PFΔ: A Benchmark Dataset for Power Flow with Load, Generator, & Topology Variations

Ana K. Rivera Him *,1 Anvita Bhagavathula *,1 Alvaro Carbonero *,1 Priya L. Donti [1]

[1]Massachusetts Institute of Technology, *These authors contributed equally

Introduction and Background

- **Power flow analysis** is a key task for operating and planning power systems.
- The power flow problem solves a **nonlinear system of equations**:

$$\text{diag}(v)\bar{Y}\bar{v} = (p_g - p_d) + (q_g - q_d)j$$

where $v = Ve^{j\theta}$ is the complex voltage (magnitude V, angle θ), Y is the grid admittance matrix, and (p_g, q_g), (p_d, q_d) are power generation and demand vectors. **Problem inputs** are Y, p_g, p_d, q_d, and V at generator nodes, while **outputs** are all remaining variables.
- High uncertainty from weather events and renewables integration demands **more and faster scenario simulation.**
- **Newton-Rhapson (NR) is too slow** for these large-scale, real-time applications.
- **Machine learning (ML)-based proxies**, esp. those based on graph neural networks, offer a **faster alternative**, but **require improvements** in generalization and feasibility.
- ML models must be **evaluated on datasets that capture realistic, diverse grid conditions** and configurations, and benchmarked using standard, field-relevant metrics.

Main Contributions

- **PFΔ:** A benchmark dataset capturing variations in load, generation, grid size, and N-1 contingencies. (See Table 1.)
- **Comparative evaluation** of a traditional solver and two GNN-based models on proposed power flow tasks.

Table 1. Comparison of benchmark datasets (OPFData, OPFLearn), custom datasets used to train GNN models, and **PFΔ**.

Dataset	Load Profile	Generator Profile	Grid Sizes	N-1	> 1000 Buses
OPFData [1]	✓	×	✓	✓	✓
OPFLearn [2]	✓	×	✓	×	✓
GraphNeuralSolver [3]	✓	✓	✓	×	×
PowerFlowNet [4]	✓	✓	✓	×	✓
TypedGNN [5]	✓	✓	✓	✓	×
PFΔ	✓	✓	✓	✓	✓

Benchmark Dataset Generation

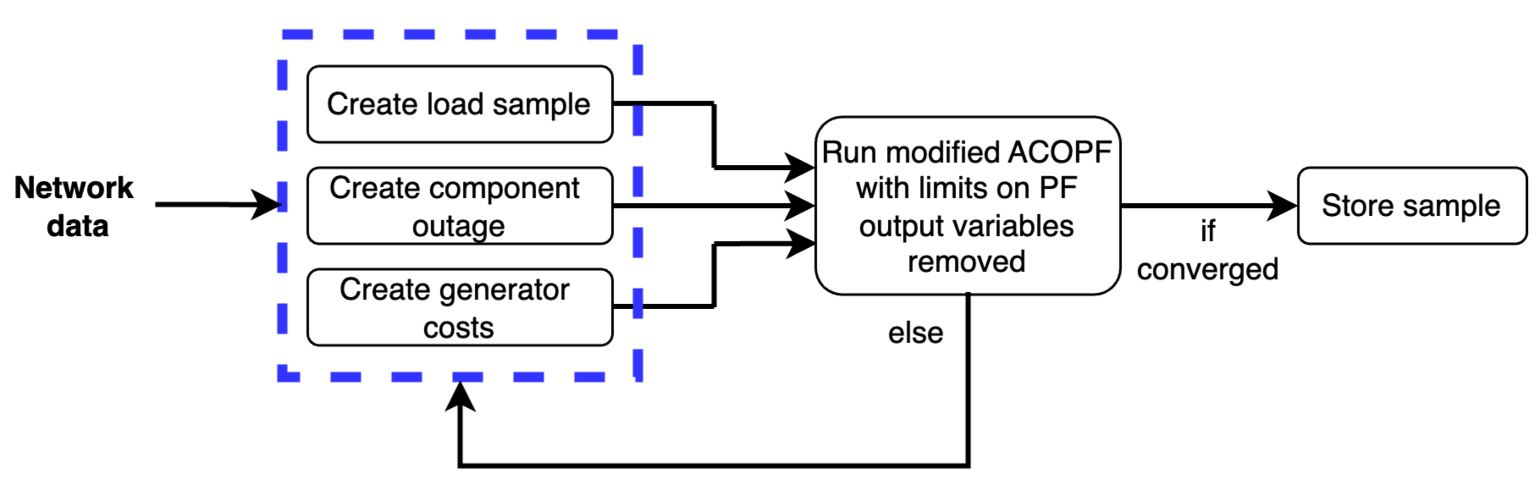

Figure 1. Data generation process in PFΔ

- **Load perturbations**: Active power demand is sampled from a uniform distribution within ± 20% of the nominal base case.
- **N-1 topological perturbations**: Each data sample is subject to one of the following equally likely events: removal of a branch, the removal of a generator, or no removal.
- **Generator cost perturbations**: Generator cost parameters are randomly permuted to induce diverse generator setpoints.
- The perturbed data sample is **run through a modified AC Optimal Power Flow (ACOPF)** solved with MATPOWER. If ACOPF converges, the sample is accepted.
- Our data generation process **introduces diversity in nodal variables that is better or comparable** to existing datasets, while capturing all perturbations of interest:

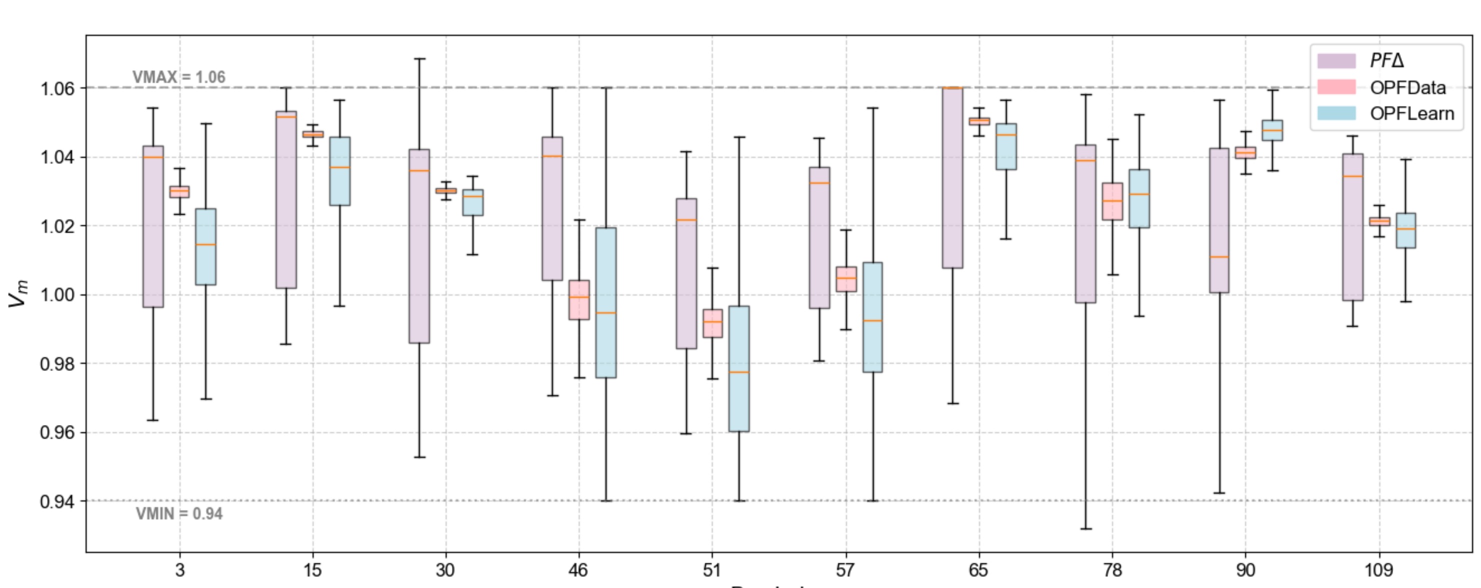

Figure 2. Box plots illustrating spread of V_m values in 10 randomly selected buses.This graphic compares feature diversity of V_m values sampled from $PF\Delta$ to other large-scale benchmark datasets.

Standardized Evaluation Tasks

We propose a set of standardized evaluation tasks to assess model performance across three regimes:

1. **In-Distribution Tasks**: Reflect scenarios where models trained on one network generalize to others of the same size—mirroring how grid operators deploy solvers tailored to specific grids.
2. **Data Efficiency Tasks**: Echo scenarios with limited historical data or distribution shifts due to evolving grid conditions.
3. **Out-of-distribution Generalization Tasks**: Capture cases where models are deployed on unfamiliar grid sizes, such as when a grid expands, or as general-purpose solvers.

Performance Metrics

We evaluate models on power balance loss (PBL) and on run time. Power balance loss measures the degree to which a model's prediction satisfies the power flow equations. We propose three ways to measure PBL:

$$PBL_\mu(\Delta S) = \frac{1}{N}\sum|\Delta S_i|, \quad PBL_{L2}(\Delta S) = \sqrt{\sum|\Delta S_i|^2}, \quad PBL_{\max}(\Delta S) = \max_i |\Delta S_i|$$

Where $\Delta S_i = \Delta P_i + j\Delta Q_i$ is the power mismatch at the i-th bus.
We compute per-instance sequential runtime for each method. For a dataset $\mathcal{D}$ split into batches $\mathbf{b}$, runtime is measured as follows:

$$\text{Runtime}(\mathcal{D}) = \frac{\sum_{\mathbf{b}} \text{Runtime}(\mathbf{b}) \cdot \text{size}(\mathbf{b})}{|\mathcal{D}|}.$$

Results and Evaluation

We evaluate three power flow solvers on our benchmark: two GNN-powered methods and one traditional iterative solver:

- **PowerFlowNet (PFNet)**: A supervised model that leverages TAGConv message passing layers [4].
- **TypedGNN (TGNN)**: A highly distilled version of the TypedGNN self-supervised model that trains to minimize power mismatches [5].
- **Newton-Raphson (NR)**: Traditional iterative power flow solver (MATPOWER).

We evaluate these models on a subset of our proposed evaluation tasks with a mean Power Balance Loss (PBL) metric to assess performance.

Table 2. Comparison of MATPOWER's Newton-Raphson (NR) solver, PowerFlowNet (PFNet), and TypedGNN (TGNN) on 3000 test samples, across various bus system sizes. GNN model performance is evaluated in low, medium, and high data efficiency regimes (LDE, MDE, HDE).

Model	PBL (118)	PBL (30)	PBL (2000)	Runtime (118)
NR (Warm-Start)	3.3e-6	1.1e-6	4.7e-4	1.3e-2
NR (Flat-Start)	3.3e-6	1.1e-6	Did not converge	1.5e-2
PFNet (LDE)	1.2e0	2.2e0	1.3e3	7.7e-3
PFNet (MDE)	1.3e0	5.6e0	1.7e3	7.6e-3
PFNet (HDE)	1.3e0	1.0e1	1.1e3	7.4e-3
TGNN (LDE)	**2.2e-1**	7.1e-1	1.1e1	6.8e-3
TGNN (MDE)	2.5e-1	7.9e-1	1.2e1	**6.4e-3**
TGNN (HDE)	5.4e-1	**5.0e-1**	**9.9e0**	7.0e-3

There are four key insights that our benchmark dataset and tasks provide, which lend themselves to key open questions that remain in the field.

- **Runtime improvements:** GNN-based models are significantly faster than Newton-Raphson (NR) solvers across all grid sizes. This speedup can be attributed to the parallelization capabilities of the hardware used to train the GNN models.
- **Self-supervised > supervised:** TGNN consistently outperforms PFNet on all tasks across all metrics, suggesting that embedding physical knowledge in the model improves generalization.
- **Feasibility remains an issue:** While ML models offer significant speed advantages, they struggle to produce feasible solutions, as shown by their far lower PBL metric values compared to those of the NR solver.
- **OOD generalization to larger grids:** Scaling to larger grid sizes without building extremely large and deep GNNs that don't suffer from oversmoothing is an open problem.

Further Work and Pathway to Impact

PFΔ provides a standardized evaluation framework to train and benchmark ML power flow solvers. Such a solver would support grid operators and planners in mitigating and adapting to climate change. Our current work is addressing the following gaps:

1. Include a broader and more diverse range of feasible load profiles.
2. Include close-to-infeasible cases that are challenging even for an NR solver.
3. Include topological perturbations that include $N-k$ contingencies.

[1] S. Lovett, M. Zgubič, S. Liguori, S. Madjiheurem, H. Tomlinson, S. Elster, C. Apps, S. Witherspoon, and L. Piloto, "OPFData: Large-scale datasets for AC optimal power flow with topological perturbations,"
[2] T. Joswig-Jones, K. Baker, and A. S. Zamzam, "OPF-Learn: An Open-Source Framework for Creating Representative AC Optimal Power Flow Datasets," in *2022 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, Apr. 2022.
[3] B. Donon, R. Clément, B. Donnot, A. Marot, I. Guyon, and M. Schoenauer, "Neural networks for power flow: Graph neural solver," *Electric Power Systems Research*, vol. 189, p. 106547, Dec. 2020.
[4] N. Lin, S. Orfanoudakis, N. O. Cardenas, J. S. Giraldo, and P. P. Vergara, "Powerflownet: Power flow approximation using message passing graph neural networks," *International Journal of Electrical Power Energy Systems*, vol. 160, p. 110112, 2024.
[5] T. B. Lopez-Garcia and J. A. Dominguez-Navarro, "Power flow analysis via typed graph neural networks," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105567, 2023.

{akrivera, abhagava, alvarocg, donti}@mit.edu