Unsupervised Physics-Informed Learning Approaches for State Estimation

Ahmed Zamzam

National Renewable Energy Laboratory
Golden, CO, USA

Small Data Learning

ANDREW NG: UNBIGGEN AI

Why Small Data Is the New Big Data

UPenn (Knowledge at Wharton)

Learning with Small Data

KDD 2020 Tutorial

- Datasets in power systems are not always big and labelled
- Can we learn from small (labeled/unlabeled) datasets?

State Estimation

Receives raw measurements (nodal injection, line flows, voltage etc.) from measuring devices.

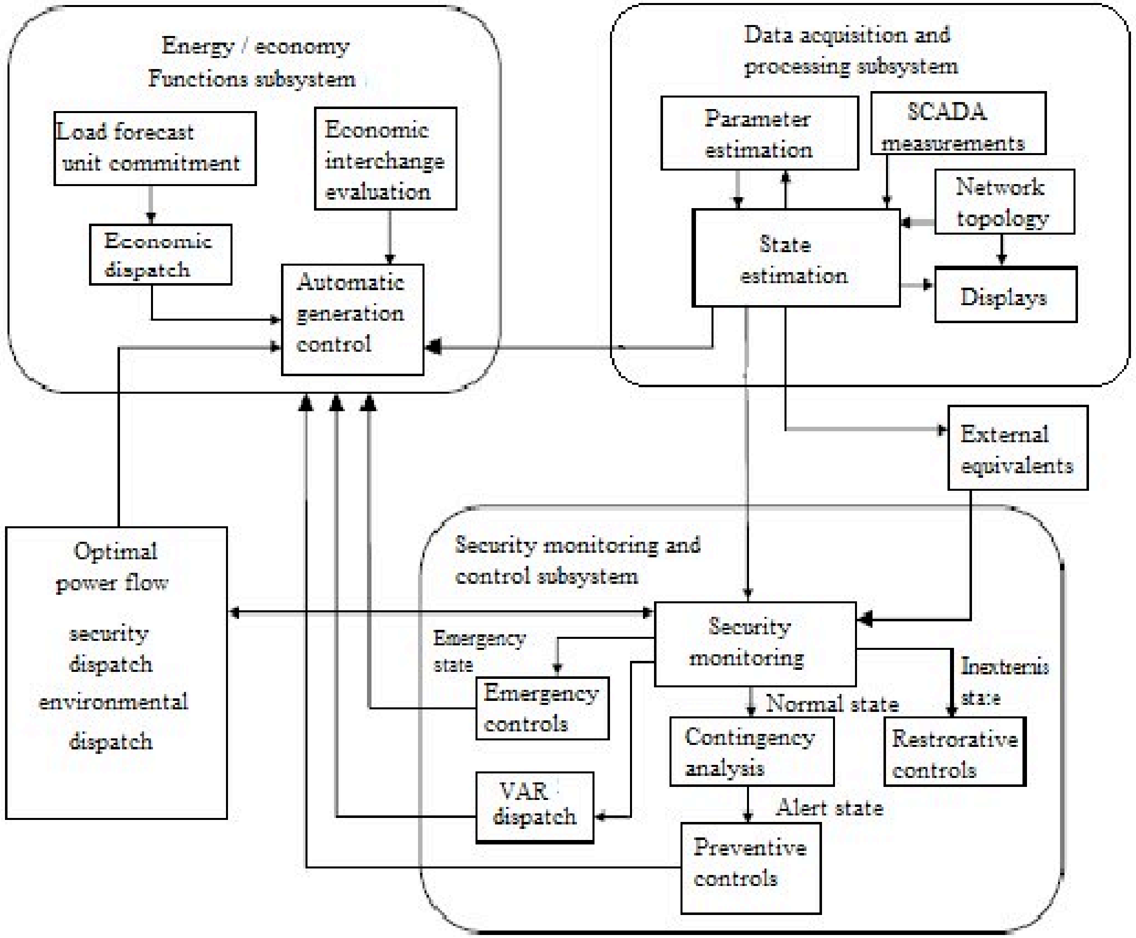
Seeks to identify the values of the bus voltage magnitudes and angles $\{|V_n|, \theta_n\}_{\forall n}$

$$z_\ell = h_\ell(\mathbf{v}) + \epsilon_\ell$$

SE:

$$\hat{\mathbf{v}} = \mathbf{F}(\mathbf{z}) := \arg\min_{\mathbf{v}} \left(\mathbf{z} - \mathbf{h}(\mathbf{v})\right)^T \mathbf{W} \left(\mathbf{z} - \mathbf{h}(\mathbf{v})\right)$$

State Estimation Block



Credit: Simson, Samson Raja, et al. "Virtual state estimation calculator model for three phase power system network." *Journal of Energy and Power Engineering* 10 (2016): 497-503.

- Challenges:
  - Data quality
  - Latency constraints
  - Robustness and reliability
  - Model identification!

SE Approaches

Model based method [1]:

- Existing GN method takes a **lot of iterations**.
- GN method may **not converge** for some scenarios.

$$\hat{\mathbf{v}} = \mathbf{F}(\mathbf{z}) := \arg\min_{\mathbf{v}} \left(\mathbf{z} - \mathbf{h}(\mathbf{v})\right)^T \mathbf{W}\left(\mathbf{z} - \mathbf{h}(\mathbf{v})\right)$$

$$\mathbf{h}(\mathbf{v}) \simeq \mathbf{h}(\mathbf{v}_k) + \mathbf{G}_k^T(\mathbf{v} - \mathbf{v}_k), \ \mathbf{G}_k : \text{Jacobian at } \mathbf{v}_k$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + (\mathbf{G}_k\mathbf{G}_k^T)^{-1}\mathbf{G}_k(\mathbf{z} - \mathbf{h}(\mathbf{v}_k))$$

Supervised learning for state estimation [2, 3, 4]:

- Requires significant training samples **with labels**, not easy to obtain.
- Ignores the knowledge of known **mathematical structure of state estimation process**.

$$\mathbf{g}_K(\mathbf{z}) = \sum_{k=1}^{K} \boldsymbol{\alpha}_k \sigma(\mathbf{w}_k^T\mathbf{z} + \beta_k)$$

$$\min_{\{\boldsymbol{\alpha}_k, \mathbf{w}_k, \beta_k\}_{k=1}^K} \ \sum_j \|\mathbf{v}^j - \mathbf{g}_K(\mathbf{z}^j)\|_2^2$$

[1] I. Dzafic, R. A. Jabr, and T. Hrnjic, "Hybrid state estimation in complex variables," IEEE Trans. Power Syst., 2018.
[2] K. Mestav et al. "Bayesian state estimation for unobservable distribution systems via deep learning," IEEE Trans. Power Syst., 2019.
[3] L. Zhang, G. Wang, and G. B. Giannakis, "Real-time power system state estimation and forecasting via deep unrolled neural networks," IEEE Trans. Signal Processing, 2019.
[4] A. S. Zamzam, N. D. Sidiropoulos, "Physics-aware neural networks for distribution system state estimation," IEEE Trans. Power Syst., 2020

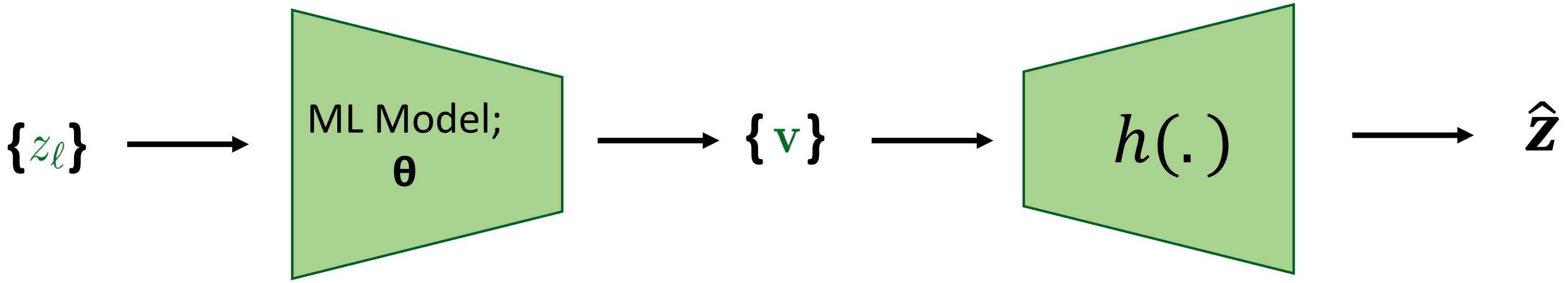
Learning vs. Optimization

- Learning approaches require very large number of diverse samples to approximate the SE mapping
  - Simulations
  - SE solver

- Most optimization approaches suffer from scalability issues especially when:
  - Majority of measurements are nonlinear
  - Not enough redundancy with noisy measurements

Learning SE Approach

$$z_\ell = \mathbf{v}^H \mathbf{D}_\ell \mathbf{v} + \mathbf{c}^H \mathbf{v} + \mathbf{v}^H \mathbf{c} \quad \longrightarrow \quad \boldsymbol{z} = \boldsymbol{h}(\mathbf{v})$$

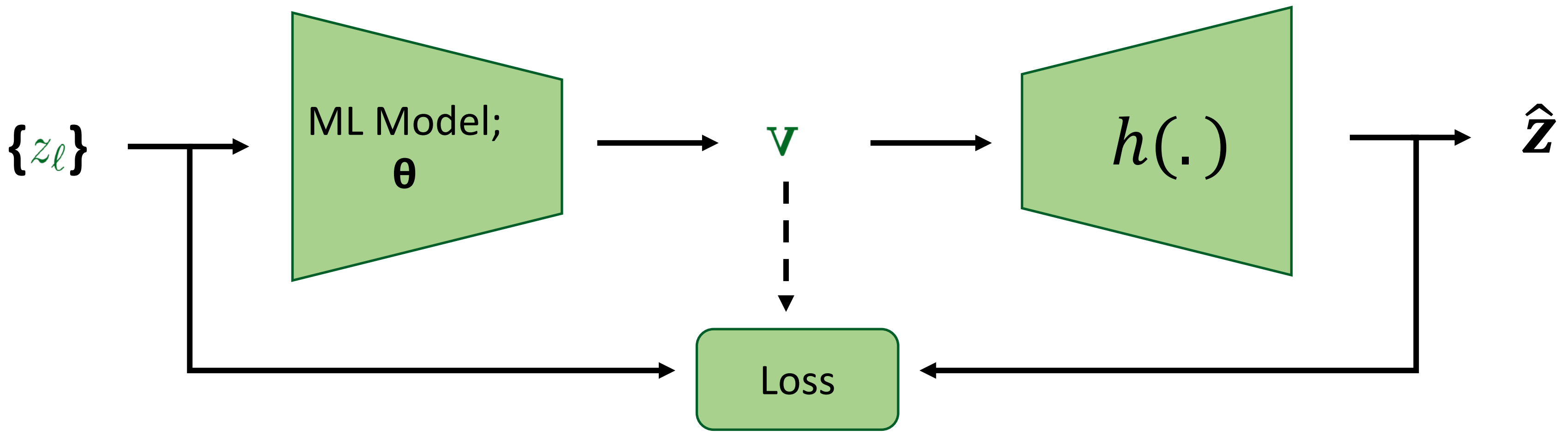


$$\hat{\boldsymbol{\theta}} \coloneqq \arg\min_{\boldsymbol{\theta}} \frac{1}{K} \sum_{i=1}^{K} \left\| \hat{\mathbf{v}}^{(i)}\left(\boldsymbol{z}^{(i)}; \boldsymbol{\theta}\right) - \mathbf{v}_{true}^{(i)} \right\|_2^2 \quad \Rightarrow \quad \hat{\boldsymbol{\theta}} \coloneqq \arg\min_{\theta} \frac{1}{K} \sum_{i=1}^{K} \left\| \boldsymbol{h}\left(\hat{\mathbf{v}}^{(i)}\left(\mathbf{z}^{(i)}; \boldsymbol{\theta}\right)\right) - \boldsymbol{h}\left(\mathbf{v}_{true}^{(i)}\right) \right\|_2^2$$

$$\Downarrow$$

$$\hat{\boldsymbol{\theta}} \coloneqq \arg\min_{\theta} \frac{1}{K} \sum_{i=1}^{K} \left\| \boldsymbol{h}\left(\hat{\mathbf{v}}^{(i)}\left(\mathbf{z}^{(i)}; \boldsymbol{\theta}\right)\right) - \boldsymbol{z}^{(i)} \right\|_2^2$$

Unsupervised Learning Approach

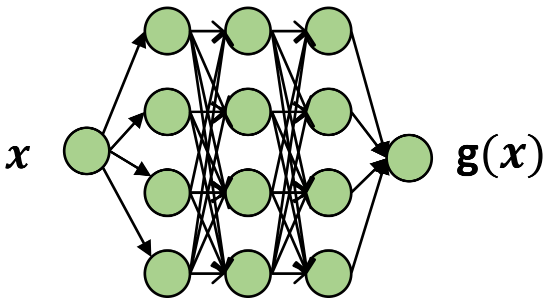


$$\min_{\boldsymbol{\theta}} \frac{1}{K} \sum_{i=1}^{K} \left\| h\left(\hat{\mathbf{v}}^{(i)}\left(\mathbf{z}^{(i)}; \boldsymbol{\theta}\right)\right) - \mathbf{z}^{(i)} \right\|_2^2 + \lambda \left\| \hat{\mathbf{v}}^{(i)}\left(\mathbf{z}^{(i)}; \boldsymbol{\theta}\right) - \mathbf{v}_0 \right\|_2^2$$

- We do NOT need ground-truth solutions/labels (unsupervised learning)
- Physics knowledge is exploited in the learning process

- Similarity with Auto-Encoders!

Unlabeled Datasets



$$f(x) = e^{x}, \quad f^{-1}(x) = \log(x)$$

$$\min_{\theta} \sum \left\| g(x^{(i)}; \theta) - e^{x^{(i)}} \right\| \Rightarrow \min_{\theta} \sum \left\| \log(g(x^{(i)}; \theta)) - x^{(i)} \right\|$$

Labeled data

- Need to generate labels (voltages) for each set of measurements
- If measurements do NOT have redundancy, getting the correct labels will be very noisy or corrupt
- Training process is straightforward

Unlabeled data

- No need for ground-truth voltage estimates
- If measurements do NOT have redundancy, the estimates quality will depend on the regularization approach used
- Training process required differentiating through the physics model

Data Generation for SE

- To be able to measure quality of our estimates, we store all ground-truth voltage profiles $(|v|, \theta)$.

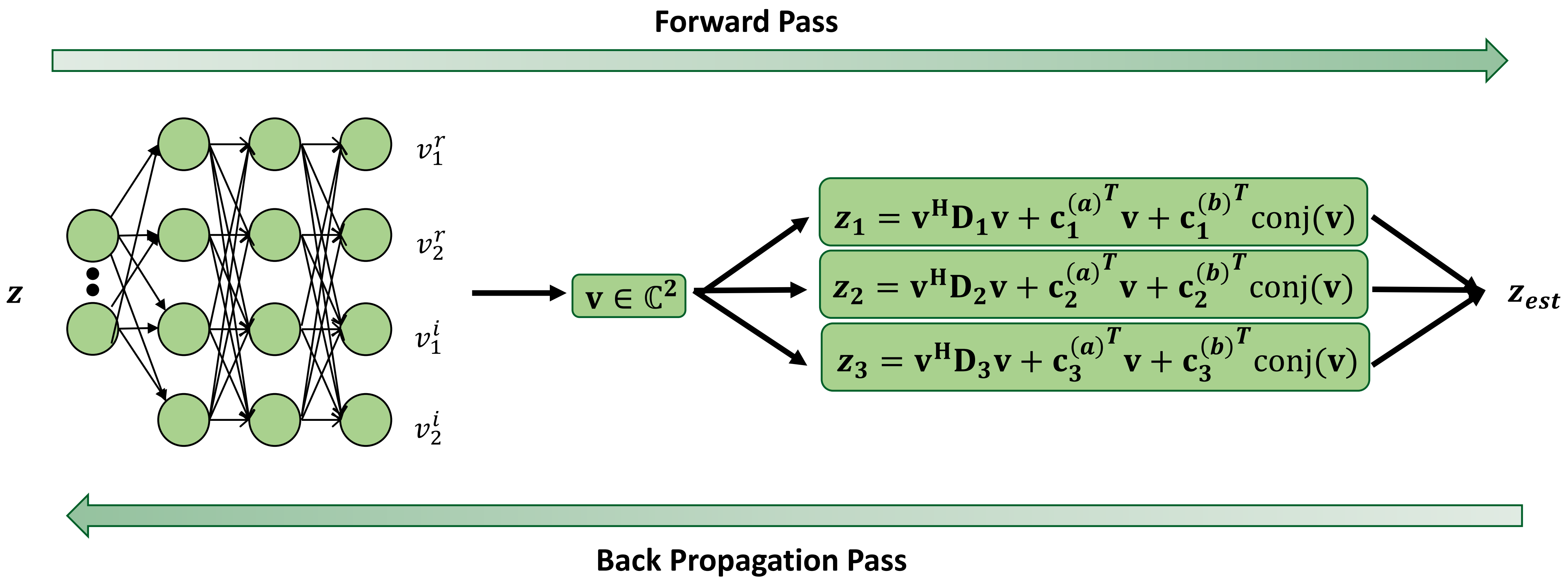
- Any set of measurements can be driven from the state of the system.

- We simulate the system under significant penetration of renewables and varying loading conditions.

- The approach can be applied essentially with only measurements.

- If a model-based SE is solvable, it will help assess the quality of the proposed estimator.

Computational Graph



Unsupervised Learning Approach

Experiment: 37-Bus Distribution system

- Insufficient measurements
- Inferior measurement precision

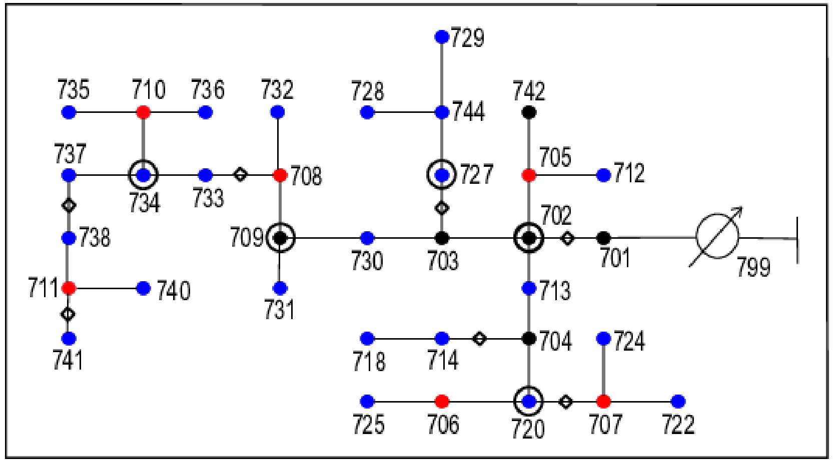
Customized loss function:

$$Loss = \sum (Z_{measured} - Z_{est})^2 + \lambda_1 [(V_{r,ph1} - \mathbf{2.8497})^2 + (V_{i,ph1} - \mathbf{0})^2 + (V_{r,ph2} - (-\mathbf{1.4623}))^2 + (V_{i,ph2} - (-\mathbf{2.4349}))^2 + (V_{r,ph3} - (-\mathbf{1.3895}))^2 + (V_{i,ph3} - \mathbf{2.4336})^2]$$

Physical equation utilized during training:

$$\mathbf{z}_{est,i} = \mathbf{v}^H * D_i * \mathbf{v} + {c_i^{(1)}}^T * \mathbf{v} + {c_i^{(2)}}^T * conj(\mathbf{v})$$

where,
v =predicted voltage
$Q, P1, and\ P2$ are the parameters of the measurement function

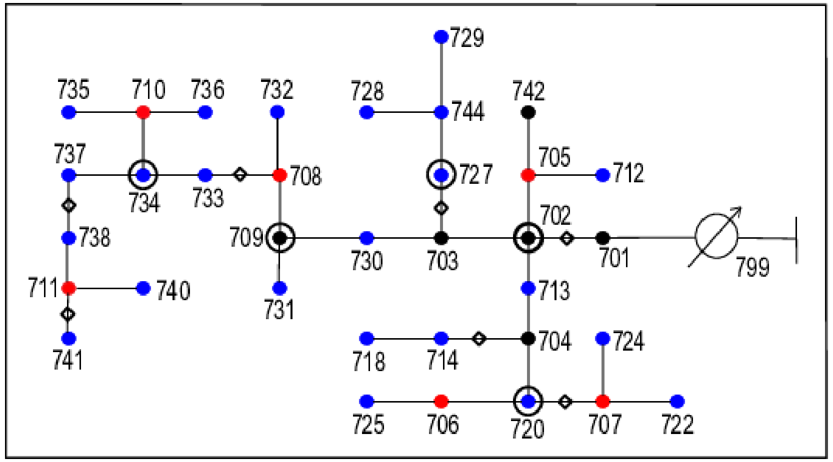


IEEE 37-bus unbalanced distribution feeder

Result Analysis

Training Statistics:

- 90,000 scenario
- 130 Epochs
- Batch size 1000
- Learning rate 0.01



Test Statistics

- 10,000 scenario
- MSE Estimation loss: 26.5851
- MSE Voltage loss: 0.0209

λ	1e-3	1e-2	1e-1	1e0	1e1	1e2	1e3
Training loss	0.0025	0.0043	0.0046	0.0074	0.0135	0.0348	0.0772
Z_loss	0.1104	0.3123	0.4548	0.4586	0.4313	0.4737	0.5998
Vr_loss	0.2390	0.0316	0.0020	0.0007	0.0005	0.0009	0.0011
Vi_loss	0.0113	0.0098	0.0096	0.0098	0.0096	0.0094	0.0095

OPF-Learn: An Open-Source Framework for Creating Representative AC Optimal Power Flow Datasets

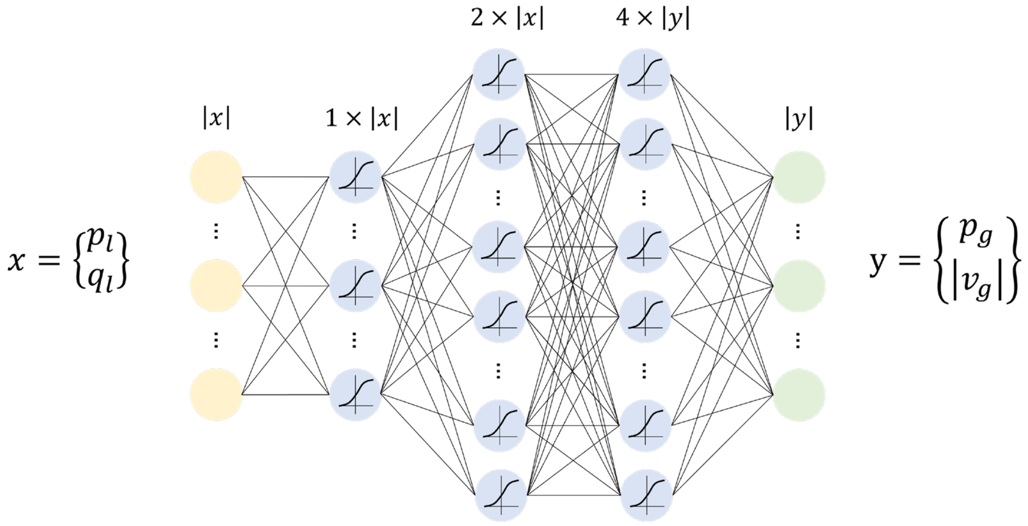
Do we have confidence in ML-based models?

- All papers in learning for OPF use different datasets
- …and different ways of generating those datasets ("we perturbed the base loading point randomly +/-10%...)
- Some different models or versions of models (MATPOWER 118-bus vs. PG-lib 118-bus)
- Impossible to compare the results in these papers

More representative datasets

→ Current datasets for power system optimization generally just contain single points

→ For supervised learning tasks, we need a lot of OPF solutions, that span a good representation of the feasible space



OPF-Learn: An Open-Source Framework for Creating Representative AC Optimal Power Flow Datasets

Trager Joswig-Jones
University of Washington

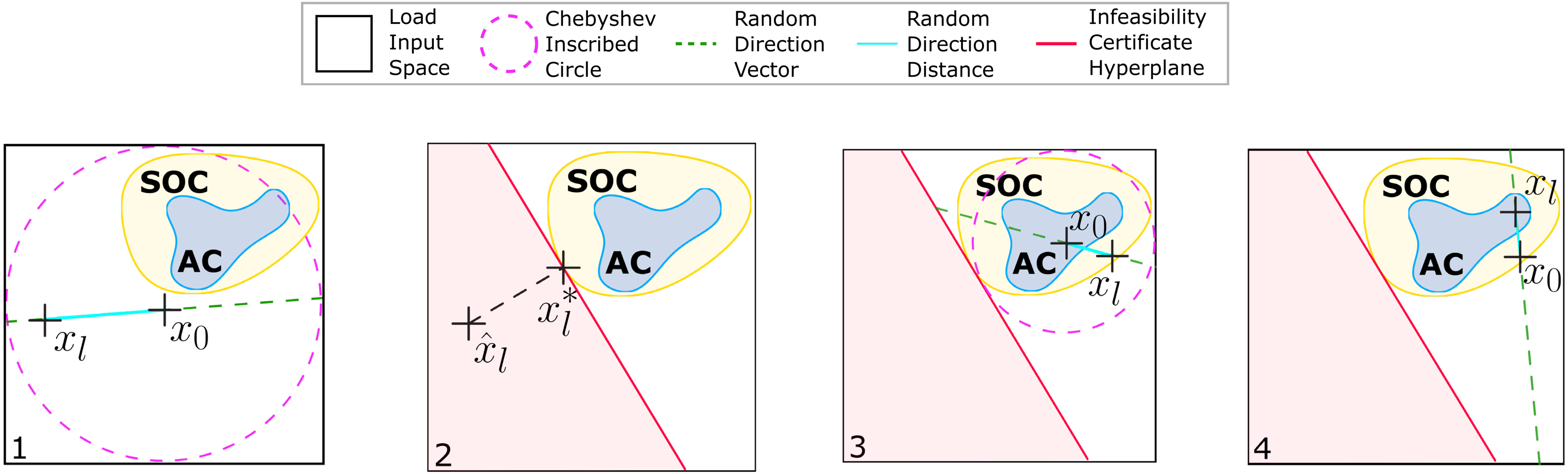
Kyri Baker
University of Colorado Boulder

Ahmed S. Zamzam
National Renewable Energy Laboratory

Will be available soon at: https://github.com/NREL/OPFLearn.jl and corresponding paper on arxiv

Kudos to the following work for N-1 security classification tasks which inspired this one: A. Venzke, D. K. Molzahn, and S. Chatzivasileiadis, “Efficient creation of datasets for data-driven power system applications,” Electric Power Systems Research, vol. 190, p. 106614, 2021.

Data Generation Technique



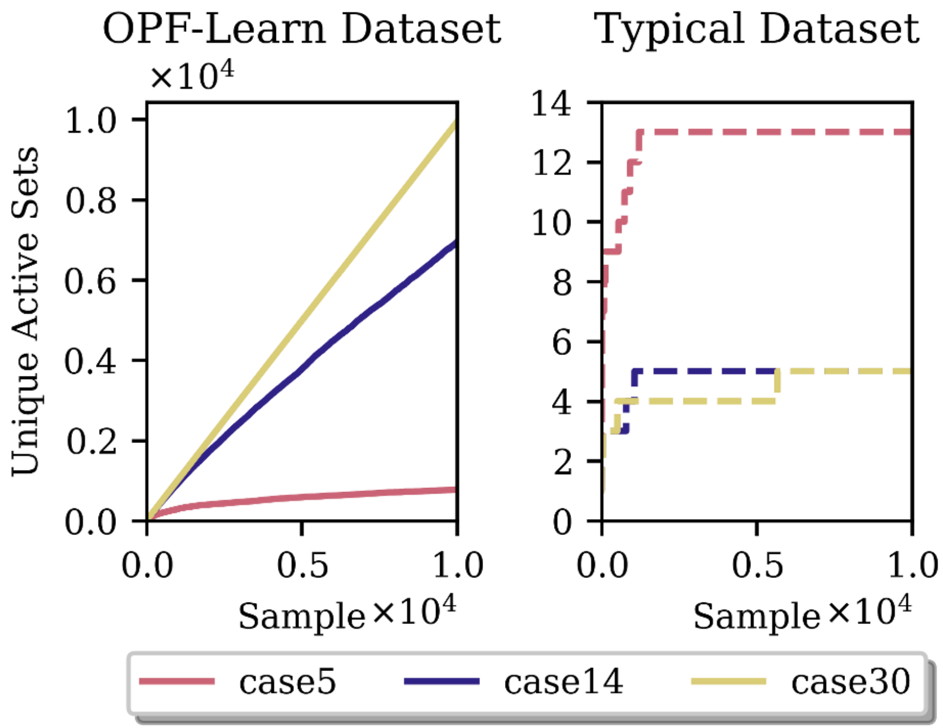
Find the Chebyshev center, x_0. Generate a random direction vector and travel a random distance along this vector to find a new load sample, x_l.

Check if x_l is AC-feasible. If not, find the nearest SOC feasible point, x_l^*. Since $\hat{x}_l \neq x_l^*$, define a new infeasibility certificate at x_l^* with normal, $\vec{n} = \hat{x}_l - x_l^*$.

Gather a new sample, x_l, as in step 1. Check if the new x_l sample is AC-feasible. Here it is not, so the nearest SOC feasible point is found. $\hat{x}_l = x_l$, so discard this sample.

Sample a new load profile, x_l, as in step 1, but starting from the last point, now x_0. Check if x_l is AC-OPF feasible. x_l is AC-OPF feasible, so store x_l and its AC-OPF optimal solution.

Finding new AC OPF solutions more efficiently



Plot of unique active sets found over time with a typical dataset creation methods and the OPF-Learn dataset creation method. Note the difference in the y-axis scale

Improved model performance

Test Dataset:	OPF-Learn Dataset Trained Model		Typical / OPF-Learn	Typical Dataset Trained Model		OPF-Learn / Typical
	OPF-Learn	Typical		OPF-Learn	Typical	
case5	2.17E-2	1.86E-3	**8.57E-2**	1.33E+0	9.08E-6	**1.46E+5**
case14	2.75E-4	1.01E-4	**3.67E-1**	3.94E-2	9.41E-7	**4.19E+4**
case30	1.55E-4	5.46E-4	**3.52E+0**	8.17E-3	1.60E-8	**5.11E+5**
case118	6.97E-2	2.35E-1	**3.37E+0**	4.47E-1	4.47E-3	**1.00E+2**

NN models trained on OPF-Learn data have less mean squared error (MSE) when tested on representative test sets compared to the same model trained on a typical dataset.

Conclusion & Future Research

Learn state estimation mapping in unsupervised fashion utilizing **knowledge of the model**.

Labeled data are substituted for using model information, at the price of increased learning complexity.

Future research:

- Adaptive penalty (linearized power flow)
- Learning model parameters in an outer loop
- Learn using distribution of measurements instead of data records

OPF-Learn.jl generates diverse datasets of loads and their optimal solutions increasing trust in ML models.

Thank you for your attention!

Many thank to collaborators:

Fouad Hasan
Amin Kargarian

Andrey Bernstein

Trager Joswig-Jones

Kyri Baker