University of Houston
Department of Electrical and Computer Engineering

Faster Optimal Power Flow Using Graph Neural Network-Assisted Methods

Thuan Pham

December 3rd, 2024

Committee Members:

Dr. Xingpeng Li (Chair)
Dr. Kaushik Rajashekara
Dr. David Jackson
Dr. Harish Krishnamoorthy
Dr. Lei Fan

UH RPG Lab

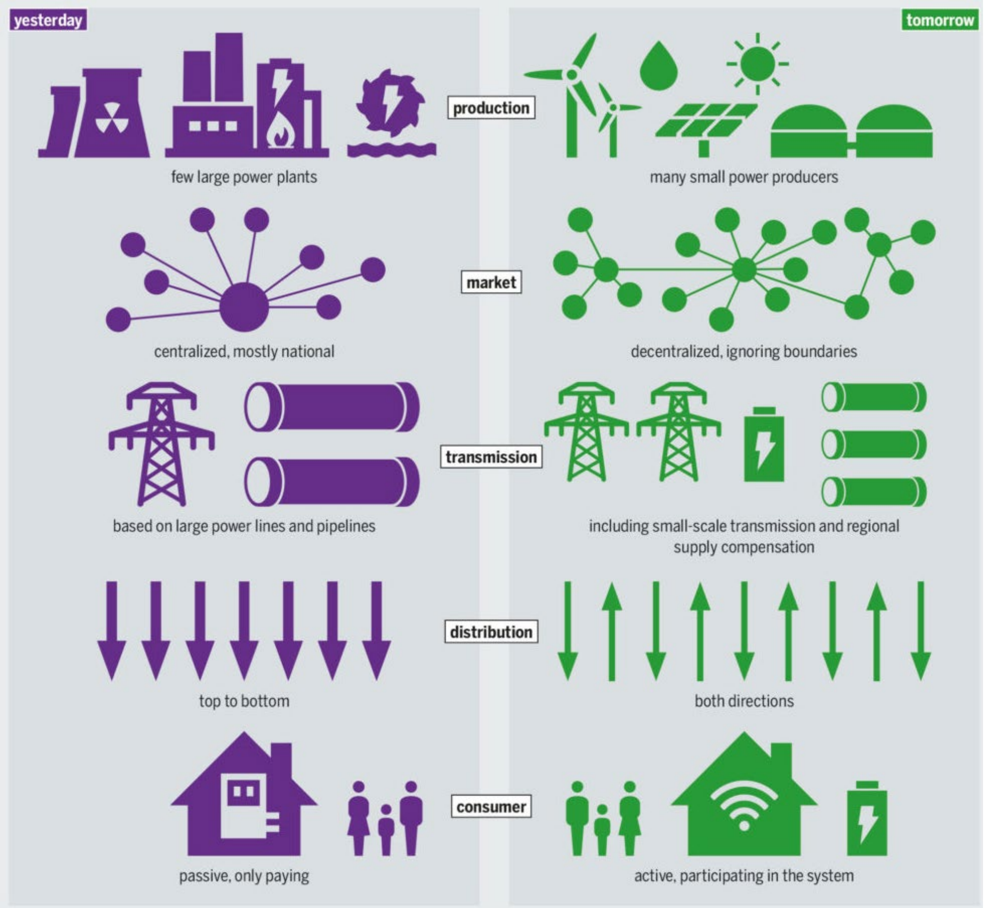
Overview

1. **Introductions**
   - Optimal Power Flow
   - Graph Neural Network
2. **Optimal Power Flow based on Graph Neural Network**
   - Reduced Optimal Power Flow
3. **N-1 Optimal Power Flow using Augmented Hierarchical Graph Neural Network**
   - Augmented Hierarchical Graph Neural Network
   - *N*-1 Reduced Optimal Power Flow
4. **Network Reconfigured Optimal Power Flow**
   - GNN-Accelerated Network Reconfigured Optimal Power Flow
   - Pre-ML Filter/Post-ML Selection
5. **Virtual Node-Splitting in Hierarchical Graph Neural Network for Optimal Power Flow**
   - Virtual Node-Splitting
6. **Conclusions & Future Work**

Chapter 1
Introduction

Power System Network

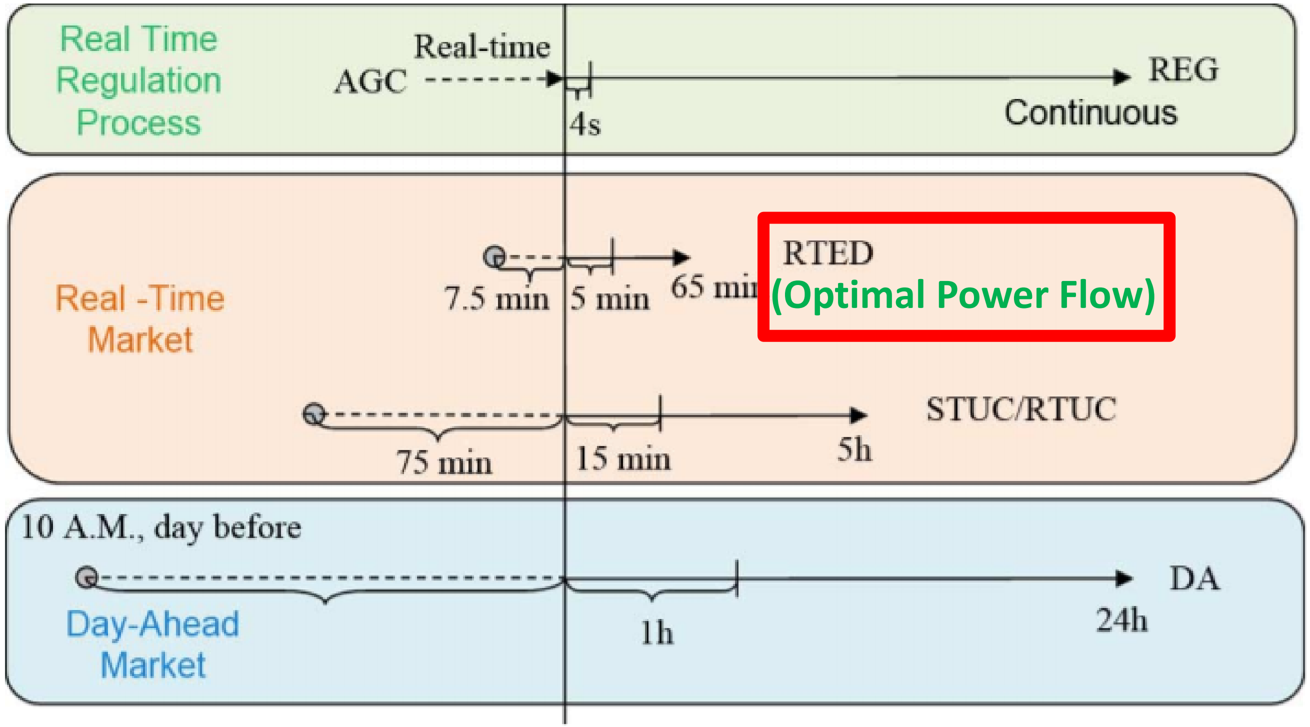
- Transitioned from a centralized model with power flowed unidirectionally to bidirectional exchange of power.
- Substantial congested lines due to increase in both generation and demand
  - Generation: increase in renewable project at both transmission and distribution level.
  - Demand: electrification of vehicle and data center investment.
- It is a challenge to operate the grid reliably and securely with a severely constrained network.



Source: K. Waehner, "The State of Data Streaming for Energy & Utilities," [Online]. Available: https://www.kai-waehner.de/blog/2023/09/01/the-state-of-data-streaming-for-energy-utilities-in-2023/.

Optimal Power Flow (OPF)

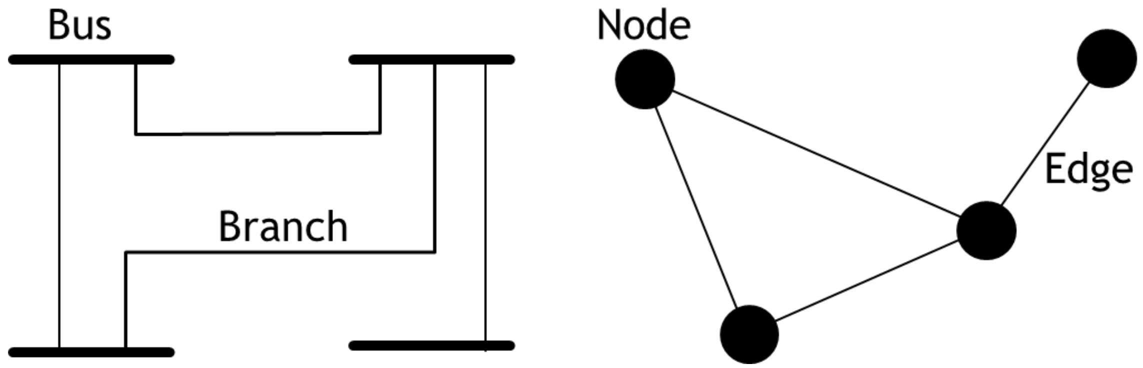
- To operate a reliable and secure grid, ISOs use different methods/strategies to control and regulate the electrical grid at different time scales.
- For real-time market, OPF determines the best operating points for online units while meeting demands and respecting other physical and reliability constraints.
- Due to the increased in congestion of the network, ***it is a challenge to solve OPF within a short timeframe for its real-time application.***



Source: Makarov, Y. V., Etingov, P. V., Ma, J., Huang, Z., & Subbarao, K. (2011). Incorporating uncertainty of wind power generation forecast into power system operation, dispatch, and unit commitment procedures. IEEE Transactions on Sustainable Energy, 2(4), 433-442.

Graph Neural Network (GNN)

- GNN is designed for network-structured data.
  - Utilize topology of the network as additional input features.
  - Usage of adjacency matrix (*nb* by *nb*)*.
  - Apply global context of the network during the training process.
- Applications of GNN in power system has been very limited.
  - Most GNN models focused on forecasting:
    - E.g., wind power prediction, or solar irradiation prediction.
  - Few GNN applications in the decision-making processes of power system.



Adjacency Matrix

0	1	1	0
1	0	1	0
1	1	0	1
0	0	1	0

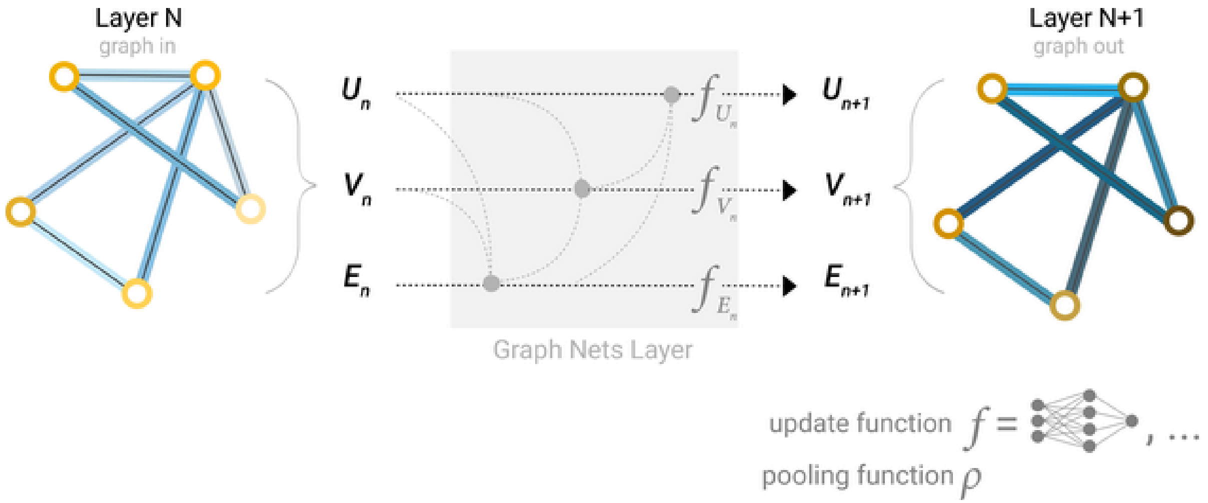
***nb = number of bus**

Graph Neural Network

- Forward propagation rule for GNN:

$$h_i^k = \sigma(W_{self}^k \cdot h_i^{k-1} + W_{neigh}^k \cdot AGG(h_j^{k-1}, \forall j \in \Omega_i))$$

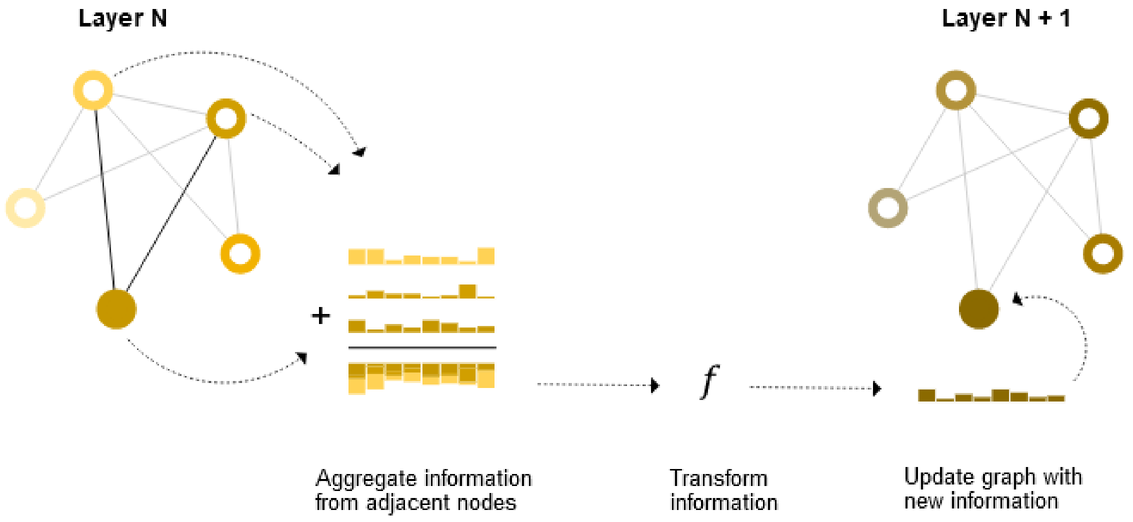
- The equation describes the relationship between each node and its neighboring nodes in each forward pass of the training stage.
  - where W_{self}^k and W_{neigh}^k are node-wise shared weight matrices
  - $\boldsymbol{AGG}(h_j^{k-1}, \forall j \in \Omega_i)$ is an aggregation function that combines feature information over all neighbor nodes (Ω_i)



Source: Sanchez-Lengeling, et al., "A Gentle Introduction to Graph Neural Networks", Distill, 2021.

Graph Neural Network

- GNN works with message passing between each layer of the neural network.
  – For each node in the graph, gather all the neighboring node/edge embeddings (messages)
  – Aggregate all messages via an aggregate function
  – All messages are passed through an update function



Source: Sanchez-Lengeling, et al., "A Gentle Introduction to Graph Neural Networks", Distill, 2021.

Research Roadmap

Introduction
- Optimal Power Flow
- Graph Neural Network

Optimal Power Flow
- Line congestion prediction using Graph Neural Network.
- Reduced Optimal Power Flow

***N*-1 Optimal Power Flow**
- Augmented Hierarchical GNN for line congestion prediction of the base case and *N*-1 cases.
- *N*-1 Reduced Optimal Power Flow

Network Reconfigured Optimal Power Flow
- GNN-Accelerated Network Reconfigured Optimal Power Flow
- Pre-ML and Post-ML processing

Virtual Node Splitting
- Virtual Node-Splitting in Hierarchical Graph Neural Network for Optimal Power Flow

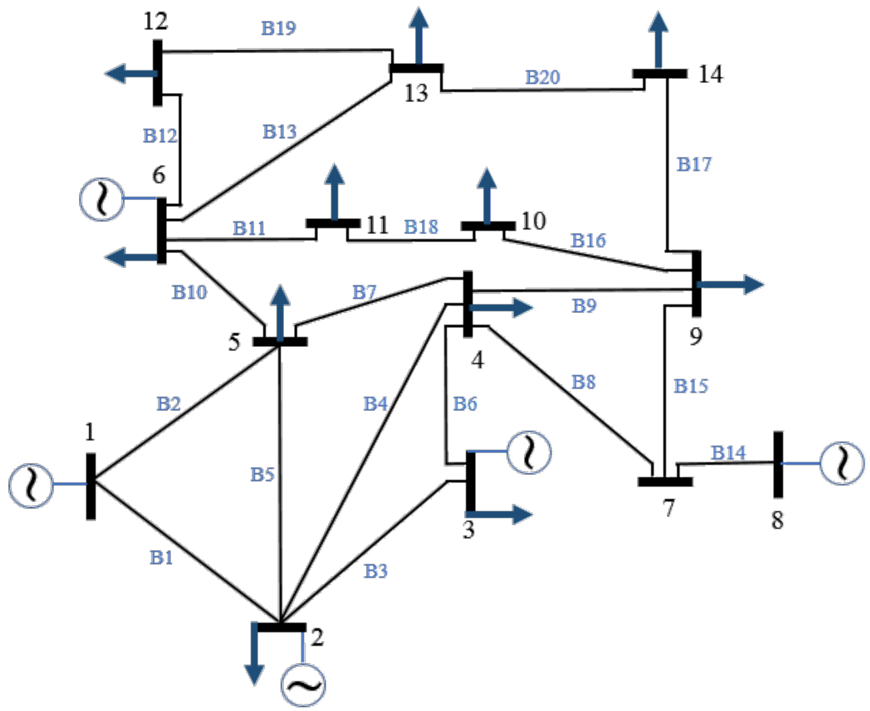
Conclusions & Future Work

Chapter 2
Optimal Power Flow based on Graph Neural Network

Optimal Power Flow (OPF)

- OPF is a constrained optimization problem that includes:
  - Variables (generation cost, load profiles…)
  - Constraints (line rating limit, generation limit…)
  - Objectives (minimum cost)
- It determines the dispatch of generating units to satisfy the electricity demand at the minimum cost while complying with the technical limits of the system.
- Computationally expensive for real time operations of the power system.
- **Simplify the original model to speed up computing time.**

Optimal Power Flow



- Objective Function:

$$min \sum_{g \in G} c_g P_g$$

- Constraints:

$P_g^{min} \leq P_g \leq P_g^{max}$	$, \ g \in G$	→ Generation Constraints
$P_k = (\theta_{f(k)} - \theta_{t(k)})/x_k$	$, \ k \in K$	→ Line Flow Equation
$-RateA_k \leq P_k \leq RateA_k$	$, \ k \in K$	→ Line Limit Constraints
$\sum_{g \in G(n)} P_g + \sum_{k \in K(n-)} P_k - \sum_{k \in K(n+)} P_k = d_n$	$, n \in N$	→ Nodal Balance Equation

G is a set of generators
K is a set of lines
N is a set of buses

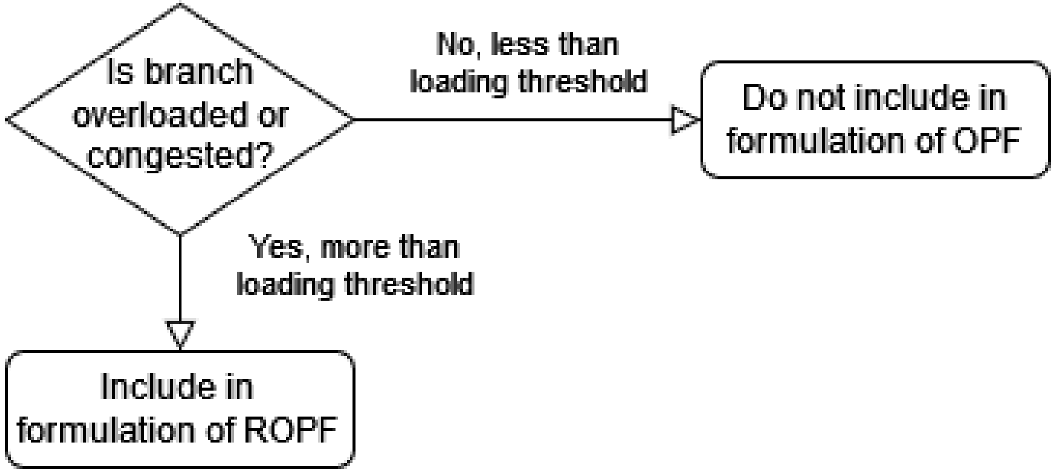
Loading Threshold for Congested Lines

- Define loading threshold as a percentage of the line rating limit.
- Based on the threshold, we will label these lines as heavily loaded or congested.
- Only need to monitor a subset of congested lines which leads to a reduced OPF problem while still maintaining solution quality.
- Ex: 80% threshold for line rating limit of 100 MW
- If line flow is above 80 MW => heavily loaded/congested
- If line flow is below 80 MW => normal

$$\cancel{-RateA_k \leq P_k \leq RateA_k \qquad k \in K}$$

$$-RateA_r \leq P_r \leq RateA_r \qquad r \in R$$

R is a set of heavily loaded or congested lines.
R is a subset of *K* that includes all lines.



Literature Review

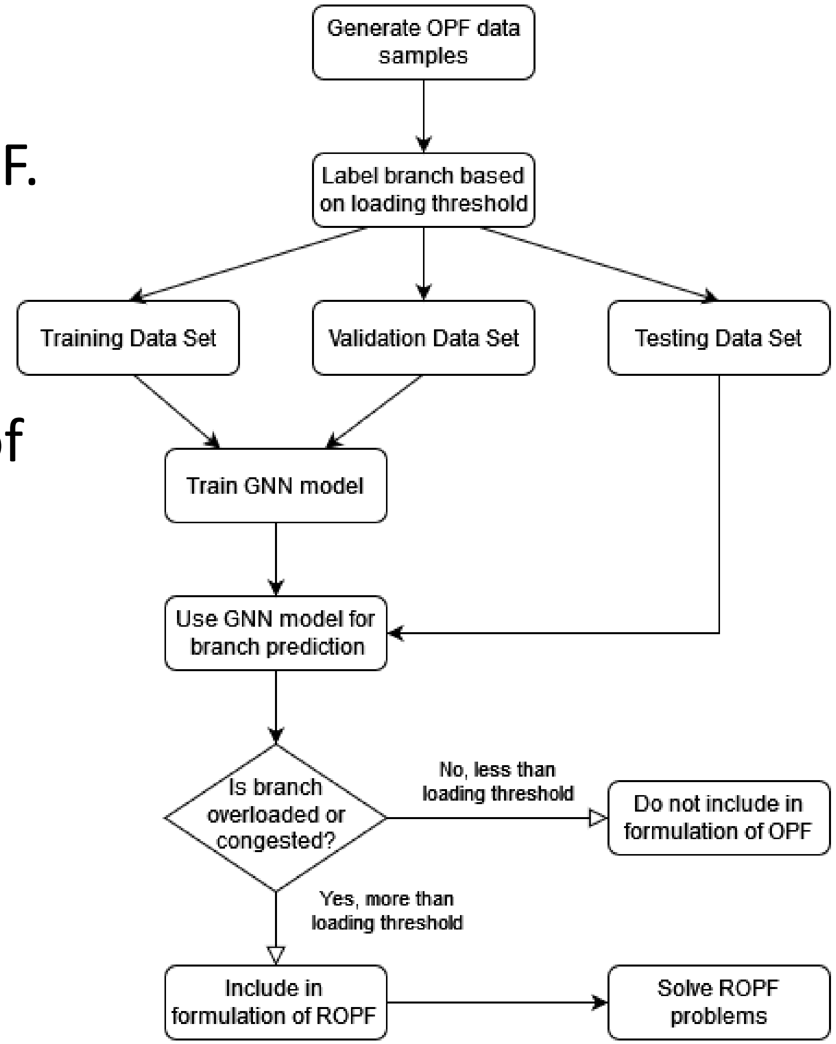
- Use GNN to directly predict the dispatch of the generators[1].
  - did not account for power flow and ignore the line rating limit in its formulation of the OPF problem.
- Convolutional neural network was built to predict generation dispatch using given load profiles [2].
- Neural network model was used to predict generations from load inputs for OPF [3].
  - Both [2] and [3] did not consider topology of the network result in lower performances from both models.

References:

1) D. Owerko, F. Gama and A. Ribeiro, "Optimal Power Flow Using Graph Neural Networks," in ICASSP, 2020.
2) K. Yang, W. Gao and R. Fan, "Optimal Power Flow Estimation Using One-Dimensional Convolutional Neural Network," in North American Power Symposium, College Station, 2021.
3) X. Pan, T. Zhao, M. Chen and S. Zhang, "DeepOPF: A Deep Neural Network Approach for Security-Constrained DC Optimal Power Flow," IEEE Transactions on Power Systems, vol. 36, no. 3, pp. 1725 - 1735, May 2021

Proposed Method for Reduced Optimal Power Flow (ROPF)

- Motivation:
  - Apply ML to improve/speed up computing time for OPF.
- Propose approach:
  - Use GNN to predict heavily loaded or congested lines.
  - Eliminate non-congested lines to reduce the number of variables & constraints in the original OPF model.
  - Less variables & constraints will lead to faster computing time.
  - Challenges:
    - Feasible solution
    - Accuracy of prediction in relation to total cost and constraints violation

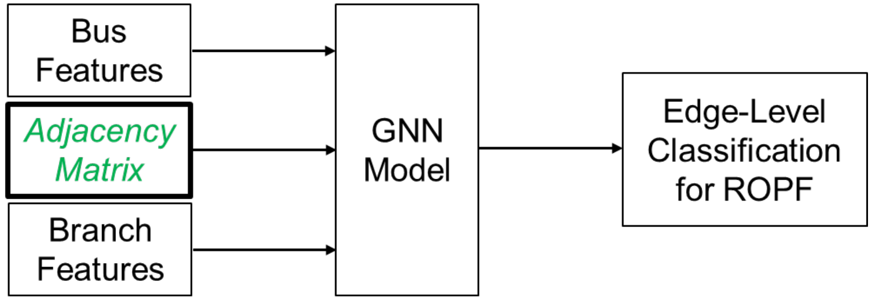


Flowchart for ROPF using GNN

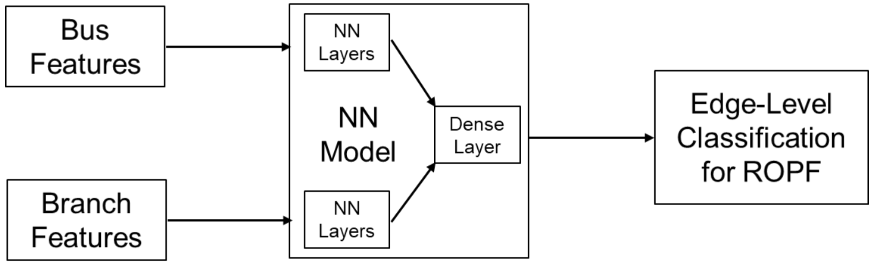
Machine Learning Models Comparison

- Evaluate effectiveness of **GNN** compared to typical ML models including:
  - Fully-connected Neural Network (**NN**).
  - Convolutional Neural Network (**CNN**).
- Analyze the results using different metrics, such as:
  - Computing time.
  - Percent error.
  - Constraint violation.

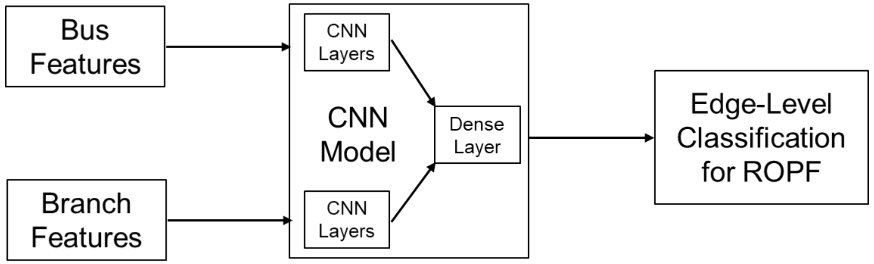
Proposed GNN Model



Benchmark NN Model

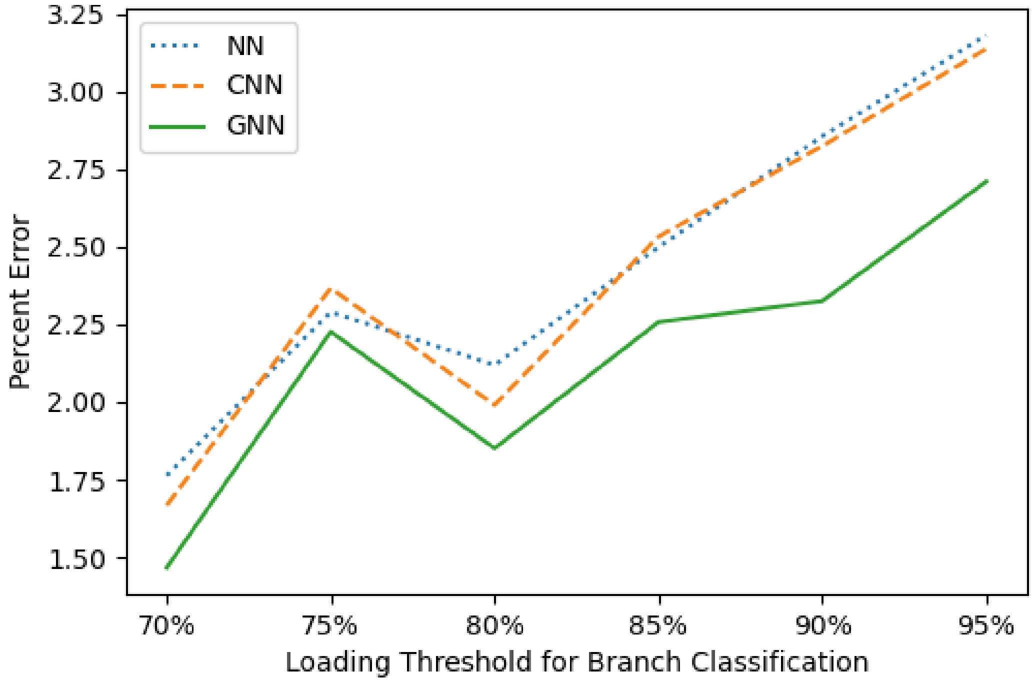


Benchmark CNN Model



Results: Percent Error

- At each threshold level, lines from samples are labeled as congested or not.
- **GNN outperforms both NN and CNN in terms of accuracy of its predictions at every level of loading threshold.**



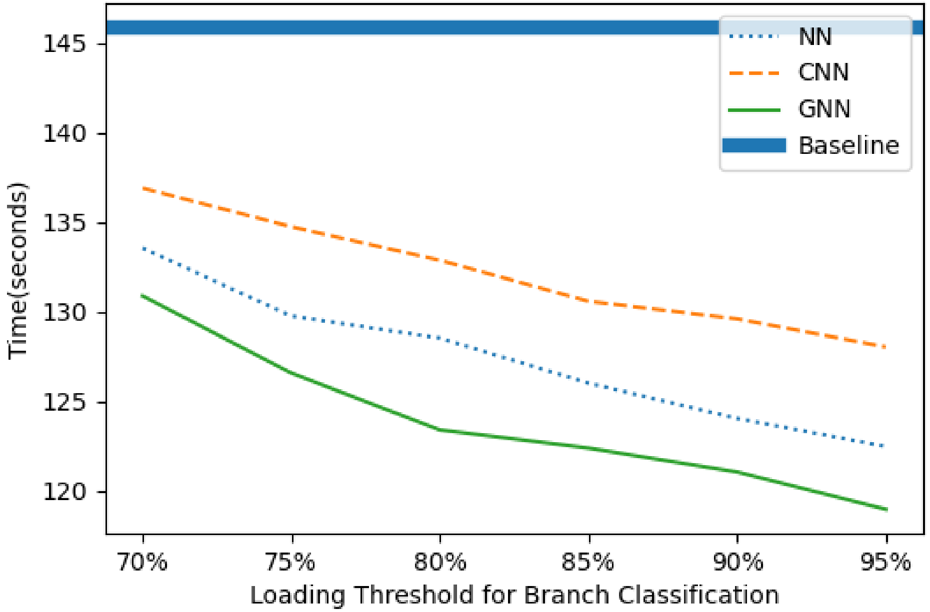
Comparison of percent errors at different loading threshold

	70%	75%	80%	85%	90%	95%
NN	1.76	2.29	2.12	2.50	2.86	3.18
CNN	1.67	2.37	1.99	2.53	2.82	3.14
GNN	1.47	2.23	1.85	2.26	2.33	2.71

- GNN has lower percent error compared to NN and CNN for each loading threshold.

Results: ROPF Solving Time

- Total time to solve 2,000 ROPF testing samples at multiple loading threshold using predictions from different ML models.



Solving time for 2000 ROPF problems using GNN at different loading threshold

GNN Model		Time (s)	Time (%)
Full OPF		145.87	100.00%
ROPF	70%	130.88	89.72%
	75%	126.57	86.77%
	80%	123.39	84.59%
	85%	122.38	83.90%
	90%	121.05	82.99%
	95%	118.97	81.56%

- GNN consistently outperforms NN and CNN across all loading threshold.
- **ROPF problem using predictions from GNN model has the fastest solving time.**

Results: ROPF Solving Time for GNN

- Comparison of average measurement metrics of 2,000 ROPF testing samples at multiple loading thresholds using GNN model for branch classification.

Threshold	Time (%)	% of Samples Over Limit	% of Lines Monitored	Prediction Error (%)
70%	89.72	0.2	27.27	1.47
75%	86.77	0.2	23.66	2.23
80%	84.59	0.25	20.55	1.85
85%	83.90	0.4	17.67	2.26
90%	82.99	9.65	15.17	2.33
95%	81.56	41.35	11.99	2.71

- The 95% threshold has the highest reduction in computing time, but over 40% of the samples did not meet the line limit constraints.
- **At 85% threshold, only 0.4% of the samples violate the line limit constraints with a 16% reduction computing time.**

Summary

- **Compared to NN and CNN, GNN is the best ML model for classifying branches that will likely be overloaded or congested.**
- Choosing the right loading threshold for branch classification is important.
- At 85% loading threshold, 16% reduction in computing time.
- Once trained (offline), the ROPF model will perform faster compared to full OPF model.

Publication:

- **Thuan Pham** and Xingpeng Li, “Reduced Optimal Power Flow Using Graph Neural Network”, *54th North American Power Symposium*, Salt Lake City, UT, USA, Oct. 2022.

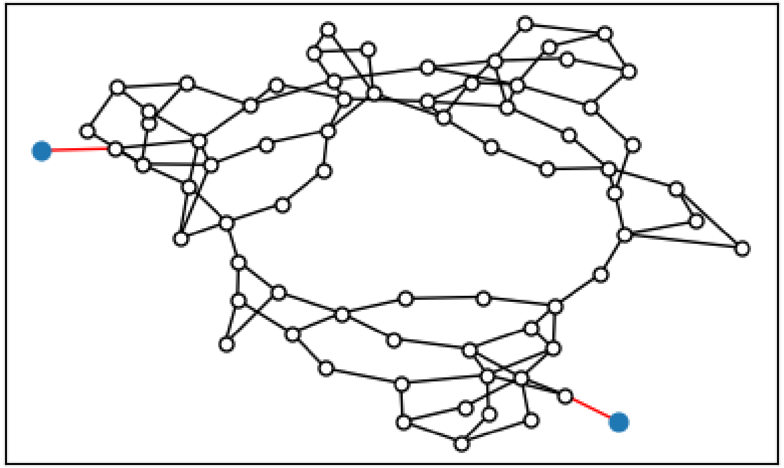
Chapter 3
N-1 Optimal Power Flow using Augmented Hierarchical Graph Neural Network

N-1 Optimal Power Flow (*N*-1 OPF)

- Modern power system must react quickly to contingency events: transmission outage, generator outage, transformer outage.
- Loss of one element (*N*-1) can significantly change operational condition of the system.
- Per NERC' guidelines, the power system must continue to operate in the normal state following the loss of one element (*N*-1).
- *N*-1 OPF evaluates system vulnerabilities by studying the outcomes of a single possible transmission outage events.

N-1 Optimal Power Flow (*N*-1 OPF)

- *N*-1 OPF performs contingency analysis by removing one element and assess its impact on violation of generation or transmission constraints.



- The goal is to minimize total generation cost while observing *N*-1 constraints.
- Operator must quickly scan through thousands of contingencies while avoiding islanding from the network.
- **Develop new strategies to solve *N*-1 OPF faster**

Optimal Power Flow

G is a set of generators
K is a set of lines
N is a set of buses

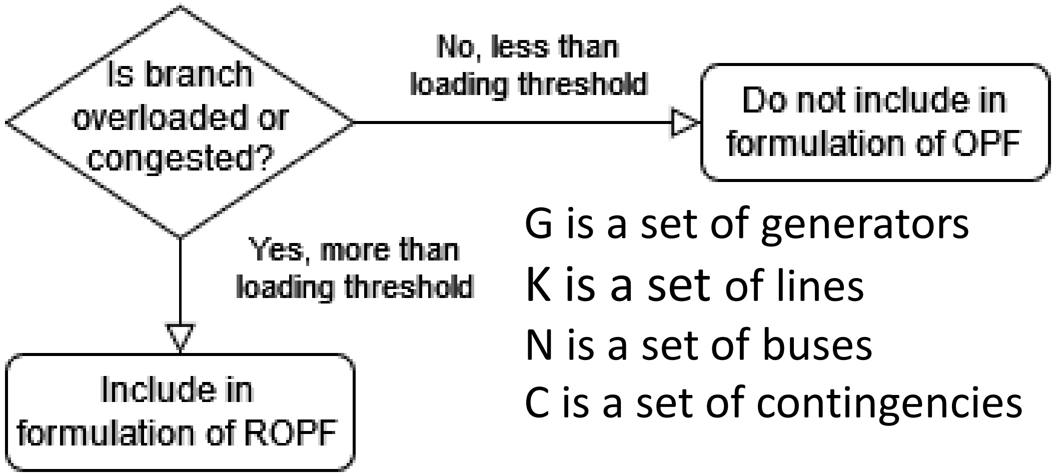
- Objective Function:

$$\min \sum_{g \in G} c_g P_g$$

- Constraints:

$\sum P_g^{res} = 0.05 \times \sum d_n$	$, g \in G$	$\longrightarrow$ Reserve
$0 \leq P_g^{res}$	$, g \in G$	$\longrightarrow$ Reserve
$P_g^{min} \leq P_g \leq P_g^{max} - P_g^{res}$	$, g \in G$	$\longrightarrow$ Generation Constraints
$P_k = \left(\theta_{f(k)} - \theta_{t(k)}\right)/x_k$	$, k \in K$	$\longrightarrow$ Line Flow Equation
$-RateA_k \leq P_k \leq RateA_k$	$, k \in K$	$\longrightarrow$ Line Limit Constraints
$\sum_{g \in G(n)} P_g + \sum_{k \in K(n-)} P_k - \sum_{k \in K(n+)} P_k = d_n$	$, n \in N$	$\longrightarrow$ Nodal Balance Equation

N-1 Optimal Power Flow



G is a set of generators
K is a set of lines
N is a set of buses
C is a set of contingencies

For *N*-1 cases, additional equations below:

- *N*-1 Constraints:

$P_{cg}^{min} \leq P_{cg} \leq P_{cg}^{max}$, $g \in G,\ c \in C$ → Generation Constraints

$-P_g^{ramp} \leq P_{cg} - P_g \leq P_g^{ramp}$, $g \in G,\ c \in C$ → Ramping Rate

$P_{ck} = \left(\theta_{f(ck)} - \theta_{t(ck)}\right)/x_k \times N1_{ck}$, $k \in K,\ c \in C$ → Line Flow Equation

$-RateC_k \leq P_{ck} \leq RateC_k$, $k \in K,\ c \in C$ → Line Limit Constraints

$\sum_{g \in G(n)} P_{cg} + \sum_{k \in K(n-)} P_{ck} - \sum_{k \in K(n+)} P_{ck} = d_n$, $n \in N,\ c \in C$ → Nodal Balance Equation

- Simplify Constraints:

$-RateA_r \leq P_c \leq RateA_r$, $r \in R$ → Subset of Line Limit Constraints

$-RateC_r \leq P_{cr} \leq RateC_r$, $r \in R,\ c \in C$ → Subset of Line Limit Constraints

R is a set of heavily loaded or congested lines.
R is a subset of *K* that includes all lines.

Literature Review

- Reduced Optimal Power Flow Using Graph Neural Network [1].
  - demonstrate that GNN can be used to calculate OPF.
- Unsupervised Optimal Power Flow Using Graph Neural Networks [2].
  - solutions of OPF contain violation constraints.
- Application of particle swarm optimization for power system operation considering N-1 contingency criteria [3].
  - did not consider computing time as one of its evaluation metrics.

References:

1. T. Pham and X. Li, "Reduced Optimal Power Flow Using Graph Neural Network," in *2022 North American Power Symposium (NAPS)*, Salt Lake City, 2022.
2. D. Owerko, F. Gama and A. Ribeiro, "Unsupervised Optimal Power Flow Using Graph Neural Networks," arXiv, 2022.
3. S. Thongkeaw, N. Rugthaicharoencheep and S. Auchariyamet, "Application of particle swarm optimization for power system operation considering N-1 contingency criteria," in 47th International Universities Power Engineering Conference (UPEC), Uxbridge, 2012.

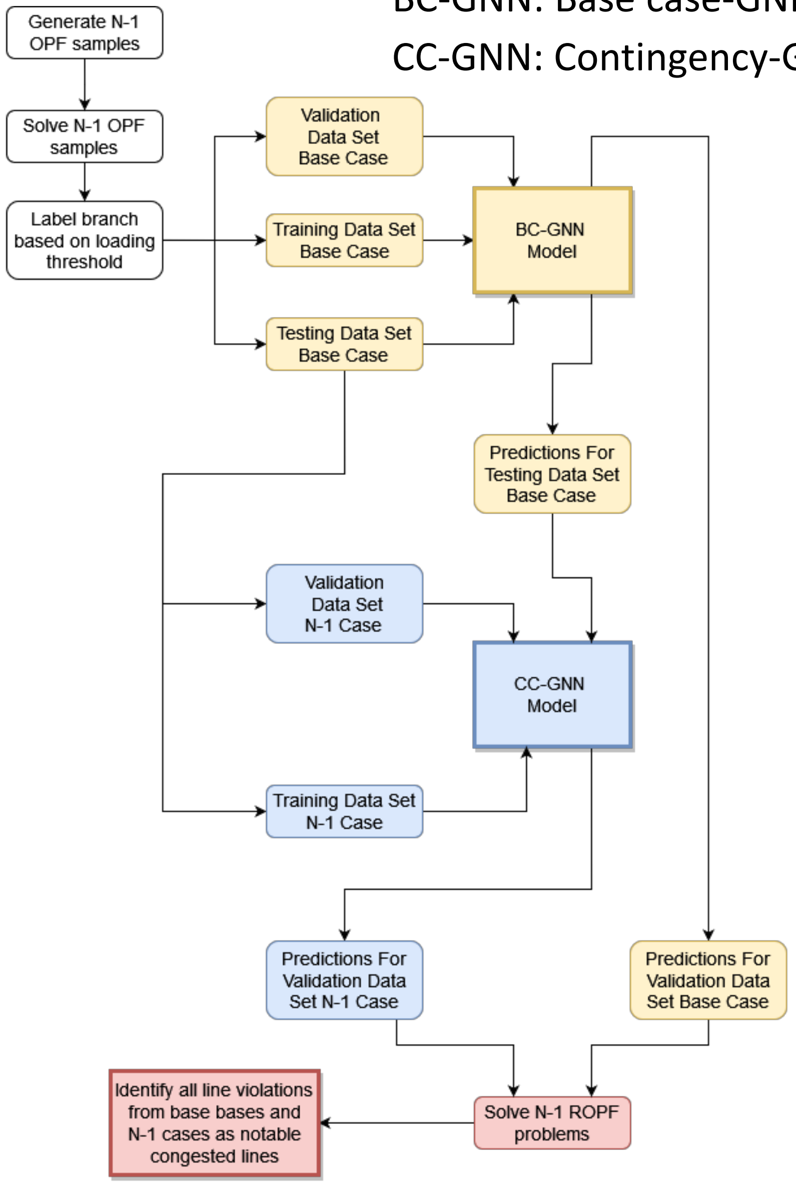
Proposed Method for *N*-1 ROPF

- Motivation:
  - Reduce computing time of *N*-1 OPF.
- Propose approach:
  - Using GNN to identify congested lines for the base cases and contingency cases.
  - Use data from the validation data set to identify notable congest lines.
  - Remove uncongested lines from the *N*-1 OPF
  - Challenges:
    - Feasible solution
    - Accuracy of prediction in relation to line constraint violation and total cost

OFFLINE – Training Mode

BC-GNN: Base case-GNN

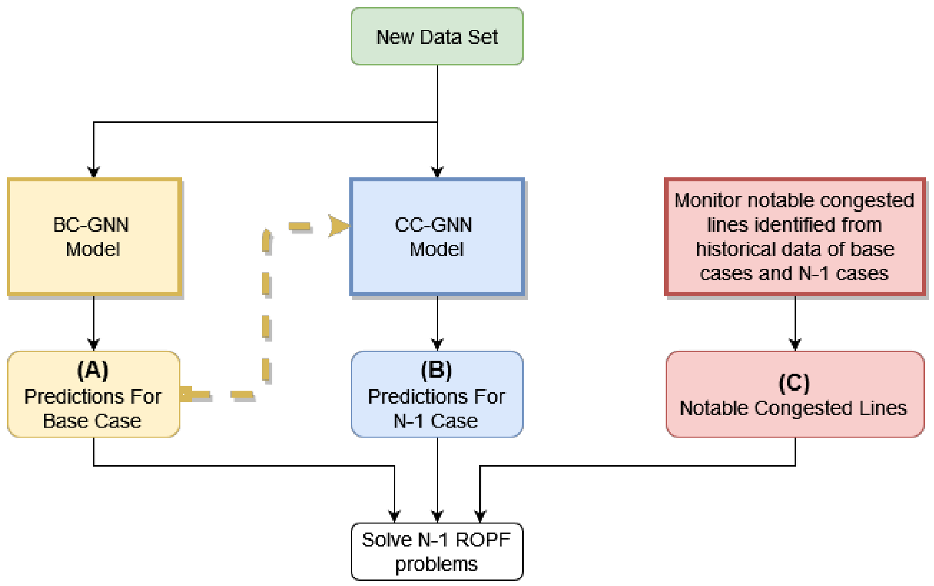
CC-GNN: Contingency-GNN



Five Proposed Models for Identification of Congested Lines

- **Notable Congested Lines [NCL]**: only use the notable congested lines identified offline [(C)].
- **Graph Neural Network [GNN]**: uses predictions from base case and *N*-1 cases, but without BC-GNN predictions as additional features for CC-GNN [(A) and (B)].
- **Augmented Graph Neural Network [AGNN]**: uses predictions from base case and *N*-1 cases [(A) and (B)], but without BC-GNN predictions as additional features for CC-GNN, along with notable congested lines [(C)].
- **Hierarchical Graph Neural Network [HGNN]**: uses predictions from base case and *N*-1 cases only, and with BC-GNN predictions as additional features for CC-GNN [(A) and (B)].
- **Augmented Hierarchical Graph Neural Network [AHGNN]** – the proposed method: uses predictions from base case and *N*-1 cases, and with BC-GNN predictions as additional features for CC-GNN, along with notable congested lines [(A), (B), (C)].

ONLINE – Prediction Mode



BC-GNN: Base case-GNN

CC-GNN: Contingency-GNN

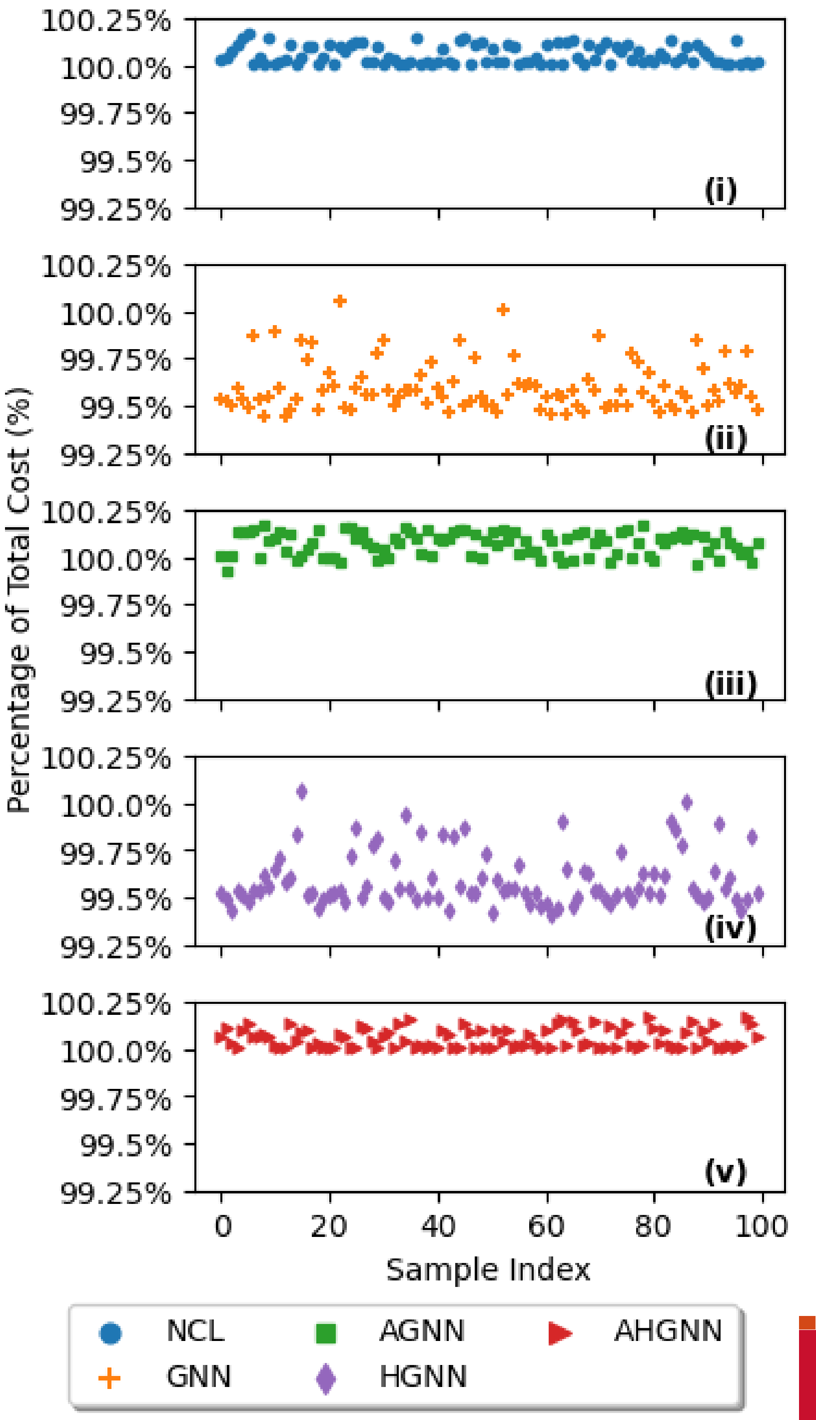
Results: Total Cost

- Majority of all models have a slight increase/decrease in the total cost of the objective function.
- **Mean for all total costs are within a margin error of $\pm$ less than 1%**

Statistical data for objective total cost for the five proposed models

Model	NCL	GNN	AGNN	HGNN	AHGNN
Mean	100.062	99.594	100.056	99.594	100.061
Median	100.043	99.551	100.046	99.551	100.043
Max	100.193	100.105	100.202	100.077	100.193
Min	100.002	99.365	99.919	99.365	100.002
Std.	0.052	0.135	0.06	0.135	0.051

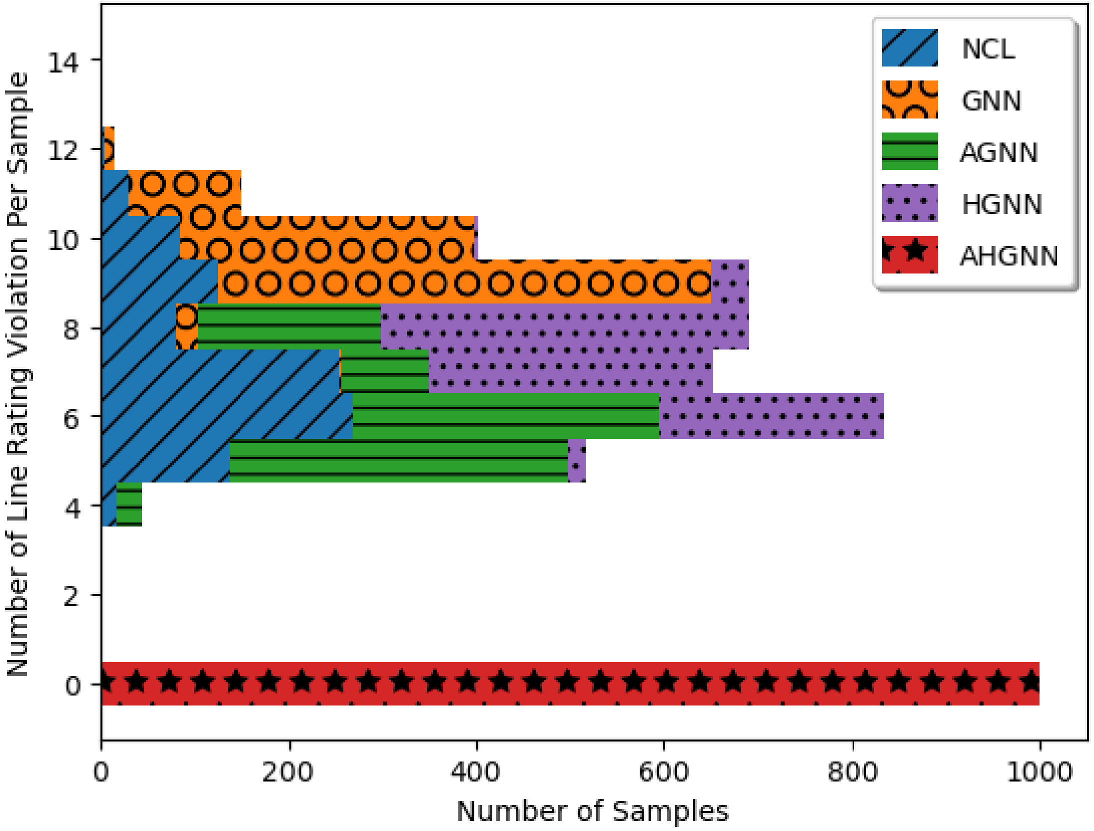
NCL: Notable Congested Lines
GNN: Graph Neural Network
AGNN: Augmented Graph Neural Network
HGNN: Hierarchical Graph Neural Network
AHGNN: Augmented Hierarchical Graph Neural Network



Results: Line Rating Violations

NCL: Notable Congested Lines
GNN: Graph Neural Network
AGNN: Augmented Graph Neural Network
HGNN: Hierarchical Graph Neural Network
AHGNN: Augmented Hierarchical Graph Neural Network

- **Only solutions from AHGNN model have zero line rating violations for all samples**



Comparison of the number of line rating violations for the five proposed models.

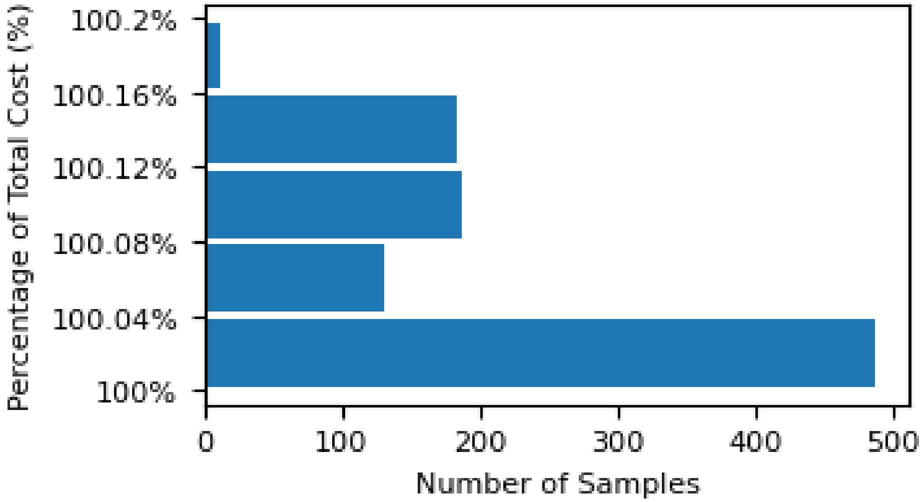
Line rating violations for the five proposed models per 1000 samples.

Number of Line Rating Violations Per Sample	NCL	GNN	AGNN	HGNN	AHGNN
0	**0**	**0**	**0**	**0**	**1000**
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	16	0	28	1	0
5	138	0	359	19	0
6	269	0	326	239	0
7	254	3	92	304	0
8	81	23	195	391	0
9	124	525	0	42	0
10	84	314	0	4	0
11	29	122	0	0	0
12	5	10	0	0	0
13	0	2	0	0	0
14	0	1	0	0	0
Total	1000	1000	1000	1000	1000

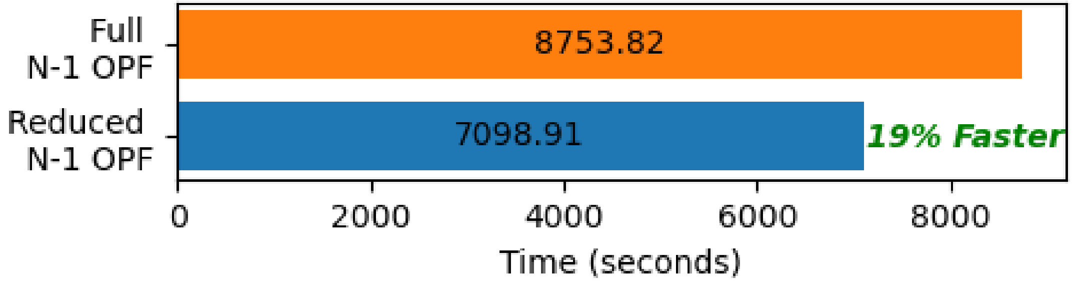
Results: Solving time for AHGNN

NCL: Notable Congested Lines
GNN: Graph Neural Network
AGNN: Augmented Graph Neural Network
HGNN: Hierarchical Graph Neural Network
AHGNN: Augmented Hierarchical Graph Neural Network

- Objective total cost for the proposed AHGNN increase by negligible **0.06%** with nearly 500 samples within **0.04%**.
- **Using AHGNN, solving time for 1000 samples with *N*-1 ROPF decrease by 19% compared to the Full *N*-1 OPF.**



Histogram of the total objective cost for AHGNN model



Comparison of solving time between full *N*-1 OPF and *N*-1 ROPF using AHGNN

NCL: Notable Congested Lines
GNN: Graph Neural Network
AGNN: Augmented Graph Neural Network
HGNN: Hierarchical Graph Neural Network
AHGNN: Augmented Hierarchical Graph Neural Network

Summary

- **AHGNN method provides the fastest solving time for optimal solution.**
- Proposed AHGNN method is very effective and accurate:
  - The predictions lead to *N*-1 ROPF solutions with zero line rating violations.
  - Total cost of objective function increases by less than 0.04% for *N*-1 ROPF.
- Proposed AHGNN method leads to enhanced computing efficiency:
  - Solving time decreases by 19% for a small-scale power system.
  - Expect a larger reduction in solving time for a larger system.

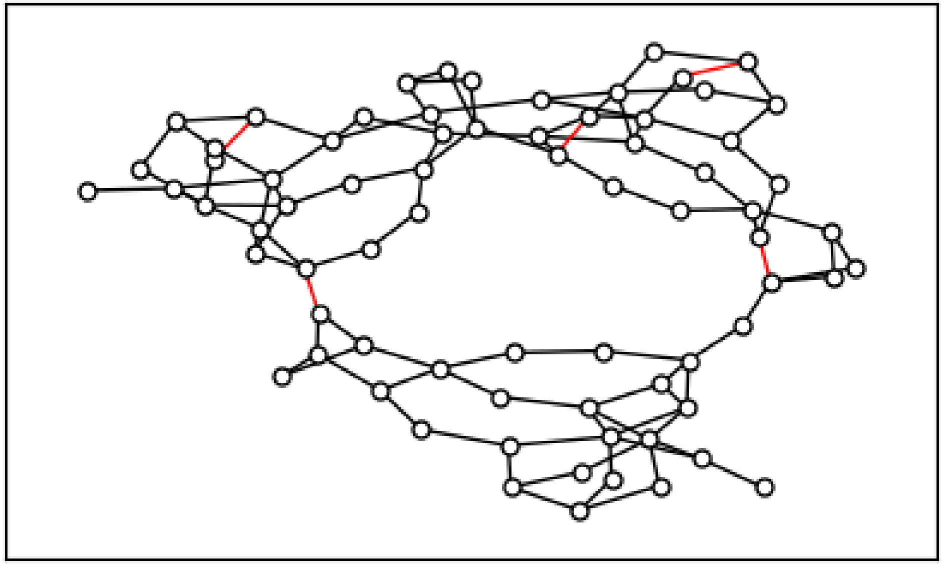
Publication:

- **Thuan Pham** and Xingpeng Li, "*N*-1 Reduced Optimal Power Flow Using Augmented Hierarchical Graph Neural Network", IEEE Transactions on Neural Networks and Learning Systems, (In Review).

Chapter 4
Network Reconfigured Optimal Power Flow

Network Reconfigured Optimal Power Flow (NR-OPF)

- Network reconfiguration strategically changes the network layout by flexibly switching transmission lines on or off.
- Allow grid operators to reduce network congestion by switching off congested lines, lead to improvement in economic efficiency and reliability.
- NR-OPF problem is solved using Mixed-Integer Linear Programming (MILP).
- Each transmission line represents a binary variable corresponding to the on/off status.
- MILP is computationally challenging problem, especially for larger system, due to the number of transmission lines.



GNN-Accelerated Network Reconfigured Optimal Power Flow (GaNR-OPF)

- For certain load profile, there is no solution using OPF, but utilizing network reconfiguration, a feasible solution can be found.
- Operators need to quickly scan all possible network reconfiguration options to find an optimal or feasible solution. Ex: 2^N where N is the number of reconfigured lines.
- Using GaNR-OPF to reduce the MILP problem into LP problem OR to reduce the number of network reconfiguration options to speed up computing time.

Optimal Power Flow

G is a set of generators
K is a set of lines
N is a set of buses

- Objective Function:

$$\min \sum_{g\in G} c_g P_g$$

- Constraints:

$\sum P_g^{res} = 0.05 \times \sum d_n$	$, g \in G$	$\longrightarrow$	Reserve
$P_g^{min} \le P_g \le P_g^{max} - P_g^{res}$	$, g \in G$	$\longrightarrow$	Generation Constraints
$0 \le P_g^{res}$	$, g \in G$	$\longrightarrow$	Reserve
$P_k = \left(\theta_{f(k)} - \theta_{t(k)}\right)/x_k$	$, k \in K$	$\longrightarrow$	Line Flow Equation
$-RateA_k \le P_k \le RateA_k$	$, k \in K$	$\longrightarrow$	Line Limit Constraints
$\sum_{g\in G(n)} P_g + \sum_{k\in K(n-)} P_k - \sum_{k\in K(n+)} P_k = d_n$	$, n \in N$	$\longrightarrow$	Nodal Balance Equation

Network Reconfigured Optimal Power Flow

G is a set of generators
K is a set of lines
N is a set of buses

- Objective Function:

$$\min \sum_{g \in G} c_g P_g$$

- Constraints:

$\sum P_g^{res} = 0.05 \times \sum d_n$	$, g \in G$	Reserve
$P_g^{min} \leq P_g \leq P_g^{max} - P_g^{res}$	$, g \in G$	Generation Constraints
$0 \leq P_g^{res}$	$, g \in G$	Reserve
$-BigM(1 - NR_k) \leq P_k - \left(\theta_{f(k)} - \theta_{t(k)}\right)/x_k \leq BigM(1 - NR_k)$	$, k \in K$	Line Flow Equation
$-NR_k * RateA_k \leq P_k \leq RateA_k * \boxed{NR_k}$	$, k \in K$	Line Limit Constraints
$\sum_{g \in G(n)} P_g + \sum_{k \in K(n-)} P_k - \sum_{k \in K(n+)} P_k = d_n$	$, n \in N$	Nodal Balance Equation

NR_k is a binary variable that determines the line status ON/OFF associated with network reconfiguration

Literature Review

- Network reconfiguration has been explored as an option to reduce operational cost and increase grid flexibility [1].
- Reinforcement learning and GNN has been used to speed up NR-OPF [2]
  - still utilize on heuristic approach to develop a ML model
- Develop a heuristics algorithm to solve NR-OPF [3]
  - proposed model did not produce precise prediction based on different topologies under given load profiles.

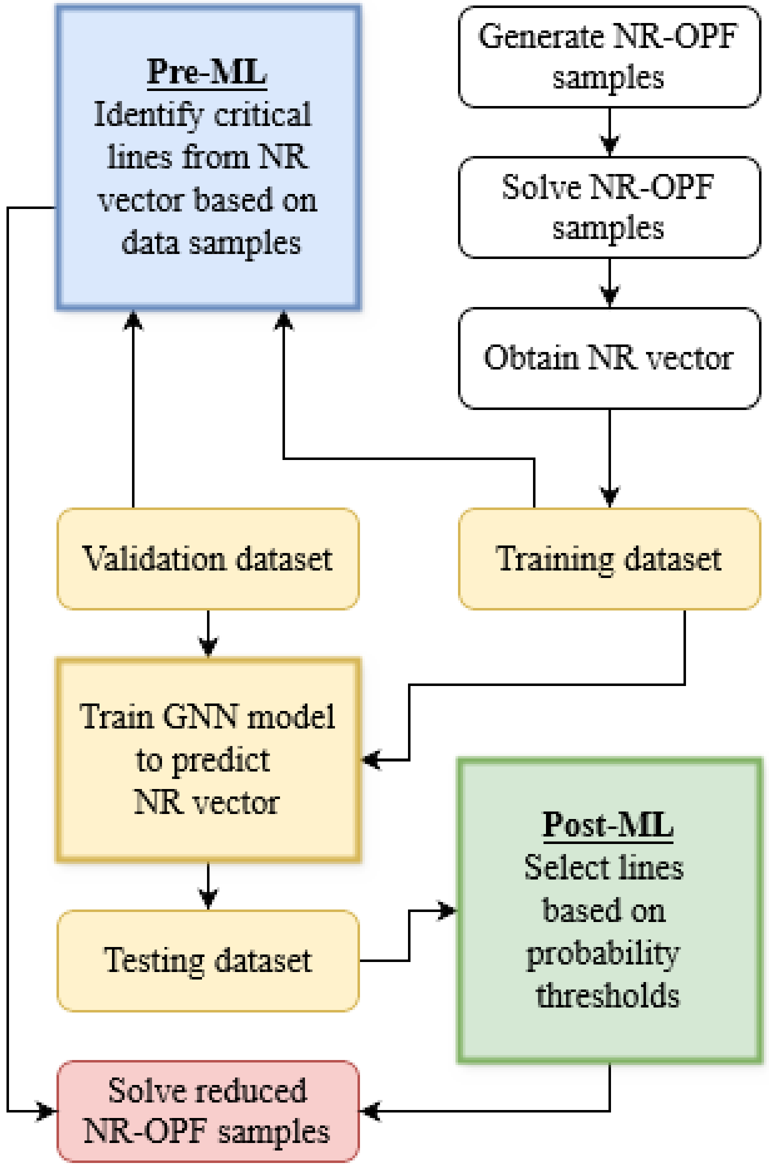
References:

1. M. Numan, M. F. Abbas, M. Yousif, S. S. M. Ghoneim, A. Mohammad and A. Noorwali, "The Role of Optimal Transmission Switching in Enhancing Grid Flexibility: A Review," in *IEEE Access*, vol. 11.
2. Crozier, C., Baker, K., & Toomey, B. (2022). Feasible region-based heuristics for optimal transmission switching. Sustainable Energy, Grids and Networks, 30, 100628.
3. T. Han and D. J. Hill, "Learning-Based Topology Optimization of Power Networks," in *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1366-1378, March 2023.

Proposed Method for GaNR-OPF

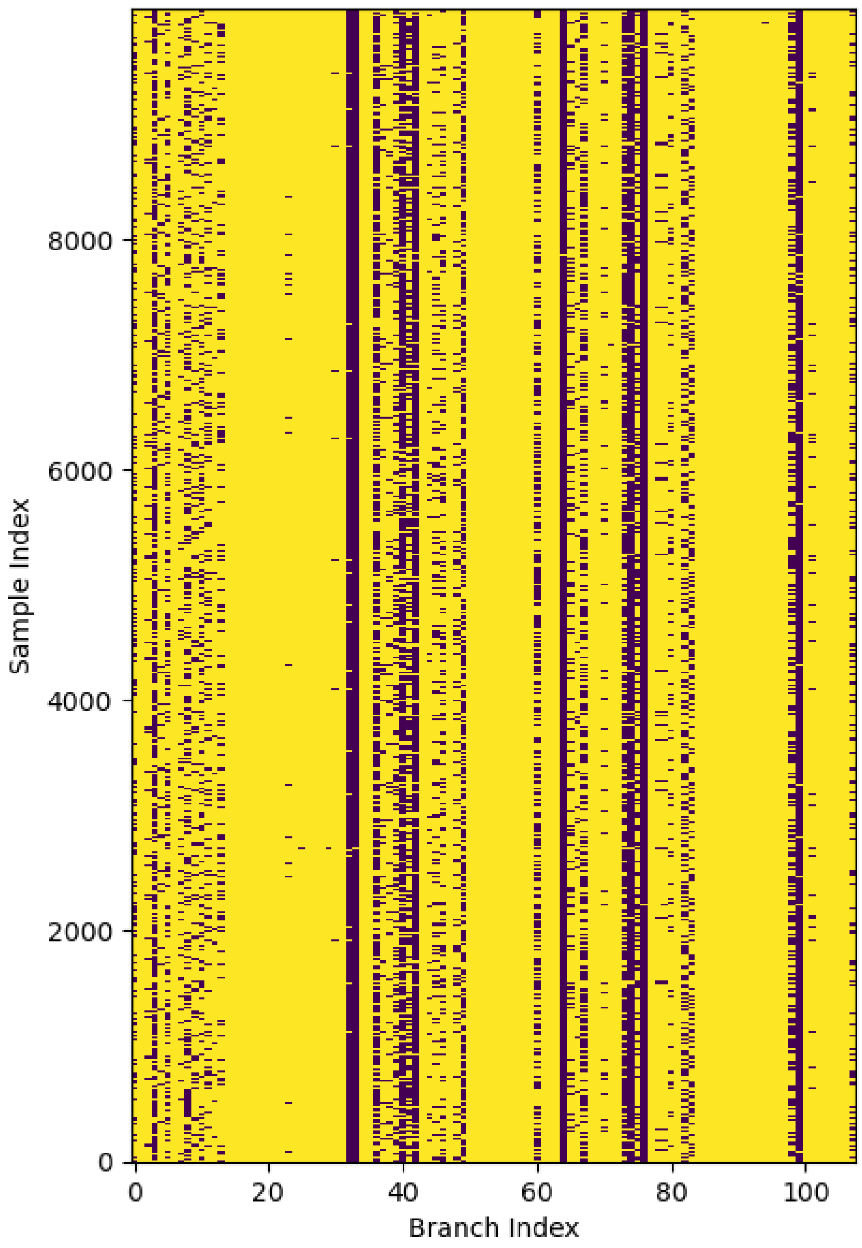
- Motivation:
  - Reduce computing time of NROPF
- Propose approach:
  - Using ML, especially GNN, to identify all line statuses
  - Turn a MILP problem into an LP problem
  - Much faster to solve, save more time
  - Challenges:
    - Feasible solution
    - Accuracy of prediction in relation to total cost

OFFLINE – Training Mode



Pre-ML Filtering Step

- Colormesh figure displays lines statuses for NROPF solutions.
- Yellow = ON Purple = OFF
- Quickly identify lines that is **ALWAYS ON** or **ALWAYS OFF** from the training samples.
- Exclude these always ON/OFF lines from predictions using GNN model.
- Focus GNN model on training “critical” lines.



Line statuses (ON/OFF) of each sample

Post-ML Selection Step

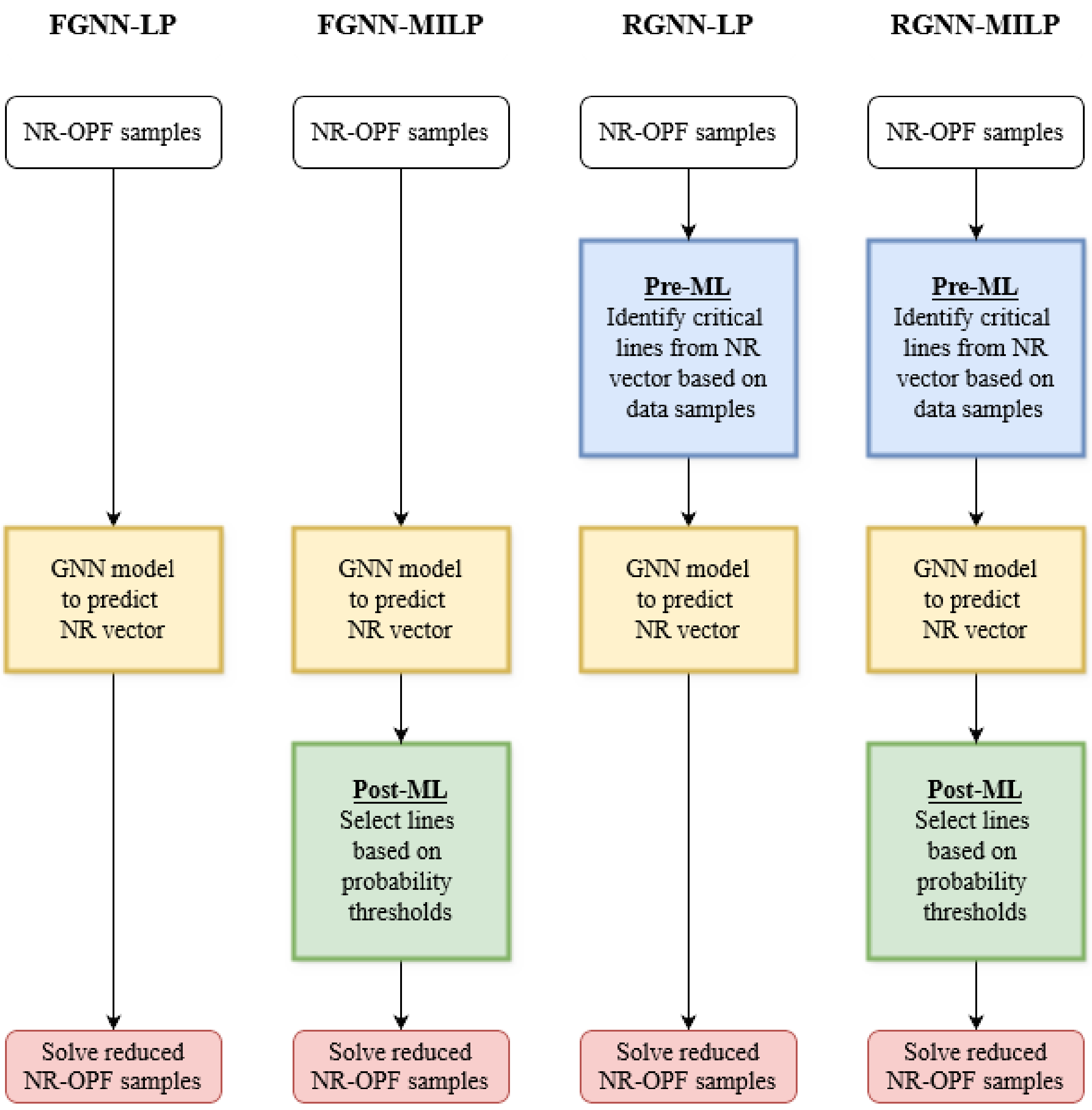
- ML prediction is a probability between 0 and 1.
- Normally, prediction with higher probability is chosen for likelihood of being corrected.
- An Upper Limit/Lower Limit of is chosen as a threshold.
- Only predictions **below 95%** or **above 5%** (highlight in gray shading) will be excluded.
- Reduce the possible number of network reconfiguration.
- Speed up time to find feasible solution.

Line	1	2	3	4	5	6
OFF	6.11%	0.04%	7.53%	81.54%	23.03%	0.15%
ON	93.89%	99.96%	92.47%	18.46%	76.97%	99.85%
Post-ML Selection	Exclude	Select	Exclude	Exclude	Exclude	Select

GaNR-ROPF Methods

- **Full GNN-Linear Problem (FGNN-LP)**: uses only GNN model for prediction of line switching status.
- **Full GNN-Mixed-Integer Linear Problem (FGNN-MILP)**: uses the post-ML selection step after using GNN model for prediction of line switching status.
- **Reduced GNN-Linear Problem (RGNN-LP)**: uses the pre-ML filtering step before using GNN model for prediction of line switching status.
- **Proposed Reduced GNN-Mixed-Integer Linear Problem (RGNN-MILP)**: uses both pre-ML filtering step and post-ML selection step along with GNN model for prediction of line switching status.

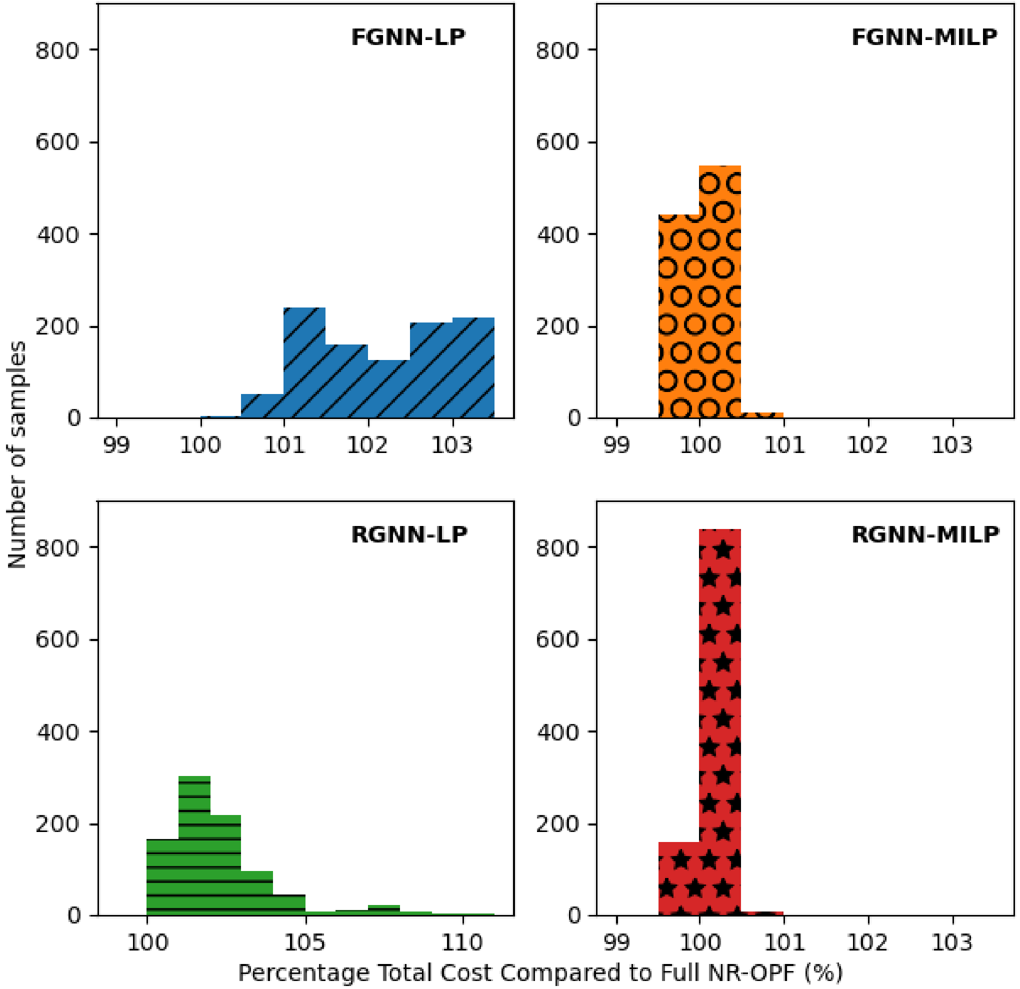
ONLINE – Prediction Mode



Results: Total Cost

FGNN-LP = No Extra Step
FGNN-MILP = Post-ML Selection
RGNN-LP = Pre-ML Filtering
RGNN-MILP= Pre-ML Filtering/Post-ML Selection

- MILP-based methods (FGNN-MILP, RGNN-MILP): Highly accurate, with results close to the optimal solution.
- LP-based methods (FGNN-LP, RGNN-LP): More variable, with deviations up to 11% from the optimal solution.
- RGNN-MILP: Stands out as the most reliable method, with over 80% of solutions perfectly matching the optimal cost.
- **Overall: MILP methods are superior in producing accurate and stable cost-optimized results.**

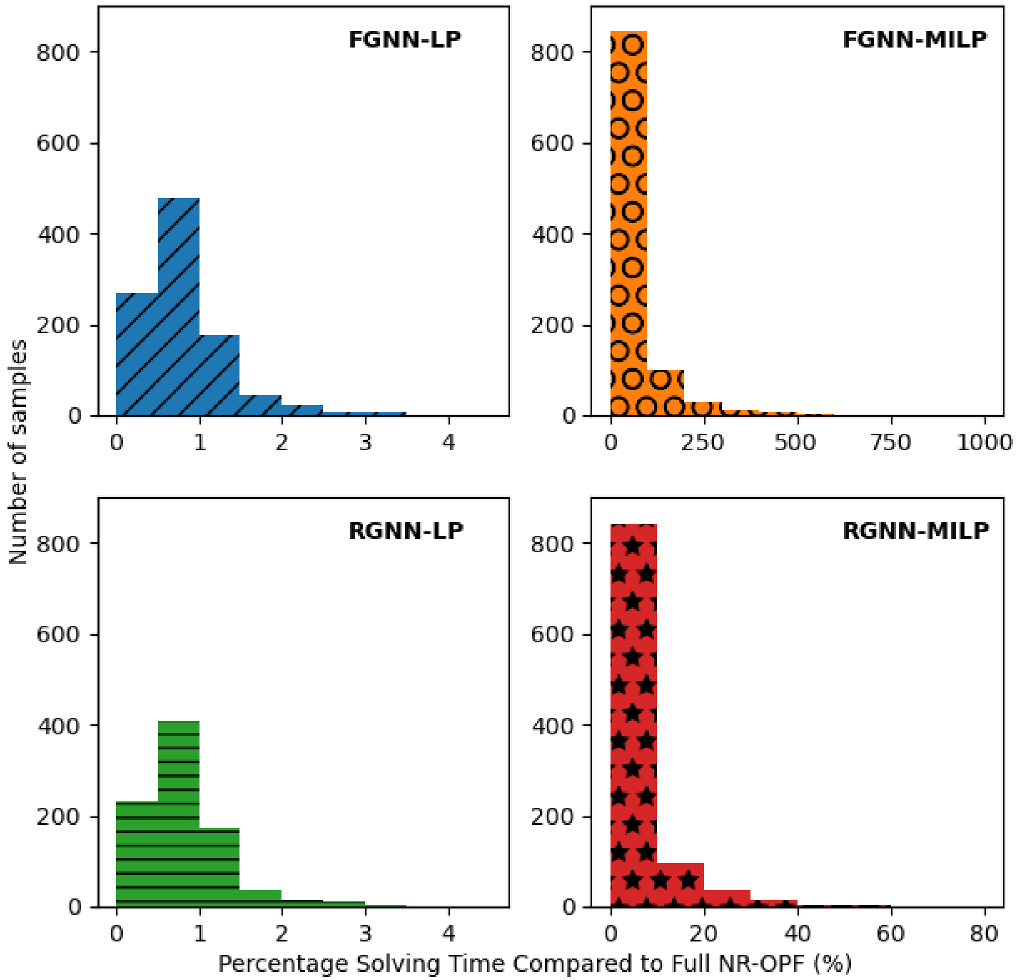


***Note: The FGNN-MILP model has 122 samples with “Infeasible” solutions out of 1,000 samples.**

Results: Solving Time

FGNN-LP = No Extra Step
FGNN-MILP = Post-ML Selection
RGNN-LP = Pre-ML Filtering
RGNN-MILP= Pre-ML Filtering/Post-ML Selection

- FGNN-LP and RGNN-LP: Achieve remarkable solving time reductions, with improvements of up to 99%.
- FGNN-MILP: Yields average solving time reductions of 40%, but some samples can take up to 10 times longer than the original problem.
- RGNN-MILP: Reduces solving time by 94%, with a median time of 3.5%, offering consistent efficiency.
- **Overall: LP-based methods are the fastest, while RGNN-MILP offers strong efficiency and consistent performance.**



***Note: The FGNN-MILP model has 122 samples with “Infeasible” solutions out of 1,000 samples.**

Results: RGNN-MILP Method

FGNN-LP = No Extra Step
FGNN-MILP = Post-ML Selection
RGNN-LP = Pre-ML Filtering
RGNN-MILP= Pre-ML Filtering/Post-ML Selection

- All four methods experience significant decrease in computing time for solution, averaging from 40% to as much as 99%.
- Based on the result, RGNN-MILP method is the ideal candidate for GaNR-OPF:
  - Total Cost: mean difference is 0.05% and median difference is 0.02% from the full NR-OPF.
  - Solving Time: mean time is 6% and median time 3.5% of the full NR-OPF.
  - **RGNN-MILP is the top performer in terms of speed and consistency among the benchmark methods.**

Statistical data for objective total cost for the four proposed methods

	Total Cost in Percent (%)				
	Mean	**Max**	**Min**	**Median**	**Std. Dev.**
FGNN-LP	102.17%	103.72%	100.29%	102.23%	0.80%
FGNN-MILP	100.02%	100.98%	99.51%	100.00%	0.11%
RGNN-LP	102.25%	111.22%	100.00%	101.83%	1.75%
RGNN-MILP	100.05%	100.56%	99.92%	100.02%	0.09%

Statistical data for solving time for the four proposed methods

	Solving Time in Percent (%)				
	Mean	**Max**	**Min**	**Median**	**Std. Dev.**
FGNN-LP	0.81%	3.38%	0.08%	0.72	0.49
FGNN-MILP	60.23%	1085.76%	2.72%	30.09	92.89
RGNN-LP	0.83%	4.01%	0.08%	0.73	0.51
RGNN-MILP	6.34%	83.77%	0.41%	3.47	8.24

FGNN-LP = No Extra Step
FGNN-MILP = Post-ML Selection
RGNN-LP = Pre-ML Filtering
RGNN-MILP= Pre-ML Filtering/Post-ML Selection

Summary

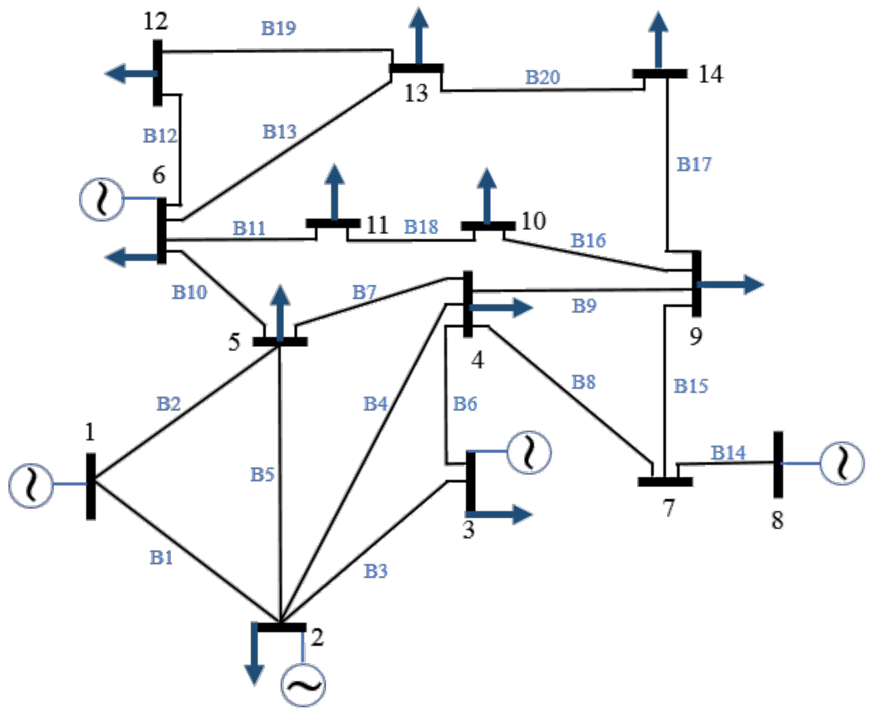
- **GaNR-OPF provides almost identical total cost to the full NR-OPF with much faster solving time.**
- Saving of 94% in solving time with all optimal solutions using predictions from RGNN-MILP method.
- Combination of both processing steps pre and post ML result in dramatic reduction in solving time while still retaining quality of solutions.
- GaNR-OPF using the proposed RGNN-MILP method is an effective tool to address a large number of network reconfiguration problems quickly.

Publication:

- **Thuan Pham** and Xingpeng Li, "Graph Neural Network-Accelerated Network-Reconfigured Optimal Power Flow", IEEE Transactions on Industrial Informatics, (In Review).

Chapter 5
Virtual Node-Splitting in Hierarchical Graph Neural Network for Optimal Power Flow

Optimal Power Flow



- Objective Function:

$$min \sum_{g \in G} c_g P_g$$

- Constraints:

$P_g^{min} \leq P_g \leq P_g^{max}$, $g \in G$ → Generation Constraints

$P_k = (\theta_{f(k)} - \theta_{t(k)})/x_k$, $k \in K$ → Line Flow Equation

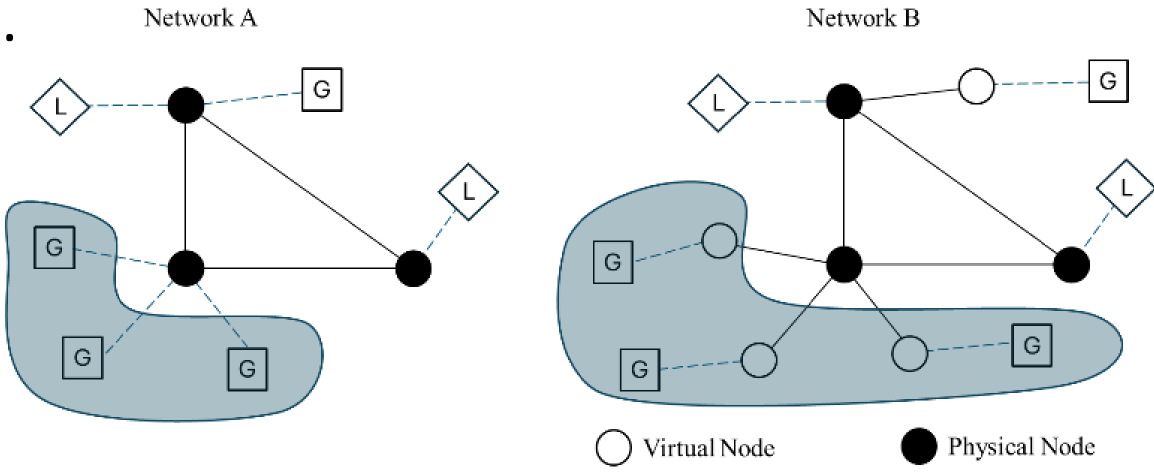
$-RateA_k \leq P_k \leq RateA_k$, $k \in K$ → Line Limit Constraints

$\sum_{g \in G(n)} P_g + \sum_{k \in K(n-)} P_k - \sum_{k \in K(n+)} P_k = d_n$, $n \in N$ → Nodal Balance Equation

G is a set of generators
K is a set of lines
N is a set of buses

Virtual Node-Splitting

- For GNN, network topologies of power systems are commonly represented as homogeneous graphs.
- When multiple generators are associated with a single node, homogeneous graph restricts the GNN model from capturing distinct node features.
- Subdividing virtual nodes from existing physical nodes:
  - Generators is connected to the virtual node.
  - Load is linked with the real node.
- Retain specific features at the node-level for generators such as:
  - Ramping rate
  - Generation cost
- Larger topology -> increase training time -> better predictions



A 3-bus system (Network A) vs. the expanded 3-bus system using virtual node-splitting (Network B).

Literature Review

- Modifying the topology of a transmission network, such as line switching and bus-splitting, has been used as a corrective mechanism to mitigate congestion issues [1].
- For transmission expansion planning, bus-splitting has demonstrated its efficacy in enhancing optimal dispatch solutions [2]
- Bus-splitting in substation has been utilized for system protection[3]
- Topology modification has been well studied for power system.

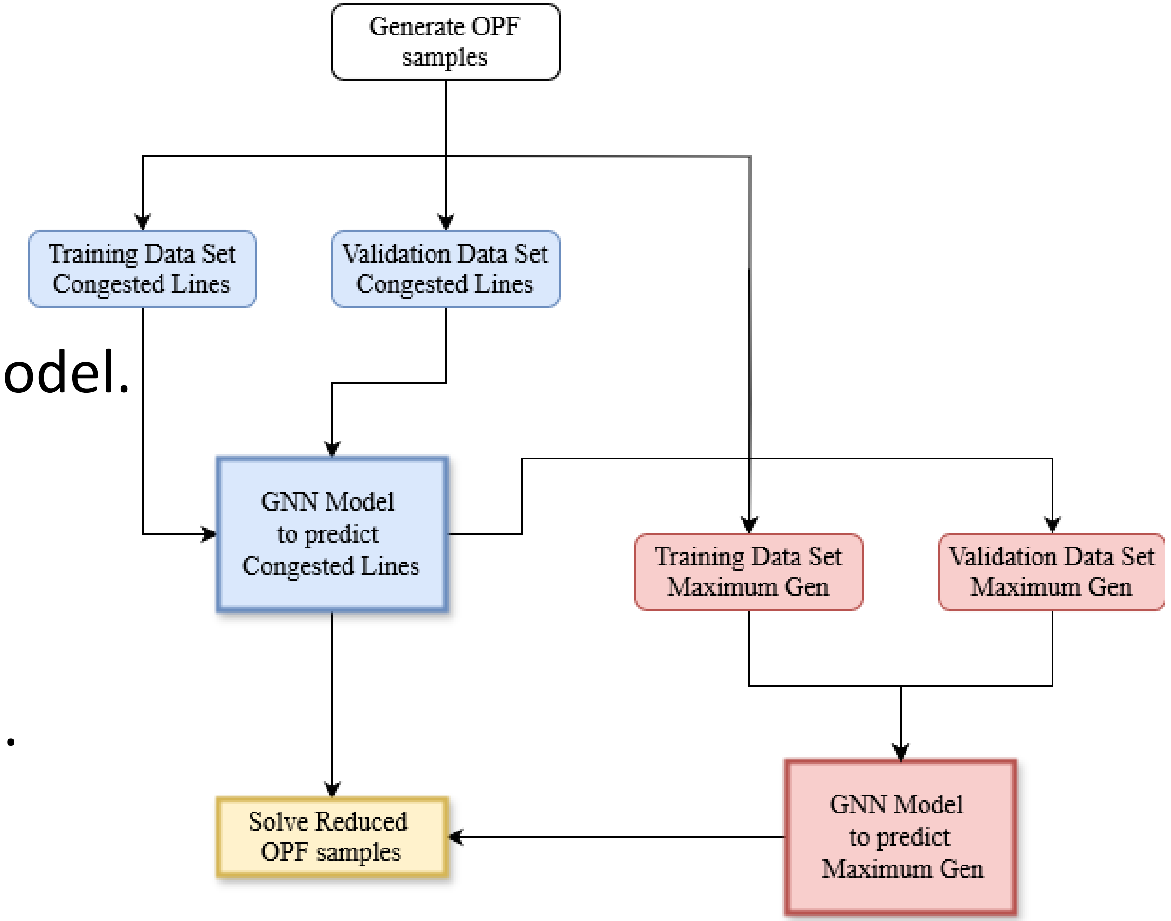
References:

1. A. Hinneck, B. Morsy, D. Pozo and J. Bialek, "Optimal Power Flow with Substation Reconfiguration," in 2021 IEEE Madrid PowerTech, Madrid, 2021.
2. M. Heidarifar, M. Doostizadeh and H. Ghasemi, "Optimal transmission reconfiguration through line switching and bus splitting," in 2014 IEEE PES General Meeting, National Harbor, 2014.
3. P. J. N. Gealone and A. E. D. Tio, "Optimized Transmission Expansion Planning with Bus-Splitting in Grids with High VRE Penetration," in SIEDS, Charlottesville, 2024.

Proposed Method for ROPFLG

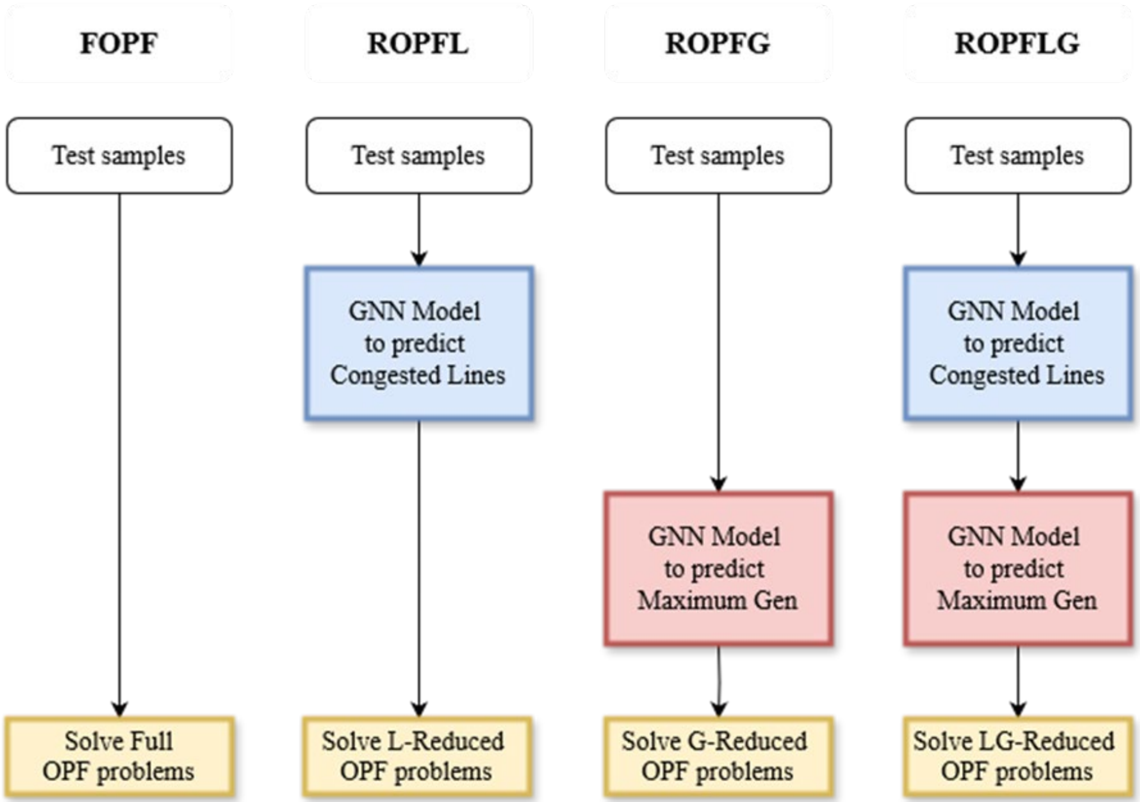
OFFLINE – Training Mode

- Motivation:
  - Reduce computing time of OPF.
- Propose approach:
  - Using virtual node-splitting to increase the number of node features for training GNN model.
  - Hierarchical GNN model:
    - 1^{st} model predicts congested lines.
    - 2^{nd} model predicts maximum-capacity generators using predictions of 1^{st} model.
  - Challenges:
    – Longer training time.
    – Feasible solution with accurate prediction in relation to objective total cost.



Four Proposed Methods

ONLINE – Prediction Mode



- **Full OPF (FOPF)**: solves the OPF problem without reducing the number of constraints.
- **Reduced OPF Lines (ROPFL)**: the GNN model predicts congested lines. Lines limit constraints for non-congested lines are removed.
- **Reduced OPF Generators (ROPFG)**: the GNN model predicts maximum-capacity generators. Parameters for maximum-capacity generators are set.
- **Reduced OPF Lines and Generators (ROPFLG)** (the proposed method): uses the first GNN model to first predict congested line to use as input features for the second GNN model, which predicts maximum- capacity generators. Line limit constraints for non-congested lines are removed, and variables for predicted maximum-capacity generators are switched to known parameters that would reduce both variables and constraints.

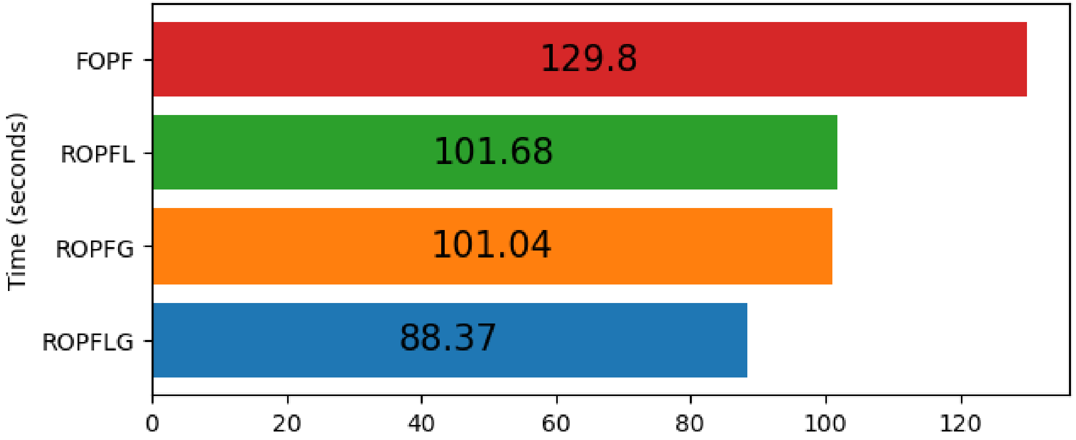
Results: Solving Time and Total Cost

FOPF = No Extra Step
ROPFL = Reduce lines constraints
ROPFG = Reduce generator constraints
ROPFLG = Reduce both types of constraints

- Mean total cost are almost identical for all methods.
- ROPFL and ROPFG have similar time saving around 22%
- ROPFLG delivers the most significant time savings, at almost 32% or 40 seconds faster, while maintaining optimal solutions comparable to the baseline FOPF.

Mean total cost and time save in (%) for solving 1000 test samples.

	Mean Total Cost (%)	**Time Saving (%)**
FOPF	100%	0
ROPFL	100.061%	21.67%
ROPFG	100.064%	22.16%
ROPFLG	100%	31.92%



Solving time for 1000 test samples using the four proposed methods.

The proposed ROPFLG method achieves significant reductions in computation time while preserving solution quality.

FOPF = No Extra Step

ROPFL = Reduce lines constraints

ROPFG = Reduce generator constraints

ROPFLG = Reduce both types of constraints

Summary

- **ROPFLG offers the highest computation time savings while preserving the optimal solution quality of FOPF.**
- All ROPF methods have nearly identical average total costs compared to FOPF, with differences below 0.1% (negligible).
- Larger systems with multiple congested lines and high generator loads would show more pronounced savings.
- The hierarchical GNN model's savings are less impactful for systems with few removable constraints.

Publication:

- **Thuan Pham** and Xingpeng Li, “Constraints and Variables Reduction for Optimal Power Flow Using Hierarchical Graph Neural Networks with Virtual Node-Splitting”, IEEE PES General Meeting, (In Review).

Chapter 6
Conclusions

Conclusions

- Machine learning, especially GNN model, enhances prediction accuracy and computational efficiency.
- GNN is the best model for classifying branches that will likely be overloaded or congested. Thus, reducing computing time for Reduced Optimal Power Flow.
- Proposed Augmented Hierarchical Graph Neural Network model's predictions lead to solutions with zero line rating violations and faster computing time.
- GNN-Accelerated Network Reconfigured Optimal Power Flow using pre-ML filtering and post-ML selection mechanisms is an effective tool to address a large number of network reconfiguration problems quickly.
- Virtual node-splitting enhances predictive capability of GNN model. Reduced Optimal Power Flow with Lines and Generators method achieves the highest computational time savings among methods while maintaining optimal solutions.

Future Work

- GNN models show potential for greater computational time savings and improved results when applied to larger, more complex systems.
- Alternative ML architectures for GNN:
  - **Transfer Learning in GNNs:** leverages pre-trained models for new tasks without retraining. Enhances deployment speed and reduces computational costs.
  - **Dual Prediction GNNs:** Predicts multiple outputs (e.g., congested lines and maximum-capacity generators) simultaneously. Optimizes resource use, improves accuracy, and reduces training time.
- Apply GNN model for AC OPF problem.
- Use line shift-factor in formulation of OPF problem.

Publications

1. **Thuan Pham** and Xingpeng Li, “Neural Network-based Power Flow Model”, IEEE Green Technology Conference, Houston, TX, USA, Mar. 2022.
2. **Thuan Pham** and Xingpeng Li, “Reduced Optimal Power Flow Using Graph Neural Network”, 54th North American Power Symposium, Salt Lake City, UT, USA, Oct. 2022.
3. **Thuan Pham** and Xingpeng Li, “*N*-1 Reduced Optimal Power Flow Using Augmented Hierarchical Graph Neural Network”, IEEE Transactions on Neural Networks and Learning Systems, (In Review).
4. **Thuan Pham** and Xingpeng Li, “Graph Neural Network-Accelerated Network-Reconfigured Optimal Power Flow”, IEEE Transactions on Industrial Informatics, (In Review).
5. **Thuan Pham** and Xingpeng Li, “Constraints and Variables Reduction for Optimal Power Flow Using Hierarchical Graph Neural Networks with Virtual Node-Splitting”, IEEE PES General Meeting, (In Review).

References

- Sanchez-Lengeling, et al., "A Gentle Introduction to Graph Neural Networks", Distill, 2021.
- Makarov, Y. V., Etingov, P. V., Ma, J., Huang, Z., & Subbarao, K. (2011). Incorporating uncertainty of wind power generation forecast into power system operation, dispatch, and unit commitment procedures. IEEE Transactions on Sustainable Energy, 2(4), 433-442.
- Li, Xingpeng (2022). ECE6327_SmartGrid. Figshare. Online resource. https://doi.org/10.6084/m9.figshare.19761268
- D. Owerko, F. Gama and A. Ribeiro, "Optimal Power Flow Using Graph Neural Networks," in ICASSP, 2020.
- K. Yang, W. Gao and R. Fan, "Optimal Power Flow Estimation Using One-Dimensional Convolutional Neural Network," in North American Power Symposium, College Station, 2021.
- X. Pan, T. Zhao, M. Chen and S. Zhang, "DeepOPF: A Deep Neural Network Approach for Security-Constrained DC Optimal Power Flow," IEEE Transactions on Power Systems, vol. 36, no. 3, pp. 1725 - 1735, May 2021
- T. Pham and X. Li, "Reduced Optimal Power Flow Using Graph Neural Network," in *2022 North American Power Symposium (NAPS)*, Salt Lake City, 2022.
- D. Owerko, F. Gama and A. Ribeiro, "Unsupervised Optimal Power Flow Using Graph Neural Networks," arXiv, 2022.
- S. Thongkeaw, N. Rugthaicharoencheep and S. Auchariyamet, "Application of particle swarm optimization for power system operation considering N-1 contingency criteria," in 47th International Universities Power Engineering Conference (UPEC), Uxbridge, 2012.
- M. Numan, M. F. Abbas, M. Yousif, S. S. M. Ghoneim, A. Mohammad and A. Noorwali, "The Role of Optimal Transmission Switching in Enhancing Grid Flexibility: A Review," in *IEEE Access*, vol. 11, pp. 32437-32463, 2023, doi: 10.1109/ACCESS.2023.3261459.
- Crozier, C., Baker, K., & Toomey, B. (2022). Feasible region-based heuristics for optimal transmission switching. Sustainable Energy, Grids and Networks, 30, 100628. https://doi.org/10.1016/j.segan.2022.100628
- T. Han and D. J. Hill, "Learning-Based Topology Optimization of Power Networks," in *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1366-1378, March 2023, doi: 10.1109/TPWRS.2022.3170083.
- A. Hinneck, B. Morsy, D. Pozo and J. Bialek, "Optimal Power Flow with Substation Reconfiguration," in 2021 IEEE Madrid PowerTech, Madrid, 2021.
- M. Heidarifar, M. Doostizadeh and H. Ghasemi, "Optimal transmission reconfiguration through line switching and bus splitting," in 2014 IEEE PES General Meeting, National Harbor, 2014.
- P. J. N. Gealone and A. E. D. Tio, "Optimized Transmission Expansion Planning with Bus-Splitting in Grids with High VRE Penetration," in SIEDS, Charlottesville, 2024.
- K. Waehner, "The State of Data Streaming for Energy & Utilities," [Online]. Available: https://www.kai-waehner.de/blog/2023/09/01/the-state-of-data-streaming-for-energy-utilities-in-2023/.

UH RPG Lab

Thank you!

Thuan Pham