Physics-aware and Risk-aware Machine Learning for Power System Operations

Hao Zhu
The University of Texas at Austin
(haozhu@utexas.edu)

PSERC Webinar
March 29, 2022

Presentation Outline

- A primer on supervised learning
- Three machine learning (ML) examples
  - Topology-aware learning for real-time market
  - Risk-aware learning for DER coordination
  - Scalable learning for grid emergency responses
- Summary

Power of AI

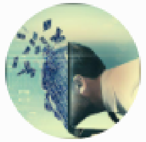
- Unprecedented opportunities offered by diverse sources of data
  - Synchrophasor and IED data
  - Smart meter data
  - Weather data
  - GIS data, ...

How to harness the power of ML to tackle problem-specific challenges in real-time power system operations?

Forbes

Mar 15, 2019, 07:37am EDT | 21,849 views

How AI Can And Will Predict Disasters



Naveen Joshi Former Contributor
COGNITIVE WORLD Contributor Group
AI

AI could put a stop to electricity theft and meter misreadings

TECHNOLOGY 23 September 2017

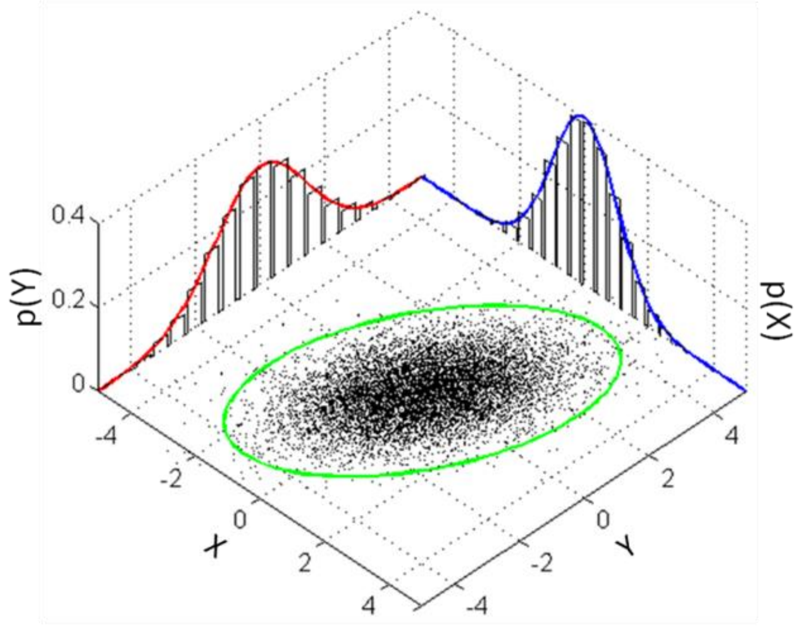
SUSTAINABLE ENERGY

Combining A.I. and human knowledge could transform how power grids work

PUBLISHED FRI, SEP 27 2019•4:55 AM EDT | UPDATED FRI, SEP 27 2019•4:55 AM EDT

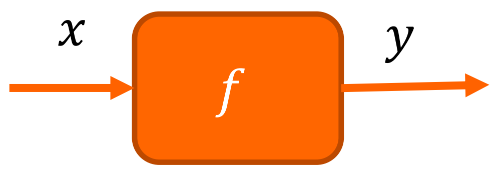
A primer on supervised learning

- *Unknown* joint distribution for $(x, y) \in \mathbb{R}^d \times Y$
  - Classification: $Y = \{\pm 1\}$ or $Y = \{1, \ldots, C\}$
  - Regression: $Y = R^b$
- Given examples, aka, *data samples* $\{(x_k, y_k)\}$
  - x_k: input **feature**
  - y_k: output **target/label**
- Without y_k => *unsupervised or semi-supervised* learning
- Samples from dynamical systems => reinforcement learning



Learning problem formulation

- Goal: construct a function $f : \mathbb{R}^d \to Y$ to map $x \to y$
  - *Predicted* value $\hat{y} = f(x) \in Y$ to be close to y
  - **Loss function**: $l(\hat{y}, y) = l(f(x), y) \geq 0$
  - For regression, use L_p norms $l(\hat{y}, y) = \left|\left|\hat{y} - y\right|\right|_p$
  - For classification, cross-entropy loss, hinge loss, etc.



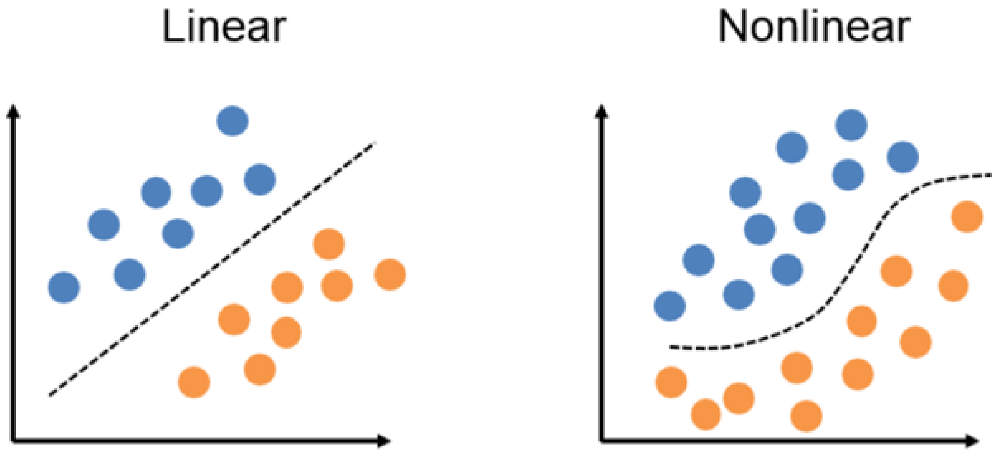
$$f^\star = \arg\min_{f \in F} \; \mathbb{E}_{(x,y)} \, l(f(x), y) \quad \xrightarrow{\text{Sample Mean}} \quad \hat{f} = \arg\min_{f \in F} \; \frac{1}{K} \sum_{k=1}^{K} l(f(x_k), y_k)$$

- Excellent generalization (error bounds on $f^\star - \hat{f}$) performance?

Vidal, Rene, et al. "Mathematics of deep learning." *arXiv preprint arXiv:1712.04741* (2017).
Bartlett, Peter L., Andrea Montanari, and Alexander Rakhlin. "Deep learning: a statistical viewpoint." *arXiv preprint arXiv:2103.09177* (2021).

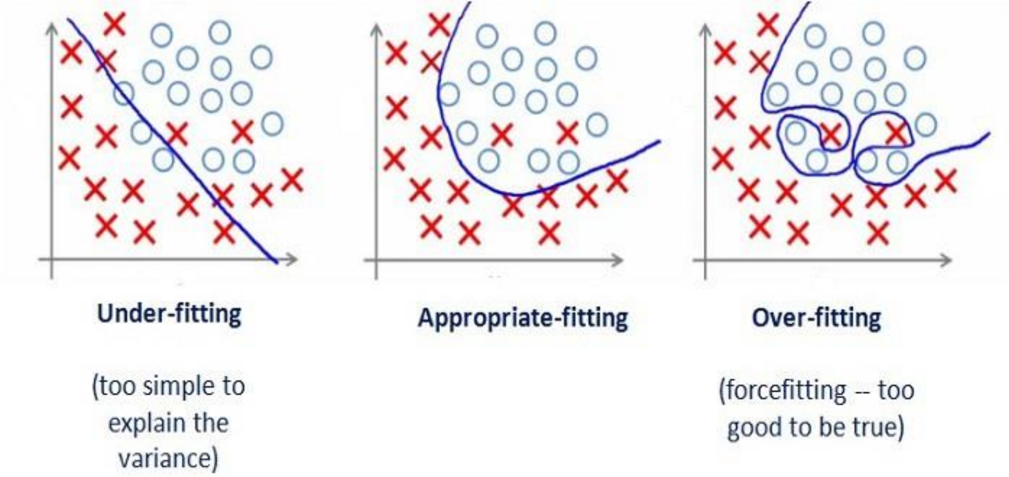
Parameterized models for *f*

- Impossible to search over any function *f* => *parameterization*
- **Linear** $f(x) = w^\top x + w_0$ parameterized by $w \in \mathbb{R}^d$ and $w_0 \in \mathbb{R}$
  - A simple model structure to use
  - Linear regression (LS, LAV)
  - Linear classification (logistic regression or SVM)



- **Nonlinear** f for better prediction
  - Polynomials, Gaussian Processes (GPs), etc.
  - Kernel learning: $f \in \mathcal{H}$ (Hilbert space for some kernel)
  - Neural networks (NN): layers of nonlinear functions.

Regularization



- Data overfitting (losses → 0)
  - Features redundant: e.g., both x_i and $-x_i$
  - Models too complex: high-order polynomials, deep neural networks
  - *We can fit any K data samples perfectly using a (K-1)-th order polynomials*

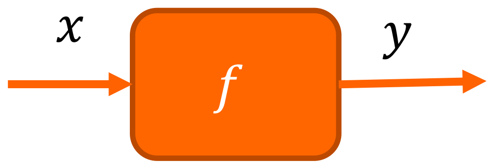
$$\hat{f} = \arg\min_{f \in F} \sum_{k=1}^{K} l(f(x_k), y_k) + \lambda \cdot \text{Reg}(f)$$

norm of parameter w

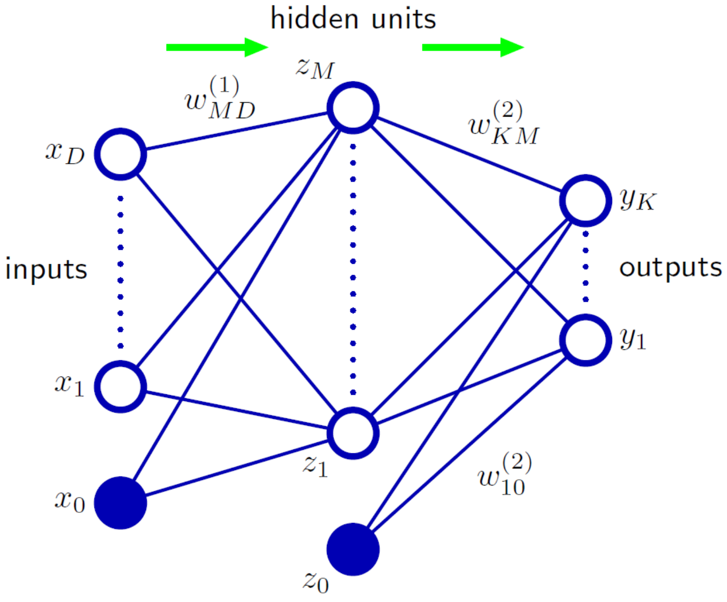
- Hyperparameter $\lambda > 0$ balances between data fitting and model complexity
- L_2 norm/Ridge: small values, or smooth using $\sum_i (w_i - w_{i-1})^2$
- L_1 norm/Lasso: sparse w (much more zero entries)

Deep (D)NN architecture

- Perceptron (single-layer NN): convert $f(x) = w^\top x$ to a nonlinear function by $f(x) = \sigma(w^\top x)$
  - nonlinear activation $\sigma(\cdot)$: sigmoid, Tanh, ReLU



- NNs: basically multi-layer perceptron (MLP)
  - Layered, feed-forward networks (input *x*, output *y*)
  - Hidden layers also called neutrons or units
  - 2-layer NNs can express all continuous functions, while for any nonlinear ones 3 layers are sufficient



Deep Learning book https://www.deeplearningbook.org/

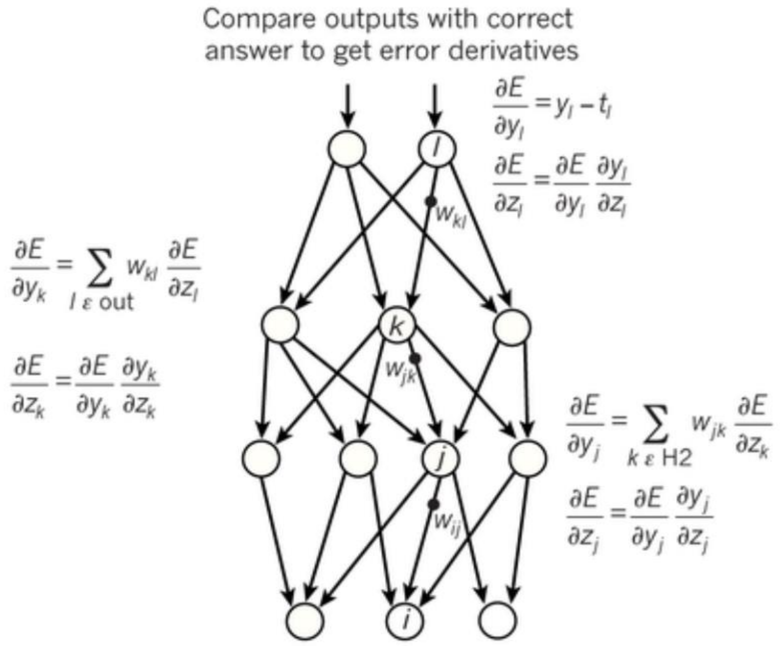
Gradient descent (GD) via *backpropagation*

$$\hat{w} = \arg\min_{w} \quad E(w) := Loss(w) + \lambda Reg(w)$$

- Nonlinear *f* => nonconvex opt. problem
- GD-based learning

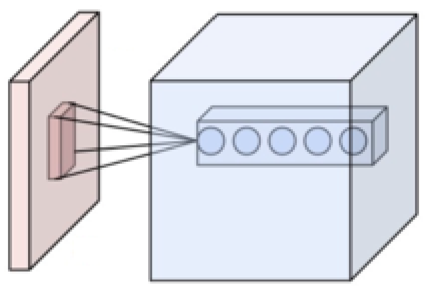
$$w \leftarrow w - \alpha \nabla E(w)$$

- In practice, local minima may not be a concern [LeCun, 2014]
- Efficient computation of gradient in a backward way using the “chain rule”

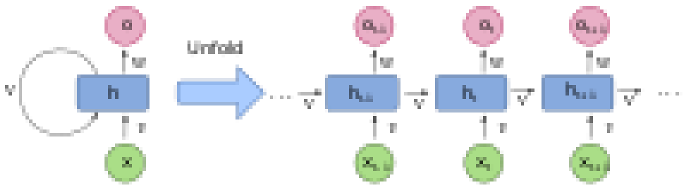


Variations of DNN

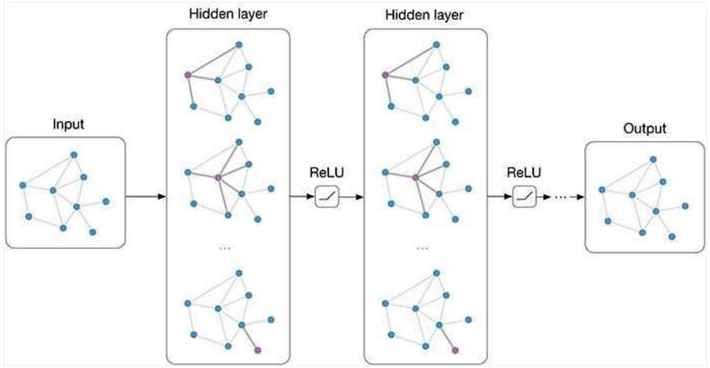
- Fully-connected NN (FCNN): weight parameters grow with data size
- Idea: reuse the weight parameters, aka, filters!



Convolutional NN (CNN):
Spatial filters for images/video



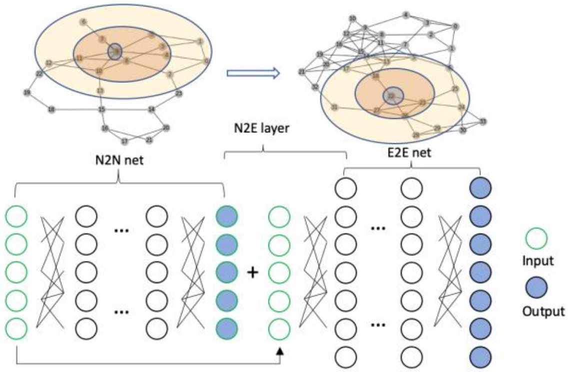
Recurrent NN (RNN):
Temporal filters for texts, speech



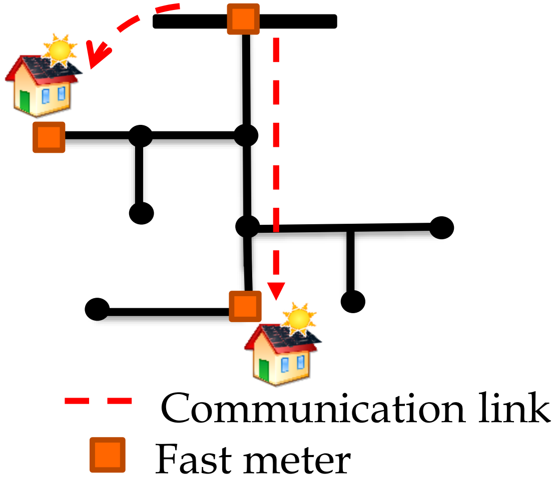
Graph NN (GNNs):
Graph filters for networked systems

Overview

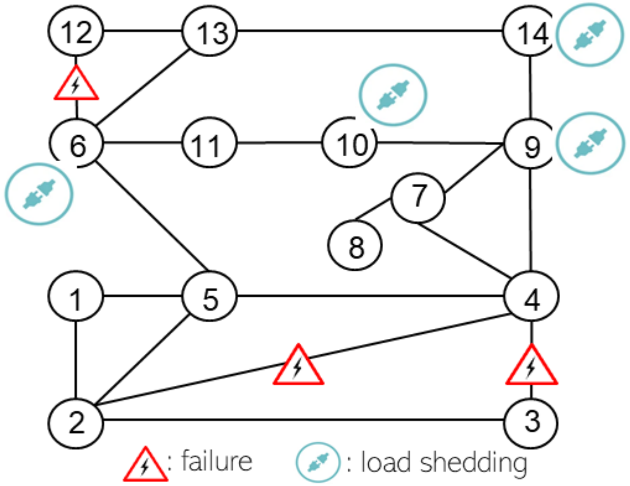
➤ We visit three problems that use domain knowledge to better design NN models that are physics-informed and risk-aware



Topology-aware learning for real-time market:
Simpler model for efficient training



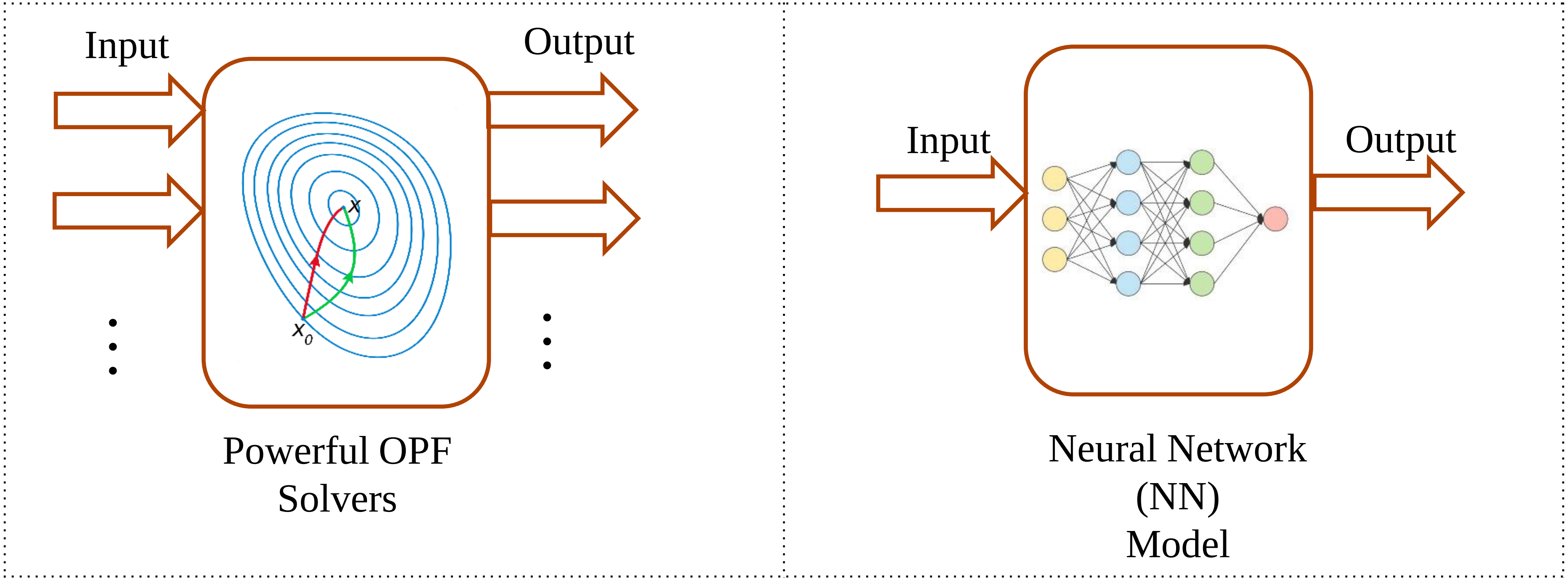
Risk-aware learning for DER coordination:
Reduced risks of voltage violations



Scalable learning for grid emergency responses:
Fast mitigations under limited data

PART I: TOPOLOGY-AWARE LEARNING FOR REAL-TIME MARKET

ML for optimal power flow (OPF)



➢ Real-time computation of the OPF solutions by learning the I/O mapping

Existing work and our focus

- Integration of renewable, flexible resources increases the grid variability and motivates real-time, fast OPF via training a neural network (NN)
  - Identifying the active constraints (for dc-OPF) [Misra et al'19][Deka et al'19]
  - Directly mapping the ac-OPF solutions [Guha et al'19]
  - Warm start the search for ac feasible solution [Baker '19] [Zamzam et al'20]
- Address the uncertainty in stochastic OPF [Mezghani et al'20]
- Connect to the duality analysis of convex OPF [Chen et al'20] [Singh et al'20]

Focus: Exploit the grid topology to *reduce the NN model complexity*

OPF for real-time market

- Power network modeled as a graph $G = (\mathcal{V}, \mathcal{E})$ with *N* nodes
- ac-OPF for all nodal injections

$$\begin{aligned} \min_{\mathbf{p},\mathbf{q},\mathbf{v}} \quad & \textstyle\sum_{i=1}^{N} c_i(p_i) \\ \text{s.t.} \quad & \mathbf{p} + \mathrm{j}\mathbf{q} = \mathrm{diag}(\mathbf{v})(\mathbf{Y}\mathbf{v})^* \\ & \underline{\mathbf{V}} \leq |\mathbf{v}| \leq \bar{\mathbf{V}} \\ & \underline{\mathbf{p}} \leq \mathbf{p} \leq \bar{\mathbf{p}} \\ & \underline{\mathbf{q}} \leq \mathbf{q} \leq \bar{\mathbf{q}} \\ & \underline{f}_{ij} \leq f_{ij}(\mathbf{v}) \leq \bar{f}_{ij}, \quad \forall (i,j) \in \mathcal{E} \end{aligned}$$

- Nodal input:

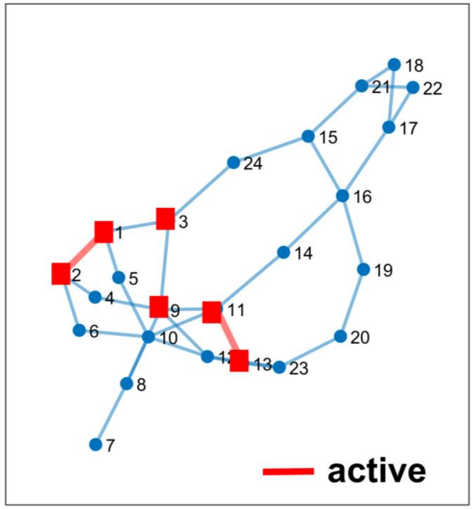
  $$\mathbf{x}_i \triangleq [\bar{p}_i, \underline{p}_i, \bar{q}_i, \underline{q}_i, \mathbf{c}_i] \in \mathbb{R}^d$$

  power limits + costs

- Nodal output: optimal p/q ?
- Fully-connected (FC)NN

FCNN layer has $\mathcal{O}(N^2)$ parameters!

Topology dependence



- [Owerko et al'20] uses graph learning to predict p/q
- Locational marginal price (LMP) from the dual problem
  - Strongly depends on the graph topology and congested lines
  - ISF (injection shift factor) matrix **S** from graph Laplacian

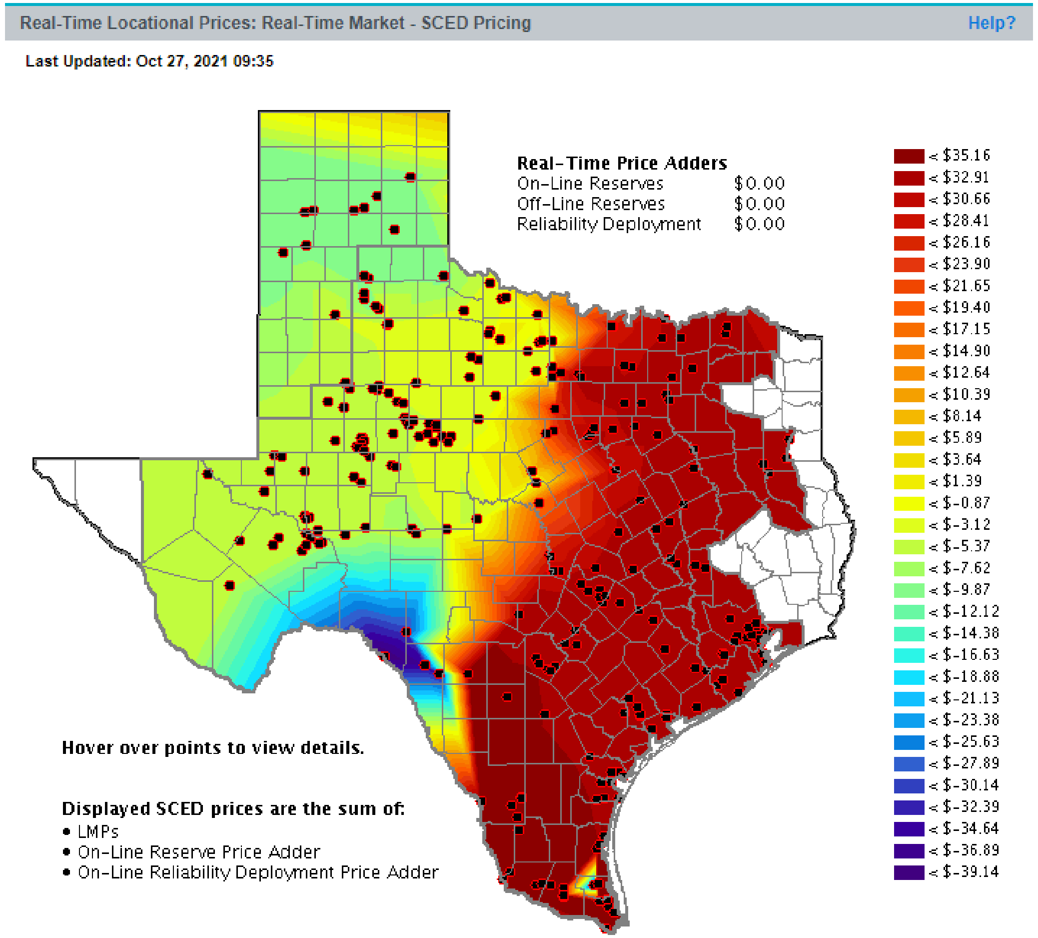
$$
\begin{aligned}
\min_{\mathbf{p}} \quad & \textstyle\sum_{i=1}^{N} c_i(p_i) \\
\text{s.t.} \quad & \mathbf{1}^\top \mathbf{p} = 0 & : \lambda \\
& \underline{\mathbf{p}} \le \mathbf{p} \le \bar{\mathbf{p}} \\
& \underline{\mathbf{f}} \le \mathbf{S}\mathbf{p} \le \bar{\mathbf{f}} & : [\underline{\boldsymbol{\mu}}; \bar{\boldsymbol{\mu}}]
\end{aligned}
$$

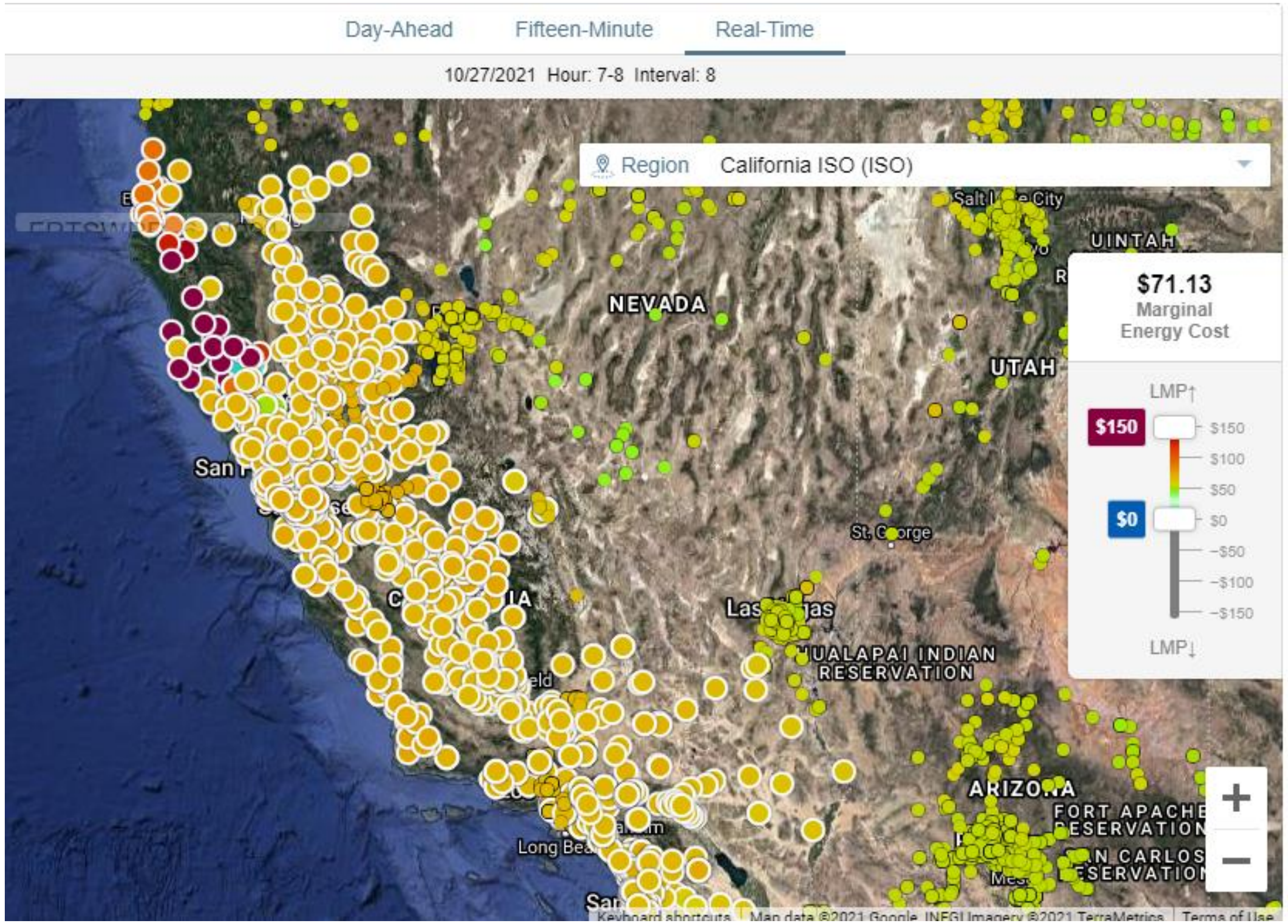
$$\boldsymbol{\pi} := \lambda^* \cdot \mathbf{1} - \mathbf{S}^\top (\bar{\boldsymbol{\mu}}^* - \underline{\boldsymbol{\mu}}^*)$$

$$\mathbf{S}^\top = \mathbf{B}_r^{-1} \mathbf{A}_r^\top \mathbf{X}^{-1}$$

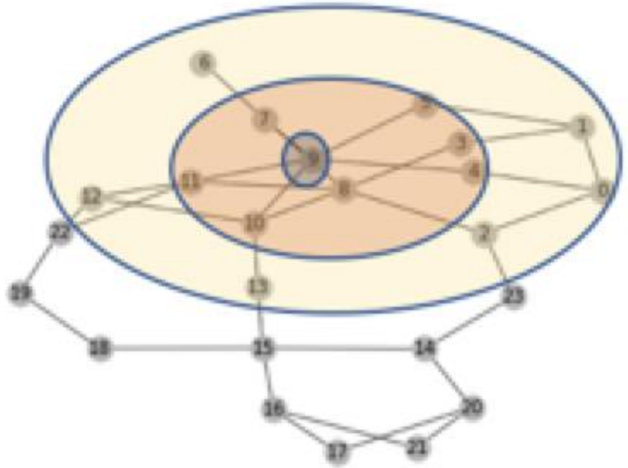
shares the same eigen-space as the graph Laplacian $\mathbf{B}_r$

LMP map with locality





Graph NN (GNN): topology-based filtering



- Input formed by nodal features as rows

$$\mathbf{X}^0 = \{\mathbf{x}_i\} \in \mathbb{R}^{N \times d}$$

- GNN layer l with learnable parameters

$$\mathbf{X}^{\ell+1} = \sigma\left(\mathbf{W}\mathbf{X}^{\ell}\mathbf{H}^{\ell} + \mathbf{b}^{\ell}\right)$$

  - Topology-based graph filter $\mathbf{W} \in \mathbb{R}^{N \times N}$

$$[\mathbf{W}]_{ij} = 0 \text{ if } (i,j) \notin \mathcal{E}$$

  - Feature filters $\{\mathbf{H}^{\ell}\}$ explore higher-dim. mapping

If lines are sparse $|\mathcal{E}| \sim \mathcal{O}(|\mathcal{V}|)$ and let $D = \max_t\{d_t\}$, then the number of parameters for each GNN layer is

$$\mathcal{O}(N + D^2)$$

Compared to FCNN $\mathcal{O}(N^2)$

Hamilton, William L. "Graph representation learning." 2020.
https://www.cs.mcgill.ca/~wlh/grl_book/

GNN for predicting LMPs

- LMP prediction [Ji et al'16, Geng et al'16]
- GNN-based LMP can determine the optimal p/f

$$\mathbf{X} \xrightarrow{f(\mathbf{X};\boldsymbol{\theta})} \hat{\boldsymbol{\pi}} \xrightarrow{\text{dispatch}} \hat{\mathbf{p}}^*(\hat{\boldsymbol{\pi}}) \xrightarrow{\mathbf{S}} \hat{\mathbf{f}}^*(\hat{\boldsymbol{\pi}})$$

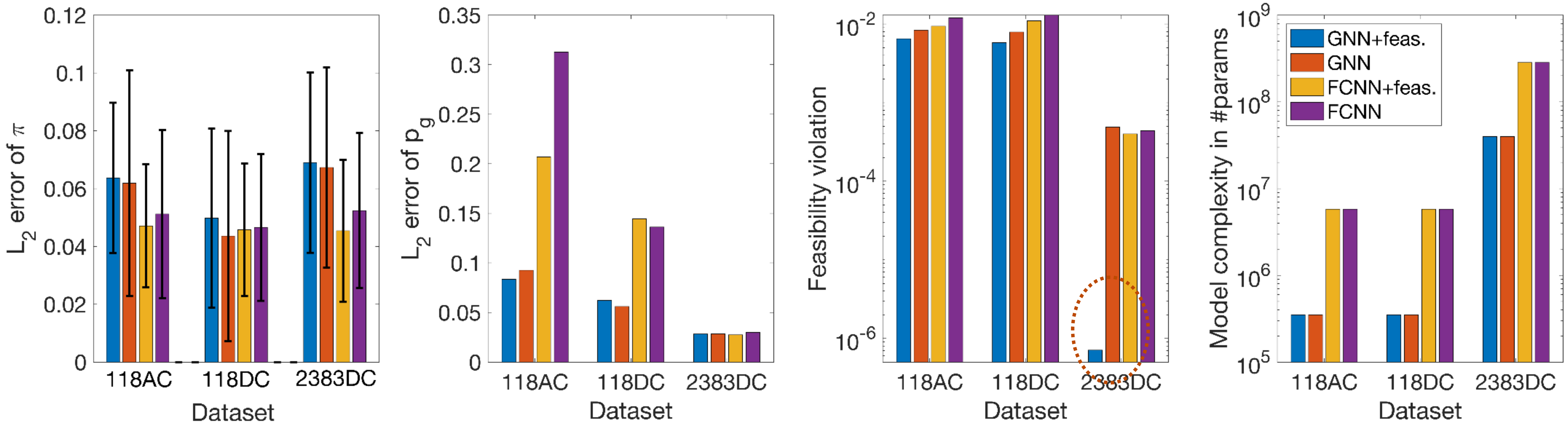
- Feasibility-regularization (FR) to reduce line flow violations

$$\mathcal{L}(\boldsymbol{\theta}) := \|\boldsymbol{\pi} - \hat{\boldsymbol{\pi}}\|_2^2 + \lambda \left\| \sigma(|\hat{\mathbf{f}}^*(\hat{\boldsymbol{\pi}})| - \bar{\mathbf{f}}) \right\|_1$$

Liu, Shaohui, Chengyang Wu, and Hao Zhu. "Graph Neural Networks for Learning Real-Time Prices in Electricity Market." *ICML Workshop on Tackling Climate Change with Machine Learning*, 2021. https://arxiv.org/abs/2106.10529

LMP prediction results

- ➢ 118-bus + ac-opf and 2382-bus + dc-opf; GNN/FCNN + feasibility regularization (FR)
- ➢ Metrics: LMP and p_g prediction error; line flow limit violation rate



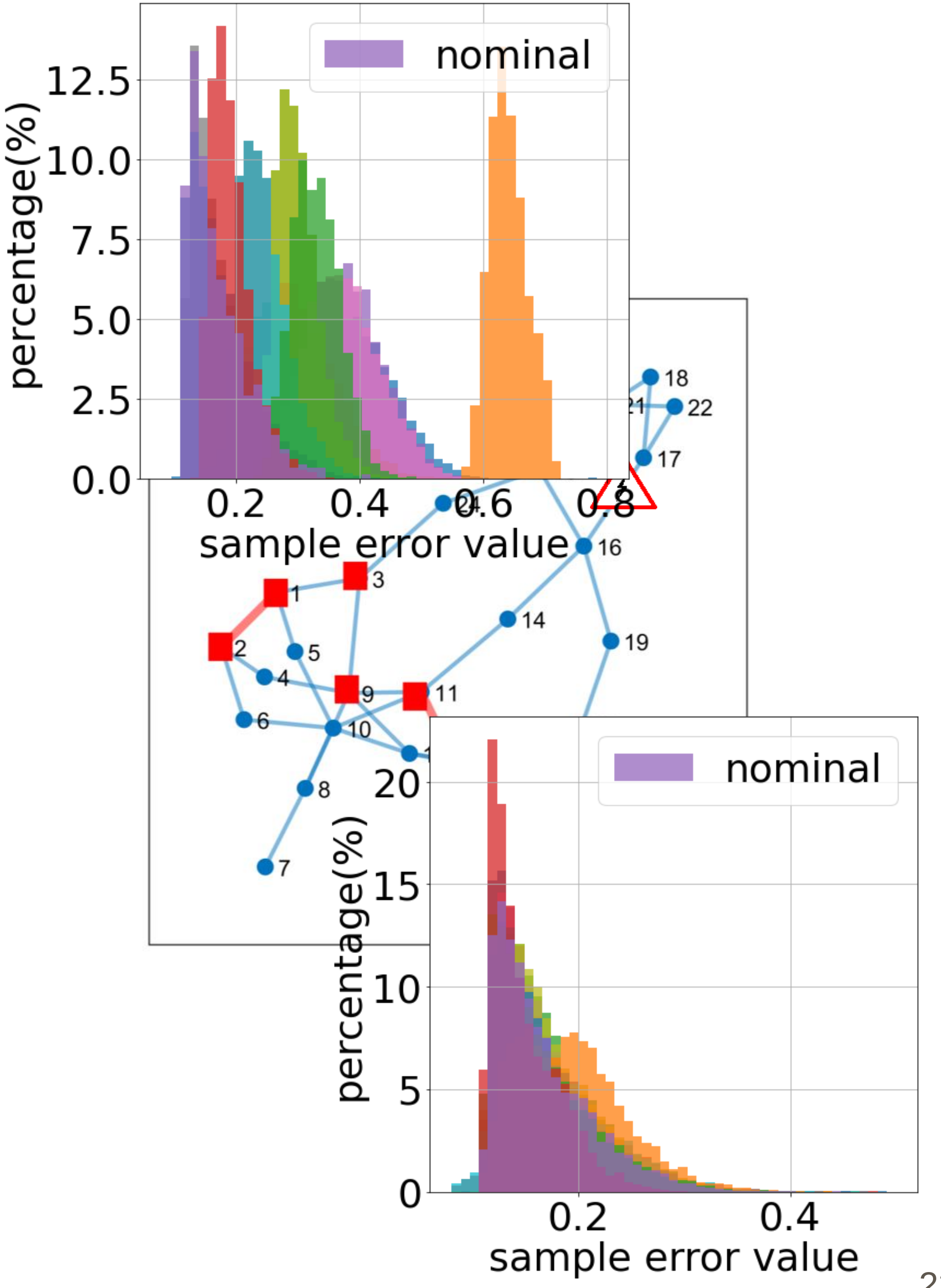
GNN for classifying congested lines

- Classifying the status for the top 10 congested lines with cross-entropy loss
- Metrics: recall (true positive rate), F1 score
- GNN better in performance scaling for large systems, thanks to reduced complexity

118ac	Recall	F1 score	**2383dc**	Recall	F1 score
GNN	98.40%	96.10%	GNN	90.00%	81.40%
FCNN	97.70%	94.60%	FCNN	87.30%	78.30%

Topology adaptivity

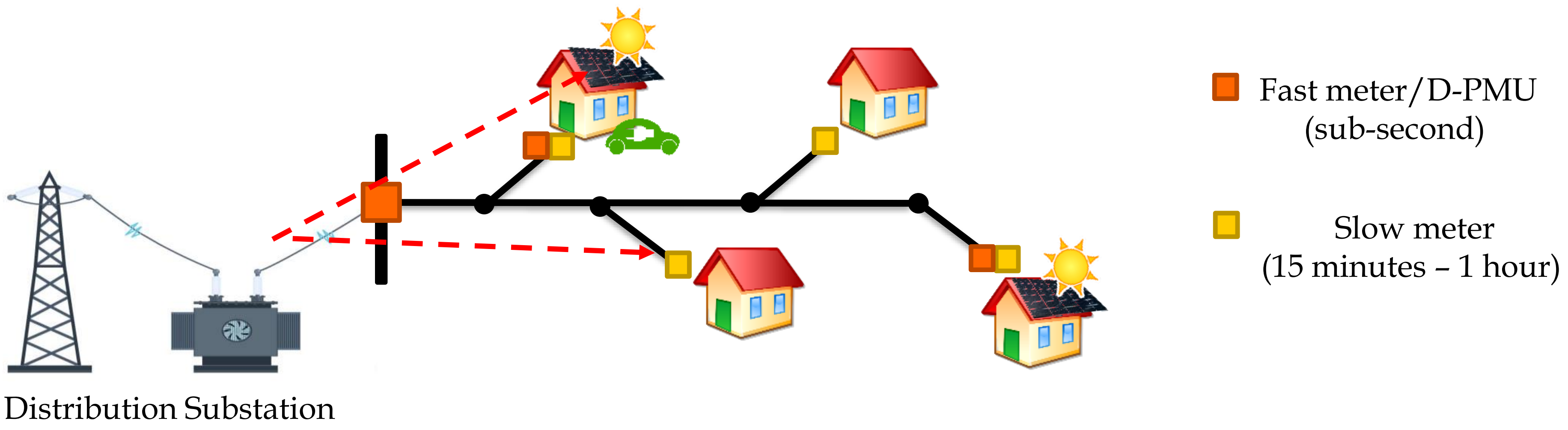
- In addition to reduced complexity, GNN-based prediction can easily adapt to varying grid topology
- Pre-trained GNN for a nominal topology can warm-start the learning for randomly selected two-line outages
- Re-trained process takes *only 3-5 epochs* to converge to good prediction
- Currently pursuing to formally analyze this transfer capability



PART II: RISK-AWARE LEARNING FOR VOLTAGE SAFETY IN DISTRIBUTION GRIDS

ML for distributed energy resources (DERs)

- Rising DERs at grid edge motivate scalable & efficient coordination to support the operations of connected distribution grids
  - Lack of frequent, real-time communications
  - Distribution control center or DMS may broadcast messages to the full system



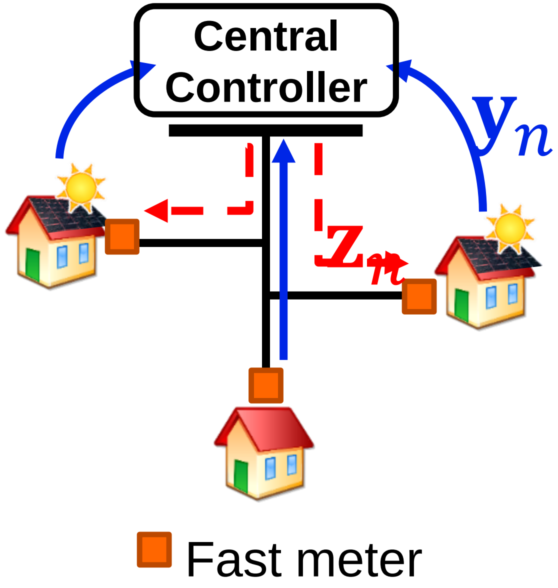
Liu, Hao Jan, Wei Shi, and Hao Zhu. "Hybrid voltage control in distribution networks under limited communication rates." *IEEE Transactions on Smart Grid* 10.3 (2018): 2416-2427.
Molzahn, Daniel K., et al. "A survey of distributed optimization and control algorithms for electric power systems." *IEEE Transactions on Smart Grid* 8.6 (2017): 2941-2962.

Existing work and our focus

- Scalable DER operations as a special instance of OPF
  - Kernel SVM learning [Karagiannopoulos et al’19],[Jalali et al’20]
  - DNNs for ac-/dc-OPF [see Part I]
  - Reinforcement learning (RL) [Yang et al’20, Wang et al’19]
- Enforcing network constraints is challenging
  - Heuristic projection or penalizing the violations

Focus: Address the statistical risks to *ensure safe operational grid limits*

Optimal DER coordination



➢ DERs for voltage regulation and power loss reduction

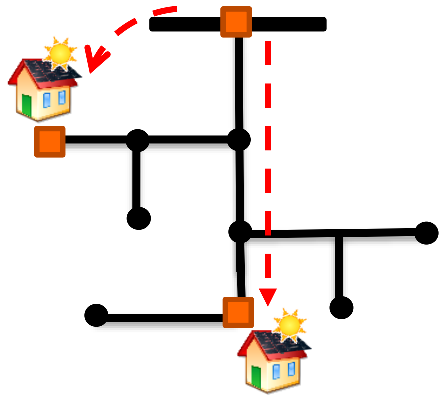
$$\mathbf{z} = \min_{\mathbf{q} \in \mathcal{Q}} \; Losses(\mathbf{q})$$
$$\text{s. to} \quad \begin{bmatrix} \mathbf{Xq} + \mathbf{h}(\mathbf{y}) - \overline{\mathbf{v}} \\ -\mathbf{Xq} - \mathbf{h}(\mathbf{y}) + \underline{\mathbf{v}} \end{bmatrix} \leq \mathbf{0}$$

- $\mathcal{Q}$: available reactive power
- $\mathbf{X}$: network matrix
- $\mathbf{y}$: operating condition
- $\underline{\mathbf{v}}, \overline{\mathbf{v}}$: voltage limits

➢ (Multi-phase) linearized dist. flow (LDF) model leads to a convex QP

➢ But a centralized solution requires high communication rates

ML for DER optimization



- Similar to OPF, want to predict $\Phi(\mathbf{y};\boldsymbol{\varphi}) \;\rightarrow\; \mathbf{z}$

- Learn a *scalable* NN model, one for each node n

$$\mathbf{y}_n^{\ell+1} = \sigma(\mathbf{W}_n^{\ell}\mathbf{y}_n^{\ell} + \mathbf{b}_n^{\ell})$$

  - $\boldsymbol{\varphi} = \{\mathbf{W}_n^{\ell}, \mathbf{b}_n^{\ell}\}$: nodal weights to be learned

- Similarly, we can use GNN architecture such that all nodes use the same filter
- *Average* loss function: mean-square error (MSE)

$$\min_{\boldsymbol{\varphi}} f(\boldsymbol{\varphi}) := \frac{1}{K}\sum_{k=1}^{K} \ell\left(\Phi(\mathbf{y}_k;\boldsymbol{\varphi}), \mathbf{z}_k\right) \quad \text{with} \quad \ell\left(\Phi(\mathbf{y}_k;\boldsymbol{\varphi}), \mathbf{z}_k\right) = \|\Phi(\mathbf{y}_k;\boldsymbol{\varphi}) - \mathbf{z}_k\|_2^2$$

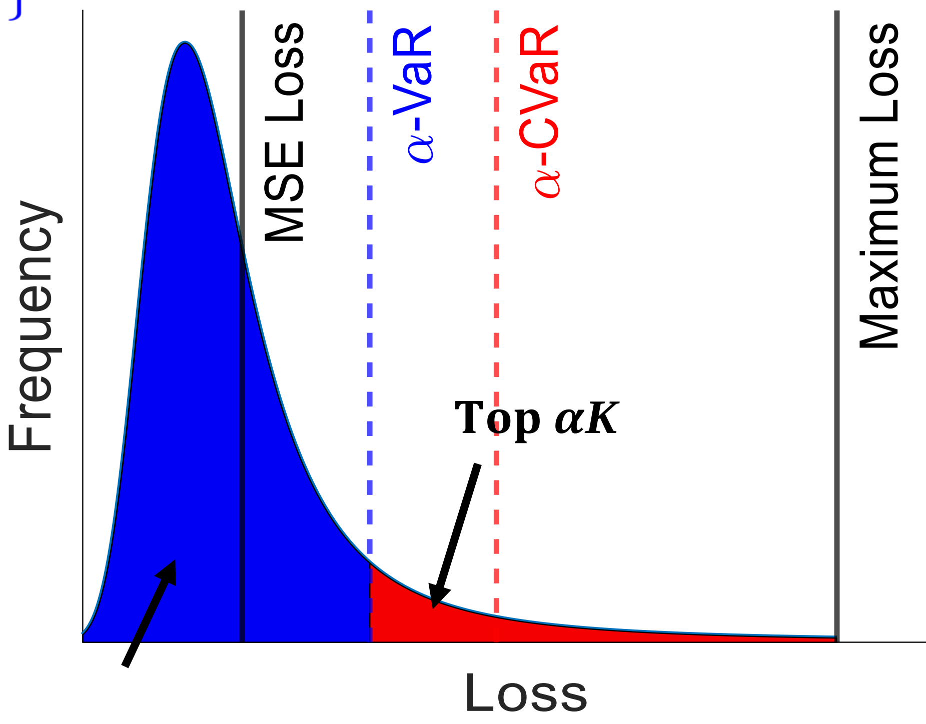
Risk-aware learning

- Consider the conditional value-at-risk (CVaR) for predicting **z**

$$\gamma_\alpha(\boldsymbol{\varphi}) := \frac{1}{\alpha K}\sum_{k=1}^{K}\ell(\Phi(\mathbf{y}_k;\boldsymbol{\varphi}),\mathbf{z}_k)\times \mathbb{1}\{\ell(\Phi(\mathbf{y}_k;\boldsymbol{\varphi}),\mathbf{z}_k)\geq v\}$$

for a given significance level $\alpha \in (0,1)$

$$\min_{\boldsymbol{\varphi}}\; f(\boldsymbol{\varphi}) + \lambda\gamma_\alpha(\boldsymbol{\varphi})$$



- λ: regularization hyperparameter
- CVaR turns out very useful for voltage constraints

Shanny Lin, Shaohui Liu, and Hao Zhu. "Risk-Aware Learning for Scalable Voltage Optimization in Distribution Grids," *Power Systems Computation Conference (PSCC) 2022 (accepted)*, https://arxiv.org/abs/2110.01490

Accelerating CVaR learning

- CVaR loss is known to preserve convexity of loss function
  - But the NN model is typically nonconvex; recent extension [Kalogerias'21]

- A key computation challenge is learning efficiency with worst-case samples

$$\gamma_\alpha(\varphi) := \frac{1}{\alpha K} \sum_{k=1}^{K} \ell(\Phi(\mathbf{y}_k; \varphi), \mathbf{z}_k) \times \mathbb{1}\{\ell(\Phi(\mathbf{y}_k; \varphi), \mathbf{z}_k) \geq v\}$$

  - Modern sampling-based ML tools reduces the accuracy of gradient computation

- We developed a straightforward mini-batch selection algorithm (Alg. 1 later) that only uses those of sufficient risk value for computing gradient

Risk of predicting q decisions

- IEEE 123-bus system with six DER nodes of flexible **q** output
  - All DERs use limited power information to learn the optimal decision
- Error performance very similar due to the high prediction accuracy
- Yet, training time accelerated by CVaR and the proposed selection algorithm

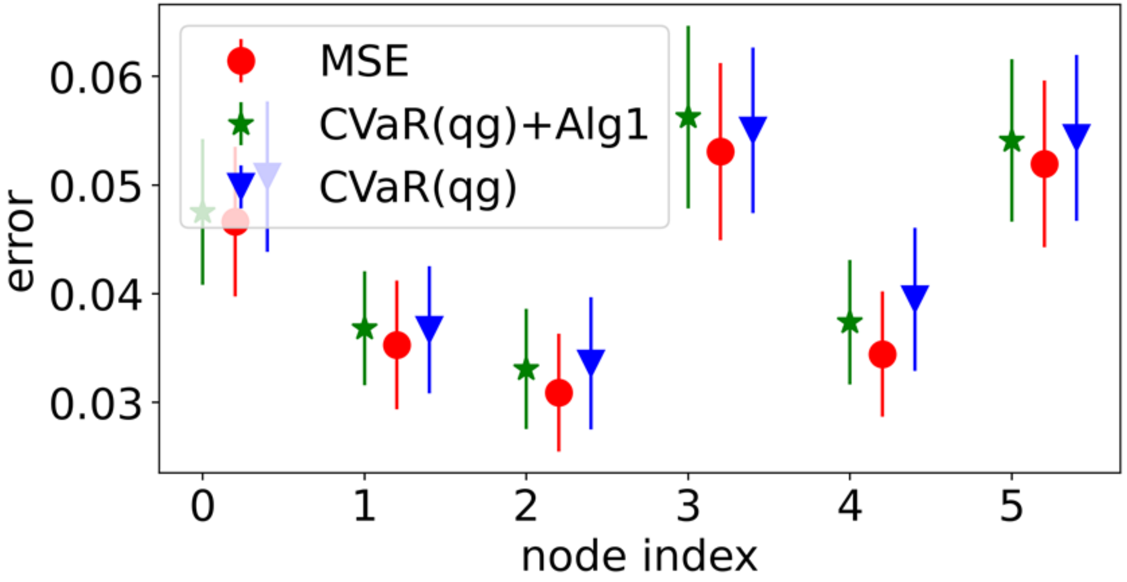


Table 1: Computation time

Loss obj.	Epoch [s]	Total [s]
MSE	0.52	46.48
CVaR(qg)	1.07	38.70
CVaR(qg)+Alg 1	0.61	35.63

Risk of voltage violation

- Further incorporating the CVaR of voltage prediction
- Reduced max voltage deviation (worst-case) -> higher operational safety
- Computational efficiency improved by the proposed selection algorithm

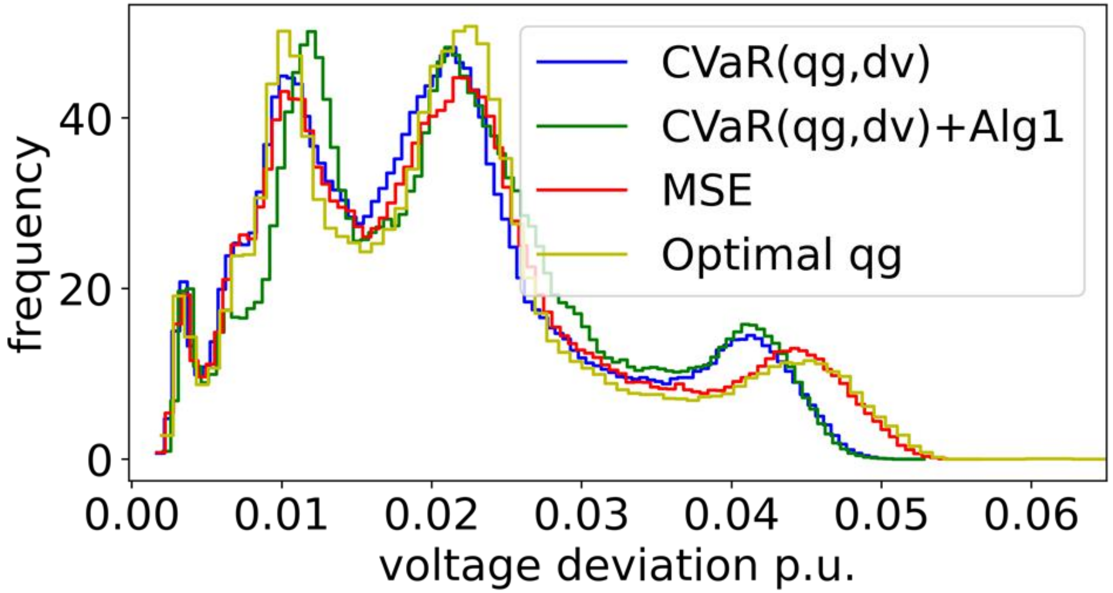


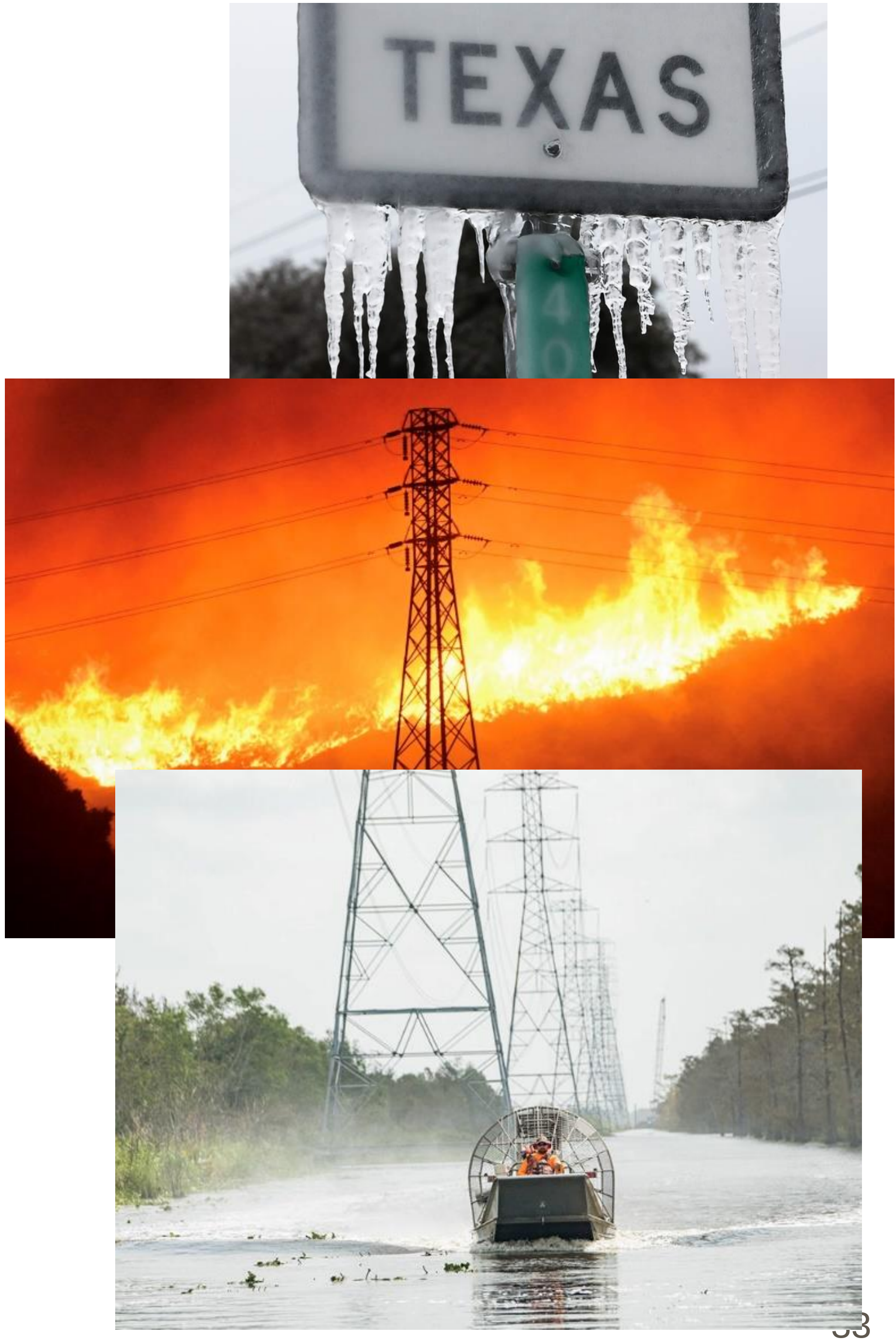
Table 1: Computation time

Loss obj.	Epoch [s]	Total [s]
MSE	0.54	44.89
CVaR(qg,dv)	0.77	31.73
CVaR(qg,dv)+Alg 1	0.51	25.93

PART III: SCALABLE LEARNING OF EMERGENCY RESPONSES FOR RESILIENCE

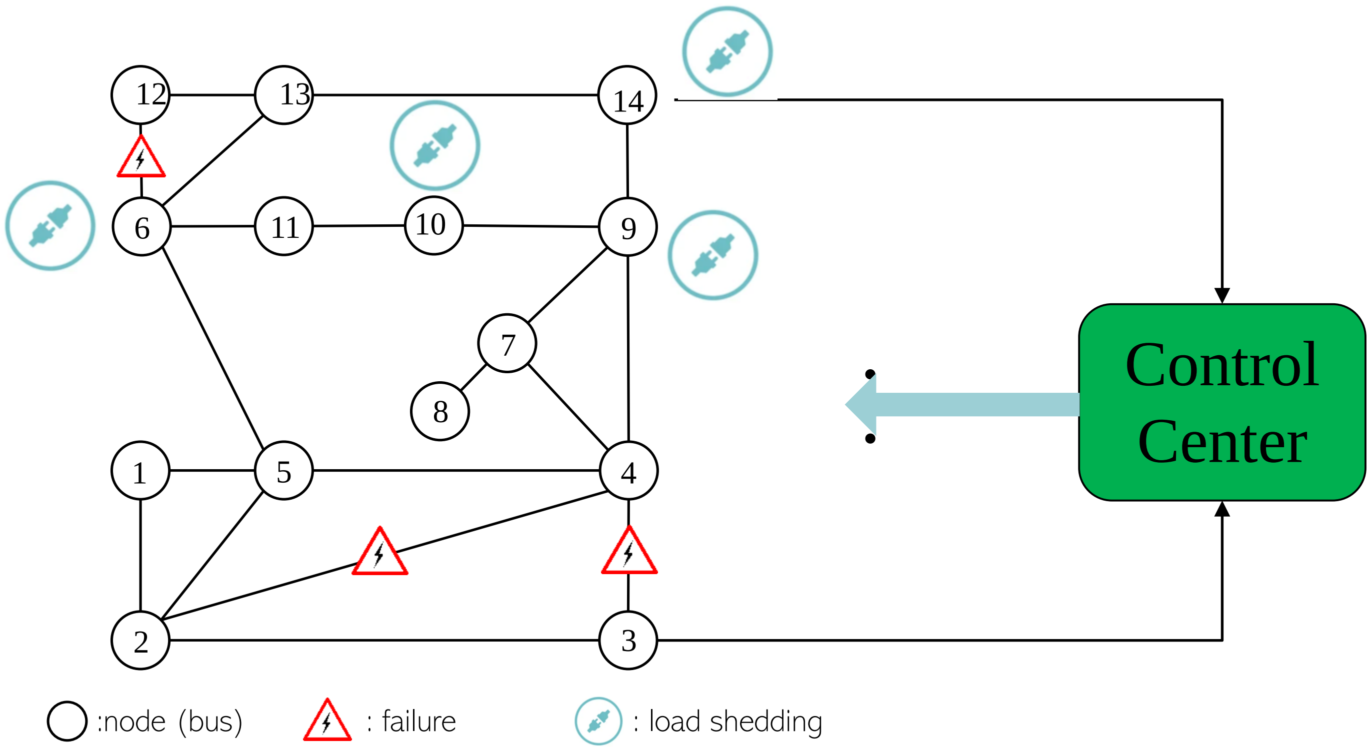
Grid emergency responses

- Grid resilience challenged by emerging types of variable energy resources (VERs), and increasingly by extreme weather events
- It imperative to design the grid operations with effective emergency responses
  - Load shedding
  - Topology optimization
  - ...
- How to attain the decisions in a scalable and safe manner?



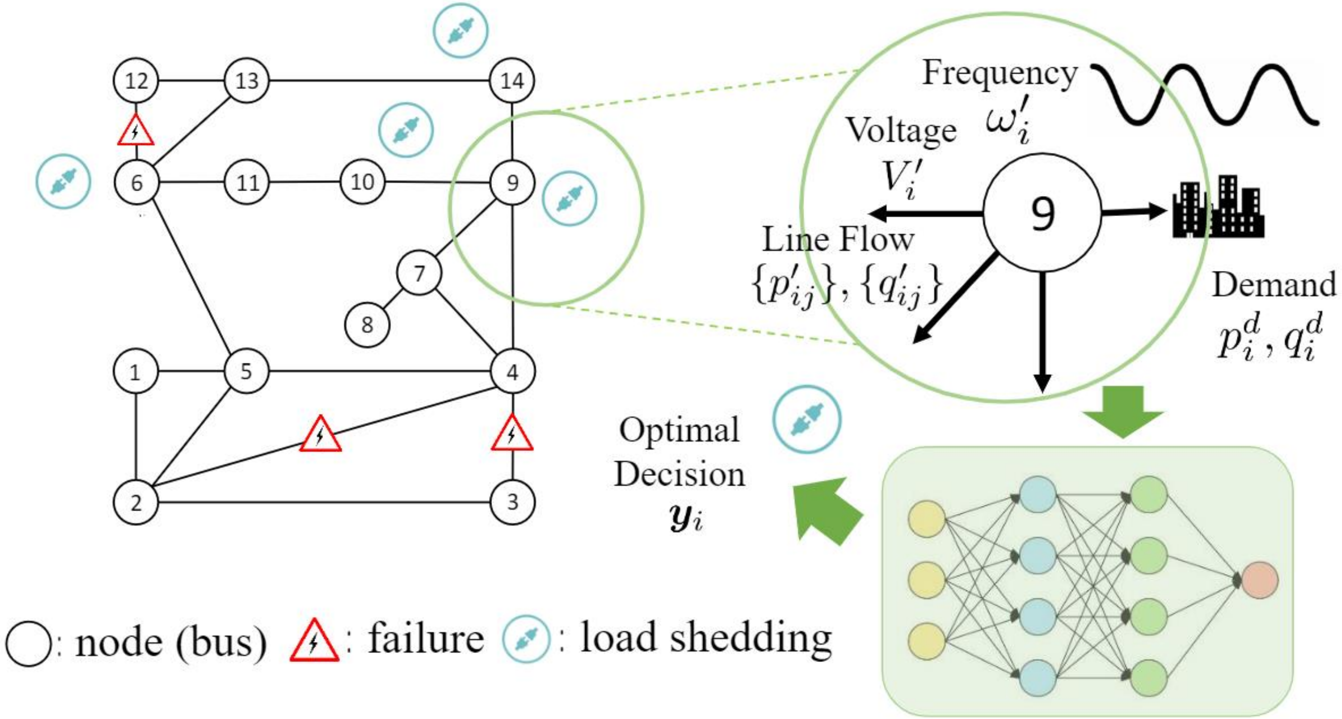
Centralized optimal load shedding (OLS)

- Load shedding determined by control center with system-wide information
- AC Optimal load shedding (OLS) program cast as a special case of AC-OPF



ML for decentralized load shedding

- Each load learns optimal decision rule from a large of historical or synthetic scenarios



- Input feature:

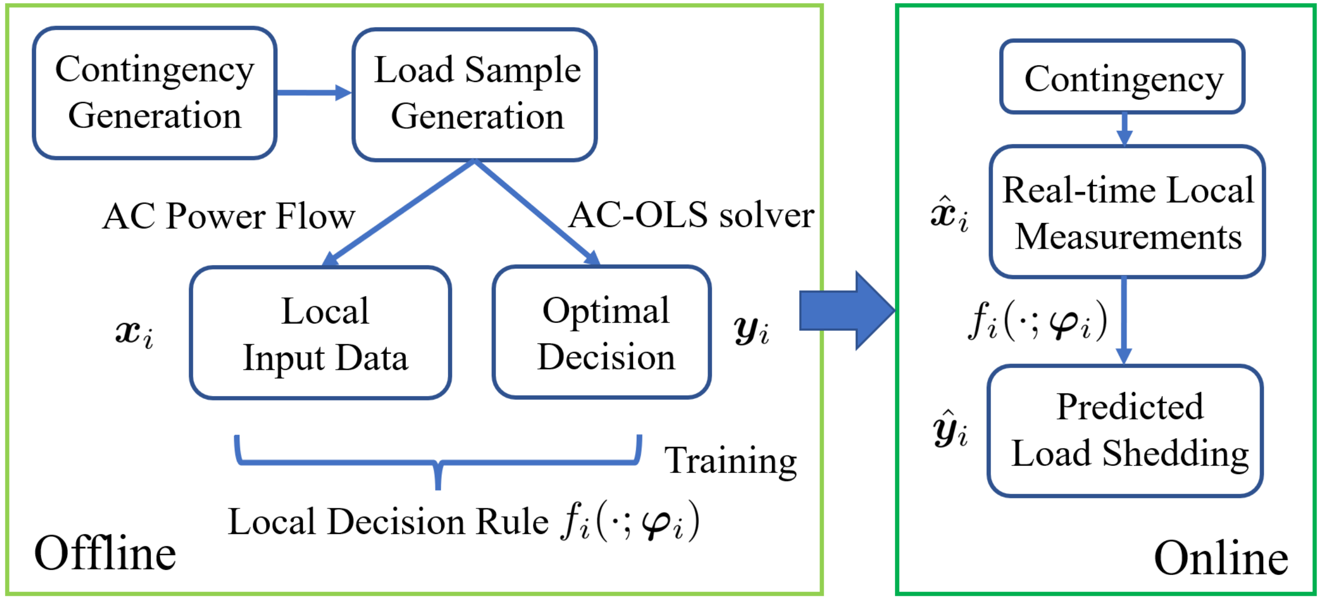
$$\boldsymbol{x}_i = [p_i^d, q_i^d, V_i', \{p_{ij}'\}, \{q_{ij}'\}, \omega_i']$$

- Local shedding solutions:

$$\boldsymbol{y}_i = [p_i^s, q_i^s]$$

Yuqi Zhou, Jeehyun Park, and Hao Zhu, "Scalable Learning for Optimal Load Shedding Under Power Grid Emergency Operations," PES General Meeting (PESGM) 2022 (accepted) https://arxiv.org/abs/2111.11980

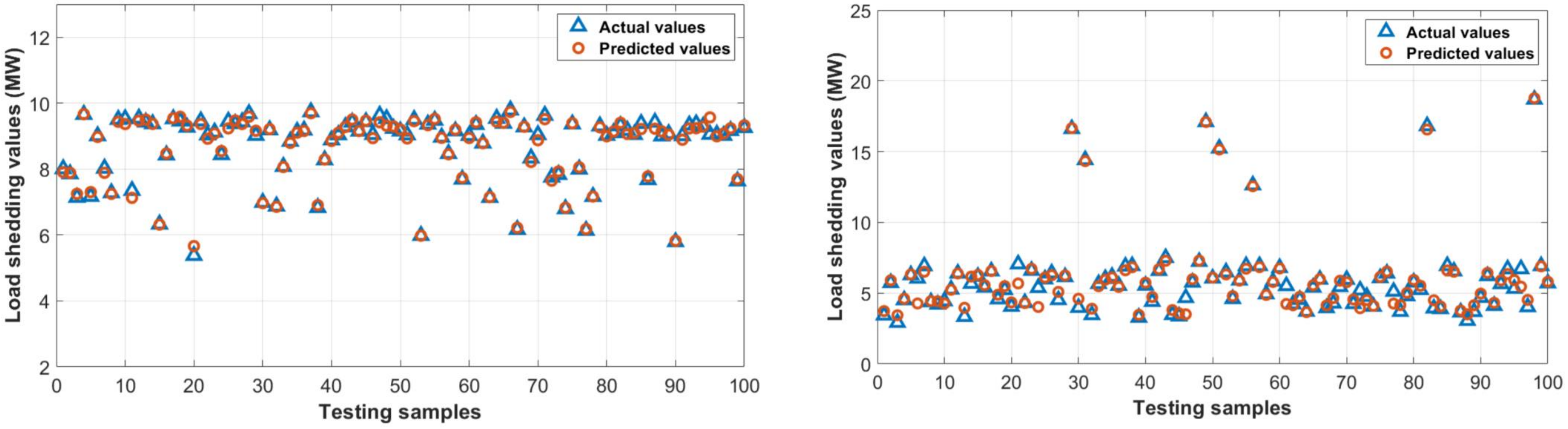
Scalable learning of load shedding



- Offline training is performed for various contingency and load conditions
- Load centers quickly make decisions during online phase in response to contingencies.

Prediction under single line outage

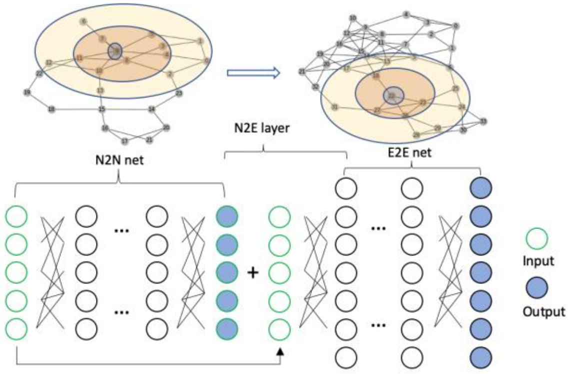
- IEEE 14-bus system; quadratic cost functions
- All $(N-1)$ contingency scenarios, under different load conditions (1000 samples for each scenario)



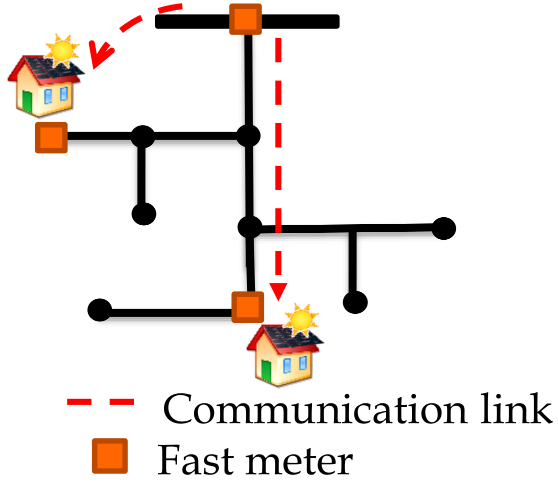
(a) Load center at bus 10

(b) Load center at bus 14

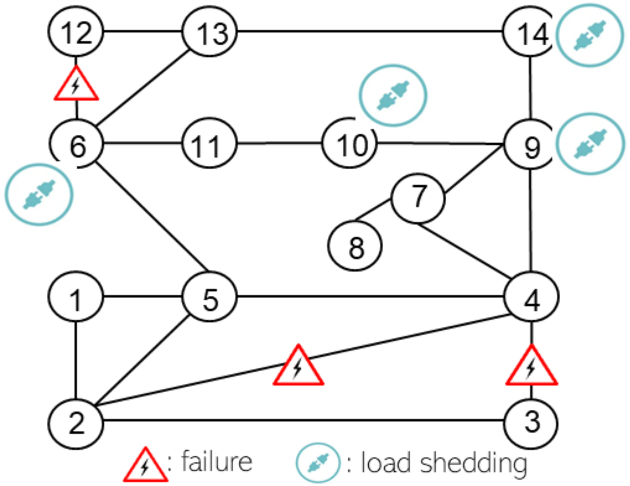
Summary



Topology-aware learning for real-time market:
Simpler model for efficient training



Risk-aware learning for DER coordination:
Reduced risks of voltage violations



Scalable learning for grid emergency responses:
Fast mitigations under limited data

- I: Topology adaptivity and other transfer learning ideas
- II: Convergence analysis and connections to safe learning
- III: Generalized emergency responses and risk-awareness

Education resources

- UT grad course *"Data Analytics in Power Systems,"* new slides available
  https://utexas.app.box.com/v/EE394VDataInPowerSys
- 2020 NSF Workshop on *Forging Connections between Machine Learning, Data Science, & Power Systems Research*
  https://sites.google.com/umn.edu/ml-ds4pes/home
- DOE-funded EPRI GEAT with Data
  https://grided.epri.com/great_with_data.html

Learning and Optimization
for Smarter Electricity Infrastructure

Hao Zhu
haozhu@utexas.edu
http://sites.utexas.edu/haozhu/
@HaoZhu6

Learning for grid resilience

Learning for dynamic resources

Learning for power electronics based resources

….

Thank you!