The University of Texas at Austin
Electrical and Computer Engineering

PSERC Summer Tutorial
July 13, 2021

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

Hao Zhu
Assistant Professor
Department of ECE
The University of Texas at Austin

Acknowledgements: Shanny Lin, Kyung-bin Kwon, Shaohui Liu, NSF Grants 1802319, 1807097, 1952193, and DOE-EPRI GREAT-Data

Power of AI

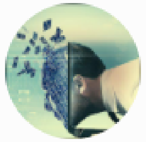
- Unprecedented opportunities offered by diverse sources of data
  - Synchrophasor and IED data
  - Smart meter data
  - Weather data
  - GIS data, ...

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

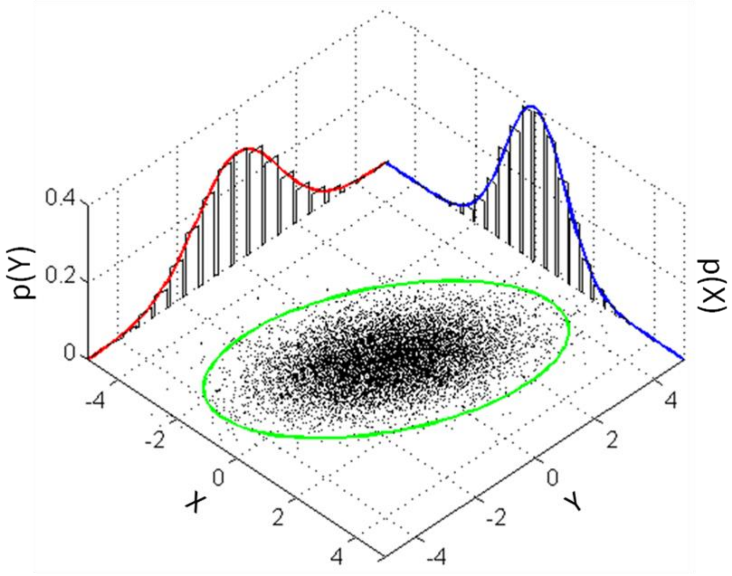
SUSTAINABLE ENERGY

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

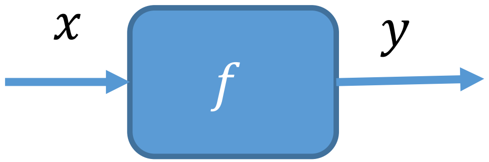
A primer on supervised learning

- *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 objective

- Goal: construct a function $f: \mathbb{R}^d \rightarrow Y$ to map $x \rightarrow y$
  - *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 $l(\hat{y}, y) = ||\hat{y} - y||_p$
  - For classification, cross-entropy loss, hinge loss, ect.



$$f^\star = \arg\min_{f \in F} \mathbb{E}_{(x,y)} l(f(x), y)$$

Sample Mean

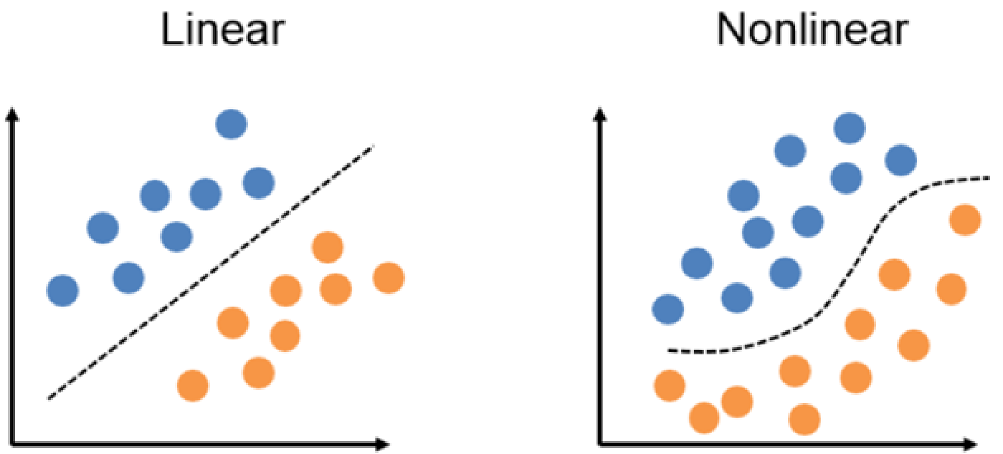
$$\hat{f} = \arg\min_{f \in F} \left(\frac{1}{K}\right)\sum_{k=1}^{K} l(f(x_k), y_k)$$

- 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).

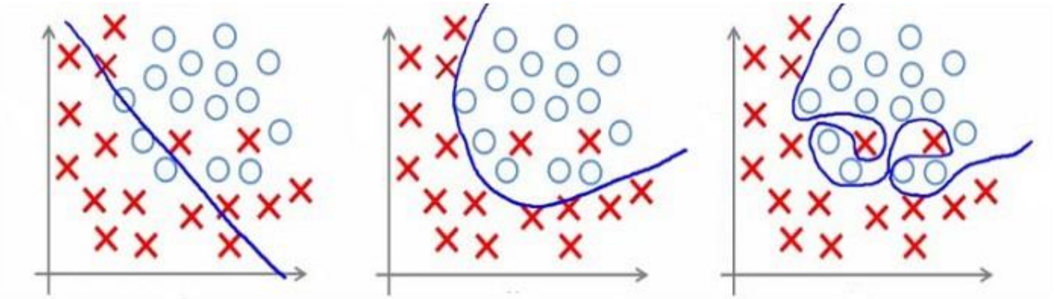
Parameterized models for f

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



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

Regularization



- Data overfitting (losses → 0)
  - Features correlated: 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

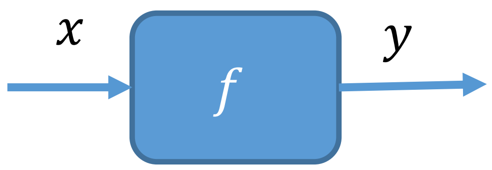
$$\hat{f} = \arg\min_{f \in F} \sum_{k=1}^{K} l(f(x_k), y_k) + \lambda \cdot \mathrm{Reg}(f)$$

norm of parameter *w*

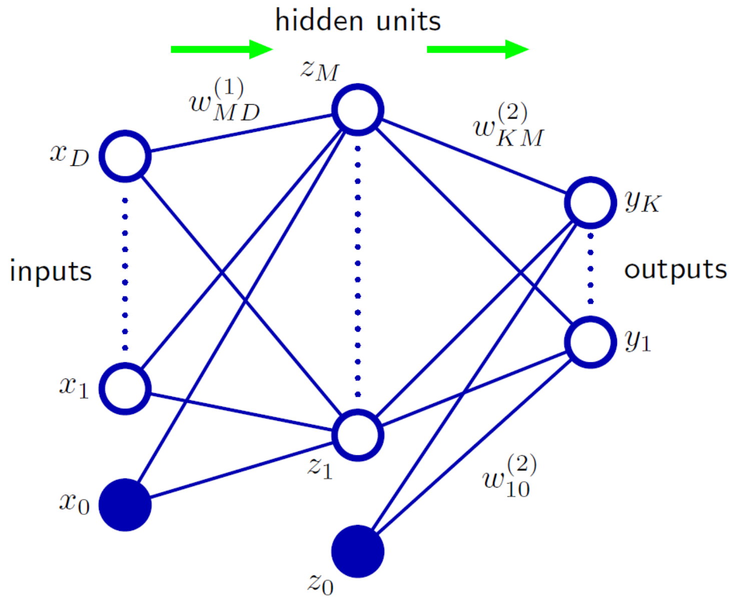
  - Hyperparameter $\lambda > 0$ balances between data fitting and model complexity
  - L_2 norm/Ridge: small values, or smooth using $\sum_i (w_i - w_{i-1})^2$
  - L_1 norm/Lasso: sparse w (much more zero entries)

Deep (D)NN architecture

- Perceptron (single-layer NN): convert $f(x) = w^\top x$ to nonlinear one 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 nonlinear ones 3 layers are sufficient



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

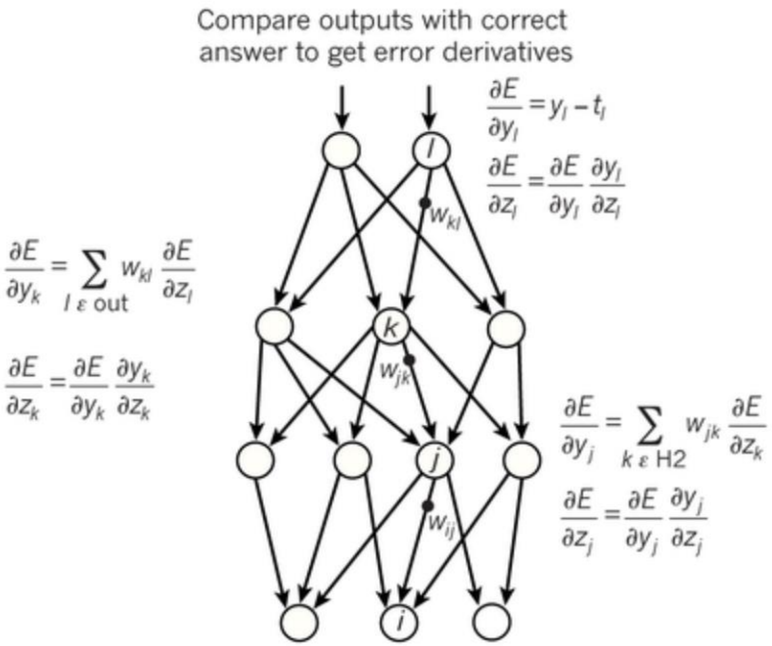
Gradient descent (GD) via *backpropagation*

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

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

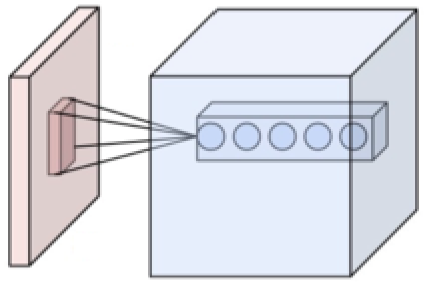
$$w \leftarrow w - \alpha \nabla E(w)$$

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

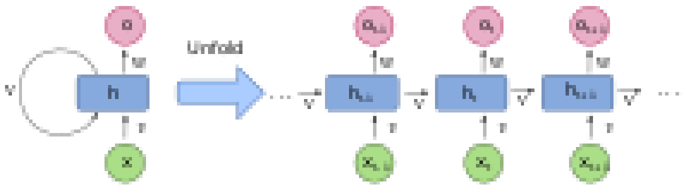


Variations of DNN

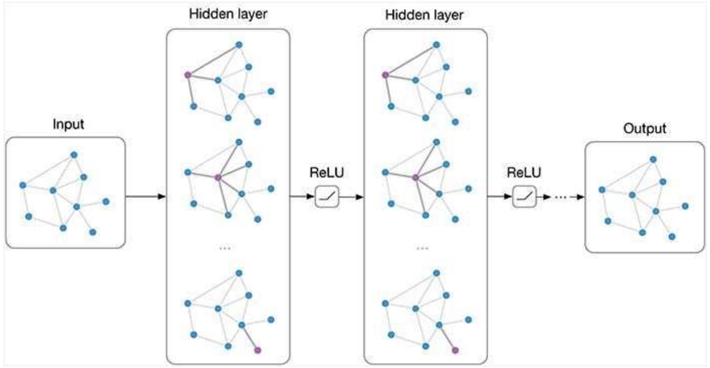
- Fully-connected NN (FCNN): weight parameters grow with data size
- Idea: reuse the weight parameters, aka, filters!



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



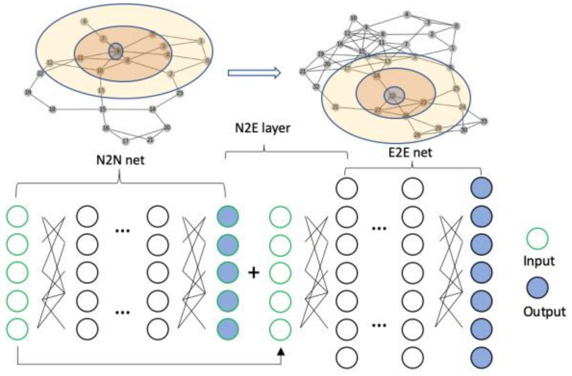
Recurrent NN (RNN):
Temporal filters for texts, speech



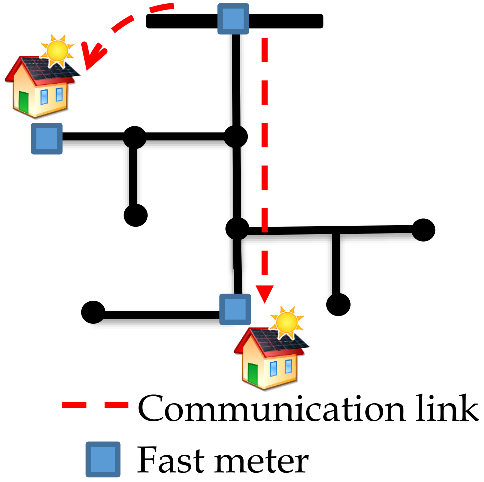
Graph NN (GNNs):
Graph filters for networked systems

Overview

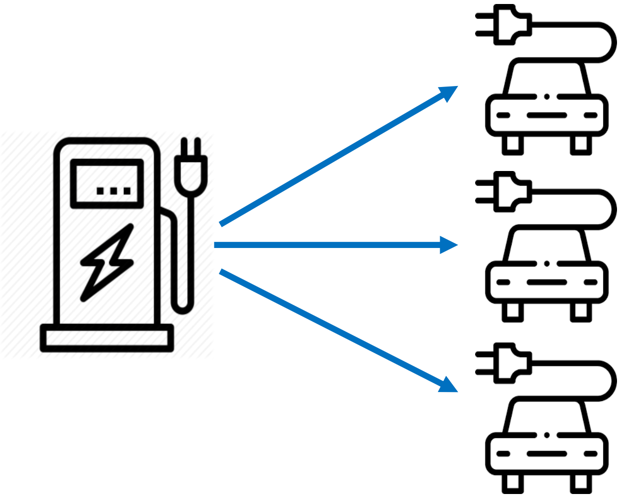
➢ We visit three problems that use domain knowledge to better design NN models that are physics-informed and risk-aware



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



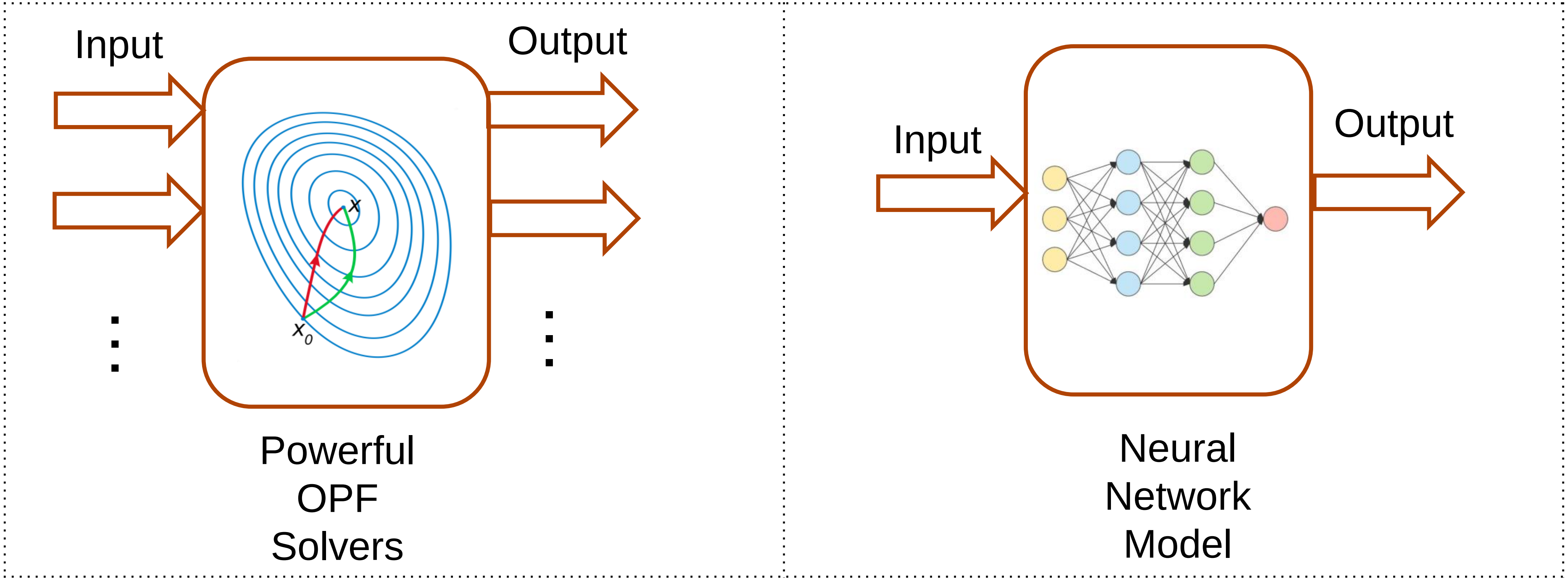
Risk-aware learning for DER coordination:
Limiting violations of voltage risks



Efficient representation for dynamical resources:
Time-varying problem dimension

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 variability of power systems and motivates real-time, adaptive, fast OPF
  - 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, without loss of generality (wlog)

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

- Nodal input:

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

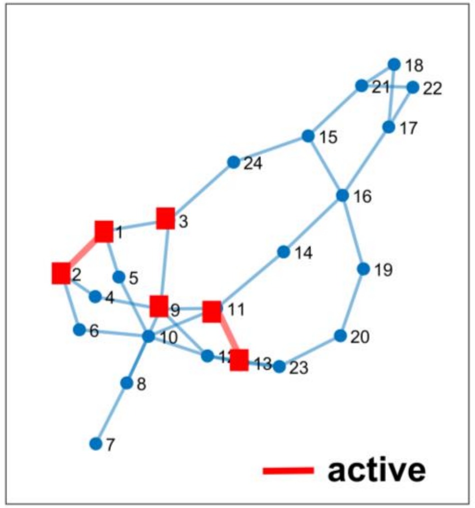
  power limits + costs

- Nodal output: optimal p/q ?

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

Topology dependence

- Earlier work [Owerko et al'20] using GNN to predict p/q
- Locational marginal price (LMP) from the dual problem
  - Typically, very few congested lines



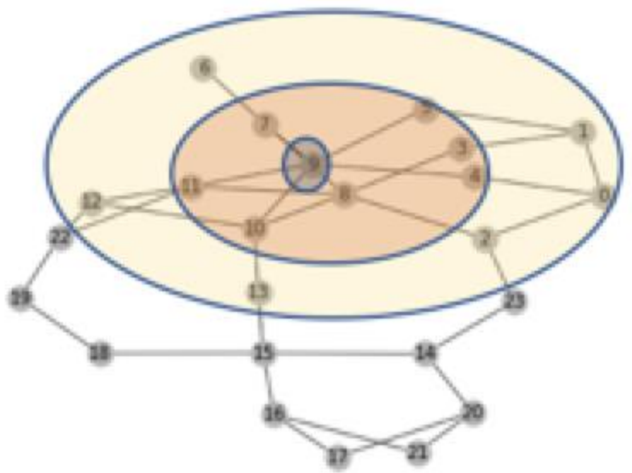
$$\begin{aligned} \min_{\mathbf{p}} \quad & \sum_{i=1}^{N} c_i(p_i) \\ \text{s.t.} \quad & \mathbf{1}^\top \mathbf{p} = 0 & : \lambda \\ & \underline{\mathbf{p}} \le \mathbf{p} \le \bar{\mathbf{p}} \\ & \underline{\mathbf{f}} \le \mathbf{S}\mathbf{p} \le \bar{\mathbf{f}} & : [\underline{\boldsymbol{\mu}};\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

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

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

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

Graph NN (GNN)



- 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 learning prices & congestion

- LMP/congestion prediction work [Ji et al'16, Geng et al'16]
- GNN-based LMP can predict 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

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

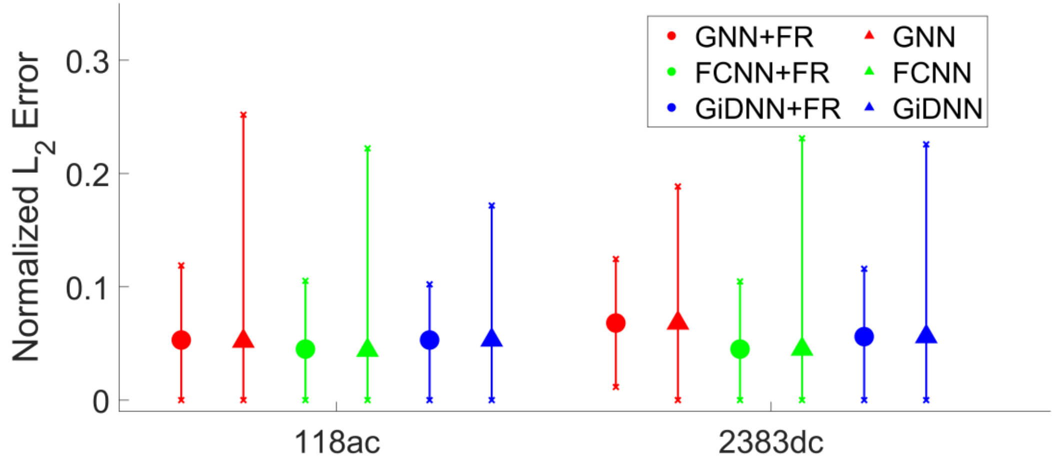
- GNN can also directly classify the status for each line
  - Cross-entropy loss, using a final fully-connected layer

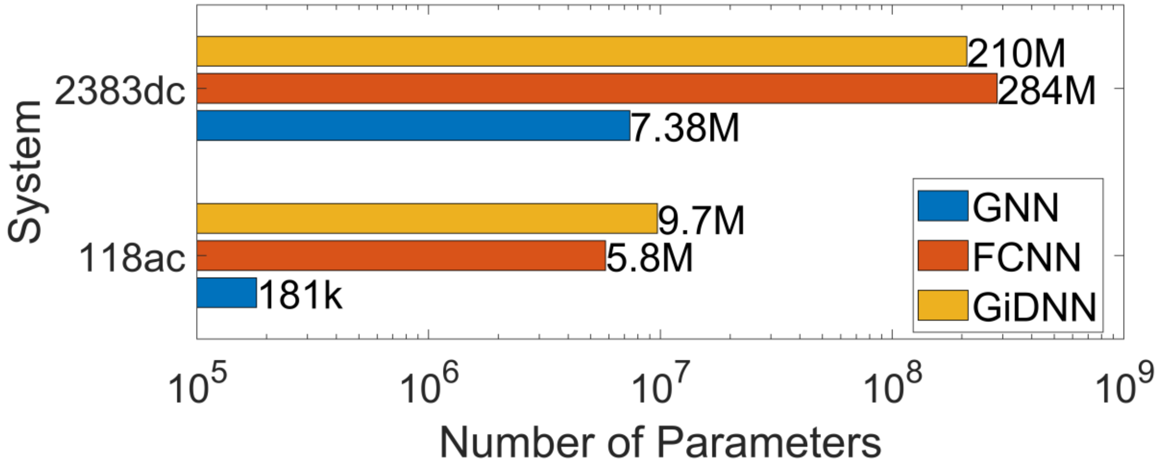
Liu, Shaohui, Chengyang Wu, and Hao Zhu. "Graph Neural Networks for Learning Real-Time Prices in Electricity Market." *ICML Workshop on Tackling Climate Change with Machine Learning*, 2021. https://arxiv.org/abs/2106.10529

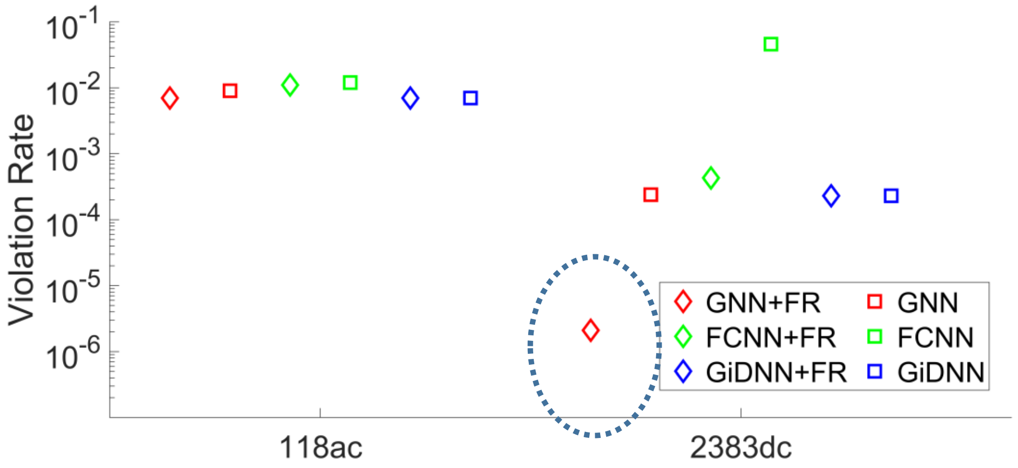
LMP prediction results

- 118-bus + ac-opf and 2382-bus + dc-opf
- Metrics: LMP prediction error;

  line flow limit violation rate
- GNN, FCNN, Graph-informed (Gi)DNN, all + feasibility regularization (FR)

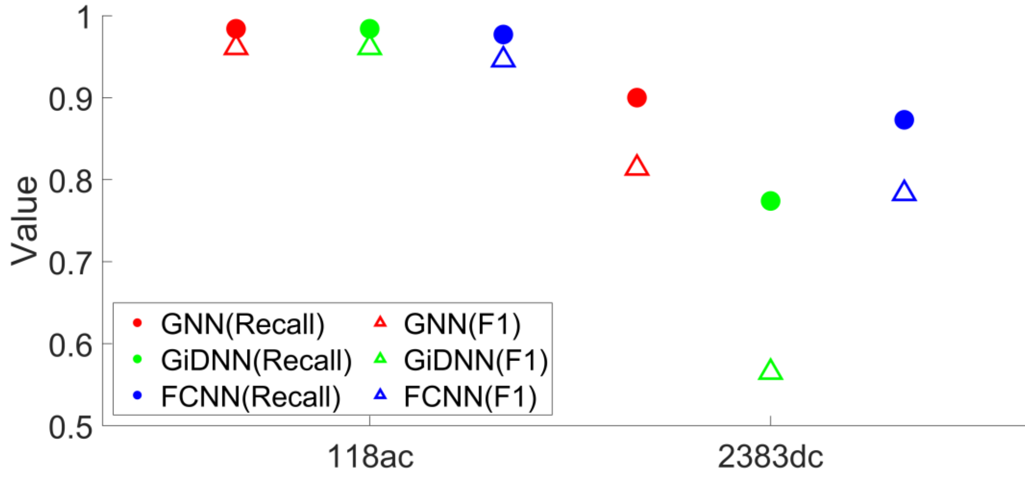






Congestion classification results

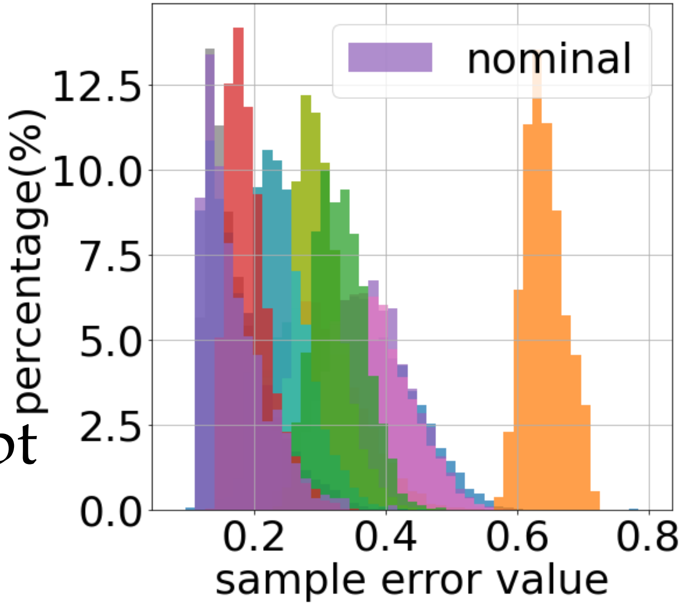
- Predicting the top 10 congested lines
- Metrics: recall (true positive rate), F1 score
- GNN maintains performance for large systems, thanks to the reduced complexity

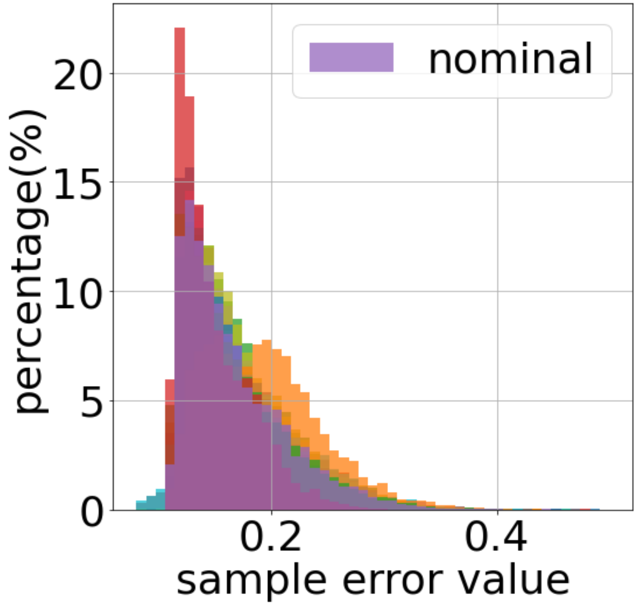


118ac	Recall	F1 score	**2383dc**	Recall	F1 score
GNN	98.40%	96.10%	GNN	90.00%	81.40%
GiDNN	98.38%	96.09%	GiDNN	77.40%	56.50%
FCNN	97.70%	94.60%	FCNN	87.30%	78.30%

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 feature

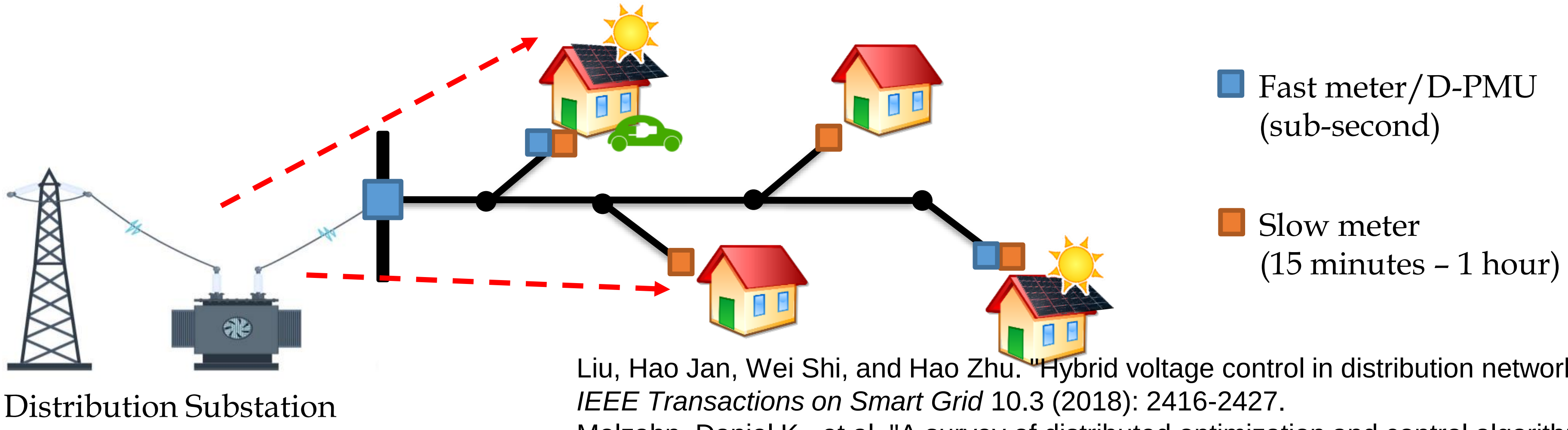




Part II: Risk-aware Learning for Voltage Safety in Distribution Grids

ML for distributed energy resources (DERs)

- Rising DERs at grid edge motivate scalable and 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



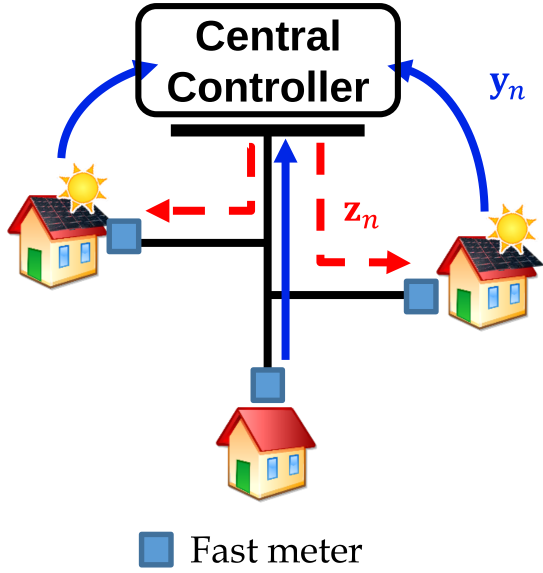
Liu, Hao Jan, Wei Shi, and Hao Zhu. "Hybrid voltage control in distribution networks under limited communication rates." *IEEE Transactions on Smart Grid* 10.3 (2018): 2416-2427.
Molzahn, Daniel K., et al. "A survey of distributed optimization and control algorithms for electric power systems." *IEEE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



- DERs for voltage regulation and power loss reduction

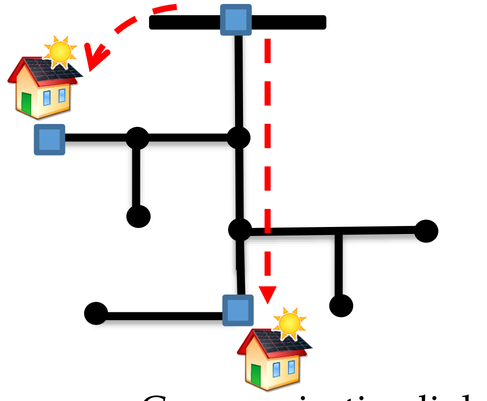
$$\mathbf{z} = \min_{\mathbf{q} \in \mathcal{Q}} \; Losses(\mathbf{q})$$

$$\text{s. to} \quad \begin{bmatrix} \mathbf{X}\mathbf{q} + \mathbf{h}(\mathbf{y}) - \overline{\mathbf{v}} \\ -\mathbf{X}\mathbf{q} - \mathbf{h}(\mathbf{y}) + \underline{\mathbf{v}} \end{bmatrix} \leq \mathbf{0}$$

- $\mathcal{Q}$: 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 $f(\mathbf{y}) \rightarrow \mathbf{z}$
- Learn a *scalable* MLP model, one for each node n

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

  - $\varphi = \{\mathbf{W}_n^\ell, \mathbf{b}_n^\ell\}$: nodal weights to be learned
  - Decentralized among all nodes (each node using local features only

- Similarly, we can use GNN architecture such that all nodes use the same filter
- Mean-squared error (MSE) based loss function: neglecting voltage limits!

$$\min_{\varphi} \ \mathcal{L}(\varphi) := \frac{1}{K} \sum_{k=1}^{K} \| f(\mathbf{y}_k; \varphi) - \mathbf{z}_k \|_2^2$$

Risk-aware Learning

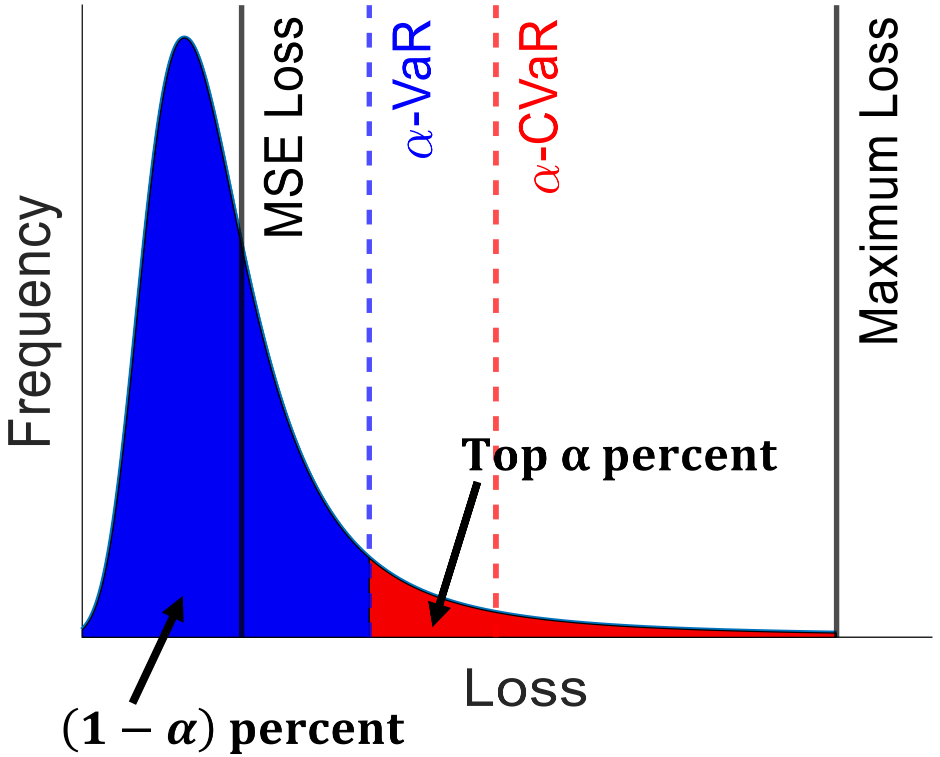
- Consider the conditional value-at-risk (CVaR) for predicting **z**

$$\gamma_\alpha(\varphi) = \mathbb{E}_{(\mathbf{y},\mathbf{z})}\left[\|f(\mathbf{y}) - \mathbf{z}\|_2^2 \,\middle|\, \|f(\mathbf{y}) - \mathbf{z}\|_2^2 \geq v_\alpha\right]$$

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

$$\min_{\varphi} \quad \mathcal{L}(\varphi) := \frac{1}{K}\sum_{k=1}^{K} \|f(\mathbf{y}_k;\varphi) - \mathbf{z}_k\|_2^2$$

$$\text{s. to} \quad \gamma_\alpha(\varphi) \leq C$$ (pre-determined threshold)

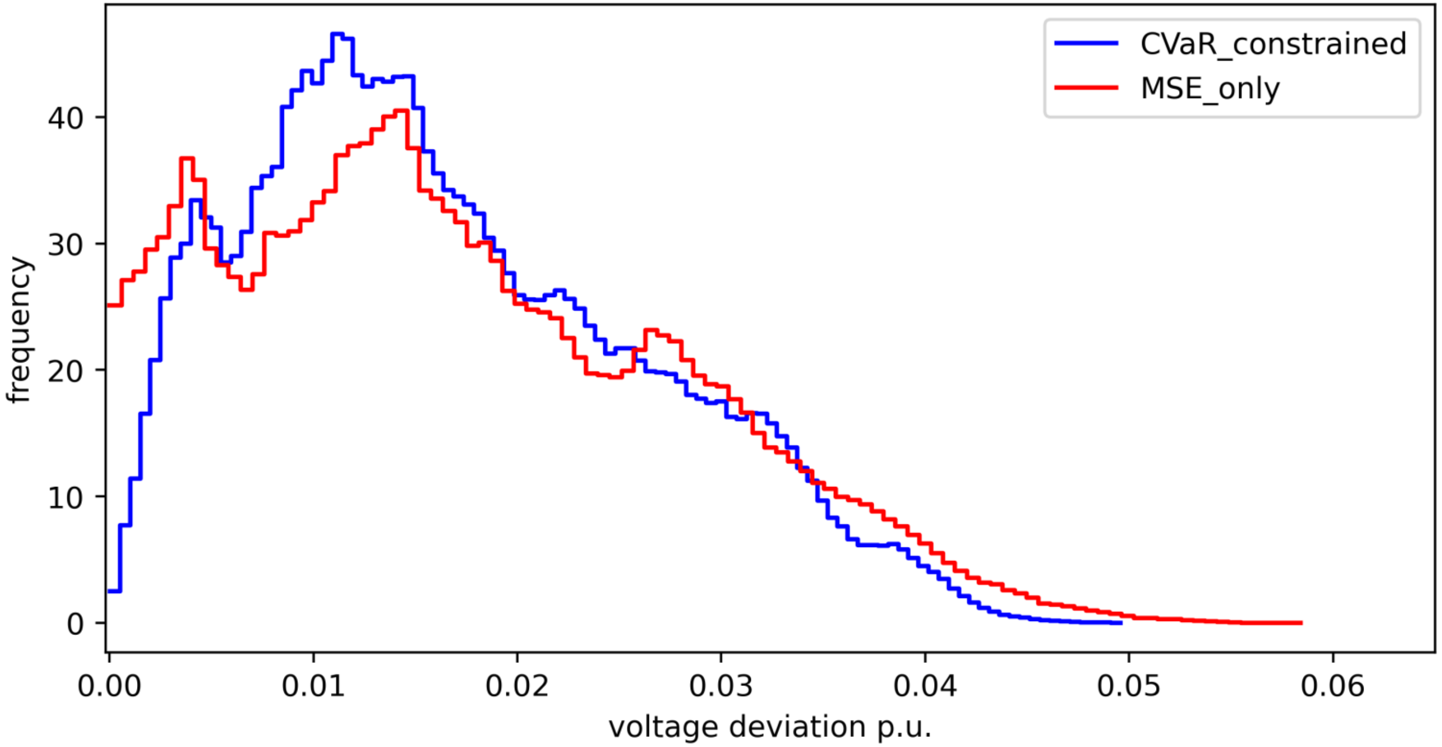


- Can consider it for the voltage constraints
- Approximate gradient can be computed

Shanny Lin, Shaohui Liu, and Hao Zhu. "Risk-Aware Learning for Scalable Voltage Optimization in Distribution Grids." *preprint, 2021.*

CVaR-constrained results

- Train NNs for 123-bus feeder using either the MSE-only loss or the risk-constrained formulation
- CVaR risk reducing the worst-case voltage limit deviations

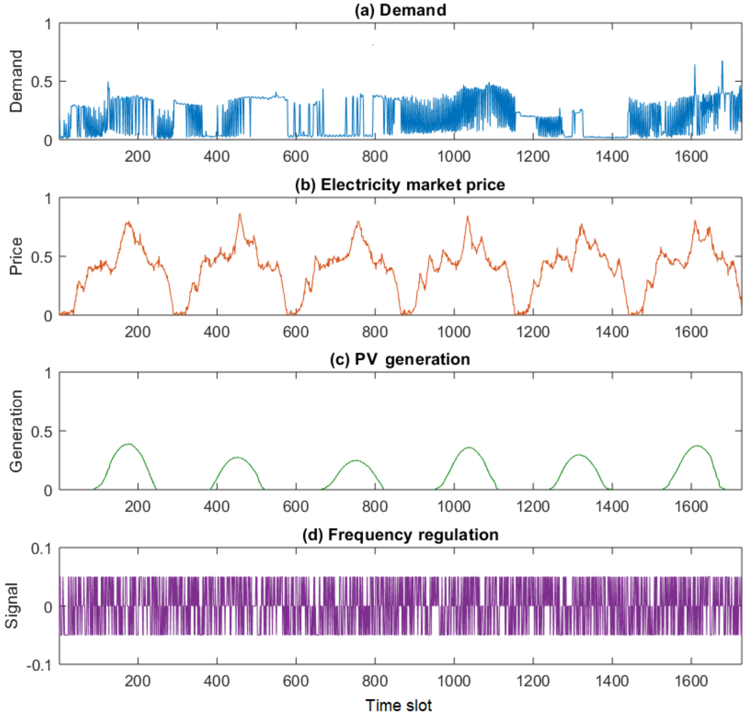


Part III: Efficient Representation for Electric Vehicle Charging using Reinforcement Learning

RL for dynamical grid resources

- So far, samples are individually generated
- Many DERs, like energy storage, EVs, or controllable loads, have internal dynamics; same for external inputs, including price/load demand/weather conditions)
- RL studies how to use past data to enhance the future decision making of a dynamical system

$$\text{learn } u_t \leftarrow \pi(x_t) \text{ from } \{x_0, u_0, x_1, u_1, \ldots, x_t, u_t, \ldots\}$$

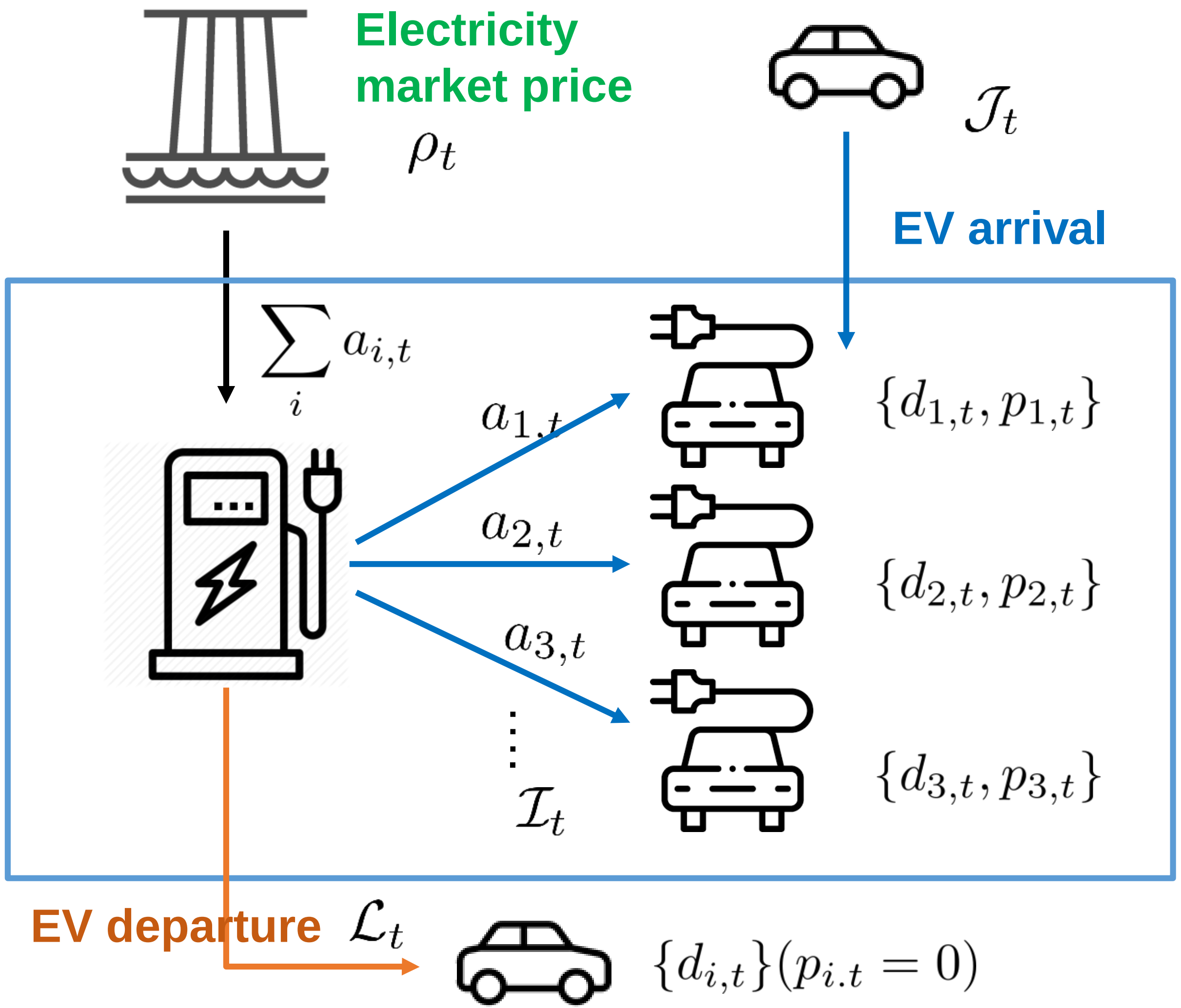


Recht, Benjamin. "A tour of reinforcement learning: The view from continuous control." *Annual Review of Control, Robotics, and Autonomous Systems* 2 (2019): 253-279. https://arxiv.org/abs/1806.09460

EV charging station (EVCS) problem

- Electrified transportation is a key to enable low-carbon energy future and address urban population issues
- Optimal operations of EVCS challenged by future uncertainty
  - Statistical modeling for EV arrival & parking time [Luo et al'18] [Huang et al'18]
  - Consideration of co-located storage/renewable [Yan et al'19] [Chen et al'17]
- RL recently advocated [Wan et'19][Li et al'20] *but incurs high complexity*
  - Existing work includes every EV's status => *time-varying problem dimension*
  - [Wang et al'21] solves this issue by designing an approximate Q-function

Problem modeling



- EVs randomly arriving
  - EV i represented by remaining demand $d_{i,t}$ and parking time $p_{i,t}$, both integer # of time slots
- EVCS decides which EVs to charge ($a_{i,t} = 1$) by purchasing electricity from grid operators
- Wlog, EV departs fully charged
  - Generalizable to a penalty on non-charged portion

Markov decision process (MDP)

- **State:** both EV-internal status and the external price

$$s_t = \{\rho_t, (d_{i,t}, p_{i,t}) \ \forall i \in \mathcal{I}_t\}$$

but $\mathcal{I}_{t+1} = \mathcal{I}_t \cup \mathcal{J}_{t+1} \backslash \mathcal{L}_{t+1}$ is time-varying and can grow quickly

- **Action:** binary decision (possibly multi-level/continuous charging)

$$a_{i,t} \in \mathcal{A} = \{0, 1\}$$

- **Transition:** EV status updates + price dynamics

$$d_{i,t+1} = d_{i,t} - a_{i,t}, \ p_{i,t+1} = p_{i,t} - 1$$

- **Reward:** related to price $r_t(s_t, a_t) = -\rho_t(\sum_{i \in \mathcal{I}_t} a_{i,t})$ (can add other costs)

Optimal EVCS policy

- Goal: find the optimal policy π for mapping $a_t \sim \pi(s_t)$ [similar to $f(\cdot)$]
- To reduce search space, consider parameterized $\pi_\mu(\cdot) = \pi(\cdot;\mu)$
  - Simple linear $\pi_\mu(s_t) = \mu^T s_t$, or nonlinear DNN, i.e., deep Q-network (DQN)

$$\max_{\mu} J(\mu) = \mathbb{E}_{a_t \sim \pi_\mu(s_t),\mathcal{P}}\left[\sum_{t=0}^{T} \gamma^t r_t(s_t,\ a_t)\middle| s_0\right]$$

- RL training complicated by the *time-varying dimensions* of both state/action
  - *How to represent state/action to allow for efficient training?*

Kyung-Bin Kwon and H. Zhu, Efficient representation for electric vehicle charging station operations using reinforcement learning," 2021 (preprint).

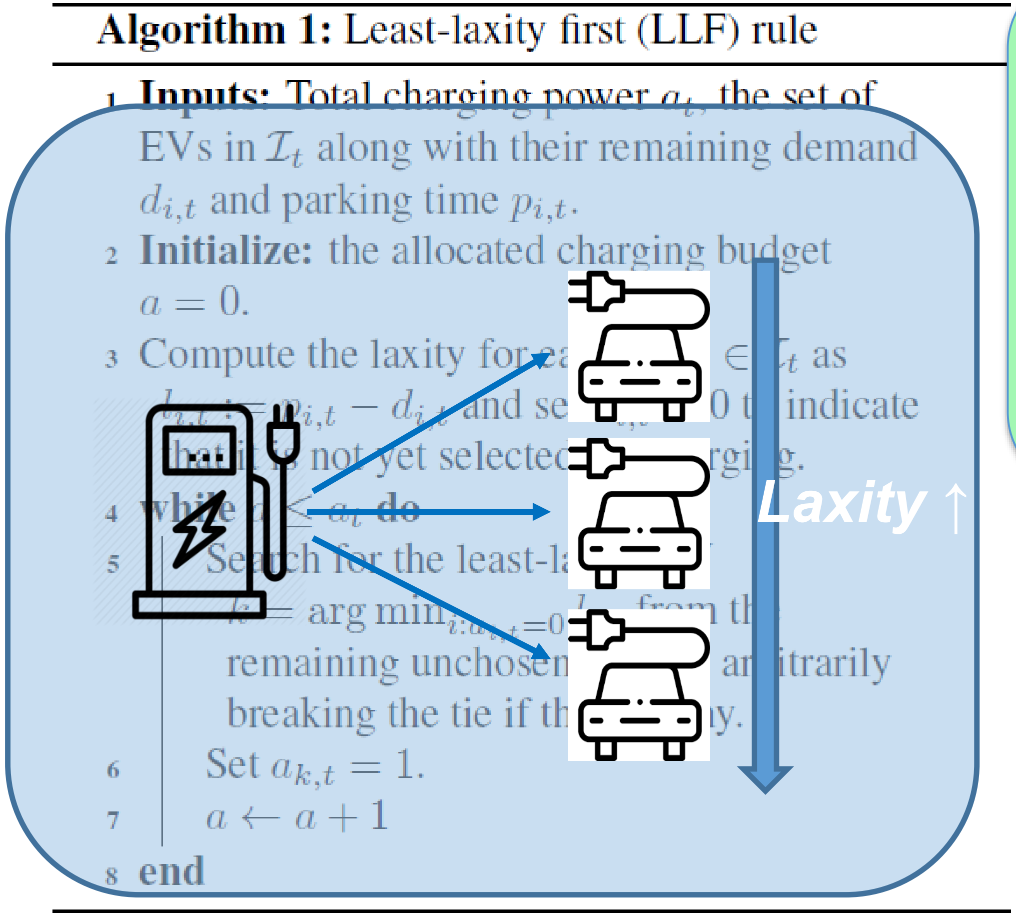
Action reduction

- As reward depends on the total charging power, how about just using it?

$$\mathcal{A}' = \{a_t := \sum_{i \in \mathcal{I}_i} a_{i,t}\}$$

  - Needs to able to recover individual actions
- Prioritize EV charging based on laxity $l_{i,t} := p_{i,t} - d_{i,t}$
  - Higher laxity => more flexibility => less priority
  - Zero laxity => has to charge this EV throughout the remaining time
  - Note that laxity ≥ 0 under our fully charged EV assumption
- Least-laxity first (LLF) rule to recover individual EV actions

Feasibility of least-laxity first (LLF)



Prop 1: *If the EVCS total charging schedule $\{a_t\}_{t \in T}$ is feasible (corresponds to some feasible schedule for individual EVs that ensure all fully charged before departure), then Algorithm 1 can produce such a feasible schedule for all EVs.*

- Basically, LLF ensures the **feasibility** of the recovered actions

Proof idea: Any feasible schedule equivalent to one satisfying LLF [Wang et al'21]

State aggregation

- **Idea:** group the EVs of the same laxity together
  - they are treated equally by the LLF rule

- Let $n_t^{(\ell)}$ collect number of EVs with laxity ℓ

$$s'_t = [\rho_t, n_t^{(0)}, n_t^{(1)}, \cdots, n_t^{(L)}]$$ with max laxity L

  - Hence, the new state is of fixed dimension $(L + 2)$
  - Reward calculation isn't affected using price
  - What can we claim for this aggregated state representation?

Equivalence of state aggregation

- Ideally, we want the new state represents the same MDP
- This equivalence requires two conditions:
  - (i) Reward homogeneity: same reward for any states aggregated into the same new state
  - (ii) Dynamic Homogeneity: same transition kernel for any aggregated states

Prop 2: *The original MDP for s_t/a_t is equivalent to the new one for s_t'/a_t using the total charging action. Accordingly, the optimal policy (or action) obtained from the new MDP through aggregation are equivalent to that for the original one.*

Intuitions for dyn. homogeneity:

Under the LLF rule, charge either one of 2 EVs at the same laxity leads to the same transition of new state or aggregated state

RL for the new MDP

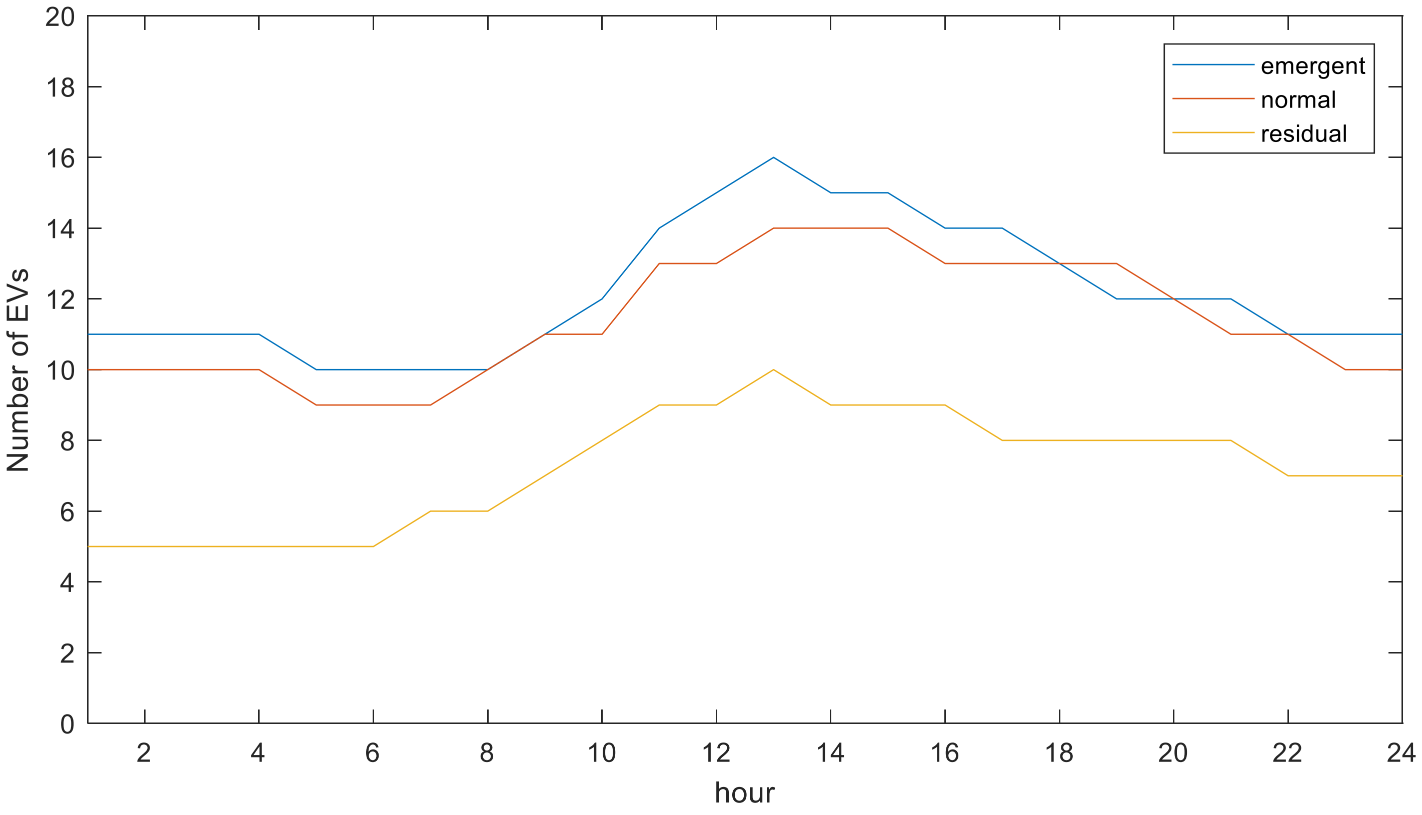
- Proposed aggregation ensures equivalence
- Policy gradient (PG) to learn the optimal linear Gaussian policy π
  - Gradient (ascent) iterations
  - Sample-based gradient estimation
- Equivalence outperforms earlier work
  - [Wang et al'21] estimates an approximate form of the Q-function (Alg. QE)

Recht, Benjamin. "A tour of reinforcement learning: The view from continuous control." 2019 https://arxiv.org/abs/1806.09460

Algorithm 2: Optimal EVCS policy search

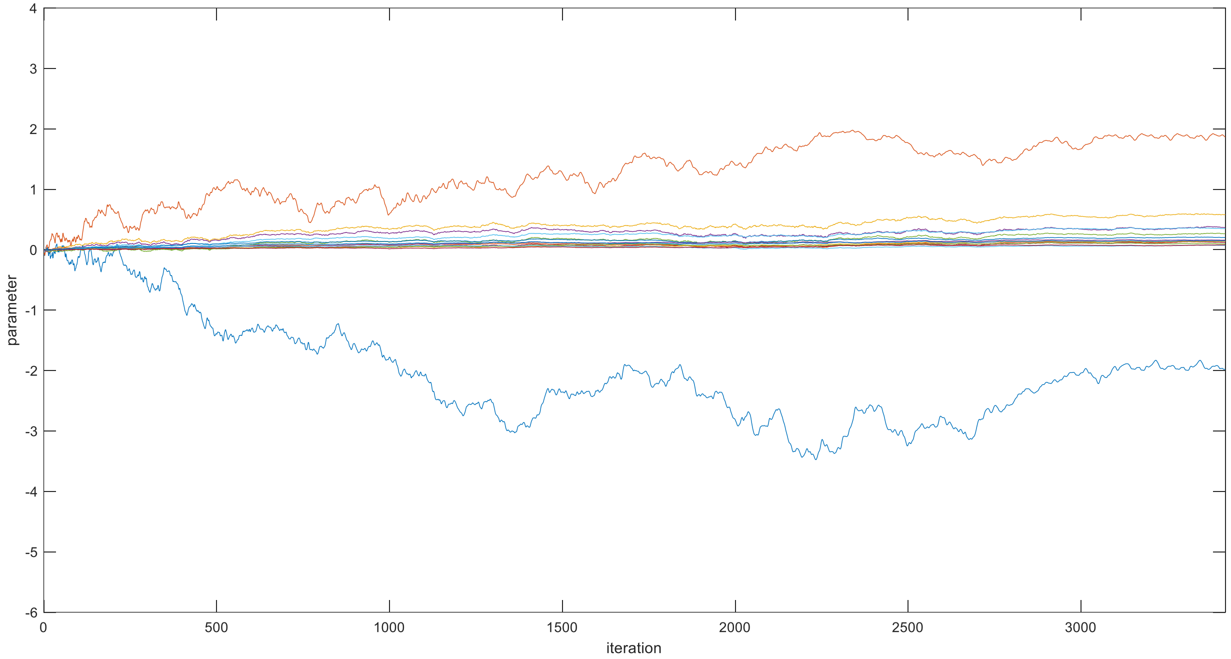
Hyperparameters: discount factor $\gamma = 1$, step-size α, and exploration time period T.
Inputs: the price sequence $\{\rho_t\}_{t=0}^{T}$, and the EV arrivals in $\{\mathcal{J}_t\}_{t=0}^{T}$ along with the initial states of EVs.
Initialize: μ^0 at iteration $n = 0$.
while μ^n *not converged* **do**
  Initialize $t = 0$ with the original state s_0.
  for $t = 0, \cdots, T-1$ **do**
    Find the aggregated state s_t' using (5);
    Sample $a_t \sim \pi_{\mu_n}(s_t')$ using (8);
    Use the LLF rule in **Algorithm 1** to recover the individual EV charging actions $\{a_{i,t}\}_{i \in \mathcal{I}_t}$;
    Compute the instantaneous reward r_t;
    Update the new state s_{t+1} using (1).
  end
  Use the sample trajectory to estimate gradient $\hat{\nabla}_\mu J(\mu^n)$ and perform the update in (15);
  Update iteration $n \leftarrow n + 1$.
end

Numerical tests



- Daily charging at 15-min intervals ($T = 96$)
  - Realistic EV arrival model
  - ERCOT real-time price
- 20 daily scenarios for training; 5 for testing
- Comparing proposed Alg 2 with Alg. QE

Convergence => further aggregation

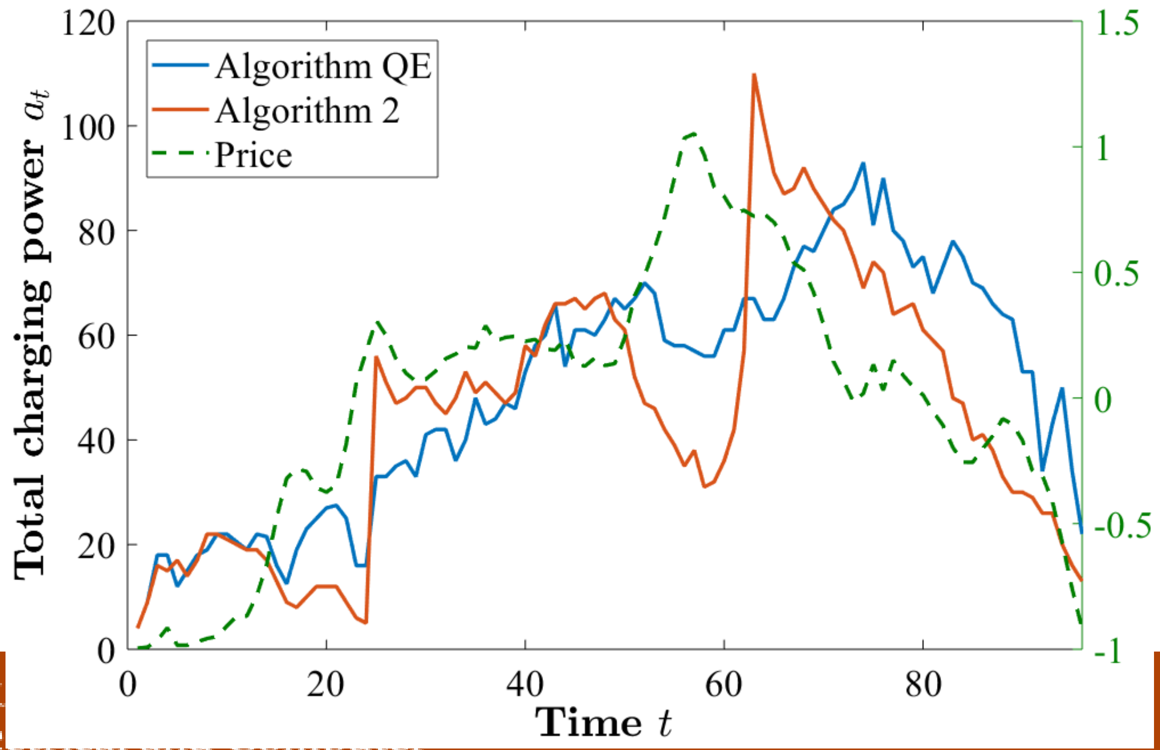


ρ_t	$n_t^{(0)}$	$n_t^{(1)}$	$n_t^{(2)}$	$n_t^{(3)}$	$n_t^{(4)}$	$n_t^{(5)}$
-1.9735	**1.8628**	0.5772	0.3674	0.2651	0.3485	0.1191
$n_t^{(6)}$	$n_t^{(7)}$	$n_t^{(8)}$	$n_t^{(9)}$	$n_t^{(10)}$	$n_t^{(11)}$	$n_t^{(12)}$
0.2021	0.1404	01386	0.1592	0.0975	0.0693	0.0797

- Parameter μ for Alg. 2 reaches convergence
- Most parameters are very small; except for ρ_t and $n_t^{(0)}$
- **Remark:** we can further reduced # of states by merging high-laxity EVs!

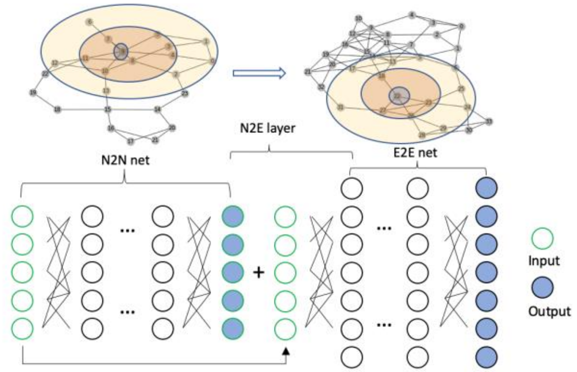
Comparison on testing performance

	Test 1	Test 2	Test 3	Test 4	Test 5	Average
Alg. 2	-5016.2	-5022.6	-5009.5	-5012.8	-5007.8	-5013.8
Alg. QE	-5240.1	-5240.3	-5234.2	-5239.3	-5230.6	-5236.9
Increase (%)	4.27	4.15	4.29	4.32	4.26	4.26

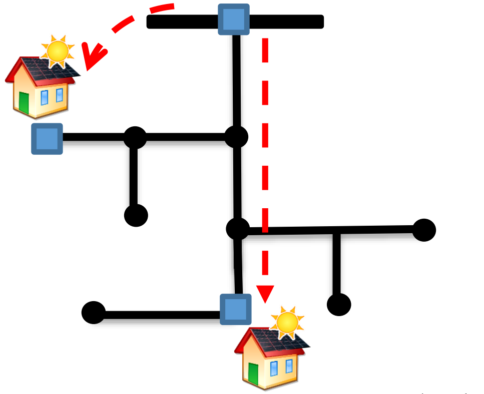


- Alg. 2 improves the reward of Alg. QE by ~4.2%
- Example charging profile indicates Alg. 2 very sensitive to price peaks and strategically reducing a_t, while Alg. QE fails to do so

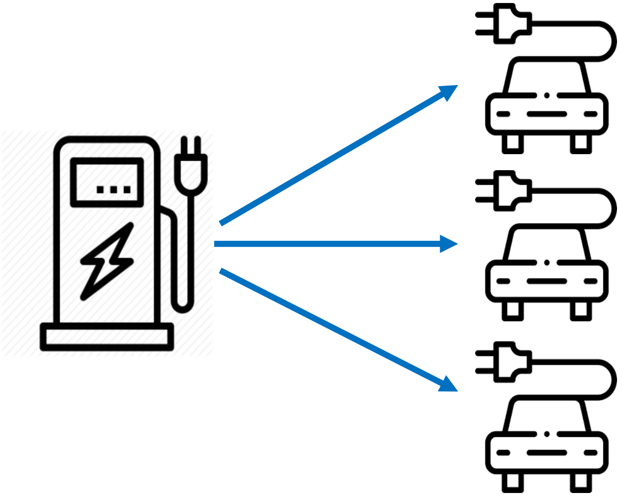
Summary



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



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



Efficient representation for dynamical resources:
Time-varying problem dimension

- I: Topology adaptivity and other transfer learning ideads
- II: GD algorithms and connection to safe RL
- III: relaxed conditions (fully charged EV) and network-coupled operations

Education resources

- UT grad course *"Data Analytics in Power Systems"*; new slides coming soon 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
  - 2021 PES GM panel on *University Gaps Assessment*, Tuesday (07/27), 7-9pm EST

Learning and Optimization for Smarter Electricity Infrastructure

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

Learning for resilient operations
Learning at the interface of power electronics
....

Thank you!