiple layers of interconnected nodes. Its input layer receives the values of the input features and then multiplies them with the weights of hidden layers. Similarly, the outputs of hidden layers are transferred to the final layer for regression. Every unit in one layer is connected to every unit in the next layer, which makes a fully connected network. The determination of the number of hidden layers relies on the dimension and complexity of the dataset. A back-propagation learning technique is used to update the weights. In this manner, the MLP model can establish a mapping between the input and the output data by approximating an activation function [10]. This can be effectively achieved after training the model with the entire dataset.

B. Convolutional Neural Network

Convolutional Neural Network (CNN), a kind of deep artificial neural network, has been widely applied to deal with high-dimensional data based on their shared-weights architecture. The architecture of CNN usually consists of convolutional layers, pooling layers, and fully-connected layers. The first layer is typically a convolutional layer that applies a filter to obtain local information. This is followed by a pooling layer that reduces the spatial size of the feature map. The output from these layers is then passed through fully connected layers for classification or regression [21]. Regarding the power system state estimation, CNN is used to analyze the complex and high-dimensional data from PMUs, and extract useful features that can be used for state estimation.

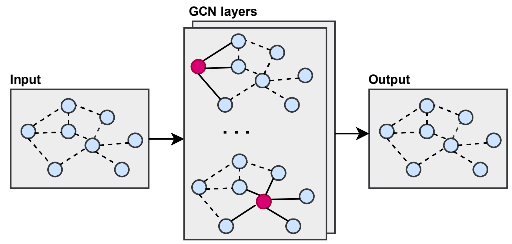


Fig. 1. A diagram of the proposed state estimation scheme

C. Graph Convolutional Network

The GCN structure includes convolutional layers for feature extraction as shown in Fig. 1. These layers are utilized to extract the spatial-temporal information among bus voltages in the power system. By employing a global average pooling, the features are aggregated from the nodes, and trained to predict state values [22]. The mathematical formulation of operational layers is simplified as follows.

$$h^i_{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}h^i_{(l)}W_{(l)}) \tag{1}$$

Where $\tilde{A} = A + I_N$ denotes the adjacent matrix, $\tilde{D}$ is the degree matrix from the matrix $\tilde{A}$, I_N is the identity matrix. $W_{(l)}$ represents the weights of layer l, and σ is a non-linear activation function. The formulation involves in stacking several layers, utilizing a localized first-order approximation of spectral graph convolutions.

III. GCN-based State Estimation Implementation

Electric power distribution systems are generally organized in a graph type. It is intuitive and logical to represent these systems as graphs. However, previous investigations neglect spatial characteristics of power systems such as network connectivity and globality due to their subdivision into multiple partitions or grids. Despite the application of 2-D convolutions on grids, the ability to capture spatial information is still limited due to compromises in data modeling. Therefore, a graph neural network is directly utilized on data structured as graphs, facilitating the extraction of significantly meaningful patterns in the spatial domain.

A. Investigated Power System

The proposed network methods are performed on the IEEE 123-bus feeder system. The system consists of overhead and underground lines, unbalanced loads, four voltage regulators, and shunt capacitor banks. The nominal voltage of the system is 4.16 kV. The regulator parameters and loads are configured following ones of the IEEE 123 Node XENDEE Test Cases.

TABLE I
123-BUS FEEDER DATASET

Parameter	Buses	Count
Measure	1, 7, 8, 13, 18, 21, 23, 25, 28, 29, 30, 35, 40, 42, 44, 47, 49, 50, 51, 52, 53, 54, 55, 57, 60, 67, 72, 78, 81, 89, 91, 93, 97, 101, 105, 108, 197	37
State	62, 63, 64, 65, 76, 77, 80, 82, 83, 86, 87, 98, 99, 100,	14
Total load change cases: 4000 samples | **Train**: 3200 | **Test**: 800		

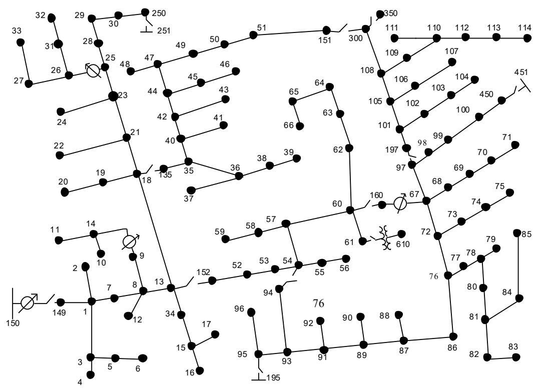


Fig. 2. The IEEE 123-bus feeder system

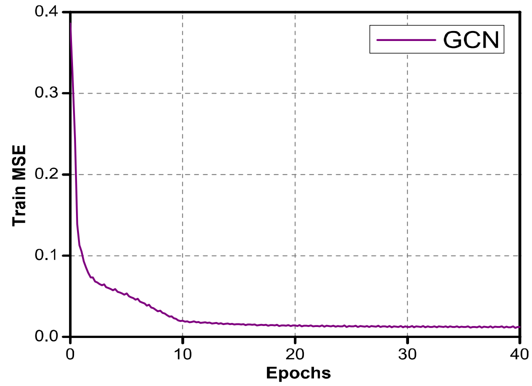


Fig. 3. Training MSE curve versus the number of training epochs

The measurements are recorded in three-phase buses distributed throughout the system as shown in Fig. 2. The bus voltages are sampled at a frequency of 1 kHz using instrument transformers at the corresponding measurement devices. Opal-RT, a real-time power system analysis tool, is utilized to simulate the microgrid system. The load-change datasets under various scenarios are produced to train and test the proposed state estimation scheme.

The graph datasets are constructed from 51 three-phase buses of the IEEE 123-bus system. To calculate load values for three-phase buses, the loads of single-phase buses are taken into account. The information of 37 three-phase buses and 14 three-phase buses are considered as measurements and states, respectively. This decision aims to test the system's observability first. Future research should investigate the best choice. Besides, we concentrate on the three-phase measured voltage as inputs to examine the effectiveness of the state estimation in the beginning. The extra single-phase inclusion in the unbalanced distribution system will also be conducted in future work.

B. Data Collecting Procedure

A Python script with RT-LAB API is written and run with RT-LAB to automatically collect the operational data of the IEEE 123 node feeder and store them in a specific folder. The collected data from Opal RT are instant voltage and current values of the main nodes in .mat format [23]. Moreover, other Python scripts are written to process the raw data in. mat format to the ready-to-use data in .pt format with Graph data class for convenient usage and storage.

C. Dataset Preparation

The dataset for a graph neural network is defined as an organized collection of a graph, node features, and label vectors [24], [25]. $D = \{(G^1, x^1, y^1), (G^2, x^2, y^2), ...(G^n, x^n, y^n)\}$, where the vertex sets is unchanged $V^i = V, \forall i \in \{1, ...n\}$, i is the graph data index. The node feature matrices $X^i \in \mathbb{R}^{N \times F \times T}$ contain three dimensions: the number of nodes $|V| = N$ that contain measurement information, the number of node features F, and the time duration T. The state vector of the graph network within the time duration T is $y^i \in \mathbb{R}^{L \times P \times T}$ with L as the number of state information-containing nodes and P as the number of state signals. The node feature matrix contains measured voltages of the three-phase buses. At the buses without measuring voltages, the missing node features are inserted by zeros. The voltage is collected from the Opal-RT-Power Sytem simulator in an interval of 500 milliseconds with a load-change case in between. Table I summarizes these parameters for data collection. All of the 4000 data are captured and processed under random load scenarios. Subsequently, the train set is collected from 3200 graphs and the rest 800 graphs are gathered as the test set. In addition, data inputs are normalized by the MinMaxScaler function of the Sklearn library.

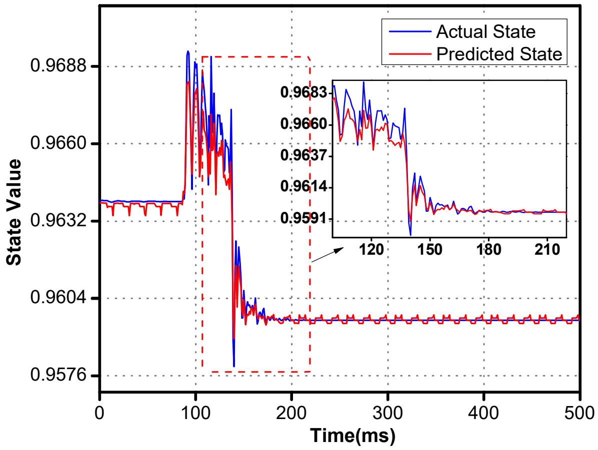
D. Response to State Changes

In real-time power system operation, the states are always changing due to the variation of load and generation. The state estimation methods are expected to capture accurately the relatively quick dynamics of state changes [9]. In this case, a load-change scenario is designed to test the performance of state estimation. The active and reactive power values are changed randomly from 70% to 130% of default loads at the 0.1^{th} second. Therefore, the actual states during the load change fluctuate transiently.

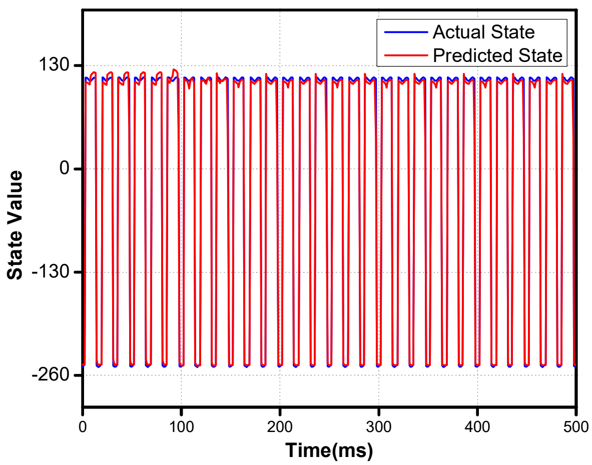
E. Configuration of Training Model

The graph datasets are trained with the Stochastic Gradient Descent (SGD) optimizer under the mean square error for estimation. To address the over-fitting problem, the dropout layers are included in hidden layers [26]. The batch size serves as a crucial parameter that significantly impacts both the execution time and model accuracy. The large batch size leads to a precise estimation of the gradient but the time consumption increases remarkably [27]. For graph datasets, the batch size should be chosen appropriately because each graph already encompasses the number of all nodes. Notably, hyper parameters are determined relatively since the training process is executed on a computer equipped with an Intel Core i7-8700 processor, 32 GB RAM, and an NVIDIA GTX 1080 GPU. The deep learning framework utilized for the study is PyTorch, specifically leveraging the PyTorch Geometric library, which is tailored for GNN model tasks [28].

To measure and compare the performance of methods, Mean Absolute Errors (MAE), Mean Squared Errors (MSE) are adopted. the significance of learning rate is crucial in achieving

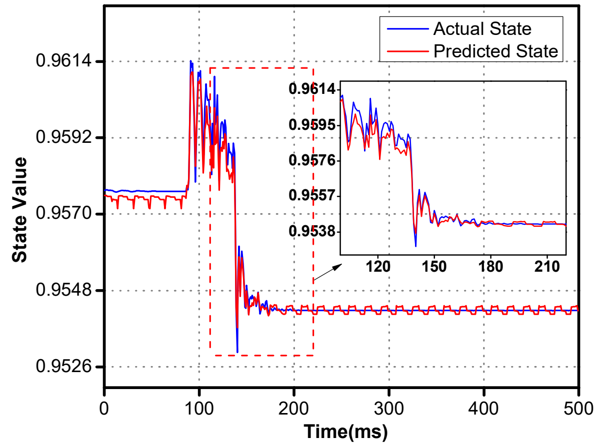


(a) Estimation of voltage magnitude

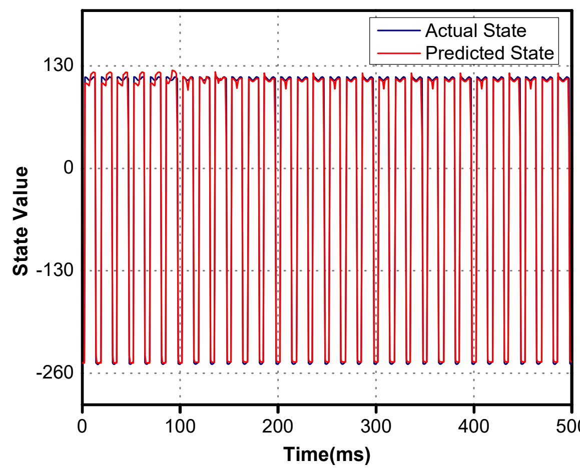


(b) Estimation of phase angle

Fig. 4. Estimation of bus 62 in topology change case.



(a) Estimation of voltage magnitude



(b) Estimation of phase angle

Fig. 5. Estimation of bus 77 in topology load-change case.

an accurate result for regression. During the training process, the learning rate initially began at 0.05 for the first 10 epochs and was subsequently adjusted to 0.01. Fig. 3 indicates the training process with a learning rate transition from 0.05 to 0.01 after 40 epochs. Mean Squared Error is saturated around 0.0122 when keeping a learning rate of 0.01.

IV. RESULTS AND ANALYSIS

The state estimation results obtained using Graph Convolutional Network are compared with other networks, including Multilayer Perceptron and Convolutional Neural Network. The GCN approach generally achieved better prediction results than traditional machine learning models. The details of these neural network structures are outlined in Table II, which indicates the functional layers and the sizes of the corresponding tensors. To ensure dimension compatibility between layers, reshaping and flattening commands are employed.

A. Comparison of Network Structures

Initially, a comparison was conducted among the different neural network structures to evaluate the network capability. The number of trainable parameters of MLP is significantly higher than other network structures. This provides an explanation for the MLP's ability to get the remarkable accuracy shown in Table III.

Mean Square Error and Mean Absolute Error during the 500-millisecond window are summarized in Table III. As a result of its ability to track the dynamics of load changes, the GCN-based estimator may have a smaller estimation error during transients. Additionally, comparing the training time required to achieve approximately 0.025 MSE loss, we observed that GCN, CNN, and MLP models took 347(s), 422(s), and 875(s) respectively, indicating that GCN exhibits greater time efficiency compared to the other models.

TABLE II
COMPARISON OF NEURAL NETWORK STRUCTURES

MLP		CNN		GCN	
Input	[500×612]	Input	[500×612]	Input	[500×51×12]
Dense	[612]	CNN+Pooling	51x[30x3]	GCN	[612×512]
Dense	[512]	CNN+Pooling	30x[30x3]	GCN	[512×256]
Dense	[512]	Dense	[120]	Dense	[256]
Dense	[512]	Dense	[60]	Dense	[256]
Dense	[14]	Dense	[14]	Dense	[14]

Fig. 4 and Fig. 5 are two examples showing the estimation of bus 62 and bus 77 during the load-change case. The predicted voltage and phase angle can be estimated well by GCN from the original steady state along the ramping of the load. Looking at these figures, we can see that the voltage magnitude deviates from the actual state much more than the phase angle does. This is because the magnitude variation is substantially more than the phase variation between samples. Besides, the training process is performed for each voltage magnitude and phase angle separately to boost the accuracy due to the large number of output and nodes in hidden layers.

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT APPROACHES

Model	**Criteria**	
	MAE	MSE
MLP	0.0362/ 0.0385/ 0.0356	0.0161/ 0.0172/ 0.0168
CNN	0.0352/ 0.0369/ 0.0334	0.0173/ 0.0171/ 0.0172
GCN	0.0258/ **0.0252**/ 0.0267	0.0146/ **0.0122**/ 0.0138

B. Discussion

An advantage of GNN compared to other data-driven approaches is its ability to reflect non-Euclidean data in the power network model. However, the pure data-reliant model requires extensive amounts of data to be accurate. This is unrealistic as real-world data, especially with rare abnormal events, are limited. To enhance the precision and robustness of state estimation under the circumstances of imbalanced data or zero-day attacks, future work could be applied to the physical equations of the power system on the graph network.

V. Conclusions

In this paper, we proposed an innovative solution to state estimation in power distribution systems. The datasets for studying load-change cases in the IEEE 123-bus feeder system were generated using the Opal-RT simulation and can be leveraged for future research purposes. The numerical results show that the GCN model outperforms other state-of-the-art methods on the same dataset, demonstrating its great potential for exploiting spatio-temporal patterns from the graph datasets. Additionally, it enables faster training, improved convergence, reduced parameter count, as well as flexibility and scalability by hyper-parameter tuning techniques. In the future, we will modify the deep learning framework to specifically integrate the system's physical information. Moreover, the effect of reduced measurements and noises should also be examined.

Acknowledgment

The information, data, or work presented herein was funded in part by the 1) U.S. Office of Naval Research under award number N000142212239, and 2) National Science Foundation (NSF-AMPS) under award number CON0002619.

References

[1] Osman Boyaci, Amarachi Umunnakwe, Abhijeet Sahu, Mohammad Rasoul Narimani, Muhammad Ismail, Katherine Davis, and Erchin Serpedin. Graph Neural Networks Based Detection of Stealth False Data Injection Attacks in Smart Grids. *IEEE Systems Journal*, 16(2):2946–2957, June 2022. Number: 2 arXiv:2104.02012 [cs, eess].

[2] Tuyen V. Vu, Bang L.H. Nguyen, Zheyuan Cheng, Mo-Yuen Chow, and Bin Zhang. Cyber-Physical Microgrids: Toward Future Resilient Communities. *IEEE Industrial Electronics Magazine*, 14(3):4–17, 2020.

[3] Daniel A. Haughton and Gerald Thomas Heydt. A Linear State Estimation Formulation for Smart Distribution Systems. *IEEE Transactions on Power Systems*, 28(2):1187–1195, May 2013.

[4] Mingming Mao, Junjun Xu, Zaijun Wu, Qinran Hu, and Xiaobo Dou. A Multiarea State Estimation for Distribution Networks Under Mixed Measurement Environment. *IEEE Transactions on Industrial Informatics*, 18(6):3620–3629, June 2022.

[5] Junbo Zhao, Marcos Netto, and Lamine Mili. A Robust Iterated Extended Kalman Filter for Power System Dynamic State Estimation. *IEEE Transactions on Power Systems*, 32(4):3205–3216, July 2017.

[6] Kaveh Dehghanpour, Zhaoyu Wang, Jianhui Wang, Yuxuan Yuan, and Fankun Bu. A Survey on State Estimation Techniques and Challenges in Smart Distribution Systems. *IEEE Transactions on Smart Grid*, 10(2):2312–2322, March 2019.

[7] Kritika Saxena and Abhijit R. Abhyankar. Agent-Based Distributed Computing for Power System State Estimation. *IEEE Transactions on Smart Grid*, 11(6):5193–5202, November 2020.

[8] Ahmed S. Zamzam, Xiao Fu, and Nicholas D. Sidiropoulos. Data-Driven Learning-Based Optimization for Distribution System State Estimation, March 2019. Issue: arXiv:1807.01671 arXiv:1807.01671 [eess].

[9] Guanyu Tian, Yingzhong Gu, Di Shi, Jing Fu, Zhe Yu, and Qun Zhou. Neural-network-based Power System State Estimation with Extended Observability. *Journal of Modern Power Systems and Clean Energy*, 9(5):1043–1053, September 2021.

[10] Behrouz Azimian, Reetam Sen Biswas, Shiva Moshtagh, Anamitra Pal, Lang Tong, and Gautam Dasarathy. State and Topology Estimation for Unobservable Distribution Systems Using Deep Neural Networks. *IEEE Transactions on Instrumentation and Measurement*, 71:1–14, 2022.

[11] Ahmed S. Zamzam and Nicholas D. Sidiropoulos. Physics-Aware Neural Networks for Distribution System State Estimation, July 2019. Issue: arXiv:1903.09669 arXiv:1903.09669 [cs, math].

[12] Bang LH Nguyen, Tuyen V Vu, Joseph M Guerrero, Mischael Steurer, Karl Schoder, and Tuan Ngo. Distributed dynamic state-input estimation for power networks of microgrids and active distribution systems with unknown inputs. *Electric Power Systems Research*, 201:107510, 2021.

[13] Junbo Zhao, Antonio Gómez-Expósito, Marcos Netto, Lamine Mili, Ali Abur, Vladimir Terzija, Innocent Kamwa, Bikash Pal, Abhinav Kumar Singh, Junjian Qi, et al. Power system dynamic state estimation: Motivations, definitions, methodologies, and future work. *IEEE Transactions on Power Systems*, 34(4):3188–3198, 2019.

[14] Kursat R. Mestav, Jaime Luengo R., and L. Tong. Bayesian State Estimation for Unobservable Distribution Systems via Deep Learning. *IEEE Transactions on Power Systems*, 34(6):4910–4920, 2019.

[15] Julio A. D. Massignan, João B. A. London, Michel Bessani, Carlos D. Maciel, Rodrigo Z. Fannucchi, and Vladimiro Miranda. Bayesian Inference Approach for Information Fusion in Distribution System State Estimation. *IEEE Transactions on Smart Grid*, 13(1):526–540, 2022.

[16] Liang Zhang, Gang Wang, and Georgios B. Giannakis. Real-time Power System State Estimation and Forecasting via Deep Neural Networks. *IEEE Transactions on Signal Processing*, 67(15):4069–4077, August 2019. Number: 15 arXiv:1811.06146 [cs, stat].

[17] R. Yarlagadda, V. Kosana, and K. Teeparthi. Power System State Estimation and Forecasting using CNN based Hybrid Deep Learning Models. In *2021 IEEE International Conference on Technology, Research, and Innovation for Betterment of Society (TRIBES)*, pages 1–6, 2021.

[18] Yi He, Songjian Chai, Zhao Xu, Chun Sing Lai, and Xu Xu. Power system state estimation using conditional generative adversarial network. *IET Generation, Transmission & Distribution*, 14, December 2020.

[19] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. 2017.

[20] Jonatan Ostrometzky, Konstantin Berestizshevsky, Andrey Bernstein, and Gil Zussman. Physics-Informed Deep Neural Network Method for Limited Observability State Estimation, February 2020. Issue: arXiv:1910.06401 arXiv:1910.06401 [cs, eess].

[21] Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. In *2010 IEEE International Symposium on Circuits and Systems*, pages 253–256, 2010. ISSN: 2158-1525.

[22] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 2020.

[23] Bang Nguyen, Tuyen Vu, Thai-Thanh Nguyen, Mayank Panwar, and Rob Hovsapian. Spatial-temporal recurrent graph neural networks for fault diagnostics in power distribution systems. *arXiv preprint arXiv:2210.15177*, 2022.

[24] B. Rozemberczki, P. Scherer, Y. He, G. Panagopoulos, A. Riedel, M. Astefanoaei, O. Kiss, F. Beres, Guzmán López, N. Collignon, et al. Pytorch geometric temporal: Spatiotemporal signal processing with neural machine learning models. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 4564–4573, 2021.

[25] Bang LH Nguyen, Tuyen Vu, Thai-Thanh Nguyen, Mayank Panwar, and Rob Hovsapian. 1-d convolutional graph convolutional networks for fault detection in distributed energy systems. *arXiv preprint arXiv:2211.02930*, 2022.

[26] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. Survey of dropout methods for deep neural networks. *arXiv preprint arXiv:1904.13310*, 2019.

[27] Dominic Masters and Carlo Luschi. Revisiting small batch training for deep neural networks. *arXiv preprint arXiv:1804.07612*, 2018.

[28] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.