TU Delft

Delft University of Technology

Document Version
Final published version

Licence
CC BY

Citation (APA)
Arowolo, O., & Cremer, J. L. (2026). Towards generalization of graph neural networks for AC optimal power flow. *Energy and AI*, *25*, Article 100842. https://doi.org/10.1016/j.egyai.2026.100842

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
In case the licence states “Dutch Copyright Act (Article 25fa)”, this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Energy and AI 25 (2026) 100842

Contents lists available at ScienceDirect

Energy and AI

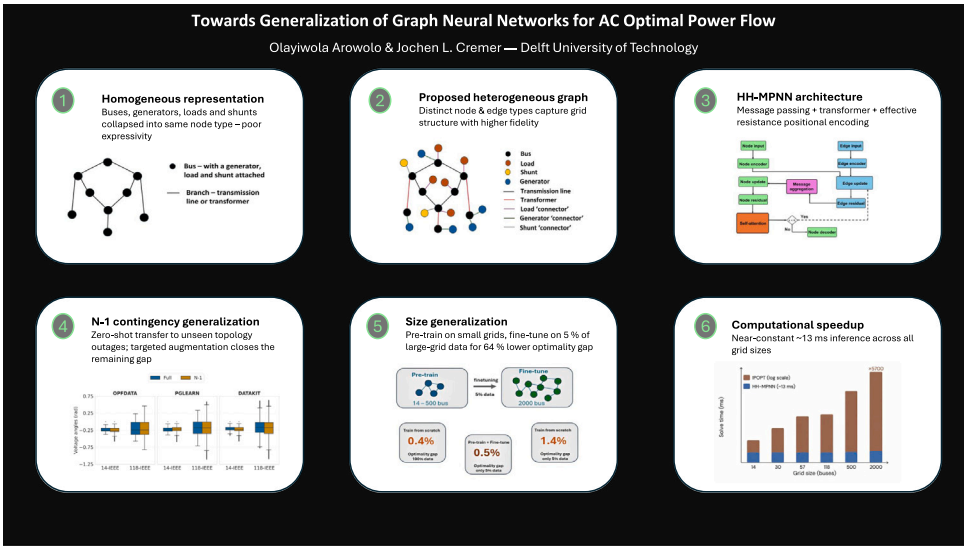
journal homepage: www.elsevier.com/locate/egyai

Towards generalization of graph neural networks for AC optimal power flow☆

Olayiwola Arowolo [a],*, Jochen L. Cremer [a,b]

[a] Delft University of Technology, Delft, the Netherlands
[b] Austrian Institute of Technology, Vienna, Austria

GRAPHICAL ABSTRACT



HIGHLIGHTS

- This paper presents HH-MPNN for approximating ACOPF solutions efficiently.
- HH-MPNN combines heterogeneous GNN and a scalar transformer using physics- informed positional encoding.
- Generalizes to N-1 contingencies; pretraining on smaller grids improves larger grids.
- Evaluated on diverse ACOPF benchmarks, including PGLearn and GridFM-Datakit datasets.

ARTICLE INFO

Keywords:
Graph neural network
Generalization
Machine learning

ABSTRACT

AC Optimal Power Flow (ACOPF) is computationally intensive for large-scale grids, often requiring prohibitive solution times with conventional solvers. Machine learning offers significant speedups, but existing models

☆ Codes available at https://github.com/Oarowolo11/TGGNN4ACOPF.

* Corresponding author.
E-mail addresses: o.a.arowolo@tudelft.nl (O. Arowolo), j.l.cremer@tudelft.nl (J.L. Cremer).

https://doi.org/10.1016/j.egyai.2026.100842
Received 4 June 2026; Received in revised form 2 July 2026; Accepted 9 July 2026
Available online 20 July 2026
2666-5468/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Optimal power flow
Topology

struggle with scalability and topology flexibility. To address these challenges, we propose a Hybrid Heterogeneous Message Passing Neural Network (HH-MPNN) that integrates a heterogeneous graph neural network (GNN) with a scalable transformer and physics-informed positional encodings. Our approach differs from previous GNN-based ACOPF approaches, which rely on local message passing alone. Our architecture explicitly models distinct power system components to capture local features while using global attention for long-range dependencies. Evaluated on diverse benchmarks including PGLearn and GridFM-DataKit datasets, HH-MPNN achieves <1% optimality gap on default topologies across grid sizes from 14 to 2000 buses on average. For N-1 contingencies, our approach demonstrates zero-shot N-1 generalization with <3% optimality gap on several test cases despite training only on default topologies. We further develop an approach that ensures robust N-1 generalization to high-impact contingencies through targeted augmentation of the training data, showing that exhaustive simulation is unnecessary for topologically flexible models. Finally, size generalization experiments demonstrate that pre-training on small grids significantly improves performance on large-scale systems. Achieving computational speedups of up to 5000 times compared to interior point solvers, these results advance practical, generalizable machine learning for real-time power system operations.

1. Introduction

The AC optimal power flow (ACOPF) problem determines the optimal generator dispatch in a grid, subject to physical and engineering constraints [1]. Despite being fundamental to power systems operations, including unit commitment and transmission expansion planning [2], ACOPF remains computationally challenging for large grids due to its non-convexity and non-linearity. No commercial solver returns globally optimal solutions for large grids within acceptable timeframes [1]. With system operators solving ACOPF every 5–15 min [1,2], and rapid demand fluctuations from Distributed Energy Resources (DERs) requiring even faster solution times, the computational burden of conventional solvers is becoming prohibitive [3,4].

While modern hardware can accelerate solvers like IPOPT [5], they remain slow for real-time use and scale poorly [6]. Operators thus rely on suboptimal relaxations [7,8] or DC approximations [9], which are never AC-feasible [10]. Suboptimal solutions for ACOPF are particularly impactful, as even a 5% improvement in efficiency can lead to savings of billions of dollars and significantly reduced carbon emissions [1].

Machine learning can accelerate ACOPF by shifting the computational burden to offline training. Existing approaches comprise: (1) end-to-end learning, which directly predicts ACOPF variables [11–14] and (2) learning-to-optimize that accelerates conventional solvers through warm-starts [15], active constraint predictions [16] or ML-aided distributed optimization [17]. See [2] for a comprehensive taxonomy on ML for ACOPF. While these approaches offer varying degrees of speedups, important challenges remain regarding topology flexibility and scalability.

A key challenge is flexibility to topology changes from outages, maintenance or transmission switching. While Fully Connected Neural Networks (FNNs) are the most widely used architecture in the literature, they struggle to adapt to topology changes [18]. Convolutional Neural Network (CNN) [19] and FNN variants [20] handle limited changes in system topology, but these remain grid-specific. Transformers have also been applied for sequential optimization problems in power systems, such as day-ahead security constrained unit-commitment. [21]. Graph Neural Networks (GNNs) are the most naturally suited to handling topology variations [22,23], but existing approaches combine GNN feature extraction with topology-specific fully connected layers, limiting their ability to handle arbitrary topology changes. Moreover, emerging cross-domain research indicates GNNs may be promising in engineering applications that require generalizing to new and larger systems [24,25]. GNN approaches have also been combined with physics-informed loss to improve constraint satisfaction [26,27], or used to predict warm start points for iterative ACOPF solvers [28]. However, pure GNN approaches using message passing have a 'locality bottleneck' problem. As such, they struggle to accurately predict variables like voltage angle, which depend on global grid properties, leading to significantly inferior performance compared to FNN baselines [18,29].

Existing GNN approaches also commonly use homogeneous graphs, treating all buses identically and modelling transmission lines and transformers as edges. Homogeneous graphs inadequately represent complex power networks where buses connect to varying numbers of generators, loads, and shunts. As such, homogeneous GNNs cannot be used for large, complex grids. [30,31] have proposed heterogeneous graph approaches for ACOPF. To overcome the 'locality bottleneck' problem, [30] requires up to 60 GNN layers to propagate information through large networks, making training computationally expensive and scalability challenging. [31] was only evaluated on small systems (up to 30 buses) with limited complexity. Furthermore, most works require retraining/finetuning for each N-1 contingency or generating exhaustive N-1 training datasets, which becomes prohibitive for large grids. It becomes important to investigate whether a single GNN model can generalize to an unseen generator or line outage without exhaustive retraining or N-1 data generation.

Moreover, emerging research [32] has highlighted the limitations of generating ACOPF data through uncorrelated bus load perturbations. Though this method of load perturbation is widely used in the literature [33], it leads to limited variation in total load demand and consequently, similar dispatch patterns across various ACOPF samples [34]. The limitation of the data generation method severely challenges the generalization assessment of several state-of-the-art ML methods for ACOPF. To gain confidence in the use of ML for ACOPF, it is important to evaluate models with challenging datasets which closely match realistic operating conditions with significant variation in total load, generator costs, network topology, etc. [35].

In this work, we address critical challenges related to topology flexibility, constraint feasibility, and scalability of ML for ACOPF. Inspired by [36], our novel architecture provides important insights into how power systems domain knowledge can be embedded into a graph-based model to improve performance on large-scale power systems. Our main contributions are:

(1) Hybrid Graph Architecture: We propose a novel architecture which integrates heterogeneous message passing, a scalable transformer, and physics-informed positional encoding. Our design overcomes the 'locality bottleneck' of standard GNNs, enabling accurate ACOPF predictions across grid sizes.
(2) Efficient N-1 Generalization: We develop an approach for robust N-1 generalization through targeted training on a small subset of high-impact contingencies, proving that exhaustive N-1 data generation is unnecessary for our topologically flexible model. We also show that our model is sufficient for accurate zero-shot prediction for a majority of N-1 contingencies.
(3) Size Generalization: We demonstrate that representations learnt on smaller, computationally "cheaper" grids can be effectively transferred to significantly larger systems (up to 2000 buses) via fine-tuning. This offers a practical strategy to reduce the prohibitive data generation costs typically associated with large-scale grids.

The rest of the paper is organized as follows: Section 2 describes our proposed model. Section 3 presents case studies and results. Section 4 discusses the results and limitations, while Section 5 concludes the paper.

2. Hybrid heterogeneous message passing neural network

We propose a Hybrid Heterogeneous Message Passing Neural Network (HH-MPNN) for predicting ACOPF variables. MPNNs are a class of GNNs in which the model learns to make predictions on graphs through an iterative exchange of information between nodes, known as message passing. The proposed HH-MPNN consists of two components. (1) A heterogeneous MPNN which aggregates local information in the graph. (2) A scalable transformer that enables global information exchange through self-attention.

2.1. Heterogeneous message passing neural network

Our heterogeneous MPNN follows the encode–process–decode framework [37]. We represent the power grid with four node types (buses, generators, loads, shunts) and three edge types (transmission lines, transformers, and connector pseudo-edges linking buses to other node types). This explicit representation of power system components improves model expressivity compared to homogeneous graphs that treat all buses identically. The architecture operates in three stages: (1) Encoding: Project node and edge features into a shared embedding space, augmented with positional encodings (see Section 2-B). (2) Processing: Iteratively update node and edge embeddings through message passing with residual connections to prevent over-smoothing. (3) Decoding: Map final node embeddings to predicted ACOPF variables using node type-specific decoders. The bus decoder predicts voltage angle and magnitudes for each bus, while the generator decoder predicts active and reactive power generation from each generator.

The encoding stage projects features through 2-layer multilayer perceptrons (MLPs):

$$h^0_{a,i} = E^a_\theta(x_{a,i}) \oplus PE_i \tag{1a}$$

$$h^0_{e,i,j} = E^e_\theta(e_{i,j}) \tag{1b}$$

where $a \in [N, G, S, L]$ denotes node type (bus, generator, shunt, load) and e denotes edge type (transmission line or transformer). $x_{i,j}$ and $e_{i,j}$ are input features, PE_i is the positional encoding and $\oplus$ denotes concatenation. Message passing updates edges using adjacent node states, then aggregates edge messages to update nodes, both with residual connections:

$$\tilde{h}^k_{e,i,j} = U^e_\theta(h^{k-1}_{a,i}, h^{k-1}_{a,j} h^{k-1}_{e,i,j}) \tag{2a}$$

$$h^k_{e,i,j} = h^{k-1}_{e,i,j} + \tilde{h}^k_{e,i,j} \tag{2b}$$

$$m^k_{a,i} = \sum_{j \in N_i} (h^k_{e,i,j}) \tag{2c}$$

$$\tilde{h}^k_{a,i} = U^a_\theta(h^{k-1}_{a,i}, m^k_{a,i}) \tag{2d}$$

$$h^k_{a,i} = h^{k-1}_{a,i} + \tilde{h}^k_{a,i} \tag{2e}$$

All E_θ and U_θ functions are 2-layer MLPs with separate parameters for each node and edge type. For connector pseudo-edges, (1b) is omitted, and edge updates in (2a) use only adjacent node features without edge-specific features.

2.2. Transformer with effective resistance positional encoding

To capture long-range dependencies in power grids, we augment the local MPNN with a scalable transformer using performer attention [38], which approximates self-attention with linear complexity. Performer is an approximation of full self-attention, which scales as $O(N)$ rather than $O(N^2)$ for full self-attention. This makes training tractable on large systems such as the tested 2000-bus system. Since graph transformers require positional encodings to distinguish node positions, we use effective resistance, a domain-informed metric that captures both local and global electrical distances as positional encoding [39].

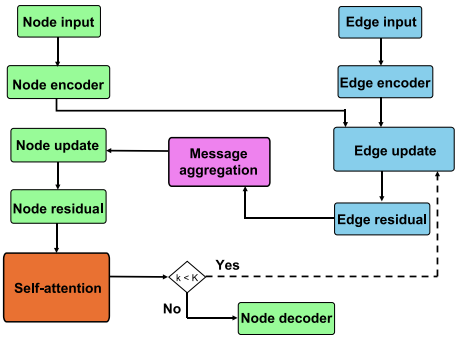


Fig. 1. The proposed HH-MPNN architecture with detailed message passing mechanism showing how information flows between different node and edge types. Skip connections and bidirectional updates enhance model expressivity. K=5 message passing layers are used. The model predicts ACOPF setpoints.

Effective resistance represents the potential difference between a pair of nodes when a unit of current is injected into one node and leaves at the other [39]. Under DC approximation, the power balance equation can be written as $P = Q\theta$, where Q is the weighted Laplacian matrix with elements:

$$Q = \begin{cases} -B_{ij}, & i \neq j \\ \sum_{j=1}^{N} B_{ij}, & i = j \end{cases} \tag{3}$$

where B_{ij} is the susceptance of the edge connecting nodes i and j. From the DC power balance equation, θ can be written as Q^+P, where Q^+ is the pseudoinverse of Q. Effective resistance ω_{ij} can now be defined as [39]:

$$\omega_{ij} = (e_i - e_j)^T Q^+ (e_i - e_j) \tag{4}$$

where $\theta_i - \theta_j = \omega_{ij} P_{ij}$ and e_i, e_j are basic vectors with 1 at positions i, j respectively and 0 elsewhere.

The effective resistance matrix, $\Omega \in \mathbb{R}^{N \times N}$, gives a notion of how difficult it is to move a unit of current from node i to j, which considers all paths in the graph [40]. However, its dimensionality scales with grid size. To maintain a consistent dimension across different grids, which is a prerequisite for size generalization [41], we summarize the N-dimensional effective resistance vector for each node into five statistical moments as shown in Eq. (5). Following the approach from [42], these moments provide a fixed-size electrical summary that captures the centrality, radius and neighbourhood structure of a node.

$$PE_i = [min, max, std, median, mean].\Omega_{[i,:]} \tag{5}$$

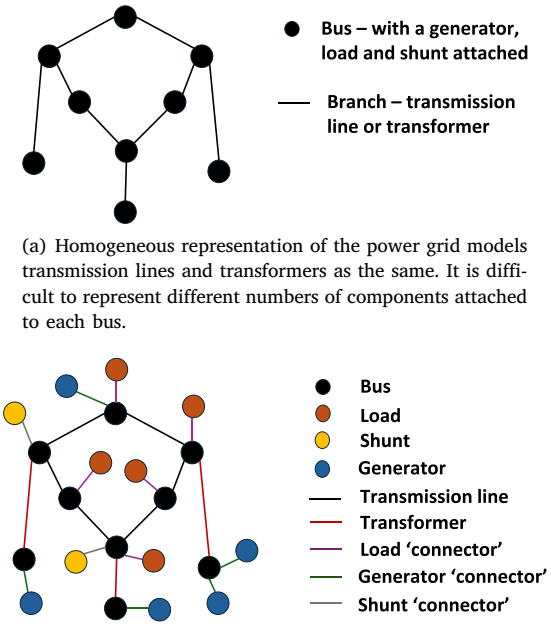
PE_i summarizes the effective resistance from bus i to other buses in the grid. This preserves distributional information about each node's electrical position while keeping the encoding dimension fixed at 5 across all grid sizes. Algorithm 1 summarizes the complete HH-MPNN forward pass. After encoding node features with positional information in line 1, we alternate between local message passing (line 3) and global attention (line 5) for K layers. The attention mechanism operates on concatenated node embeddings from all types (line 4), enabling information exchange across the entire graph. We combine local MPNN updates with global transformer updates via element-wise addition (line 7), with a 2-layer MLP projection T_θ to integrate the representations. The transformer output also employs a residual connection (line 6). Type-specific 2-layer MLP decoders (buses and generators) produce final predictions (line 9). We use K=5 message passing layers in all experiments(see Fig. 1).

Algorithm 1: Algorithm for HH-MPNN. X is the input feature matrix across all 4 node types, and PE is the positional encoding.

$X^0 = E^a_\theta(X) \oplus PE$
for $k = 0, 1, \ldots, K-1$ **do**
  $(X_M^{k+1}, E^{k+1}) = \text{MPNN}(X^k, E^k)$
  Aggregate node embeddings for all node types
  $\hat{X}_T^{k+1} = \text{AT}(X^k)$
  $X_T^{k+1} = X_T^k + \hat{X}_T^{k+1}$
  $X^{k+1} = T_\theta(X_T^{k+1} + X_M^{k+1})$
  Disaggregate node embeddings by node types
$Y = D^a_\theta(X^K)$

Table 1
Comparison of the 3 datasets used in this study.

Setting	Dataset		
	OPFData	PGLearn	Datakit
Global load scaling	×	✓	✓
Independent bus load sampling	✓	✓	✓
Branch admittance perturbation	×	×	✓
Gen cost perturbation	×	×	✓
Topology perturbation	N-1 (line, transformer, generator)	N-1 (line)	N-1 (line, transformer, generator)



(a) Homogeneous representation of the power grid models transmission lines and transformers as the same. It is difficult to represent different numbers of components attached to each bus.

(b) Proposed heterogeneous graph with different node and edge types enables a more accurate representation of the power grid.

Fig. 2. Heterogeneous versus Homogeneous graph representation of the power grid.

3. Case study

3.1. Dataset

We use 3 open-source ACOPF datasets for validating our approach. The OPFData [43] and PGLearn [34] datasets are publicly available, and we generate the Datakit dataset with the GridFM-Datakit tool [35]. Each dataset has two variants: a 'full topology' dataset with the default grid topology and an 'N-1 topology' dataset, which has a randomly disconnected component (branch or generator). Where available, we use 6 grids from each dataset for our experiments: 14, 30, 57, 118, 500 and 2000-bus grids. In addition to N-1 topology variations, GridFM-datakit allows variation in generator costs and network parameters, potentially increasing the diversity of dispatch patterns. We refer to [35] for a quantitative comparison of the diversity of the 3 datasets. Table 1 shows a comparison of the data generation settings for the 3 datasets. We provide details on the settings we use for GridFM-datakit data generation in the Appendix C.

3.2. Baselines

We compare the performance of our proposed approach with the following baseline machine learning models:

3.2.1. Fully connected neural network (FNN)

The network comprises 5 linear layers with a hidden dimension of 256 and ReLU activations. Input and output vectors are flattened, making the architecture tied to a specific grid topology.

3.2.2. Convolutional neural network (CNN)

We use 5 1D convolutional layers with ReLU activation and max pooling, followed by adaptive average pooling and a linear output layer with sigmoid activation. Like FNN, this architecture is topology-specific.

Table 2
The hyperparameters used for training HH-MPNN. x denotes the variable input dimension, which depends on node or edge type. Batch sizes are adjusted to grid size.

Parameter	Value
Encoder layers	x–256–256
Decoder layers	256–256–2
Update layers	x–256–256
No. of GNN layers	5
No. of attention layers	5
No. of attention heads	1
Normalization type	LayerNorm
Learning rate	1×10^{-5}
Weight decay	5×10^{-8}
Optimizer	Adam
No. of epochs	100
Batch size	256/64/16

3.2.3. Graph convolutional neural network (GCN)

We represent buses as nodes and branches as edges, assuming uniform components at each bus (generator, load, shunt) and using generator masks to indicate presence/absence. This homogeneous representation is less flexible than our heterogeneous approach, as shown in Fig. 2. The model comprises 5 graph convolution layers with a hidden dimension of 256. We apply GCN only to grids for which homogeneous representation is tractable (14, 30, 57, and 118-bus systems). In the case of 500 and 2000-bus systems, multiple generators with different costs may be connected to a bus, making a homogeneous representation difficult.

3.3. Experiment settings

The ACOPF prediction is a supervised multi-output node regression task on power system graphs. The inputs are per-node features for buses, generators, loads and shunts, e.g., active and reactive power demand for loads, generator costs for generators, etc., as well as branch

Table 3
Mean Squared Error (MSE) of the models on the full topology dataset. Errors are in pu except for θ in radians. The best (lowest) error for each dataset is highlighted.

System	Model	OPFData ($\times 10^{-4}$)				PGLearn ($\times 10^{-4}$)				DataKit ($\times 10^{-4}$)			
		θ	V	P_G	Q_G	θ	V	P_G	Q_G	θ	V	P_G	Q_G
14-IEEE	FNN	0.40	0.02	3.00	3.00	0.42	0.18	11.0	3.0	0.35	0.09	2.00	5.00
	CNN	0.03	0.02	**0.08**	2.00	0.10	0.04	0.50	0.94	0.09	0.07	1.00	3.00
	GCN	3.00	0.43	17.00	3.00	6.00	2.00	72.00	13.00	3.00	1.00	50.00	11.00
	HH-MPNN	**0.01**	**0.00**	0.21	**0.06**	**0.01**	**0.01**	**0.04**	**0.04**	**0.06**	**0.01**	**0.12**	**0.05**
30-IEEE	FNN	0.32	0.06	13.00	3.00	0.27	0.06	3.00	4.00	0.21	0.08	2.00	6.00
	CNN	0.07	0.05	**0.32**	2.00	0.09	0.03	0.41	3.0	0.08	0.05	1.00	3.00
	GCN	2.00	0.60	3.00	0.98	5.00	1.00	28.00	5.00	2.00	0.75	15.00	3.00
	HH-MPNN	**0.05**	**0.01**	0.37	**0.24**	**0.01**	**0.01**	**0.02**	**0.13**	**0.06**	**0.01**	**0.01**	**0.19**
57-IEEE	FNN	0.63	0.06	142.00	44.00	0.26	0.06	43.0	22.0	0.43	0.11	18.0	26.0
	CNN	0.64	0.06	139.00	44.00	0.20	0.04	9.00	16.00	0.34	0.09	29.00	23.00
	GCN	3.00	1.00	**60.00**	**6.00**	3.00	1.00	47.00	11.00	31.00	3.00	408.00	34.00
	HH-MPNN	**0.37**	**0.02**	106.00	20.00	**0.03**	**0.01**	**4.00**	**0.84**	**0.15**	**0.03**	**6.00**	**3.00**
118-IEEE	FNN	2.00	0.10	55.00	75.00	3.0	0.07	37.00	35.00	7.00	**0.44**	273.00	156.00
	CNN	**0.83**	**0.04**	**12.00**	**22.00**	**0.58**	**0.05**	**7.00**	20.00	21.00	0.58	277.00	238.00
	GCN	9.00	0.77	318.00	68.00	26.00	1.00	479.00	246.00	186.00	4.00	2167.00	585.00
	HH-MPNN	3.00	0.07	42.00	33.00	0.85	0.07	**7.00**	**11.00**	**6.00**	0.57	**79.00**	**96.00**
500-GOC	FNN	3.00	0.05	13.00	11.00	N/A	N/A	N/A	N/A	4.00	0.11	30.00	24.00
	CNN	3.00	0.05	13.00	11.00	N/A	N/A	N/A	N/A	9.00	0.43	440.00	88.00
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	**1.00**	**0.03**	**6.00**	**7.00**	N/A	N/A	N/A	N/A	**1.57**	**0.22**	**7.63**	**17.64**
2000-GOC	FNN	0.88	0.09	10.00	17.00	N/A	N/A	N/A	N/A	3.00	0.27	62.00	42.00
	CNN	6.00	0.09	10.00	17.00	N/A	N/A	N/A	N/A	12.00	0.46	141.00	69.00
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	**0.34**	**0.06**	**5.00**	**9.00**	N/A	N/A	N/A	N/A	**2.03**	**0.34**	**16.05**	**18.76**

features such as line parameters. We predict the voltage angles θ, voltage magnitude V, active generator power PG and reactive generator power QG. Following [6], we apply a sigmoid function to bound outputs and satisfy inequality constraints (see Appendix A, (7b)–(7d)). Branch flows are computed using power flow Eqs. (7g)–(7h) after the model prediction. Angle difference (7e), branch thermal limit (7f), and power balance (7i) constraints may be violated. We train the models with supervised mean square error (MSE) loss between predicted and ground truth variables. All models use Adam optimizer with learning rate 1 x 10^{-5} and weight decay 5 x 10^{-8}, trained for 100 epochs. We use 270,000 samples for training (90%), 15,000 for validation (5%) and 15,000 for testing (5%) for the OPFData and GridFM datasets. For the PGLearn datasets, we also train on 270,000 samples, but we evaluate on the entire test dataset provided by the dataset. Batch size is 256 for small grids (14–118 buses), 64 for the 500-bus system, and 16 for the 2000-bus system. We used a random seed of 42 for splitting the datasets in all the experiments. All models have 5 layers with a hidden dimension of 256. All experiments were performed on a Linux machine with 32 CPU cores and 256 GB of memory and a NVIDIA A30 GPU with 24 GB of memory. The hyperparameters used for the experiments are shown in Table 2.

3.4. Metrics

We evaluate models using three metrics: (1) **Mean Squared Error (MSE)** between predictions and ground truth for each ACOPF variable. (2) **Optimality gap**: percentage difference between model and ground truth objective costs. (3) **Constraint satisfaction**: average and maximum violations of constraints not strictly enforced ((7e),(7f),(7i)). Generator power and voltage magnitude constraints (7b)–(7d) are satisfied by design through sigmoid activation. We report constraint violations over the whole test dataset.

3.5. Prediction on 'full topology' dataset

Tables 3–6 summarize the results on the full topology datasets. **Prediction accuracy** (Table 3): We make three observations about the results. First, model errors generally increase with increasing grid size. The pattern is consistent across the three datasets, suggesting that larger grids with more generators have more complex dispatch patterns, which make learning more challenging. However, model performance does not decrease monotonically; for example, models achieve higher accuracy on the 2000-bus system than on the 118-bus system for the OPFData and Datakit datasets. Second, the GCN model consistently has the worst performance, with orders of magnitude larger errors on the 118-bus system compared to other models. This highlights the poor scaling behaviour of the baseline graph model and its difficulty with predicting generator dispatches, which have little to no locality dependence. HH-MPNN provides significant improvements over the GCN model, with the hybrid architecture achieving similar or better accuracy compared to the global architectures (FNN and CNN) even on the largest dataset. Third, the same model trained on different datasets can achieve accuracy of different orders of magnitude. Across the 3 datasets, similar error values can be observed on the smaller grids (14, 30 and 57-bus), but on the larger datasets (500 and 2000-bus), errors of different orders of magnitude can be observed between the OPFData and the Datakit datasets. This confirms the concern in the literature that the data sampling method (particularly correlated versus uncorrelated loads) strongly influences machine learning benchmark outcomes. The Datakit dataset is the most diverse of the three and constitutes the most challenging learning problem for the machine learning models. Interestingly, our proposed HH-MPNN model achieves the lowest error for all variables and for all the tested grids on the Datakit dataset, highlighting the superior performance of our approach.

Constraint violations and optimality gap (Tables 4–6): We also present the optimality gap and constraint violations of the models since the predicted solutions are often infeasible. All models satisfy angle difference constraints (θ_{ij}) completely, so we do not include these in the constraint violations. Branch flow limits and power balance are the only remaining constraints that can be violated. First, there are no branch flow violations for the 14 and 57-bus systems across models and datasets, which suggests that lines are not congested in these systems. Once again, we see the GCN model incurs orders of magnitude higher line violations than the other models on the 30-bus and 118-bus systems, a pattern which holds across datasets. However, the HH-MPNN

Table 4
Constraint violations and optimality gap for PGLearn Dataset. The best (lowest) violations and optimality gaps are highlighted.

System	Model	Average constraint violation ($\times 10^{-4}$ p.u.)				Max constraint violation (p.u.)				Optimality gap (%)
		$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	
14-IEEE	FNN	0.00	0.00	6.00	8.00	0.00	0.00	0.41	**0.14**	0.68
	CNN	0.00	0.00	**0.16**	**3.00**	0.00	0.00	**0.37**	**0.14**	**0.03**
	GCN	0.00	0.00	6.00	17.00	0.00	0.00	1.05	0.46	2.84
	HH-MPNN	0.00	0.00	0.29	1.15	0.00	0.00	0.68	0.16	0.05
30-IEEE	FNN	1.00	**0.00**	5.00	**0.30**	0.10	0.10	0.49	0.22	1.81
	CNN	**0.25**	0.07	**0.04**	1.00	**0.08**	**0.06**	**0.38**	**0.18**	**0.01**
	GCN	188.00	186.00	18.00	27.00	0.79	0.78	1.81	1.17	7.28
	HH-MPNN	3.44	0.30	0.36	2.00	0.09	**0.06**	0.63	0.27	0.14
57-IEEE	FNN	0.00	0.00	**0.05**	2.00	0.00	0.00	1.00	0.67	**0.08**
	CNN	0.00	0.00	3.00	**0.43**	0.00	0.00	**0.85**	**0.50**	0.22
	GCN	0.00	0.00	2.00	15.00	0.00	0.00	3.58	1.46	0.99
	HH-MPNN	0.00	0.00	2.85	0.70	0.00	0.00	2.57	0.68	0.15
118-IEEE	FNN	**2.00**	**2.00**	**0.86**	**4.00**	**1.97**	**1.97**	**6.37**	**1.21**	**0.04**
	CNN	5.00	5.00	2.00	**4.00**	3.10	3.10	10.40	1.58	**0.04**
	GCN	925.00	924.00	255.00	514.00	26.72	26.87	30.52	9.69	21.44
	HH-MPNN	9.00	10.13	0.90	22.70	3.23	3.26	7.34	3.10	0.32

model frequently obtains lower average and maximum constraint violations than the FNN and CNN models. The performance of the HH-MPNN is also remarkably consistent, with optimality gaps $< 1\%$ on all systems and datasets. The superior performance of the model is more apparent on the most complex Datakit dataset. The Datakit dataset incorporates changes to generator cost, topology and admittance variation, hence it produces more diverse dispatch patterns. The HH-MPNN, being a more complex model than the baselines, can learn these complex patterns, hence, the improvement is more pronounced on the Datakit dataset compared to the OPFData data with only load-perturbation. With the Datakit dataset, the HH-MPNN model achieves superior or equal optimality gaps while maintaining comparable violations to the global models. The average line flow violations of the model for the three datasets reflect how heavily loaded the systems are for the datasets. For example, HH-MPNN and the global models have line violations in the order of $1x10^{-4}$pu on the 118-bus system for PGLearn and OPFData, but violations 1–2 orders of magnitude higher for the same system on the GridFM-Datakit dataset. This reflects how close the lines are to the limits in the samples contained in each dataset on average. Note that line flows are analytically derived from the predicted voltage magnitudes and angles. Power balance constraint violations in the order of 10^{-4} to 10^{-2} are the most significant violations incurred by the model. But this is acceptable for approximate solvers and comparable to state-of-the-art results [6,30]. (See Appendix D for comparison with additional baselines.) Our approach can already be used for applications which require an approximate solver, where DCOPF is currently used, such as network topology optimization. However, for applications which require the strict satisfaction of all AC constraints, the model prediction may be used as a warm start for the full ACOPF optimization process or may be combined with feasibility restoration/projection steps.

3.6. Zero-shot generalization to N-1 cases

A key property of graph-based models is adaptability to topology changes. We demonstrate zero-shot generalization whereby a model trained only on full topology datasets (*mfull*) is applied to the test subset of the N-1 dataset without retraining. We compare this zero-shot experiment with models trained on the N-1 training subset (*mnminusone*) to establish an upper bound, which we refer to as the base experiment.

Optimality gap (Fig. 3): HH-MPNN demonstrates robust zero-shot performance, maintaining $< 3\%$ optimality gap in experiments on both the OPFData and PGLearn datasets despite training only on the full topology data. This shows the model's flexibility to handle topological variations. However, results on the Datakit dataset show the fundamental limit of zero-shot generalization. Specifically, the 118-bus shows

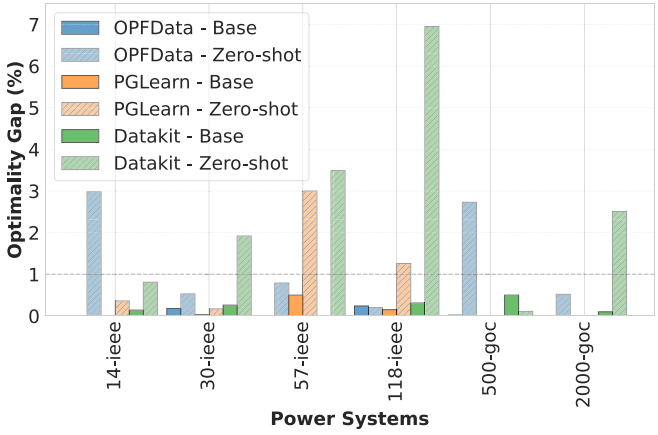


Fig. 3. Comparison of average optimality gaps of the proposed HH-MPNN model in the base and zero-shot experiments. Zero-shot optimality gaps show moderate degradation with increasing system size. Zero-shot generalization is weakest for the Datakit 118-bus system.

a 6.9% optimality gap in the zero-shot setting. This underscores that a topology-aware architecture may still struggle when the underlying power flow patterns in unseen scenarios deviate significantly from the training distribution. We illustrate this in Fig. 4, where the distribution shift on the 118-bus datakit dataset is the most pronounced. Large voltage angles may also indicate that lines may be operating near their thermal limits, and many constraints are likely active — in other words, the out-of-distribution cases in the N-1 dataset are operationally challenging high-impact contingencies. Despite this challenge, the zero-shot optimality gaps remain comparable to those of DCOPF approximations ($3-4\%$ optimality gap) [30], demonstrating that the model retains practical utility even under distribution shift. Strong generalization to N-1 scenarios and fast inference time enables real-time N-1 contingency analysis. Interestingly, the zero-shot performance does not degrade monotonically with increasing system size; zero-shot performance in the 500-bus case, for example, is better than the 118-bus case. This suggests that the difficulty of the OPF problem may depend more on the number of active constraints in a scenario than on the grid size. As one might expect, zero-shot experiments show larger violations than the base models, due to the distribution shift between the full-topology training data and the N-1 topology test data. Constraint satisfaction can be further improved through post-processing: we demonstrate the application of power flow post-processing on the 118-bus system in

Table 5
Constraint violations and optimality gap for OPFData Dataset. The best (lowest) violations and optimality gaps are highlighted.

System	Model	Average constraint violation ($\times 10^{-4}$ p.u.)				Max constraint violation (p.u.)				Optimality gap (%)
		$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	
14-IEEE	FNN	0.00	0.00	5.60	0.56	0.00	0.00	0.12	0.08	0.28
	CNN	0.00	0.00	**0.21**	1.22	0.00	0.00	0.11	0.06	**0.01**
	GCN	0.00	0.00	8.08	40.13	0.00	0.00	0.90	0.46	0.67
	HH-MPNN	0.00	0.00	0.53	**0.22**	0.00	0.00	**0.10**	**0.05**	0.07
30-IEEE	FNN	**0.00**	**0.00**	0.56	0.65	**0.00**	**0.00**	**0.21**	**0.08**	0.36
	CNN	4.42	0.05	**0.14**	1.14	**0.00**	**0.00**	0.27	0.10	**0.06**
	GCN	226.50	217.70	8.53	36.29	0.92	0.88	1.89	0.84	0.61
	HH-MPNN	0.25	0.00	0.19	**0.62**	0.04	0.00	0.32	0.12	0.13
57-IEEE	FNN	0.00	0.00	**1.68**	**0.46**	0.00	0.00	**0.36**	**0.18**	0.13
	CNN	0.00	0.00	1.83	3.26	0.00	0.00	0.44	0.25	**0.08**
	GCN	0.00	0.00	4.08	4.73	0.00	0.00	3.03	1.41	0.36
	HH-MPNN	0.00	0.00	5.51	4.50	0.00	0.00	0.91	0.41	0.25
118-IEEE	FNN	**0.28**	**0.06**	6.86	**1.53**	**0.47**	**0.47**	**2.61**	**0.67**	0.28
	CNN	1.57	1.18	**0.15**	5.60	1.17	1.16	3.71	1.28	**0.01**
	GCN	477.10	478.70	225.10	310.00	4.98	5.01	8.33	5.62	9.94
	HH-MPNN	11.12	11.35	3.04	13.75	2.44	2.45	4.39	1.89	0.05
500-GOC	FNN	3.65	3.45	1.47	**0.88**	0.31	0.30	5.12	1.20	0.05
	CNN	**0.00**	**0.00**	**1.18**	2.34	**0.00**	**0.00**	**3.21**	**0.64**	**0.04**
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	48.32	48.42	3.38	37.55	19.55	19.50	66.16	17.59	0.17
2000-GOC	FNN	**0.00**	**0.00**	**0.73**	**1.36**	**0.00**	**0.00**	3.27	0.71	0.02
	CNN	**0.00**	**0.00**	1.26	8.48	**0.00**	**0.00**	**2.49**	**0.60**	**0.01**
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	0.85	0.86	2.19	9.59	3.84	3.84	24.80	8.64	**0.01**

Table 6
Constraint violations and optimality gap for GridFM-datakit Dataset. The best (lowest) violations and optimality gaps are highlighted.

System	Model	Average constraint violation ($\times 10^{-4}$ p.u.)				Max constraint violation (p.u.)				Optimality gap (%)
		$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	$S_{ij}(+)$	$S_{ji}(-)$	P_b	Q_b	
14-IEEE	FNN	0.00	0.00	2.00	**0.74**	0.00	0.00	0.50	0.50	0.23
	CNN	0.00	0.00	**0.20**	3.00	0.00	0.00	**0.38**	**0.38**	**0.04**
	GCN	0.00	0.00	8.00	12.00	0.00	0.00	1.26	0.76	3.77
	HH-MPNN	0.00	0.00	0.72	4.00	0.00	0.00	0.42	0.30	0.15
30-IEEE	FNN	**0.20**	0.35	3.00	**2.00**	0.52	0.52	0.79	0.48	1.11
	CNN	1.00	**0.13**	3.0	**2.00**	**0.18**	**0.18**	**0.51**	0.35	0.53
	GCN	199.00	190.00	3.00	26.00	1.11	1.07	2.13	1.05	5.73
	HH-MPNN	3.00	1.00	**0.55**	3.00	0.28	0.24	0.58	**0.29**	**0.13**
57-IEEE	FNN	0.00	0.00	2.00	**1.00**	0.10	0.00	2.09	1.72	0.23
	CNN	0.00	0.00	**0.01**	5.00	0.05	0.00	1.74	0.87	0.11
	GCN	0.00	0.00	44.00	2.00	0.00	**0.00**	5.33	1.52	7.05
	HH-MPNN	0.02	0.00	2.00	5.00	0.36	0.04	**1.28**	**0.66**	**0.10**
118-IEEE	FNN	208.00	209.00	33.00	79.00	26.29	26.27	40.84	8.43	0.30
	CNN	**27.00**	29.00	**29.00**	**18.00**	**3.42**	**3.43**	**12.36**	**3.81**	1.16
	GCN	3071.00	3079.00	260.00	1266.00	19.69	19.72	26.77	10.79	8.55
	HH-MPNN	38.24	39.23	**3.91**	58.20	4.89	4.90	16.90	4.53	**0.10**
500-GOC	FNN	**73.00**	**73.00**	**0.87**	**64.00**	42.32	42.33	188.91	52.68	0.60
	CNN	123.00	123.00	26.00	114.00	**10.70**	**10.70**	62.16	**17.34**	1.15
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	104.18	104.19	24.08	81.10	28.54	28.70	**53.11**	22.52	**0.42**
2000-GOC	FNN	74.00	75.00	**27.00**	**140.00**	39.38	39.39	151.58	59.54	0.57
	CNN	121.00	121.00	57.00	254.00	37.70	37.70	114.24	**27.90**	**0.02**
	GCN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	HH-MPNN	**68.28**	**68.81**	39.65	198.00	**30.20**	**30.14**	**100.73**	41.86	0.39

Appendix B, and note that other feasibility projection techniques may also be employed to ensure full constraint satisfaction.

3.7. Theoretical limit of N-1 generalization

While HH-MPNN achieves stable zero-shot N-1 performance on most grids, the zero-shot generalization degrades for high-impact contingencies. This can be observed in the optimality gap of about 6.9% on the 118-bus Datakit dataset. The sharp degradation of the optimality gap for some test cases suggests the zero-shot transfer has a distributional limit. Here, we characterize the theoretical limit of such N-1 generalization. N-1 generalization of our proposed approach, or any graph-based model, can be seen as the stability of the learned mapping $f_\theta : G \to Y$ under a branch outage. Let $G = (V, \mathcal{E})$ be the default topology and $G' = (V, \mathcal{E}^-)$ be the N-1 topology. If f_θ is the

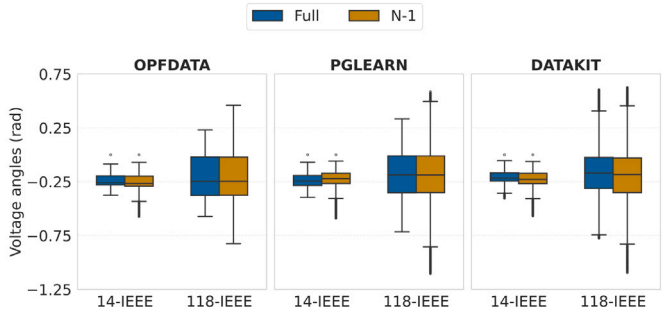


Fig. 4. Distribution of true voltage angles in the full-topology training data subset and N-1 topology testing data subset, for the 14 and 118-bus systems, across the 3 datasets.

neural network function and y^* is the true ACOPF solution, the N-1 generalization error of the neural network is bounded by:

$$\left\| f_\theta(G') - y^*(G') \right\| \leq \underbrace{\left\| f_\theta(G') - f_\theta(G) \right\|}_{\text{structural shift}} + \underbrace{\left\| f_\theta(G) - y^*(G') \right\|}_{\text{distributional shift}} \tag{6}$$

Notice that the first term, the structural shift, depends entirely on the neural network architecture and the function it learns, f_θ. The network f_θ may be designed such that $f_\theta(G') \approx f_\theta(G)$, thus the error due to the first term becomes small. However, the distributional shift cannot be accounted for by f_θ alone. For a well-trained model on the default grid, $f_\theta(G) \approx y^*(G)$. Thus, for a high-impact contingency where $\|y^*(G) - y^*(G')\|$ is large, the model error due to the distributional shift is large too. This is because the mapping y^* is non-smooth where the set of active constraints $\mathcal{A}$ changes [44]. If a high-impact outage e changes G such that $\mathcal{A}(G') \neq \mathcal{A}(G)$, the prediction error becomes high because f_θ extrapolates to an unseen scenario. This is what we observe empirically in the zero-shot experiment for the 118-bus system. Our results indicate that while the proposed HH-MPNN is structurally flexible and robust to structural shifts, its distributional robustness is limited by the active constraint sets $\mathcal{A}$ implicitly encountered during training.

3.8. Robustness through high-impact contingency training

To address the degradation in the optimality gap during N-1 generalization for some test cases, we develop a training strategy which uses the full topology dataset augmented by a few high-impact contingency samples from the N-1 training dataset. We selected the top 100 most expensive samples from the N-1 training dataset as measured by objective cost, representing <0.1% of the N-1 training subset. We use the most expensive samples by objective cost here as a proxy for the most severe contingencies. The model trained on this augmented full topology dataset is evaluated on the N-1 test dataset. This training approach reduced the average zero-shot optimality gap from **6.9%** to **1.4%** on the 118-bus Datakit dataset. Our findings reveal that exhaustive N-1 data generation is unnecessary for a topologically flexible model such as HH-MPNN; robust N-1 generalization requires only a small, targeted subset of high-impact N-1 scenarios to bridge the gap between topological flexibility and distribution shifts. In Table 7, we show the sensitivity of the model's optimality gap to the number of high-impact N-1 samples selected from the N-1 training dataset for augmented training with the full topology dataset. The optimality gap decreases monotonically as the number of N-1 samples increases. We selected 100 samples as it captures most of the reduction in the optimality gap while keeping the data generation cost low. We provide additional detail on the distribution of the N-1 contingencies for the 118-bus Datakit dataset in Appendix E. Where no N-1 dataset exists, identifying high-impact contingencies for targeted training can be done using existing security analysis tools or fast linear proxies. For example, the most severe contingencies often involve outages in areas which are typically heavily congested, low impedance tie-lines, and parallel circuits.

Table 7
Sensitivity of the optimality gap to the number of N-1 training samples used in the augmented training. Optimality gap decreases monotonically, but with diminishing returns. Increasing N-1 samples from 50 to 100 more than halves the gap, but a fivefold increase to 500 samples only reduces the gap by a further 0.4%.

No. of N-1 samples	Optimality gap (%)
50	3.35
100	1.36
300	0.95
500	0.89

Table 8
Average constraint violations and optimality gap for the proposed model trained from scratch (HH-MPNN-S), pretrained model without finetuning (HH-MPNN-NF) and pretrained model with finetuning (HH-MPNN-WF).

Model	Violations ($\times 10^{-4}$ pu)				Opt-gap (%)
	$S_{ij}(+)$	$S_{ij}(-)$	P_b	Q_b	
HH-MPNN-S	698.79	704.08	349.85	1595.00	1.40
HH-MPNN-NF	4117.27	4149.95	3396.74	10953.09	57.44
HH-MPNN-WF	**275.35**	**276.70**	**124.21**	**575.10**	**0.47**

3.9. Size generalization

Since HH-MPNN operates on power grids of arbitrary size, we investigate size generalization: whether models can generalize to larger unseen grids. We train a model on combined full topology Datakit training datasets from smaller (14, 30, 57, 118, and 500-bus) grids. We then test on the 2000-bus grid in 3 experiments: (1) Training from scratch with only 5% training data from the 2000-bus grid (HH-MPNN-S). (2) Pretraining on smaller grids and testing the 2000-bus grid with no finetuning. (HH-MPNN-NF). (3) Fine-tuning the pretrained model on the same 5% 2000-bus training samples (HH-MPNN-WF). Tables 8 show that HH-MPNN-WF achieves lower average constraint violations and 64% lower optimality gap (1.4% versus 0.5%) than HH-MPNN-S. HH-MPNN-NF leads to orders of magnitude worse predictions, showing the necessity of fine-tuning with a few samples from the target grid to learn scale-dependent features. We note that the model trained on the full training dataset still slightly outperforms HH-MPNN-WF, but HH-MPNN-WF achieves a comparable result with 95% less training data.

3.10. Ablation studies

3.10.1. Impact of transformer and positional encoding

We further investigate the effectiveness of the transformer with the effective resistance positional encoding component of the proposed architecture. To this end, we compare 5 experiments on the 118-bus full topology Datakit dataset. The first experiment uses the proposed model (HH-MPNN) as previously described. The second experiment uses the same heterogeneous message passing neural network (H-MPNN) but without the transformer and positional encoding. In the third experiment, we use the homogeneous graph representation with the topology-aware graph convolution neural network model (TAG-Conv) [45]. TAGConv is a more expressive GNN model than the GCN baseline. We also increase the number of graph convolution layers

Table 9
Average constraint violations and optimality gap for the proposed model (HH-MPNN), the same heterogeneous MPNN model without the transformer and effective resistance positional encoding (H-MPNN), a homogeneous TAGConv model with 14 layers, HH-MPNN* with node degree as PE and the transformer with no MPNN (Xformer).

Model	Violations ($\times 10^{-4}$ pu)				Opt-gap (%)
	$S_{ij}(+)$	$S_{ij}(-)$	P_b	Q_b	
HH-MPNN	**38.24**	**39.23**	3.91	**58.20**	0.10
H-MPNN	790.70	790.70	51.80	312.80	1.24
TAGConv-14	1207.00	1209.00	23.84	677.50	5.15
HH-MPNN*	81.00	83.00	**1.00**	80.00	0.34
Xformer	129.00	131.00	9.0	62.00	**0.05**

to 14, to match the diameter of the 118-bus graph. In theory, this makes it possible for the model to propagate information from any node throughout the graph. In the fourth experiment, we replace the proposed effective resistance-based PE with node degree in the proposed architecture (HH-MPNN*). In the final experiment, the MPNN component is removed. We only use the transformer and PE components of the proposed architecture (Xformer). We report the performance of the 5 models in Table 9. The proposed model outperforms the H-MPNN and TAGConv-14 models by a large margin, which shows the effectiveness of our approach. HH-MPNN* incurs significantly more constraint violations and worse optimality gap than HH-MPNN. Xformer can achieve a lower optimality gap than HH-MPNN by overfitting the fixed architecture. But this comes at the cost of increased constraint violations. It is noteworthy that increasing the number of layers of an expressive homogeneous GNN model still gives inferior performance. This can be attributed to the well-known oversquashing problem in GNN. In contrast, the heterogeneous graph representation mitigates some of the local representation bottleneck problems. Overall, the HH-MPNN model with transformer and effective-resistance positional encoding achieves the best balance between optimality gap and constraint violations by enabling long-range information propagation throughout the graph.

3.10.2. Impact of training data size

We also explore the impact of the training dataset size on the model performance. For this, we use the PGLearn 118-bus dataset due to the large number of available samples. The dataset is split into a training set and a testing set with approximately 800,000 and 200,000 samples, respectively. We further split the training set into a training and a validation set. We carry out 5 experiments using 5, 25, 40, 75 and 99% of the original training samples as training data and the rest as the validation set. We evaluate the trained model on the separate test dataset in each case. Fig. 5 shows the scaling behaviour of the proposed HH-MPNN model. Increasing the dataset improves constraint violations, in particular, the maximum or worst-case constraint violations. However, a moderate dataset of about 25 – 40% is already sufficient for near-optimal performance with all the constraints except Qb, which shows sensitivity to dataset distribution. Beyond a training sample size of about 300,000, diminishing performance returns are observed.

3.11. Analysing computational time

Table 10 compares solution times between IPOPT and HH-MPNN. IPOPT computational time scales as $\mathcal{O}(n^{0.3-1.9})$ with grid size, making real-time solution impractical for large grids. HH-MPNN maintains nearly constant inference time (about 13 ms) across all grid sizes, providing speedups of 10^3 to 10^4 compared to IPOPT. Beyond inference speedups, the graph-based approach reduces dataset generation costs. Zero-shot N-1 generalization eliminates the need to generate exhaustive N-1 training data, while size generalization enables pre-training on smaller grids with lower-cost data generation, reducing dataset requirements for new large systems. For example, the 15,000-sample fine-tuning dataset requires only 350 CPU hours to generate,

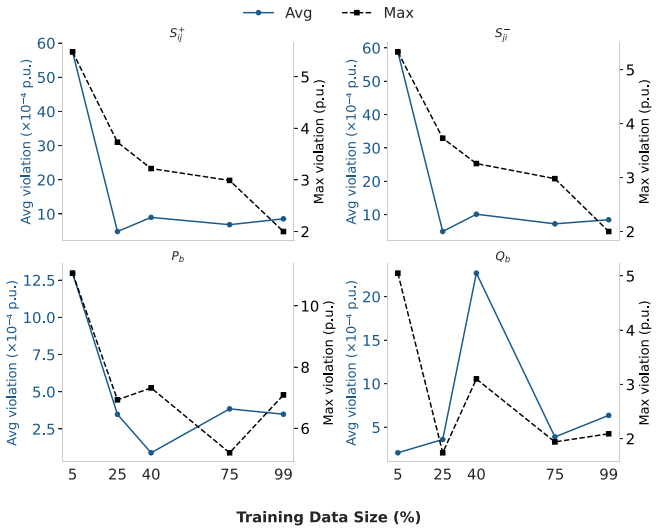


Fig. 5. Average and maximum constraint violations with different training dataset sizes on the PGLearn 118-bus dataset. About 25% of available data (200,000 samples) in this case is sufficient to give a near-optimal result.

Table 10
Average run time per instance comparing IPOPT and HH-MPNN, with computed average speed-up factors.

Average run time per instance ($\times 10^{-3}$ s)			
System	IPOPT	HH-MPNN	Average speed-up
14-ieee	33.2[a]	12.4	×3
30-ieee	102.3[a]	12.5	×8
57-ieee	328.8[a]	12.2	×27
118-ieee	420.7[a]	12.8	×32
500-goc	6486.0[b]	13.7	×473
2000-goc	83781.0[b]	14.7	×5700

[a] IPOPT timing results from [46].
[b] IPOPT timing results from [30].

compared to 6300 CPU hours for the full 270,000-sample training set, demonstrating substantial computational savings. HH-MPNN training for the largest 2000-bus system took about 163 h on a single Nvidia A30 GPU with 24 GB memory.

4. Discussions

The proposed HH-MPNN provides accurate but approximate ACOPF solutions across various grid sizes. For grids with fixed topology, our approach achieves performance comparable to FNN and CNN models while demonstrating the flexibility to handle N-1 topology changes that topology-specific architectures like FNN and CNN cannot. Our results challenge the assumption that graph-based models perform poorly when predicting generator setpoints and voltage angles. With sufficiently expressive architecture — combining heterogeneous graphs and transformers — both local and globally-dependent variables can be approximated accurately. We demonstrated topological flexibility through zero-shot prediction on N-1 topology variations and fine-tuning pretrained models on larger grids. We explore N-1 generalization limits, showing how distribution shifts affect model performance for operationally challenging contingency cases. Through this insight, we develop an approach for efficient and robust N-1 generalization through the inclusion of a few challenging contingency cases in the training dataset. As power systems are critical infrastructure, ML-based ACOPF approaches must be used with the appropriate safety measures in place. Our proposed HH-MPNN may be deployed for a fast approximate solution of the ACOPF problem. It may be used as a decision support

tool for operators or for warm-starting optimization solvers. Predictions still need to be verified or post-processed for full feasibility before operational use.

Limitations: We cannot guarantee satisfaction of all constraints. However, other feasibility restoring frameworks, such as physics-informed loss or differentiable constrained optimization [47, 48], may be combined with our proposed approach to satisfy all constraints. We have not considered more extreme contingencies, such as N-2 up to N-k, uncertain load demands due to extreme weather, renewable integration, etc., in this work. N-k and size generalization to even larger grids should be investigated. Pre-computation of effective resistance matrices can be computationally intensive for large grids. Computing the effective resistance matrices is $O(N^2)$ for sparse graphs such as those in power systems, where N is the number of nodes. For a fixed topology, this computation only needs to be done once. For N-1 changes which leave the graph connected, the matrix pseudoinverse can be quickly updated via the Sherman–Morrison formula for rank-one updates. For example, computing the ER matrix once for the 500-bus system using a *SciPy* sparse solver took 33 ms on a standard *Intel(R) Core i7 1265U 1.80 GHz* computer. Explainability remains an important aspect of operationally useful AI. Explaining the outputs of complex graph-based models such as HH-MPNN remains an ongoing work in the broader AI community. Our current approach uses HH-MPNN as a black-box approximator of ACOPF setpoints. Future work should consider adapting existing explainable AI frameworks, such as one proposed by [49], for ACOPF and other safety-critical power system applications. Another promising direction is replacing the MLP layers in our encoders, updater, and decoders with Kolmogorov–Arnold Network (KAN) layers [50], which use learnable spline-based activations and have shown improved parameter efficiency and interpretability. This may reduce model size while improving the explainability of the model predictions. Future work should explore hybrid ML-optimization approaches for feasibility guarantees and rigorous theoretical analysis of generalization bounds. Finally, future works would extend the proposed approach to multi-period, security-constrained and stochastic ACOPF problems.

5. Conclusion

We proposed a Hybrid Heterogeneous Message Passing Neural Network (HH-MPNN) for AC Optimal Power Flow that addresses topology adaptability and scalability challenges in existing machine learning approaches. Our architecture achieves $< 1\%$ optimality gaps on challenging test cases with variable loads, generator cost, line parameters and system topologies, while maintaining constraint violations comparable to state-of-the-art models. We show the strong generalization ability of the model through zero-shot transfer to N-1 contingency cases. Pre-training on smaller grids also improves performance on larger systems. Computational speedups of 10^2 to 10^3 compared to interior point solvers make the approach practical for real-time operations. The proposed approach may be used as a decision support tool for operators, and the favourable inference time makes practical integration with existing energy management systems possible, even when a feasibility restoration step is carried out after the model prediction. These results demonstrate the potential for unified, generalizable graph-based models for diverse power system applications.

CRediT authorship contribution statement

Olayiwola Arowolo: Writing – original draft, Software, Methodology, Investigation, Data curation, Conceptualization. **Jochen L. Cremer:** Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was funded by the Dutch Research Council, Veni Talent Program Grant 19161. We thank Ali Rajaei, Balthazar Donon, Jochen Stiasny, and Megha Khosla for insightful discussions during this work. We also thank Alban Puech for help with using the GridFM-Datakit tool.

Appendix A. Problem formulation

For a power system with N buses, $\mathcal{E}$ branches, G generators, L loads and S shunts, the ACOPF problem is defined as follows:

$$\min \sum_{i \in G} a(PG_i^2) + b(PG_i) + c \tag{7a}$$

$$PG_i^{min} \le PG_i \le PG_i^{max} \quad \forall i \in G \tag{7b}$$

$$QG_i^{min} \le QG_i \le QG_i^{max} \quad \forall i \in G \tag{7c}$$

$$V_i^{min} \le V_i \le V_i^{max} \quad \forall i \in N \tag{7d}$$

$$\theta_{ij}^{min} \le \theta_{ij} \le \theta_{ij}^{max} \quad \forall (i,j) \in \mathcal{E} \tag{7e}$$

$$|S_{ij}| \le S_{ij}^{max} \quad \forall (i,j) \in \mathcal{E} \tag{7f}$$

$$S_{ij} = (Y_{ij} + Y_{ij}^c)^* \cdot \frac{|V_i|^2}{|T_{ij}|^2} - Y_{ij}^* \frac{V_i V_j^*}{T_{ij}} \quad \forall (i,j) \in \mathcal{E} \tag{7g}$$

$$S_{ji} = (Y_{ij} + Y_{ij}^c)^* \cdot |V_j|^2 - Y_{ij}^* \frac{V_i^* V_j}{T_{ij}^*} \quad \forall (i,j) \in \mathcal{E} \tag{7h}$$

$$\sum_{k \in G} SG_k - \sum_{k \in L} SL_k - \sum_{k \in S} SS_k = \sum_{(i,j) \in \mathcal{E}} S_{ij} \quad \forall i \in N \tag{7i}$$

Here, a, b and c are the quadratic, linear and constant cost coefficients of generators, respectively. PG_i and QG_i are the generators' active and reactive powers, respectively. V_i is the voltage magnitude at a bus i, while θ_{ij} is the voltage angle difference between buses i and j respectively. S_{ij} is the complex power over branch (i,j), in the forward direction, S_{ji} in the complex branch power in reverse direction. SG_k is the complex power of generator k, SL_k is the complex power at load k and SS_k is the complex power at shunt k. Y_{ij} is the admittance of a branch, Y_{ij}^c is the charging admittance of the branch while T_{ij} is the complex transformation ratio of the branch.

Appendix B. Post processing model predictions with power flow

Power flow can be used to process the predictions from the ML models further, to ensure satisfaction of power balance constraints and make the obtained solution AC-feasible. However, we note that correcting the model predictions with power flows may now lead to violations of inequality constraints, such as voltage and power bounds. In the power flow formulation, only the slack bus has its active generation adjusted. This can lead to a violation of active power bound at the slack bus. Generations at other PV buses remain the same as predicted by the ML model. The reactive generation at PQ buses may be changed. We do not enforce reactive power limits; however, this can be achieved by converting PV buses to PQ buses at the reactive power limit. Table 11 shows the changes in constraint violations and the optimality gap after power-flow postprocessing on the 118-bus grid using the Datakit dataset.

Post-processing with power flow reduces active power balance violations to the precision of the numerical solver. The maximum active power balance violation after post-processing is in the order of 10^{-5} pu. However, the active and reactive generation limits may now be

Table 11
Constraint violations before and after power flow (PF).

	Constraint violation ($\times 10^{-4}$ p.u.)								
	$S_{ij}(+)$	$S_{ij}(-)$	P_b	Q_b	V	θ	P_G	Q_G	Opt. gap (%)
Pre PF	38.24	39.23	3.91	58.2	0.00	0.00	0.00	0.00	0.10
Post PF	1.00	1.00	0.00	591.0	0.06	0.00	3.0	492.0	0.04

Table 12
Comparison of the proposed HH-MPNN with recent GNN and physics-informed approaches.

Method – System	Metric	Value
CANOS – 2000 [30]	Average MSE	0.74×10^{-4} pu
HH-MPNN – 2000 (ours)		0.92×10^{-4} pu
TYPED-GNN – 118 [27]	Average MSE	4.77×10^{-2} pu
HH-MPNN – 118 (ours)		2.97×10^{-3} pu
PINCO – 118 [26]	Optimality gap	1.10%
HH-MPNN – 118 (ours)		0.10%

exceeded. The maximum active generation violation is 1.06 pu while the maximum reactive generation violation is 3.97 pu. Power flow post-processing also significantly reduces line flow violations. In total, the power flow step only takes 466 ms per scenario for the 118-bus grid, making it a scalable option for post-processing of the ML ACOPF predictions.

Appendix C. Datakit data generation

Load perturbation: We use the default aggregate load profile from EIA provided by Datakit. The aggregate load profile is scaled by ref_t, a global scaling factor. This factor is obtained by scaling the load profile within the range $[l, u]$, where u is obtained by incrementally increasing the active and reactive load from the nominal value by a factor of 0.1 until OPF does not converge. u is the last factor for which opf converges. $l = (1 - r)u$ where r is set to 0.4. The active and reactive load at any instance t is given by:

$$P_{i,t} = P_i \times ref_t \times E^p_{i,t} \tag{8a}$$

$$Q_{i,t} = Q_i \times ref_t \times E^q_{i,t} \tag{8b}$$

where $E^p_{i,t}, E^q_{i,t}$ are sampled from $\mathcal{U}(1-0.2, 1+0.2)$ for per-bus load variations Admittance perturbation: Each branch reactance and resistance is sampled from $\mathcal{U}(max(0, (1-0.2)), 1+0.2)$. Generator cost perturbation: Each generator's cost coefficient is perturbed by scaling the base cost by a factor sampled from $\mathcal{U}(1-0.5, 1+0.5)$ N-1 topology perturbation: For the N-1 dataset, the admittance perturbation is replaced by a topology perturbation, where a generator or branch may be selected at random to be dropped.

Appendix D. Additional baselines

We compare the proposed HH-MPNN to additional baselines from the literature in Table 12. We note that we have used reported results for baselines. We compared CANOS [30] to HH-MPNN, where both are evaluated on the 2000-bus OPFData. We also compare HH-MPNN to other recent physics-informed GNN variants, Typed-GNN [27] and PINCO [26], on the 118-bus grid. Reported results for Typed-GNN and PINCO are based on custom datasets used in the respective studies. We have compared this with our HH-MPNN variant trained on the 118-bus Datakit dataset. Full benchmarking of different ML models for ACOPF using comprehensive and consistent evaluation metrics, and high-quality open-source datasets remains an important future work.

Table 13
The proportion of branch outages and generator outages sum to 1. Each proportion is fairly consistent from the top 50 to the top 500 most severe N-1 contingencies.

Statistic	Complete	Top 50	Top 100	Top 300	Top 500
Proportion of branch outages	0.72	0.68	0.75	0.71	0.71
Proportion of gen outages	0.28	0.32	0.25	0.29	0.29
No. of unique branch outages	83	28	50	72	81
No. of unique gen outages	29	11	15	25	28

Appendix E. N-1 contingency analysis

We provide additional detail on the distribution of N-1 contingencies for the 118-bus Datakit dataset in Table 13. The dataset has a total of 112 unique component outages or contingencies. About 72% of the outages correspond to branch outages while 28% are generator outages. We compare the number of unique contingencies in the full N-1 dataset, to the number of unique contingencies in the top 50,100,300,500 most severe contingencies in the N-1 training data, ranked by the objective costs. It is interesting to observe that the top 100 contingencies already contain 65 of the 122 unique contingencies in the dataset, while the top 500 is missing only 3 unique contingencies. This highlights the fact that severe contingencies arise from a range of different outage scenarios for this grid and is not limited to a small subset of unique contingencies.

Data availability

Data will be made available on request.

References

[1] Cain MB, O'Neill RP, Castillo A. History of optimal power flow and formulations. 2012, Federal Energy Regulatory Commission (FERC).

[2] Khaloie H, Dolanyi M, Toubeau J-F, Vallée F. Review of machine learning techniques for optimal power flow. SSRN Electron J 2024.

[3] Panciatici P, Bareux G, Wehenkel L. Operating in the fog: Security management under uncertainty. IEEE Power Energy Mag 2012;10(5):40–9.

[4] Hamann HF, et al. Foundation models for the electric power grid. Joule 2024;8(12):3245–58.

[5] Shin S, Anitescu M, Pacaud F. Accelerating optimal power flow with GPUs: SIMD abstraction of nonlinear programs and condensed-space interior-point methods. Electr Power Syst Res 2024;236:110651.

[6] Pan X, Chen M, Zhao T, Low SH. DeepOPF: A feasibility-optimized deep neural network approach for AC optimal power flow problems. IEEE Syst J 2023;17(1):673–83.

[7] Yang Z, Zhong H, Bose A, Zheng T, Xia Q, Kang C. A linearized OPF model with reactive power and voltage magnitude: A pathway to improve the MW-only DC OPF. IEEE Trans Power Syst 2018;33(2):1734–45.

[8] Molzahn DK, Holzer JT, Lesieutre BC, DeMarco CL. Implementation of a large-scale optimal power flow solver based on semidefinite programming. IEEE Trans Power Syst 2013;28(4):3987–98.

[9] Chatzos M, Fioretto F, Mak TWK, Hentenryck PV. High-fidelity machine learning approximations of large-scale optimal power flow. 2020, arXiv:2006.16356 [eess].

[10] Baker K. Solutions of DC OPF are never AC feasible. In: Proceedings of the twelfth ACM international conference on future energy systems. Virtual Event Italy: ACM; 2021, p. 264–8.

[11] Zamzam AS, Baker K. Learning optimal solutions for extremely fast AC optimal power flow. In: 2020 IEEE international conference on communications, control, and computing technologies for smart grids. Tempe, AZ, USA: IEEE; 2020, p. 1–6.

[12] Park S, Chen W, Mak TW, Van Hentenryck P. Compact optimization learning for AC optimal power flow. IEEE Trans Power Syst 2024;39(2):4350–9.

[13] Huang W, Chen M, Low SH. Unsupervised learning for solving AC optimal power flows: Design, analysis, and experiment. IEEE Trans Power Syst 2024;39(6):7102–14.

[14] Owerko D, Gama F, Ribeiro A. Unsupervised optimal power flow using graph neural networks. In: ICASSP 2024 - 2024 IEEE international conference on acoustics, speech and signal processing. Seoul, Korea, Republic of: IEEE; 2024, p. 6885–9.

[15] Baker K. Learning warm-start points for ac optimal power flow. In: 2019 IEEE 29th international workshop on machine learning for signal processing. Pittsburgh, PA, USA: IEEE; 2019, p. 1–6.
[16] Crozier C, Baker K. Data-driven probabilistic constraint elimination for accelerated optimal power flow. In: 2022 IEEE power & energy society general meeting. Denver, CO, USA: IEEE; 2022, p. 1–5.
[17] Chatzos M, Mak TWK, Hentenryck PV. Spatial network decomposition for fast and scalable AC-OPF learning. IEEE Trans Power Syst 2022;37(4):2601–12.
[18] Falconer T, Mones L. Leveraging power grid topology in machine learning assisted optimal power flow. IEEE Trans Power Syst 2023;38(3):2234–46.
[19] Jia Y, Bai X, Zheng L, Weng Z, Li Y. ConvOPF-DOP: A data-driven method for solving AC-OPF based on CNN considering different operation patterns. IEEE Trans Power Syst 2023;38(1):853–60.
[20] Zhou M, Chen M, Low SH. DeepOPF-FT: One deep neural network for multiple AC-OPF problems with flexible topology. IEEE Trans Power Syst 2023;38(1):964–7.
[21] Liu M, Kong X, Xiong K, Wang J, Lin Q. Multi-scale spatio-temporal transformer: A novel model reduction approach for day-ahead security-constrained unit commitment. Appl Energy 2025;380:124963.
[22] Owerko D, Gama F, Ribeiro A. Optimal power flow using graph neural networks. In: ICASSP 2020 - 2020 IEEE international conference on acoustics, speech and signal processing. Barcelona, Spain: IEEE; 2020, p. 5930–4.
[23] Gao M, Yu J, Yang Z, Zhao J. A physics-guided graph convolution neural network for optimal power flow. IEEE Trans Power Syst 2024;39(1):380–90.
[24] Wu J, Sun C, Yang C. On the size generalizibility of graph neural networks for learning resource allocation. Sci China Inf Sci 2024;67(4):142301.
[25] Fan Y, Liu J, Tao Y, Sun Y, Li Z, Li J, Wang K, Ma J, Fan D, Xu L. A transfer-learning framework to alleviate data scarcity in cross-slope wind pressure modeling. Results Eng 2026;29:109376.
[26] Varbella A, Briens D, Gjorgiev B, D'Inverno GA, Sansavini G. Physics-informed GNN for non-linear constrained optimization: PINCO a solver for the AC-optimal power flow. 2024, arXiv:2410.04818 [eess].
[27] Lopez-Garcia TB, Domínguez-Navarro JA. Optimal power flow with physics-informed typed graph neural networks. IEEE Trans Power Syst 2025;40(1):381–93.
[28] Deihim A, Apostolopoulou D, Alonso E. Initial estimate of AC optimal power flow with graph neural networks. Electr Power Syst Res 2024;234:110782.
[29] Liu S, Wu C, Zhu H. Topology-aware graph neural networks for learning feasible and adaptive AC-OPF solutions. IEEE Trans Power Syst 2023;38(6):5660–70.
[30] Piloto L, et al. CANOS: A fast and scalable neural AC-OPF solver robust to N-1 perturbations. 2024, arXiv:2403.17660 [cs].
[31] Ghamizi S, Ma A, Cao J, Rodriguez Cortes P. OPF-HGNN: Generalizable heterogeneous graph neural networks for AC optimal power flow. In: 2024 IEEE power & energy society general meeting. Seattle, WA, USA: IEEE; 2024, p. 1–5.
[32] Baù M, Perbellini L, Grillo S. A principled framework to evaluate quality of AC-OPF datasets for machine learning: Benchmarking a novel, scalable generation method. 2025, arXiv:2508.19083 [eess].
[33] Li Y, Memon Z, Jin H, Fenu S, Song K, Sharma SB, Gasana P, Kim H, Zhao L, Kim K. LUMINA: Foundation models for topology transferable ACOPF. 2026, arXiv:2603.04300 [cs].
[34] Klamkin M, Tanneau M, Hentenryck PV. PGLearn – An open-source learning toolkit for optimal power flow. 2025, arXiv:2505.22825 [cs].
[35] Puech A, et al. Gridfm-datakit-v1: A python library for scalable and realistic power flow and optimal power flow data generation. 2025, arXiv:2512.14658 [cs].
[36] Rampášek L, Galkin M, Dwivedi VP, Luu AT, Wolf G, Beaini D. Recipe for a general, powerful, scalable graph transformer. 2023, arXiv:2205.12454 [cs].
[37] Battaglia PW, Pascanu R, Lai M, Rezende D, Kavukcuoglu K. Interaction networks for learning about objects, relations and physics. 2016, arXiv:1612.00222 [cs].
[38] Choromanski K, et al. Rethinking attention with performers. In: International conference on learning representations. Austria; 2021.
[39] Cetinay H, Kuipers FA, Van Mieghem P. A topological investigation of power flow. IEEE Syst J 2018;12(3):2524–32.
[40] Koç Y, Warnier M, Mieghem PV, Kooij RE, Brazier FM. The impact of the topology on cascading failures in a power grid model. Phys A 2014;402:169–79.
[41] Yehudai G, Fetaya E, Meirom E, Chechik G, Maron H. From local structures to size generalization in graph neural networks. 2021, arXiv:2010.08853 [cs].
[42] Cai C, Wang Y. A simple yet effective baseline for non attributed graph classification. In: The sixth international conference on learning representations. 2019.
[43] Lovett S, Zgubič M, Liguori S, Madjiheurem S, Tomlinson H, Elster S, Apps C, Witherspoon S, Piloto L. OPFData: Large-scale datasets for AC optimal power flow with topological perturbations. 2024.
[44] Misra S, Roald L, Ng Y. Learning for constrained optimization: Identifying optimal active constraint sets. 2019, arXiv:1802.09639 [math.OC].
[45] Du J, Zhang S, Wu G, Moura JMF, Kar S. Topology adaptive graph convolutional networks. 2018, arXiv:1710.10370 [cs].
[46] Fioretto F, Mak TW, Van Hentenryck P. Predicting AC optimal power flows: Combining deep learning and Lagrangian dual methods. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, (01):2020, p. 630–7.
[47] Nguyen HT, Donti PL. FSNet: Feasibility-seeking neural network for constrained optimization with guarantees. 2025, arXiv:2506.00362 [cs].
[48] Donti PL, Rolnick D, Kolter JZ. DC3: A learning method for optimization with hard constraints. In: International Conference on Learning Representations. Austria; 2021.
[49] Calzarossa MC, Giudici P, Zieni R. An assessment framework for explainable AI with applications to cybersecurity. Artif Intell Rev 2025;58(5):150.
[50] Liu Z, Wang Y, Vaidya S, Ruehle F, Halverson J, Soljacic M, Hou TY, Tegmark M. KAN: Kolmogorov arnold networks. In: The thirteenth international conference on learning representations. 2025.