The Generality-Accuracy Trade-off in Neural State Estimation

Aleksandr Berezin, Eric MSP Veith and Stephan Balduin
R&D Division Energy
OFFIS – Institute for Information Technology
Oldenburg, Germany
e-mail: {aleksandr.berezin | eric.veith | stephan.balduin}@offis.de

Thomas Oberließen and Sebastian Peter
Institute of Energy Systems, Energy Efficiency and Energy Economics
Technical University of Dortmund
Dortmund, Germany
e-mail: {thomas.oberliessen | sebastian.peter}@tu-dortmund.de

***Abstract*—This paper addresses the challenge of neural state estimation in power distribution systems. We identified a research gap in the current state of the art which lies in the inability of models to adapt to changes in the power grid, such as loss of sensors and branch switching, in a zero-shot fashion. We designed benchmarks to evaluate the robustness of models to different changes in grid topology and used them to test models with different architectures. The observed results strongly suggest the existence of a trade-off between accuracy and robustness.**

Keywords-neural state estimation; zero-shot learning; transfer learning; graph neural networks.

I. Introduction

This work extends the results of our conference paper [1], which considered the problem of zero-shot Neural State Estimation (NSE) with a focus on the relationship between model complexity and performance. This extension provides better benchmarks and evaluations, focusing on the apparent trade-off between accuracy and generality of NSE models. It also expands the set of tested models to include non-graph-based architectures and a non-parametric baseline.

We begin by reviewing relevant prior work in Section II, followed by a formal statement of our research question in Section III. Section IV details the methodology, model selection, and experimental setup. The results of the experiments are presented in Section V. Finally, Section VI summarizes our findings and suggests directions for future work.

II. State of the art

Power System State Estimation (PSSE) is the task of inferring the "state" of an electrical power grid from real-time data collected by various sensors distributed throughout the system. The "state" in this context generally refers to the voltage magnitudes and phase angles at each bus in the grid.

For many years, PSSE was mainly performed for transmission grids using simplifying assumptions such as near-DC power flow and computational methods with poor scalability [2]. This is enabled by balanced operation with a relatively simple, predominantly linear topology of transmission grids, given their scale and structure.

This approach cannot be extended to distribution grids that transport electricity from substations to end consumers. Their unbalanced nature, radial or weakly meshed topology, high R/X ratios, and above all, cost inefficiency to achieve sufficient sensor coverage complicate the state estimation process. Initially designed with transmission systems in mind, conventional methods often struggle to provide accurate state estimation in more complex, dynamic, and less predictable distribution systems [2].

However, with the proliferation of Distributed Energy Resources (DERs) and other complex consumers, grid operators are faced with the need to perform PSSE for distribution grids. Furthermore, §14a of the German Energy Industry Act effectively requires operators to develop observability in distribution grids in order to align consumption with production from renewable energy sources, which requires PSSE.

A. Conventional methods

The traditional and most widely used approach for PSSE is the Weighted Least Squares (WLS) method [3]. This algorithm minimizes the sum of the squared differences between the observed and estimated measurements, with each term being weighed inversely proportionally to the square of the measurement error standard deviation.

What limits the direct application of WLS in distribution systems is the minimum number of measurements required for the convergence of WLS. Assuming the grid contains n buses, it is then described by $2n$ variables, namely n voltage magnitude values and n voltage angles. A slack bus serves as the reference; its voltage angle is set to zero or a known constant, and therefore does not need to be estimated. The voltage angles of the other network buses are relative to the voltage angle of the connected slack bus. Therefore, the state estimation must find $2n-k$ variables, where k is the number of defined slack buses. The minimum amount of measurements m_{min} needed for the WLS method to work is therefore:

$m_{min} = 2n - k$

However, in order to perform well, the number of redundant measurements should be higher. A value of $m \approx 4n$ is often considered reasonable for practical purposes. This level of observability is unachievable in distribution grids due to

economic constraints and the sheer number of elements that must be monitored.

Another problem is that the WLS algorithm is computationally intensive. Assuming a dense system matrix, its time complexity is generally considered to be $\mathcal{O}(N^3)$, where N is the number of buses. This is due to the need for matrix inversions and solving linear equations. This complexity becomes a limitation for large-scale distribution grids with thousands of buses, leading to significant computational burden and time constraints, especially when real-time estimates are required. Additionally, WLS assumes that all error distributions are Gaussian, a condition that may not always hold in practice.

B. Feed-forward methods

The most promising path to overcome these limitations and provide observability in distribution grids is currently believed to be NSE: data-driven methods that utilize historical data in addition to real-time measurements. Artificial Neural Networks (ANNs) may be able to perform the calculation faster while being robust to insufficient measurements [4][5]. However, like all Machine Learning (ML) methods, the performance of ANNs is contingent on the quality and quantity of the available training data. Therefore, NSE approaches are usually valid only for the grid they have been trained on. Once the topology or characteristics of nodes change, the ANN needs to be retrained. This is known as the problem of Transfer Learning (TL).

When designing an ANN-based system for solving PSSE, it is tempting to start with Multi-Layer Perceptrons (MLPs). Not only are they the easiest ANNs to implement, but they can approximate any function guaranteed by the universal approximation theorem. In practice, it means that, given sufficient computing power and training data, such models can achieve an arbitrarily high level of accuracy. However, feed-forward models underutilize two important properties of power grids. The first property is that the grid can be represented as a graph. The second is that a power grid, like any physical system, is local, meaning that any interaction between two elements must propagate through the lines and buses between them. Together, these properties enable a drastic reduction in the number of possible interactions that a model needs to consider. The feed-forward models learn about this locality implicitly, inferring the grid topology from the data. However, this is a disadvantage when a topology change occurs, because the model's knowledge of the grid structure and the physical processes (graph signals) are entangled, leading to huge performance degradation when anything in the underlying system changes.

The same is true for other fully-connected feed-forward models that build upon the MLP, such as transformers; hence the title of this subsection.

C. Geometric methods

A sensible way to overcome this limitation is to use models that incorporate information about the graph topology into their calculations. Such models are known under an umbrella term Graph Neural Networks (GNNs), or, less commonly, "geometric models". They are specifically designed to separate the graph structure from the graph signals and only model the latter. This means that they also require the topology of the power grid as input. This can be a barrier if that topology is not known, as in this case a separate topology estimation system is required. However, this design drastically reduces the number of trainable parameters and allows a model trained on one graph to perform inference on another with little to no adjustments [1]. In the context of power systems, there are situations where the grid topology changes due to alterations in switch states or maintenance of elements, up to and including islanding, when parts of the grid become isolated from the greater system. Some of these events can occur suddenly and the control system must adapt to them in real time, which necessitates an ability to generalize provided by GNNs. However, the downside is that GNNs cannot generally approximate an arbitrary function, which in practice limits their maximum accuracy.

In this study, we will be using a subset of GNNs known as Message Passing Networks (MPNs). The most common examples of MPNs are Graph Convolutional Networks (GCNs), Graph Attention Networks (GATs), and Graph Isomorphism Networks (GINs). All of these models are built on the graph message passing operation, which has the locality property with respect to the graph geometry.

Expectedly, recent years have seen a high volume of publications that propose utilizing GNNs for NSE in various ways.

A recent study on GNNs for state estimation [6] came out of a collaboration between TenneT and Radboud University. It demonstrates a GCN-based topology control system to mitigate grid congestion. The model, trained on historical fault data, dynamically reconfigures the grid topology by opening or closing circuit breakers in response to overload warnings.

A similar project combining state estimation with active control is described in [7]. The researchers aimed to develop data analytics services for predicting localized grid congestion caused by excessive distributed renewable generation and eventually prevent it by issuing bids for purchasing energy flexibility on the market. To achieve this, they used a GNN model for both state estimation (to detect congestion) and for generating control signals (in the form of bids). The data used are live voltages and energy profiles from prosumers with PV systems, as well as the known grid topology. They note that GNNs are far more efficient than other tested models and are more capable of adapting to grid changes, while being slightly less accurate.

However, to our knowledge, none of these research projects specifically studied the problem of Zero-Shot Learning (ZSL) in PSSE. The contribution of this work is in setting up multiple evaluation scenarios for ZSL in NSE and using them to evaluate the performance of GNNs against other models.

D. Theoretical limitations of GNNs

While our paper investigates the capabilities and limitations of GNNs experimentally, several notable papers take a more rigorous route of building a theoretical understanding of them.

The capability that is critical to our use case is generalization across different topologies, which is analyzed by two papers.

The first study to successfully incorporate non-trivial graph similarities, architectural choices, and loss functions into generalization analysis is [8]. They analytically derive the bounds of generalization for GNNs as a function of these factors. The theoretical results are then verified on real datasets. The conclusion relevant to this study is that the generalization ability of GNNs leverages the correlation between graph structure and node labels.

A later study [9] pushes the theoretical analysis further by incorporating model complexity into the calculation of generalization bounds and shows that there is a trade-off between the generalization capability of a model and its complexity. However, increasing complexity does not necessarily degrade generalization if it aligns with the task at hand. Quantifying this alignment is another major contribution of the study, as it gives new tools for choosing better GNN configurations for a given problem.

However, model complexity also affects performance of GNNs in scenarios without topology changes. This brings us to another important capability of GNNs: scaling, i.e., the possibility of increasing the accuracy of the model by adding more layers, as is usually done in feed-forward models.

Unfortunately, this mode of scaling in GNNs is limited by the oversmoothing phenomenon [10]. It is a consequence of GNN layers acting as low-pass filters, which effectively averages the output values over multiple iterations. Eventually, the model converges to an output where the values at all nodes of the graph are identical. Therefore, GNN models have a finite optimal depth that can differ between graphs and model architectures and, therefore, is usually found empirically.

A deep analysis of the oversmoothing phenomenon is presented in [11]. They find that the onset of oversmoothing is related to the graph diameter, which is usually small for real-life graphs. After the number of layers surpasses the diameter, for each node, there will be no nodes that have not been encountered before in message passing and hence the node representations will tend to homogenize. In contrast to most other graph datasets, power grids are characterized by long linear branches and therefore large diameters, which should make oversmoothing less likely to occur. However, this structure presents challenges of its own for GNNs.

III. Research question

When discussing the ability of a model to generalize to different grid topologies, it is important to differentiate between *homogeneous* and *heterogeneous* modes of TL. In general, the homogeneous TL mode means that the source and target data are in the same feature space, while in the heterogeneous TL mode they are represented in different feature spaces.

In the context of power grids, this is the difference between two use cases. In the homogeneous case, the power grid remains the same, but some connections between its nodes appear or disappear due to changes in switch states or elements going in and out of service. In the heterogeneous case, the model trained on one grid is used to make predictions about a completely different grid [12].

This distinction becomes very important in real-life deployments. Integrating a new model into the control system of a real grid naturally takes time, and training the model on that specific grid could be incorporated into this process without noticeably slowing it down. On the other hand, changes in grid topology due to switching can happen suddenly and unpredictably, and the model must adapt to them in real time.

There is also another way in which the data distribution can shift in the context of PSSE: the observable subset of buses can change, which changes the amount of input data points available to the model. This can also be considered a form of homogeneous TL.

A subset of TL is Zero-Shot Learning (ZSL). This scenario excludes the possibility of fine-tuning the model on the new distribution and evaluates its performance directly after the transfer. In this project, we specifically focus on ZSL because it is more representative of real-life situations where a model must make predictions immediately after a topology change without access to any training data for fine-tuning. In other words, the model should be *robust* to distributional shifts.

Of course, in practice, a model can be fine-tuned to provide the best performance for the new topology. Still, until this process is complete, the previous version of the model has to substitute for it and provide sufficiently good estimates, even if they are of lower quality.

The research question for this paper is what existing models in application to the PSSE problem are robust to changes in the data distribution, specifically:

A To the reduction of the subset of observable buses;
B To grid topology changes resulting from changing switch states;
C To transfer to a completely different power grid.

IV. Methodology

Before we proceed to describe the models we investigate, please note that our model implementations may not be ideal and therefore may not provide the most accurate results in absolute terms. This study should not be taken as an attempt to rank different models and determine the best performing ones, but rather to observe the impact of graph topology changes on the models' performance. This metric should theoretically be more robust to imperfections in model implementations, since it reflects the more fundamental structural properties of the models in question.

A. GNN models

We are comparing four GNN models using the implementations provided by the PyTorch Geometric framework [13]:

1) Graph Convolutional Network (GCN) as proposed in [14]
2) Graph Attention Network (GAT) as proposed in [15]
3) Graph Isomorphism Network (GIN) as proposed in [16]
4) Graph Sample and Aggregate (GraphSAGE) as proposed in [17]

Each model is tested with a variable number of layers ranging from 1 to 10 as a hyperparameter. This is needed to empirically determine the optimal depth of a GNN where it is sufficiently expressive but not yet affected by oversmoothing. Later in Section V the models will be labeled by a concatenation of the architecture name with the number of layers, i.e., "GAT3" is a GAT model with 3 layers.

The models are trained to predict two features: the real and imaginary parts of the complex voltage for every node. The number of features in the hidden layers is the same. We use the Huber loss function [18], a dropout probability of 0.5 and the GraphNorm normalization method from [19]. The optimizer is Adam with a learning rate of 0.001.

B. MLP models

Although the most recent research on NSE focuses primarily on GNNs, the classic MLP architecture is also considered for this role, and experiments with them provide a valuable perspective.

One of the most cited models of this type is Physics-Aware Neural Network (PAWNN), proposed in [20]. The idea of it is to use the classic perceptron as a building block but prune its synapses according to the graph adjacency matrix, i.e., the grid topology. These perceptrons are stacked in a variable number of layers, equal to the maximum diameter of a vertex-cut partition of the original graph.

There also exists an improved derivation of this model, proposed in [21]. The improvement is based on the observation that designing the ANN architecture based on the adjacency matrix, as in the original PAWNN, may lead to unnecessary connections between layers. The Pruned Physics-Aware Neural Network (P2N2) cuts out those unnecessary connections and uses separate weight matrices for the individual parts of the ANN, depending on the grid topology.

Another approach is the Prox-Linear Network (PLN) model proposed in [22], which is based on a prox-linear solver for state estimation using the Least Absolute Value (LAV) method. The main idea is to split the nonlinear state estimation problem into several blocks that are proximally linear. The PLN is built by unfolding these blocks. In practice, this structure reduces to a MLP.

For this project, we reproduced the P2N2 and PLN models from their descriptions in the corresponding papers. This means that the implementation may not be entirely faithful to what the authors intended, but this is an unavoidable limitation caused by the lack of reference implementations.

C. Baseline

Choosing a baseline method for NSE is made difficult by the absence of a single commonly accepted method that works under the condition of partial observability (which excludes WLS). The solution we chose is the non-parametric feature propagation algorithm from [23], which interpolates missing node-level features by solving a heat equation with known features as boundary conditions. This results in a smooth interpolation of features between known nodes. Of course, this algorithm is not designed for PSSE and is not expected to perform well, but it gives a deterministic solution that is easy to grasp intuitively, which makes it a suitable baseline.

D. Graph representations of power systems

A successful application of GNN models naturally depends on how well the underlying data can be represented in the graph format. The most obvious representation, and the one used in this paper, is known as the bus-branch model. In it, buses are represented as nodes of the graph, while lines and transformers (branches) are its edges, with branch admittances as edge weights. Voltages normalized to local reference values are the node features.

Admittances are chosen as edge weights because the graph Laplacian operator assumes higher edge weights to mean a higher correlation between nodes. This operator is, in turn, used in both the GNN models and the feature propagation algorithm. It should be noted that the models in question support neither complex-valued weights nor multidimensional weights, so we have to use the magnitude of the true complex impedance.

However, using admittance instead of impedance as edge weights becomes a problem for representing closed switches, which have zero impedance and, therefore, infinite admittance. This problem is solved by fusing buses connected by closed bus-to-bus switches into one bus. This is complicated because multiple closed switches are often connected to the same bus, so a naive approach of fusing adjacent buses in random order does not work. Instead, we use an iterative algorithm inspired by [24]. Firstly, we build an auxiliary graph of just the closed bus-to-bus switches with buses as nodes and switches as edges. In this graph, nodes with a degree of one can be safely removed (fused with their adjacent buses). This, in turn, will lower the degree of the adjacent node. Eventually, every node will reach a degree of one and can be fused until every connected component of the auxiliary graph is fused into a single node.

However, it should be noted that the bus-branch model is not the only power grid representation that exists in the literature. For example, in [25], the authors used a more granular representation: they model each grid element as a separate node, with the addition of extremity nodes for connecting elements such as lines or transformers. The main limitation of this approach and the reason we choose not to use it is that it requires training data to include voltage values not only for the buses but also for all grid elements and their extremities, which is rarely available in real datasets; therefore, it is confined to simulations in practice.

An even more interesting factor-graph-like representation was developed in [26]. In general, a factor graph is a bipartite graph consisting of factor and variable nodes, where factor nodes represent measurement types (e.g., bus voltage and branch current), while variable nodes capture state variables (voltages). This structure allows for easy inclusion/exclusion of different measurement types and sidesteps problems with

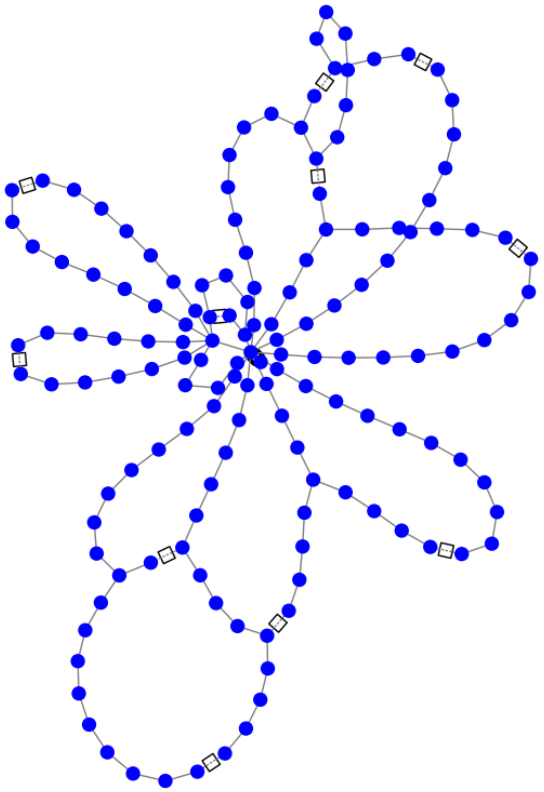

Figure 1. A visualization of the SimBench 1-MV-urban–1-sw grid.

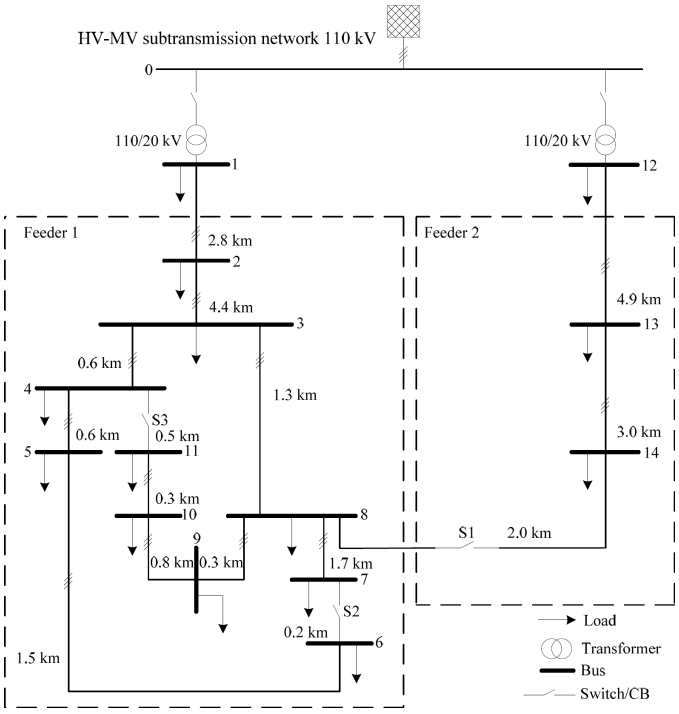

Figure 2. A visualization of the CIGRE medium voltage distribution network.

the initialization of missing features. However, it is outside the scope of our study.

E. Datasets

The main dataset used in this project is the SimBench *1-MV-urban–1-sw*, a 147-node, 10 kV medium voltage grid [27] depicted in Figure 1. It is composed of a grid model and a per-bus complex (active and reactive) power yearly time series. To calculate the resulting grid state, we performed a power flow calculation using the SIMONA energy system simulation software [28]. The resulting dataset comprises the base data and a year of complex voltage time series with a 15-minute temporal resolution. This dataset is hereafter called PQ.

Most grid branches in this model are of the open loop type, which means an open switch (depicted as a square in Figure 1) connects two separate branches. To simulate a realistic topology change, we made a line in one of the open loop branches inoperable, resembling a line fault, and closed the loop switch to resupply all nodes. Performing this operation on different branches resulted in multiple variations of the base grid topology. Afterward, we reran the simulation for each variation to obtain a topology change dataset, which is hereafter referenced as TC.

Unfortunately, the base dataset did not contain information about measurement devices. Therefore, we had to choose observable nodes randomly based on an observability level of 50%, which we assume is realistic for distribution grids. This means that the state estimator has access to true voltage values for half of the grid buses.

An auxiliary dataset used in the heterogeneous ZSL experiments is based on the CIGRE medium voltage distribution network from [29], pictured in Figure 2. It is a much smaller grid with only 15 nodes, which allows us to study how the complexity of the grids affects the performance of ZSL. The voltage data for it are generated using the Midas simulation framework [30]. The shorthand name for this dataset is MV.

V. Experiments

Our experiments are composed of three benchmarks that we call use cases. They correspond to the three subquestions of the main Research question.

Our main evaluation criterion is Median absolute deviation (MAD) across all snapshots in a dataset, which is used to estimate the average performance of models. For the ZSL experiments, we also define another metric called Performance Drop (Degradation) Ratio (PDR) as

$$\frac{MAD_{\text{train}} - MAD_{\text{test}}}{MAD_{\text{train}}}.$$

It normalizes the generalization gap, making it comparable between models or datasets. Lower PDR indicates strong generalization; higher values indicate poor generalization.

A. Static performance

Before experimenting with ZSL, we first evaluate the models without it. Here, the PQ dataset is split equally into training and testing subsets. After training on the first subset, we calculate the MAD on the second one and the PDR between

TABLE I. BEST MODEL CONFIGURATIONS FOR PQ.

Model	MAD	PDR
P2N2	0.03	-0.00
PLN	0.15	0.01
Baseline	0.36	-0.01
GIN1	0.47	-0.01
GraphSAGE1	0.53	-0.01
GraphSAGE2	0.62	-0.01
GIN2	0.63	0.00
GIN5	0.63	0.00
GraphSAGE3	0.65	-0.00
GIN6	0.67	-0.00

TABLE II. BEST MODEL CONFIGURATIONS FOR MV.

Model	MAD	PDR
P2N2	0.03	0.01
PLN	0.07	-0.00
GIN1	0.31	0.00
GraphSAGE1	0.31	0.00
GCN2	0.34	0.01
GCN3	0.35	0.00
GraphSAGE2	0.35	0.00
GAT2	0.37	0.00
GAT4	0.37	0.01
GraphSAGE10	0.37	0.00

them. The resulting values are presented in Table I. The same evaluation for the MV dataset can be found in Table II.

The near-zero PDR values for all models indicate that they can generalize to unseen data with the same topology. We can also observe that the MLP models are starting with a huge advantage, being an order of magnitude more accurate compared to GNN models, which all fall below the baseline. We acknowledge that this is not necessarily representative, as many other papers discussed in Section II are able to achieve much better performance by using different graph representations and other methods. As already mentioned, the goal of this study is not to replicate the state of the art, but rather to examine the performance changes in ZSL scenarios, which brings us to the next experiments.

As for the baseline feature propagation method, we hypothesize that it works better in higher-resolution grids where the voltage levels between nodes change more smoothly, which in our case is the PQ grid.

Comparing the performance between PQ and MV datasets, we can see that MV presents an easier task for all methods except the baseline one.

B. Observability degradation

In the first use case corresponding to subquestion A, we train the models on the grid with a baseline level of observability and then linearly reduce it to zero at testing time. Of course, the model performance decreases along with this reduction. The shorthand name of this use case is observability degradation (OD). This process is pictured in Figure 3. For readability, we limit the displayed GNN models to only the four best performing ones.

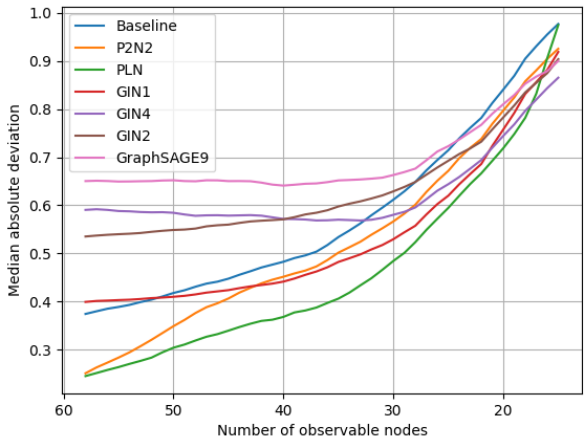

Figure 3. Performance degradation with observability reduction.

The results are unsurprising: all models behave similarly, and their performance smoothly drops from the reference values shown in Table I to the same final value at the end. GNN models maintain the reference performance longer, whereas other models lose performance immediately as the number of observable nodes decreases.

C. Homogeneous topology changes

The second use case corresponds to subquestion B and tests ZSL for homogeneous topology changes. In it, we split the TC dataset into 11 subsets according to the number of distinct topologies, meaning that the topology within each subset is static. We then perform a full 10-fold cross-validation, training the models on 10 subsets and testing on the remaining one. We then calculate the PDR for each fold. The resulting values are displayed as a box plot in Figure 4 to visualize the reliability of models in a ZSL setting.

The main observation from this graph is that the models that showed the best static performance before are now the worst performing, in terms of both averages and variation. To test if there is indeed a negative correlation, we use a scatter plot (Figure 5) of model performance on the two metrics (static MAD and median PDR in the current scenario.

The point spread suggested the existence of a Pareto front. To investigate further, we selected the non-dominated points with respect to both metrics, obtaining a bounding line that is plotted in red in Figure 5. We then computed Spearman's rank correlation on these points, which confirmed a strong negative monotonic correlation ($\rho \approx -1$, $p \approx 0$). The existence of such a clear empirical Pareto front strongly suggests a trade-off between the static accuracy of models and their robustness to homogeneous topology changes.

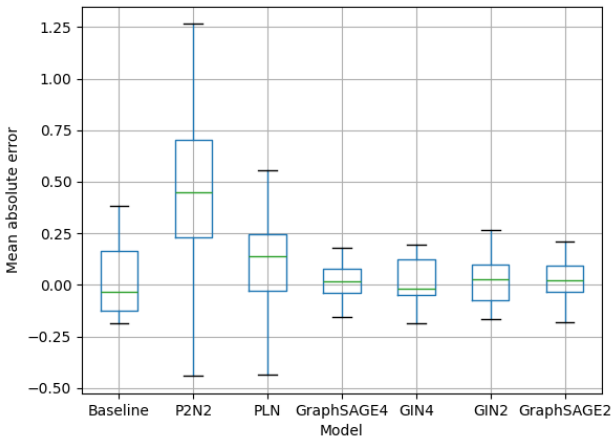

Figure 4. Cross-validation box plot.

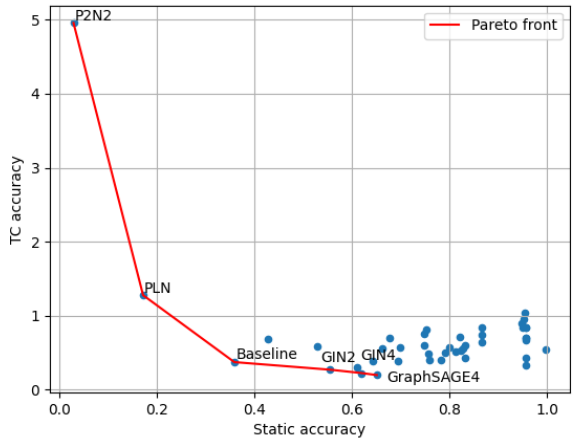

Figure 5. Cross-validation scatter plot.

The next aspect of this trade-off we wanted to analyze is the spatial error distribution that the aggregate metrics above do not capture. For this purpose, in Figure 6, we plot the comparison of the nodal error distribution between a static topology (labeled as PQ) and the switching scenario (labeled as TC). Since these plots are less space-efficient, we opted to show three cases: the MLP models (P2N2 and PLN) and an average across all GNN models, since the error distributions for these models were similar. Observable nodes are plotted as white. Note that the TC topology looks different and has a lower number of nodes due to the bus fusion transformation explained in Section IV-D.

From the figure, we can see different patterns of error propagation between the MLP and GNN models. For P2N2, errors under topology change conditions increase not just in magnitude but also in variance: the distribution of errors between nodes becomes noisy. For other models, the increase is more uniform, although certain nodes adjacent to switches pose a much harder challenge than the others. In general, failures of GNNs are less localized and instead "smeared" across the entire graph.

All models exhibit lower accuracy in long branches, which is explainable by two factors. First, switch changes mainly affect outward branches, because that is where most switches are located. Second, the outward branches are represented as path graphs, which are difficult to efficiently sample from and therefore require more sensors for robust signal reconstruction with GNNs [31].

D. Heterogeneous topology changes

The third use case corresponds to subquestion C and covers the heterogeneous ZSL scenario. Here, we transfer the model between the PQ and MV datasets in both directions, that is, training on one and then testing on another. The scenario where the model is trained on PQ and tested on MV has the shorthand name "PQ2MV", and the other "MV2PQ". We compute PDR for both transfers and present the results in the form of a scatter plot in Figure 7.

Note that it is impossible to test the MLP models in this scenario because their number of input and output features is fixed at training time, but the number of nodes between the two topologies is different. Therefore, we only test the GNN models here.

We can immediately see a significant difference between the two scenarios. Most GNNs handle PQ2MV, i.e., the transfer from a larger to a smaller network, much better than the reverse. Of course, this is in large part explainable by the fact that the MV dataset is simply an easier task, as shown above in the static performance evaluation.

A possible conjoint explanation is that the effective number of data points in a dataset for a GNN is equal to the number of snapshots multiplied by the number of nodes in the graph. Although the number of snapshots in the PQ and MV datasets is the same, the former contains more training data (and likely also more diverse data) than the latter. To test this hypothesis, we trained another series of models on a reduced PQ dataset where the number of data points is equalized with the MV dataset, and then reran the PQ2MV experiment. This increased PDR on average by 0.187, which supports the hypothesis but is not enough to fully explain the gap between PQ2MV and MV2PQ, suggesting that both factors are contributing to it.

VI. Conclusion and Future Work

Overall, the current state of the art can be represented as a three-way trade-off between conventional methods, feed-forward, and geometric models (GNNs). Conventional methods like WLS, based on the physical equations governing power systems, can provide reliably accurate results and are not affected by topology changes, as opposed to data-driven methods that infer the current state of the system from historical data. However, as we established in Section II, the amount of measurement data required for them to work is unattainable in distribution grids. But as we switch to NSE methods, we are faced with a choice between accurate feed-forward models that cannot generalize to different topologies,

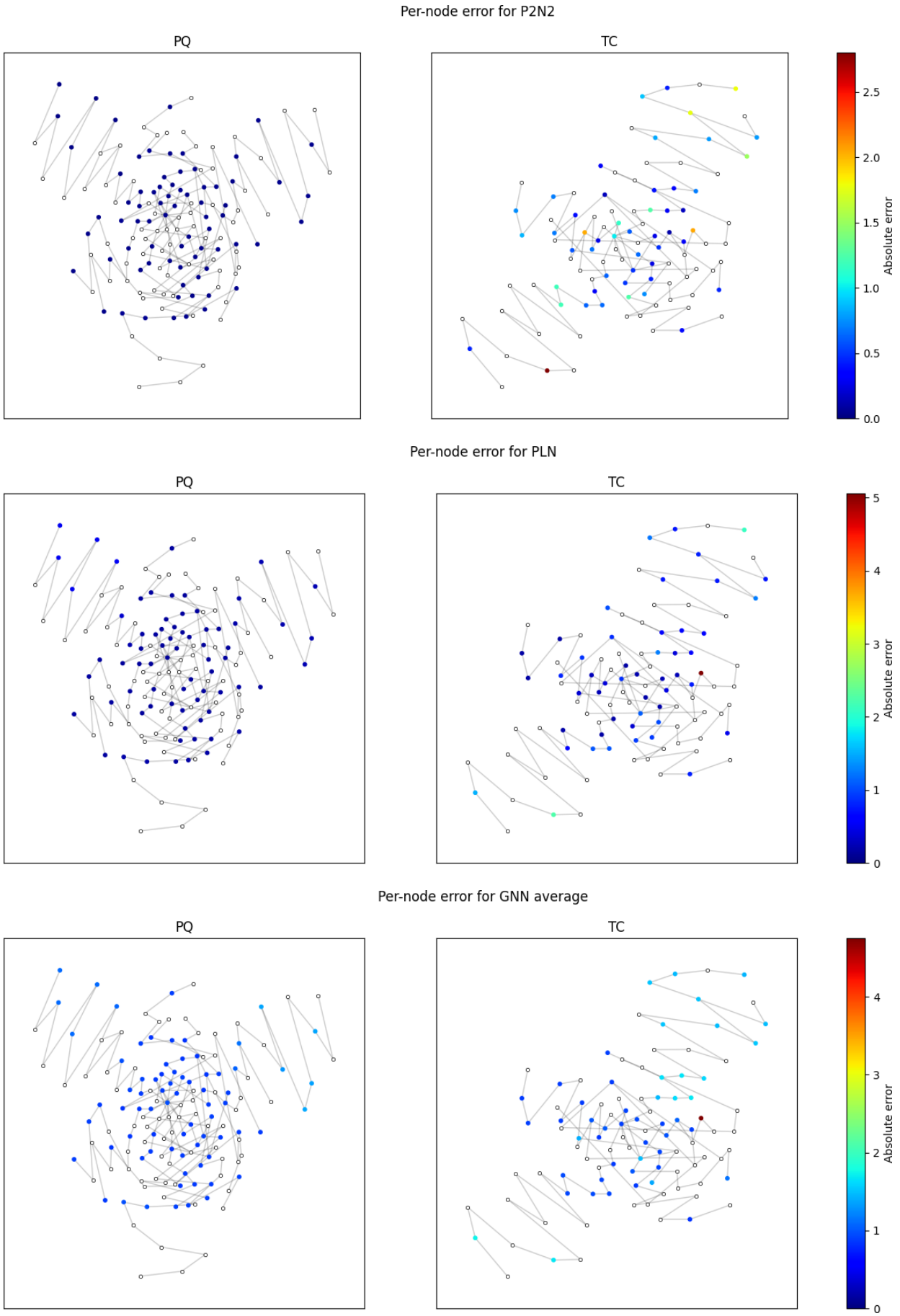

Figure 6. Nodal error comparison between static and dynamic topologies.

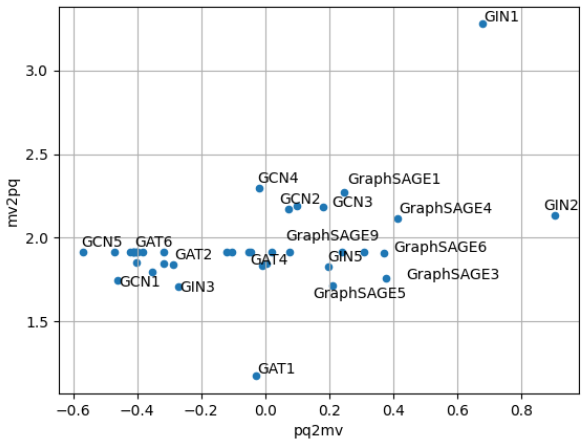

Figure 7. Heterogeneous transfer scatter plot.

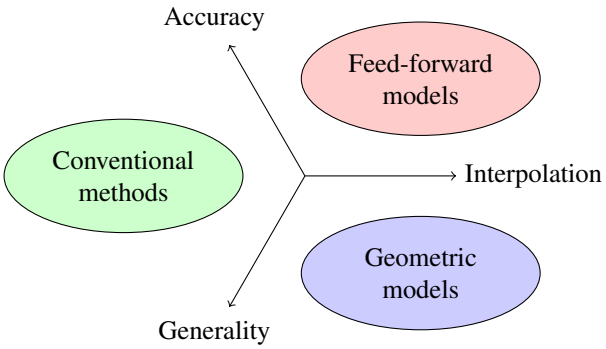

Figure 8. The three-way trade-off between conventional methods, feed-forward, and geometric models.

and GNNs, which can generalize but are not as accurate. We can therefore identify three desirable characteristics of PSSE methods:

- **Robustness to low observability**: how well does the method cope with limited measurement data, or how many sensors can be lost before the method becomes unreliable. This characteristic will be referred to as "Interpolation" in Figure 8.
- **Accuracy**: how closely can the method approximate ground truth data, assuming that there are enough data to fully utilize its expressive capacity.
- **Generality**: how well can the method adapt to changes in the grid topology without requiring retraining.

The trade-off between these characteristics is illustrated in Figure 8 and arises because no currently existing method has all three characteristics simultaneously.

Representing the problem in this way outlines the research gap: to create a method that combines performance, accuracy, and generality. This will be the direction of our future work.

Availability of Data and Source Code

The source code and datasets for this project are publicly available at the following repository:

https://gitlab.com/transense/nse-tl-paper

Acknowledgment

This research is partially supported by project TRANSENSE, funded by the German Federal Ministry for Economic Affairs and Climate Action (FKZ 03EI6044A).

References

[1] A. Berezin, S. Balduin, E. M. Veith, T. Oberließen, and S. Peter, "On zero-shot learning in neural state estimation of power distribution systems," in *ENERGY 2025, The Fifteenth International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies*, ser. IARIA Conference, Mar. 2025, pp. 47–52. [Online]. Available: https://www.thinkmind.org/library/ENERGY/ENERGY_2025/energy_2025_2_50_30034.html.

[2] F. F. Wu, "Power system state estimation: A survey," *International Journal of Electrical Power & Energy Systems*, vol. 12, no. 2, pp. 80–87, 1990, ISSN: 0142-0615. DOI: 10.1016/0142-0615(90)90003-T.

[3] A. Abur and A. G. Expósito, *Power System State Estimation*. CRC Press, Mar. 2004, ISBN: 9780203913673. DOI: 10.1201/9780203913673.

[4] K. R. Mestav, J. Luengo-Rozas, and L. Tong, "State estimation for unobservable distribution systems via deep neural networks," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, 2018, pp. 1–5. DOI: 10.1109/PESGM.2018.8586649.

[5] S. Balduin, T. Westermann, and E. Puiutta, *Evaluating different machine learning techniques as surrogate for low voltage grids*, Oct. 2020. DOI: 10.1186/s42162-020-00127-3.

[6] M. de Jong, J. Viebahn, and Y. Shapovalova, *Generalizable graph neural networks for robust power grid topology control*, 2025. arXiv: 2501.07186 [cs.LG].

[7] F. Fusco, B. Eck, R. Gormally, M. Purcell, and S. Tirupathi, *Knowledge- and data-driven services for energy systems using graph neural networks*, 2021. arXiv: 2103.07248 [cs.LG].

[8] A. Vasileiou, B. Finkelshtein, F. Geerts, R. Levie, and C. Morris, *Covered forest: Fine-grained generalization analysis of graph neural networks*, 2024. arXiv: 2412.07106 [cs.LG].

[9] S. Maskey, R. Paolino, F. Jogl, G. Kutyniok, and J. F. Lutzeyer, *Graph representational learning: When does more expressivity hurt generalization?* 2025. arXiv: 2505.11298 [cs.LG].

[10] K. Oono and T. Suzuki, "Graph neural networks exponentially lose expressive power for node classification," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=S1ldO2EFPr.

[11] X. Wu, Z. Chen, W. Wang, and A. Jadbabaie, *A non-asymptotic analysis of oversmoothing in graph neural networks*, 2023. arXiv: 2212.10701 [cs.LG].

[12] S.-G. Yang, B. J. Kim, S.-W. Son, and H. Kim, "Power-grid stability predictions using transferable machine learning.," *Chaos*, vol. 31 12, p. 123 127, 2021. DOI: 10.1063/5.0058001.

[13] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch Geometric," in *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[14] T. Siameh, *Semi-supervised classification with graph convolutional networks*, Dec. 2023. DOI: 10.13140/RG.2.2.22993.71526.

[15] P. Veličković *et al.*, *Graph attention networks*, 2018. [Online]. Available: https://openreview.net/forum?id=rJXMpikCZ.

[16] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, *How powerful are graph neural networks?* 2019. [Online]. Available: https://openreview.net/forum?id=ryGs6iA5Km.

[17] W. L. Hamilton, R. Ying, and J. Leskovec, *Inductive representation learning on large graphs*, Long Beach, California, USA, 2017.

[18] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, Mar. 1964, ISSN: 0003-4851. DOI: 10.1214/aoms/1177703732.

[19] T. Cai *et al.*, *Graphnorm: A principled approach to accelerating graph neural network training*, 2021. arXiv: 2009.03294 [cs.LG].

[20] A. S. Zamzam and N. D. Sidiropoulos, "Physics-aware neural networks for distribution system state estimation," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4347–4356, 2020.

[21] M.-Q. Tran, A. S. Zamzam, and P. H. Nguyen, *Enhancement of distribution system state estimation using pruned physics-aware neural networks*, 2021. DOI: 10.48550/ARXIV.2102.03893. [Online]. Available: https://arxiv.org/abs/2102.03893.

[22] L. Zhang, G. Wang, and G. B. Giannakis, "Real-time power system state estimation and forecasting via deep unrolled neural networks," *IEEE Transactions on Signal Processing*, vol. 67, no. 15, pp. 4069–4077, 2019.

[23] E. Rossi *et al.*, "On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features," Proceedings of Machine Learning Research, vol. 198, B. Rieck and R. Pascanu, Eds., 11:1–11:16, Dec. 2022.

[24] L. Thurner *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, 2018. DOI: 10.1109/TPWRS.2018.2829021.

[25] M. Ringsquandl *et al.*, "Power to the relational inductive bias: Graph neural networks in electrical power grids," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, ser. CIKM '21, ACM, Oct. 2021, pp. 1538–1547. DOI: 10.1145/3459637.3482464.

[26] O. Kundacina, M. Cosovic, and D. Vukobratovic, "State estimation in electric power systems leveraging graph neural networks," in *2022 17th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, IEEE, Jun. 2022, pp. 1–6. DOI: 10.1109/pmaps53380.2022.9810559.

[27] S. Meinecke *et al.*, "Simbench—a benchmark dataset of electric power systems to compare innovative solutions based on power flow analysis," *Energies*, vol. 13, no. 12, p. 3290, Jun. 2020, ISSN: 1996-1073. DOI: 10.3390/en13123290.

[28] J. Hiry, "Agent-based discrete-event simulation environment for electric power distribution system analysis," Ph.D. dissertation, 2021. DOI: 10.17877/DE290R-22549.

[29] K. Strunz, E. Abbasi, R. Fletcher, R. Iravani, and G. Joos, *Benchmark Systems for Network Integration of Renewable and Distributed Energy Resources*. Apr. 2014, ISBN: 9782858732708.

[30] S. Balduin, E. Veith, and S. Lehnhoff, "Midas: An open-source framework for simulation-based analysis of energy systems," in *Simulation and Modeling Methodologies, Technologies and Applications*. Springer International Publishing, 2023, vol. 780, pp. 177–194, ISBN: 978-3-031-43823-3. DOI: 10.1007/978-3-031-43824-0_10.

[31] F. Wang, Y. Wang, and G. Cheung, "A-optimal sampling and robust reconstruction for graph signals via truncated neumann series," *IEEE Signal Processing Letters*, vol. 25, no. 5, pp. 680–684, May 2018, ISSN: 1558-2361. DOI: 10.1109/lsp.2018.2818062.