Learning to Solve Optimal Power Flow with Graph Neural Networks

Yassine Guennoun
yguenn@stanford.edu

Hippolyte Wallaert
hippowal@stanford.edu

Edouard Rabasse
erabasse@stanford.edu

1 Motivation

Powergrid operators have to solve the **Optimal Power Flow (OPF)** problem repeatedly as loads and renewable outputs vary. Each solution determines generator set-points that meet demand at minimal cost while respecting nonlinear network constraints. However, conventional solvers are computationally intensive, limiting real-time control and market responsiveness.

To overcome this bottleneck, we propose training a **Graph Neural Network (GNN)** that learns the mapping from grid states to the OPF solution, including bus voltages and power flows. The learned model can provide near-instantaneous predictions or high-quality warm starts for classical solvers, combining the speed of learning-based inference with the physical rigor of traditional optimization.

2 Problem Definition

The **Optimal Power Flow (OPF)** problem seeks to determine generator outputs and bus voltages that minimize total generation cost while satisfying power balance and network constraints. It is a nonlinear and nonconvex optimization problem central to power system operation [1].

The problem can be formulated on a power network represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where buses correspond to nodes and transmission lines to edges. Each node $i \in \mathcal{V}$ is characterized by local electrical quantities: the net active power injection $P_i = P_{G_i} - P_{D_i}$, the net apparent power $S_i = \sqrt{P_i^2 + Q_i^2}$, and the voltage magnitude $|V_i|$. Each edge $(i, j) \in \mathcal{E}$ encodes the physical coupling between buses, described by the active and reactive power flows (P_{ij}, Q_{ij}), the line reactance X_{ij}, and the thermal capacity limit lr_{ij}. The OPF constraints enforce Kirchhoff's laws on this graph, coupling each bus i only to its neighbors $\mathcal{N}(i)$. Hence, the OPF task can be interpreted as learning the nonlinear operator mapping grid states to optimal power and voltage configurations over $\mathcal{G}$.

Aspect	**Specification**
Task type	Node-level regression on graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.
Input graph	$\mathcal{V}$: set of buses i with node features $\mathbf{x}_i = [P_i, S_i, \|V_i\|]$
	$\mathcal{E}$: set of transmission lines (i, j) with edge features $\mathbf{e}_{ij} = [P_{ij}, Q_{ij}, X_{ij}, lr_{ij}]$
Output variables	Predicted node-level quantities $\hat{\mathbf{y}}_i = [\hat{P}_{G_i}, \hat{Q}_{G_i}, \hat{V}_i, \hat{\theta}_i]$.

3 Dataset: PowerGraph (2024)

We use the PowerGraph (2024) dataset [1], a public benchmark designed for graph based learning in power systems. It includes standard IEEE test networks such as the IEEE 24, IEEE 39, and IEEE 118 bus systems, each simulated under thousands of random load and generation scenarios. Each instance provides detailed node and edge features and corresponding OPF solutions. The dataset is fully compatible with PyTorch Geometric via an `InMemoryDataset` interface.

[1] The mathematical definition of the problem is given in the Appendix

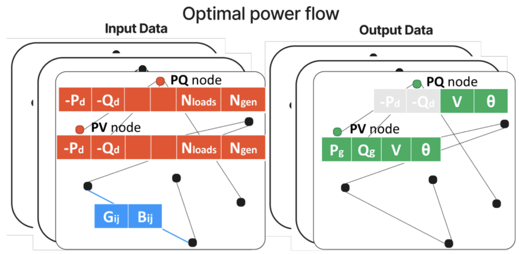

Figure 1: Instance of the PowerGraph dataset for optimal power flow. The input node features are in red, and output node-level predictions are in green (taken from [1])

4 Methodology

4.1 Graph ML Techniques and Model Choice

- **Model selection:** We plan to start with a Physics-Informed GNN based on the **GraphSAGE** architecture [2], due to its simplicity, scalability, and stable training on large graphs. We will later experiment with **PNA** [3] for richer neighborhood aggregation and better physical consistency.
- **Message passing:**

$$h_i^{(k+1)} = \sigma\Big(W^{(k)} \cdot \big[\, h_i^{(k)} \,\|\, \mathrm{AGG}\big(\{h_j^{(k)} : j \in \mathcal{N}(i)\}\big) \,\big]\Big)$$

  where $h_i^{(k)}$ is the node embedding at layer k, AGG is typically a mean or max operator and $W^{(k)}$ are learnable weights.

4.2 Training Objective and Physical Regularization

The training objective combines a data-driven regression term with a physics-based regularization term to ensure that predictions remain consistent with the physical laws governing power systems:

$$\mathcal{L} = \mathrm{RMSE}(P_g, \hat{P}_g) + \mathrm{RMSE}(Q_g, \hat{Q}_g) + \mathrm{RMSE}(V, \hat{V}) + \mathrm{RMSE}_{\mathrm{circ}}(\theta, \hat{\theta}) + \lambda \underbrace{\left(\sum_i \Delta P_i{}^2 + \Delta Q_i{}^2 \right)}_{\text{Kirchhoff's loss term}}.$$

The final term introduces a *physics-based regularization* that enforces Kirchhoff's Current Law (KCL) at each node of the power grid. This regularization does not impose Kirchhoff's laws as a hard constraint, but rather encourages the model to converge toward physically feasible solutions.

4.3 Why this model is appropriate

In the PowerGraph paper [1], the authors benchmarked standard GNNs (GCNConv [4], GATConv [5], GINEConv [6], and TransformerConv [7]) on their dataset without physics-based constraints. They tackled node-level regression and graph-level tasks. While TransformerConv and GINEConv performed well for classification, all models struggled with regression, highlighting the limitations of purely data-driven approaches for predicting physical quantities.

In contrast, physics-informed approaches in recent literature address this issue more directly. For instance, KCLNet [8] implements a hard projection of predicted node injections onto the feasible space defined by Kirchhoff's laws. Eeckhout et al. [9], on the other hand, train their model solely using a soft physics-informed loss that directly penalizes node-wise power imbalance. Their formulation is the first to incorporate real physical line losses (due to the Joule effect) into this imbalance calculation, which leads to better generalization.

In our work, we adopt a hybrid loss function that combines a data-driven term to ensure accurate prediction with a physics-informed regularization term enforcing Kirchhoff's laws. This formulation leverages both physical consistency and predictive performance, enabling a fair comparison across multiple GNN architectures. We plan to start with GraphSAGE due to its simplicity, inductive capability, and efficiency on large graphs.

References

[1] Anna Varbella, Kenza Amara, Blazhe Gjorgiev, Mennatallah El-Assady, and Giovanni Sansavini. PowerGraph: A power grid benchmark dataset for graph neural networks. Dataset available at `https://figshare.com/articles/dataset/PowerGraph/22820534`, October 2024. arXiv:2402.02827 [cs].

[2] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive Representation Learning on Large Graphs, September 2018. arXiv:1706.02216 [cs].

[3] Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. Principal neighbourhood aggregation for graph nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 13260–13271. Curran Associates, Inc., 2020.

[4] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[5] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[6] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

[7] Chao Shi, Bin Li, Muhan Li, Bo Wang, and Yao Sun. Masked label prediction: Unified message passing model for semi-supervised classification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[8] Azad Deihim, Dimitra Apostolopoulou, and Eduardo Alonso. Initial estimate of AC optimal power flow with graph neural networks. *Electric Power Systems Research*, 234:110782, September 2024.

[9] Victor Eeckhout, Hossein Fani, Md Umar Hashmi, and Geert Deconinck. Improved physics-informed neural network based ac power flow for distribution networks, 2024.

Appendix

The Optimal Power Flow problem can be written as follows :

$$
\begin{aligned}
\min_{\{P_G, Q_G, V, \theta\}} \quad & \sum_i C_i(P_{G_i}) \\
\text{s.t.} \quad & P_{G_i} - P_{D_i} = \sum_{j \in \mathcal{N}(i)} \frac{|V_i||V_j|}{X_{ij}} \sin(\theta_i - \theta_j), \\
& Q_{G_i} - Q_{D_i} = -\sum_{j \in \mathcal{N}(i)} \frac{|V_i||V_j|}{X_{ij}} \cos(\theta_i - \theta_j), \\
& |S_{ij}| = \sqrt{P_{ij}^2 + Q_{ij}^2} \leq lr_{ij}, \\
& V_i^{\min} \leq |V_i| \leq V_i^{\max}, \quad P_{G_i}^{\min} \leq P_{G_i} \leq P_{G_i}^{\max}.
\end{aligned} \tag{1}
$$

where $C_i(P_{G_i})$ is the generation cost function of generator i; P_{G_i} and Q_{G_i} denote the active and reactive power generated at bus i; P_{D_i} and Q_{D_i} are the active and reactive power demands; $|V_i|$ and θ_i are the voltage magnitude and phase angle at bus i; X_{ij} is the line reactance between buses i and j; and lr_{ij} represents the thermal capacity limit of the corresponding transmission line. The first two constraints enforce active and reactive power balance (Kirchhoff's laws), while the remaining constraints bound line flows, voltages, and generation capacities.