A Convex Design in Structural Deep Neural Network for Controlling Renewables

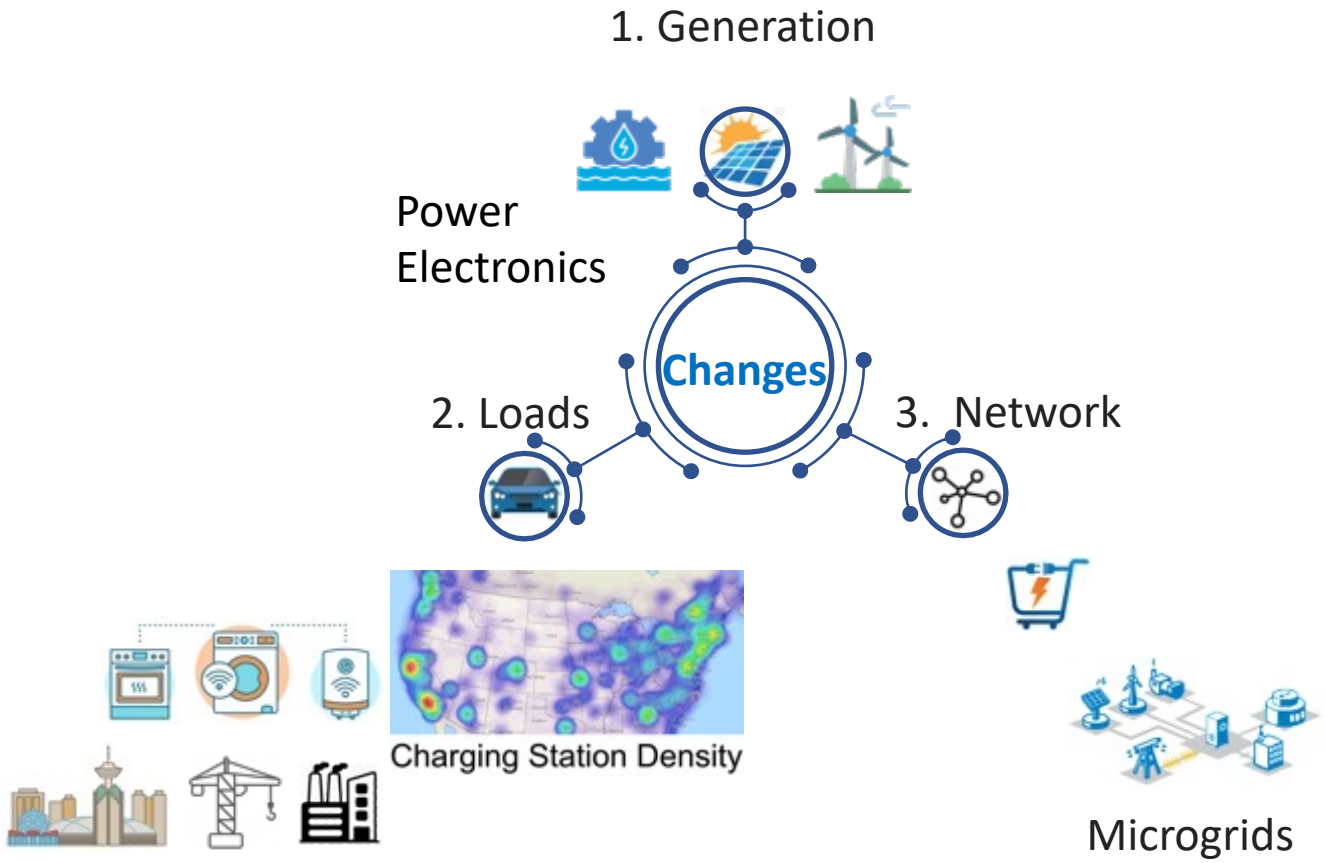
Yang Weng

Arizona State University

4/24/2024

PSERC Webinar

Energy Grid Transformation



Challenges:

- Intermittency and Variability
- Voltage Fluctuations
- Capacity Limits and Grid Congestion
- Grid Stability and Reliability
- Reverse Power Flow
- Cyber Attacks

Important Solution:

- Data-Driven Operation with Guarantees

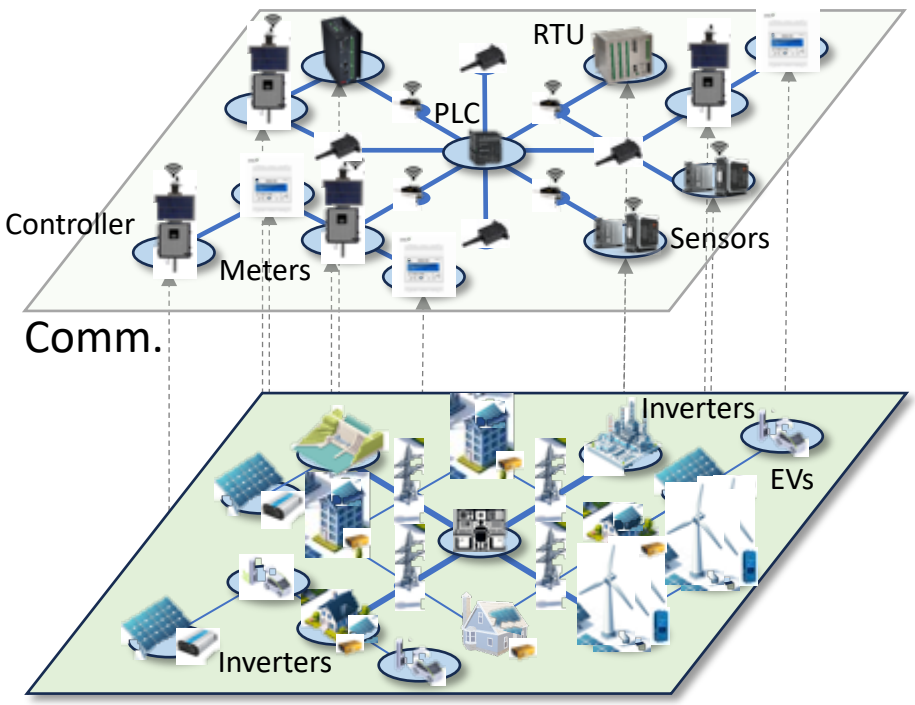
Research Questions: Reliable and Secured Operation?

1. Normal Conditions

AI-Enhanced Optimal Control

Good: Now

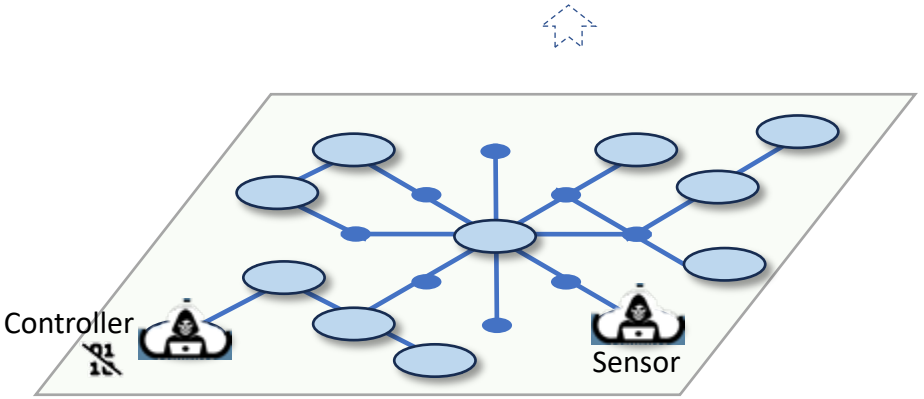
Future



Attacks

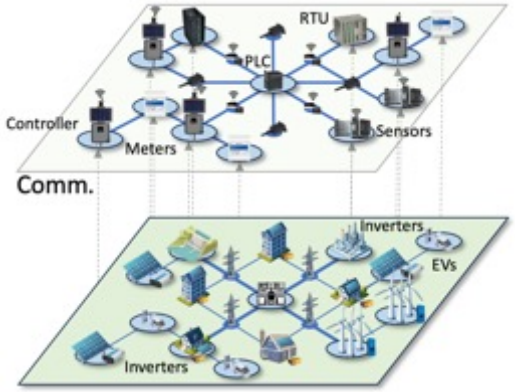
2. Under Attack

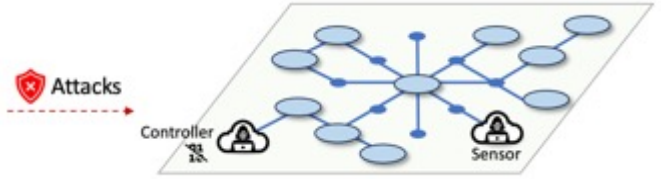
Operate Through Compromise



Outline

- Optimal Voltage Control
  - Now: Input Convex Neural Network for Optimality
  - Future: Generalizability Design
- Operate with Limited System Information
  - Deep Learning Twin for Physical Consistency
- Operate Through Compromise
  - Inject False Data without System Information
  - Defense Strategy and its Optimal Control
- Conclusion





Problem Definition for Voltage Regulation and Past Work

Minimize Voltage Deviation

$$\min_{q} \sum_{i=1}^{N} |V_i - V_{i,ref}|$$

s. t. $p_{min} \le p \le p_{max}$ Active power

$q_{min} \le q \le q_{max}$ Reactive power

$V_{min} \le V \le V_{max}$

$$p_i = \sum_{k=1}^{N} V_i V_k (G_{ik} \cos\theta_{ik} + B_{ik} \sin\theta_{ik})$$

$$q_i = \sum_{k=1}^{N} V_i V_k (G_{ik} \sin\theta_{ik} - B_{ik} \cos\theta_{ik})$$

Power flow equations

With System Information:

Linearization of Nonlinear Power Flow Constraints [Chamana 18] [Changfu 19]

Convexification

- Semi-Definite Programming (SDP) [Wang 17]
- Penalty Technique [Lin 22]

Without System Info: Learning-based Methods

- Generative Learning
  - Power Flow-based Learning [Zhang et al 24] [Du 21]
  - Prior Knowledge like Radial Networks and X/R Ratios [Xu 19]
- Discriminative Learning
  - Machine Learning Model-based [Ayyagari 19] [Hong 23]

Limited or No System Information: How to Do Optimal Control?

Minimize Voltage Deviation

$$\min_{q} \sum_{i=1}^{N} |V_i - V_{i,ref}|$$

$$|\boldsymbol{V} - \boldsymbol{V_{ref}}| = f(\boldsymbol{p}; \boldsymbol{q})$$

Data-Driven Opt. Modeling

$$\min_{q} \quad \|f(\boldsymbol{p}, \boldsymbol{q})\|$$

s. t.
$$q_{min} \le q \le q_{max}$$
$$V_{min} \le V \le V_{max}$$
$$p_{min} \le p \le p_{max}$$

Linear Constraints

s. t. $p_{min} \le p \le p_{max}$ Active power

$q_{min} \le q \le q_{max}$ Reactive power

$$V_{min} \le V \le V_{max}$$

$$p_i = \sum_{k=1}^{N} V_i V_k (G_{ik} \cos\theta_{ik} + B_{ik} \sin\theta_{ik})$$

$$q_i = \sum_{k=1}^{N} V_i V_k (G_{ik} \sin\theta_{ik} - B_{ik} \cos\theta_{ik})$$

Power flow equations

→ $f_{DNN}((\boldsymbol{p}, \boldsymbol{q})|\boldsymbol{\Omega})$ Non-convex

No Sys Info but Data?

Can We Design Convex Data-Driven Opt. Modeling?

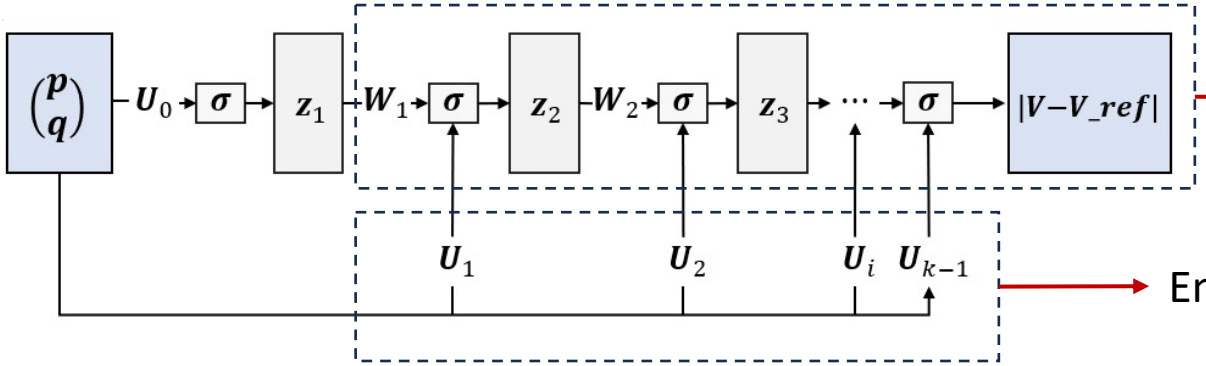
→ $f_{Convex\ NN}((\boldsymbol{p}; \boldsymbol{q})|\boldsymbol{\Omega}) = |\boldsymbol{V} - \boldsymbol{V_{ref}}|$

Answer: Embed Convexity into NNs by Input Convex Neural Network

One possible solution to approximate $|V - V_{ref}|$ with $f_{ICNN}((p; q)|\Omega)$ [*B. Amos et al, 2017*].

Proposition [Input *Convexity* of Neural Networks]

A neural network is convex w.r.t input x, given that all weights in $W_{1:k-1}$ are **non-negative**, and all activation functions $\sigma(\cdot)$ are **convex and non-decreasing (e.g. ReLU)**.



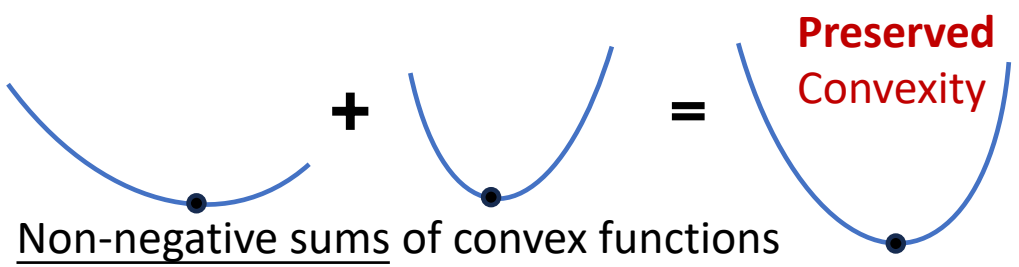
→ Non-negative Layers: embed convexity

$$|V - V_{ref}| = f_{ICNN}((p; q)|\Omega)$$

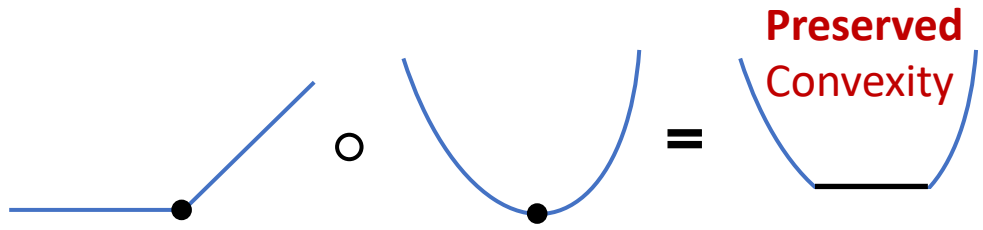
→ Enhance representation ability

f_{ICNN} is convex in $(p; q)$ → Guarantee *convexity* in data-driven model for *control* target

Proof: Convexity is maintained in all feed-forward steps



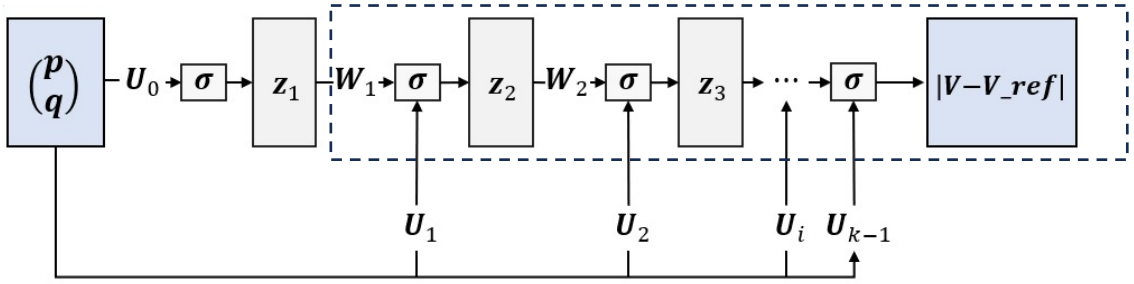
Non-negative sums of convex functions



Composition of nonlinear ReLU and convex function

Improvement?

Guarantee *convexity* in data-driven model for *control* target → f_{ICNN} is convex in $(\boldsymbol{p};\boldsymbol{q})$

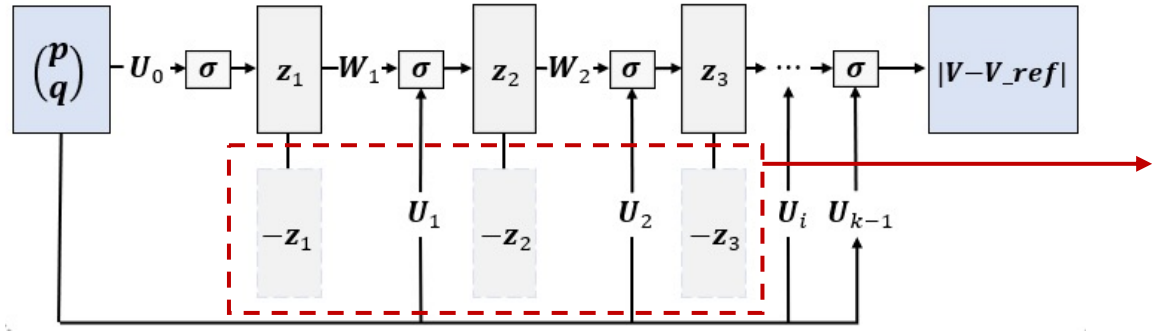


Non-negative Layers: embed convexity

$\boldsymbol{W_{1:i} \geq 0}$ ❑ Harder to model decreasing function

❑ Push weights against 0 boundary (stuck)

→ Limit Approximation Capacity

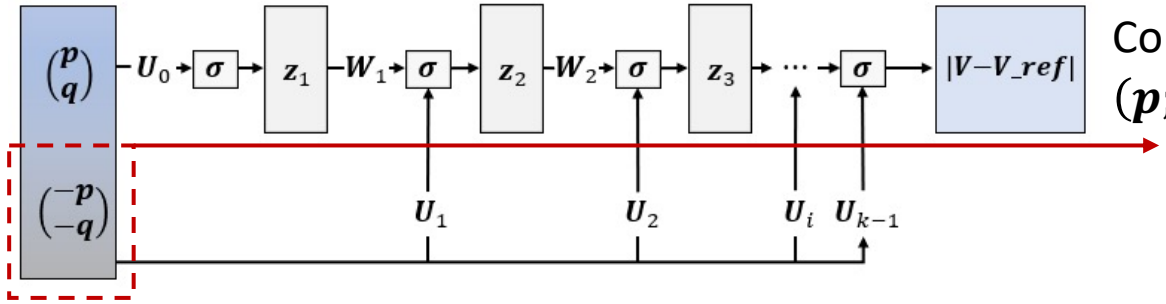


Let $\boldsymbol{W}$ be able to be negative?

$\boldsymbol{z_{1:i}} \rightarrow (\boldsymbol{z_{1:i}}; -\boldsymbol{z_{1:i}})$

$\Leftrightarrow \boldsymbol{W^T_{1:i\{<0\}}} \cdot \boldsymbol{z_{1:i}} := \boldsymbol{W^T_{1:i\{\geq 0\}}} \cdot -\boldsymbol{z_{1:i}}$

[*Y. Chen et al, 2019; 2020*]



Compensate for Capacity by Expanding Inputs Only

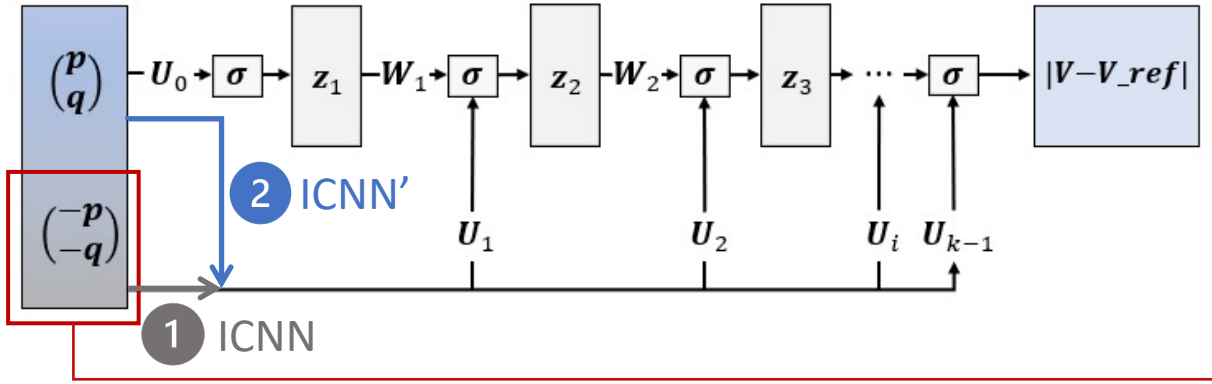
$(\boldsymbol{p};\boldsymbol{q}) \rightarrow \big((\boldsymbol{p};\boldsymbol{q});(-\boldsymbol{p};-\boldsymbol{q})\big)$

$\rightarrow f_{ICNN}\left(\big((\boldsymbol{p};\boldsymbol{q});(-\boldsymbol{p};-\boldsymbol{q})\big)|\boldsymbol{\Omega}\right) = |\boldsymbol{V}-\boldsymbol{V_{ref}}|$

→ Too much Computation Burden

The Problem of Redundancy Design

While ensuring convexity with $|\boldsymbol{V} - \boldsymbol{V_{ref}}| = f_{ICNN}\left(((\boldsymbol{p};\boldsymbol{q});(-\boldsymbol{p};-\boldsymbol{q}))|\boldsymbol{\Omega}\right)$ [*Y. Chen et al, 2019; 2020*].



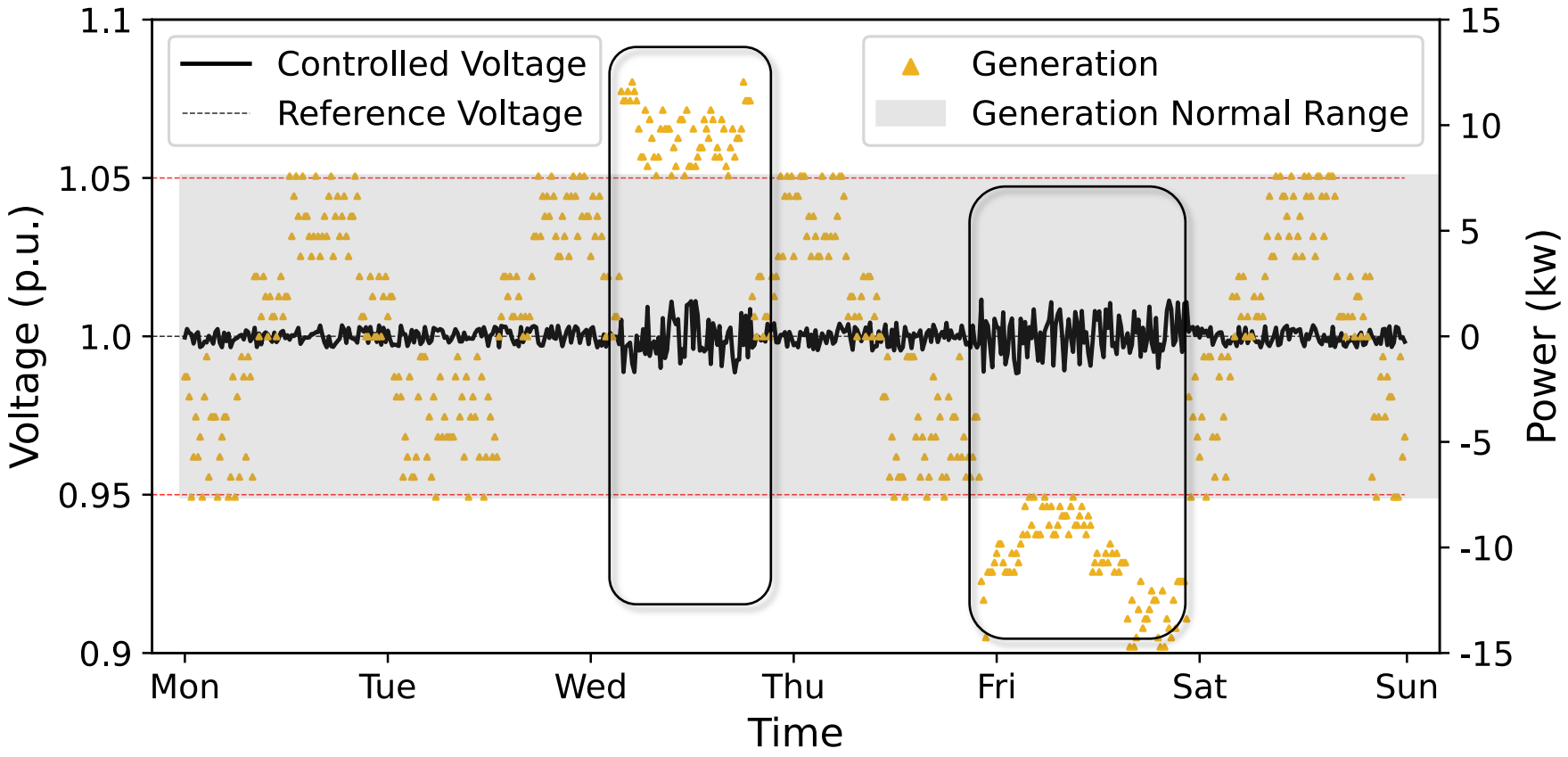
	Model Variation	System Size (Input Dimension)	Error (MAPE)	R2 Score
1	ICNN with Expanded Inputs $[(\boldsymbol{p};\boldsymbol{q});(-\boldsymbol{p};-\boldsymbol{q})]$ [*Y. Chen et al, 2019; 2020*]	20	0.26	0.97
		30	0.24	0.96
		50	0.04	0.86
2	ICNN' with Regular Inputs $(\boldsymbol{p},\boldsymbol{q})$	20	0.24 ↓	0.98 ↑
		30	0.23 ↓	0.97 ↑
		50	0.03 ↓	0.88 ↑

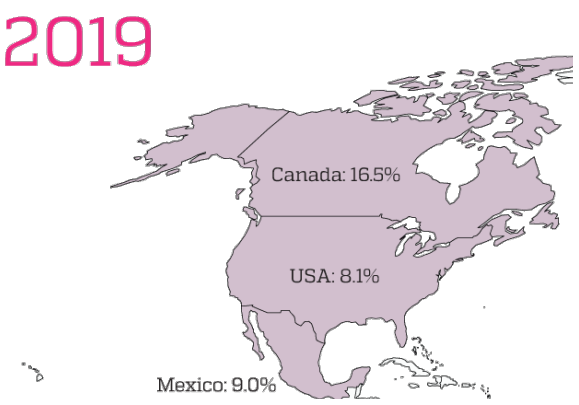
Expanded Inputs Unnecessary

≥18% less computation time
≥25% less memory occupied

→ Better for grid edge computing

ICNN: Good for the Past and Bad for the Future



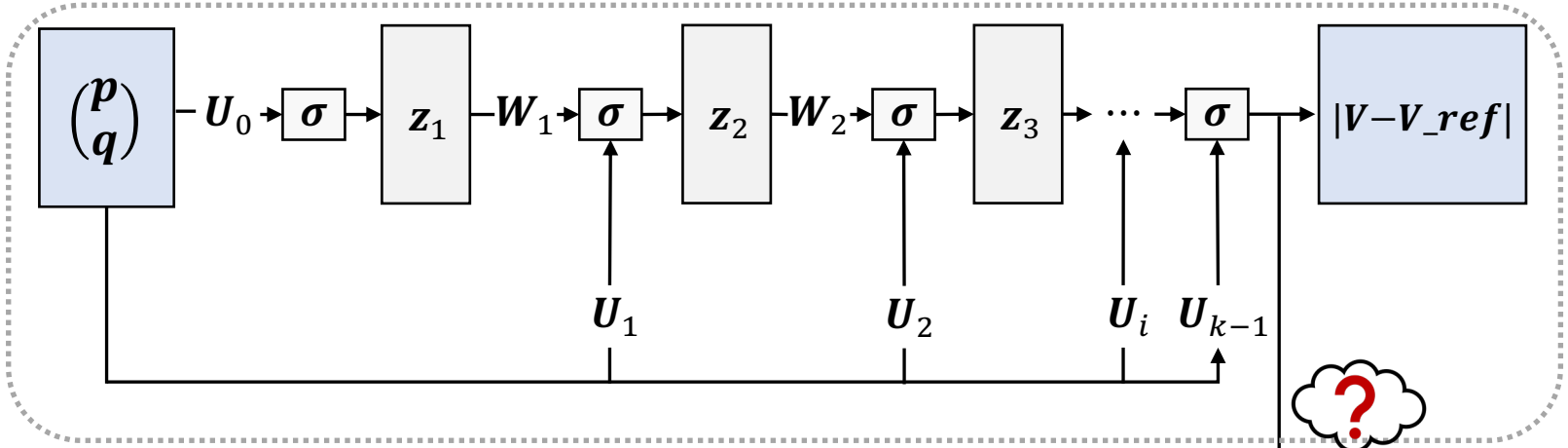


DER Growth

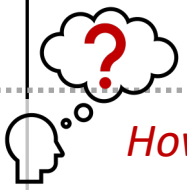
New Operation Points: with More Fluctuation on DER Generations

Calling → Generalizability

Redesigned ICNN with Generalizability

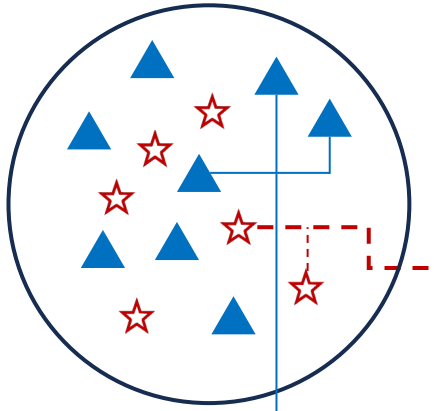


***G**eneralized* **I**nput **C**onvex **N**eural **N**etwork (GICNN)



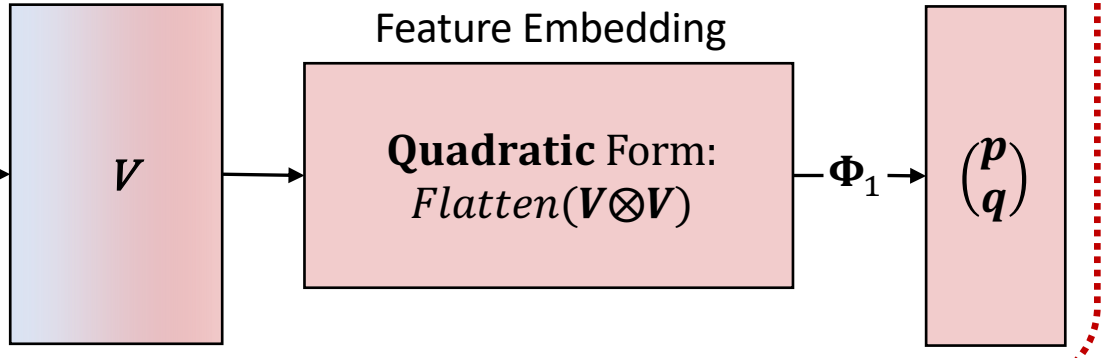
How to add physical guidance?

Unseen Operating Points with More inverter-based DERs

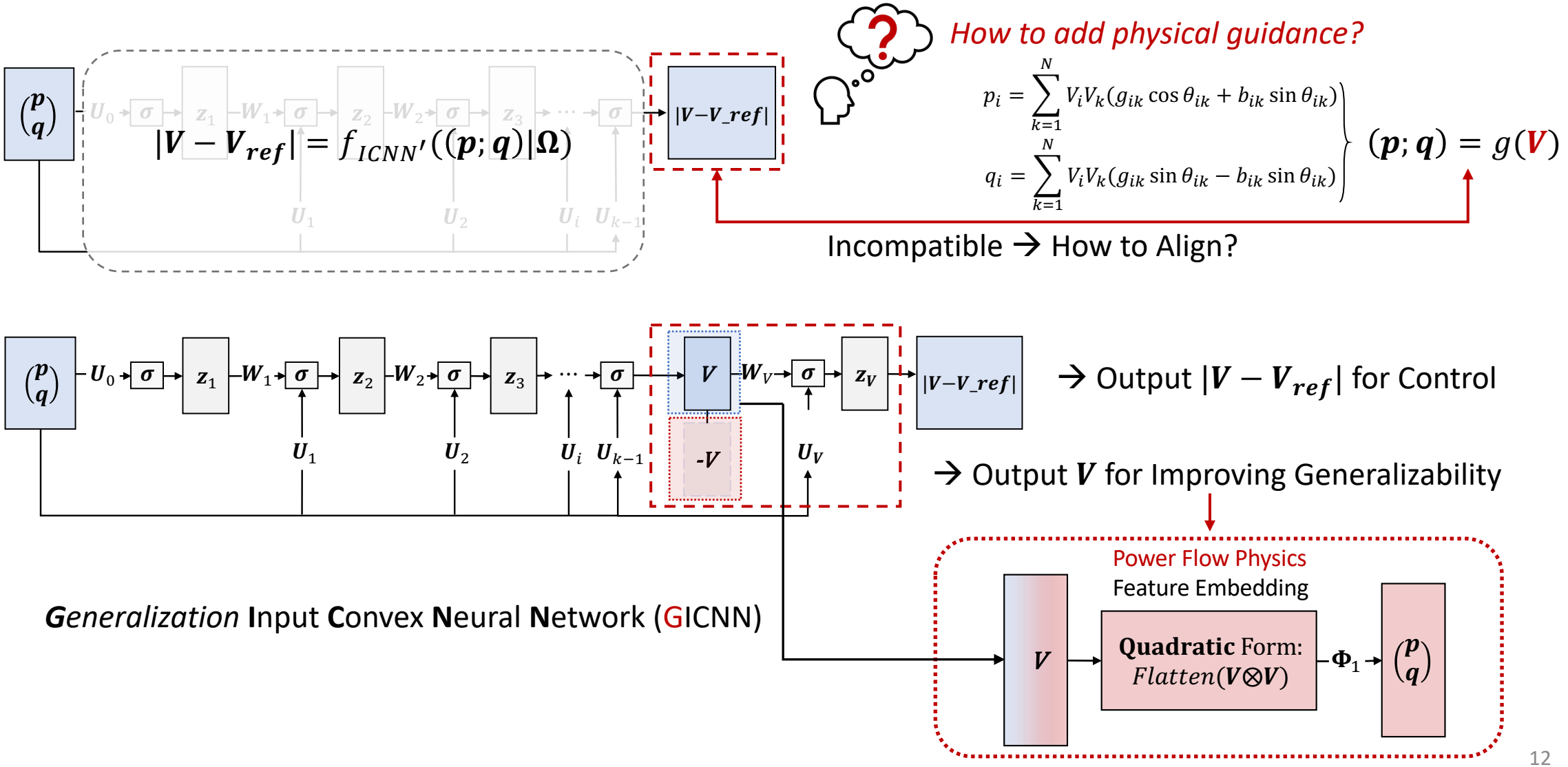


Historical Operating Points $(\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{V})$

Power Flow Physics Feature Embedding



Redesigned ICNN with Generalizability



Redesigned ICNN with Generalizability

Regularization on Physics Consistency

$$\min_{\Omega,\Phi} \sum \left[\boldsymbol{V} - f_{ICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q})\right]^2 + \lambda[(\boldsymbol{p};\boldsymbol{q}) - g_\Phi(\boldsymbol{V})]^2$$

$$\min_{\Omega,\Phi} \sum \left[\boldsymbol{V} - f_{ICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q})\right]^2 + \lambda\left[\boldsymbol{V} - f_{ICNN'_\Omega}(g_\Phi(\boldsymbol{V}))\right]^2$$

$$\min_{\Omega,\Phi} \sum \left[\boldsymbol{V} - f_{ICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q})\right]^2 + \lambda\left[(\boldsymbol{p};\boldsymbol{q}) - g_\Phi(f_{ICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q}))\right]^2$$

GICNN: Cyclic Structure of Bidirectional Mapping

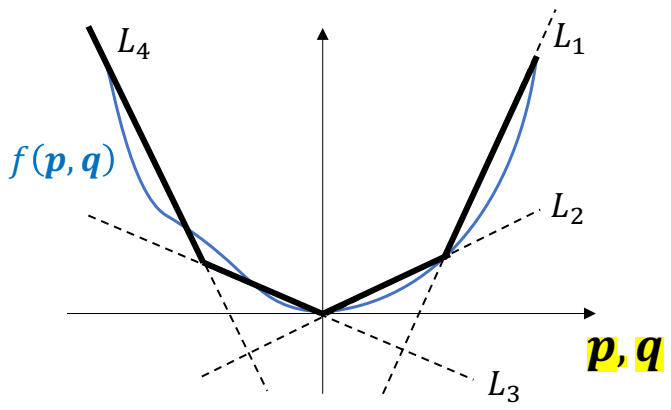
$$\min_{\Omega,\Phi} \sum \left[\boldsymbol{V} - f_{GICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q})\right]^2 + \lambda\left[(\boldsymbol{p};\boldsymbol{q}) - g_\Phi(f_{GICNN'_\Omega}(\boldsymbol{p};\boldsymbol{q}))\right]^2 + \lambda'\left[\boldsymbol{V} - f_{GICNN_\Omega}(g_\Phi(\boldsymbol{V}))\right]^2$$

Representation Capacity of Generalization ICNN (GICNN)

Theorem 1. For the objective $|\boldsymbol{V} - \boldsymbol{V}_{\mathrm{ref}}| \coloneqq f(\boldsymbol{p}, \boldsymbol{q})$ and arbitrary $\varepsilon > 0$, there exist a GICNN network $f_{GICNN_\Omega}(\boldsymbol{p}, \boldsymbol{q})$ such that

$$\sup_{\boldsymbol{p},\boldsymbol{q}} \left\| f_{GICNN_\Omega}(\boldsymbol{p}, \boldsymbol{q}) - f(\boldsymbol{p}, \boldsymbol{q}) \right\| < \varepsilon$$

Proof: $\exists K > 0$, *and* K *affine functions* $L_1, \cdots, L_k$, *such that the maximum of these affine functions can approximate* f *with accuracy* ε, *i.e.,* $\sup_{\boldsymbol{p},\boldsymbol{q}} |f(\boldsymbol{p}, \boldsymbol{q}) - \max\{L_1, L_2, \cdots, L_K\}| \leq \varepsilon$.



$$\| \quad L_K + \sigma(L_{K-1} - L_K + \cdots + \sigma(\cdots + \sigma(L_1 - L_2)))$$

→ GICNN

Generalization Guarantee of GICNN

Definition 2. [*Generalizability* Gap] To quantify the generalization capability of a model f, the generalization gap is defined as $\epsilon(f) = \underbrace{\mathbb{E}_{(x,y)\sim\mathfrak{D}}[f(x)-y]^2}_{\text{Error on training set + unseen data}} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}[f(x_i)-y_i]^2}_{\text{Error on training set}}$

Theorem 3. [GICNN *Generalizability* Improvement]

Suppose L is the number of hidden layers, P is the number of parameters, and n is the size of training set. The **ICNN'** model $\min_{\Omega}\left[V - f_{ICNN_\Omega}(p)\right]^2$ has generalization gap bounded as $\epsilon\left(f_{ICNN_\Omega}\right) \le \mathcal{O}\left(\sqrt{LP/n}\right)$ [Imaizumi 22].

The **GICNN** model $\min_{\Omega,\Phi}\ \left[V - f_{GICNN_\Omega}(p)\right]^2 + \lambda\left[p - g_\Phi(f_{GICNN_\Omega}(p))\right]^2 + \lambda'\left[V - f_{GICNN_\Omega}(g_\Phi(V))\right]^2$ has generalization gap bounded as $\epsilon\left(f_{GICNN_\Omega}\right) \le \mathcal{O}\left(\sqrt{L^{1-2L}\log P\,/n}\right) \le \epsilon\left(f_{ICNN_\Omega}\right)$ [Taheri 22].

Idea:

handle f^{-1}?

$\min_{\Omega}\ [V - f_\Omega(p)]^2$

s. t. f^{-1} is mpoly(2)

second-degree multivariate polynomial

leverage topology information

dual network

constraint → penalty?

$\min_{\Omega}\ [V - f_\Omega(p)]^2$

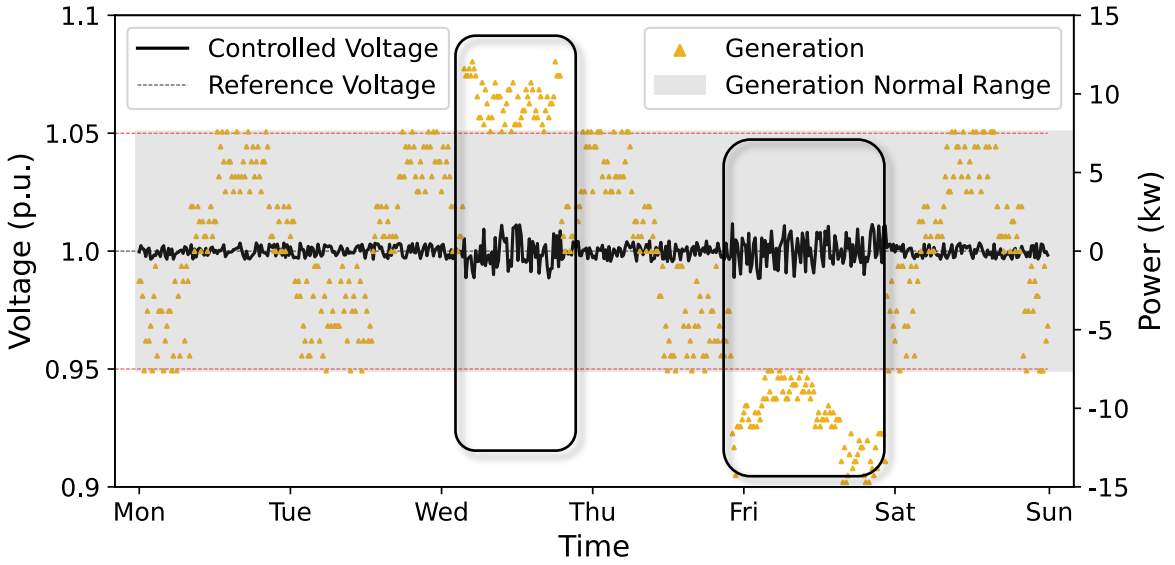
s. t. $f(g)$ is identity

g is mpoly(2)

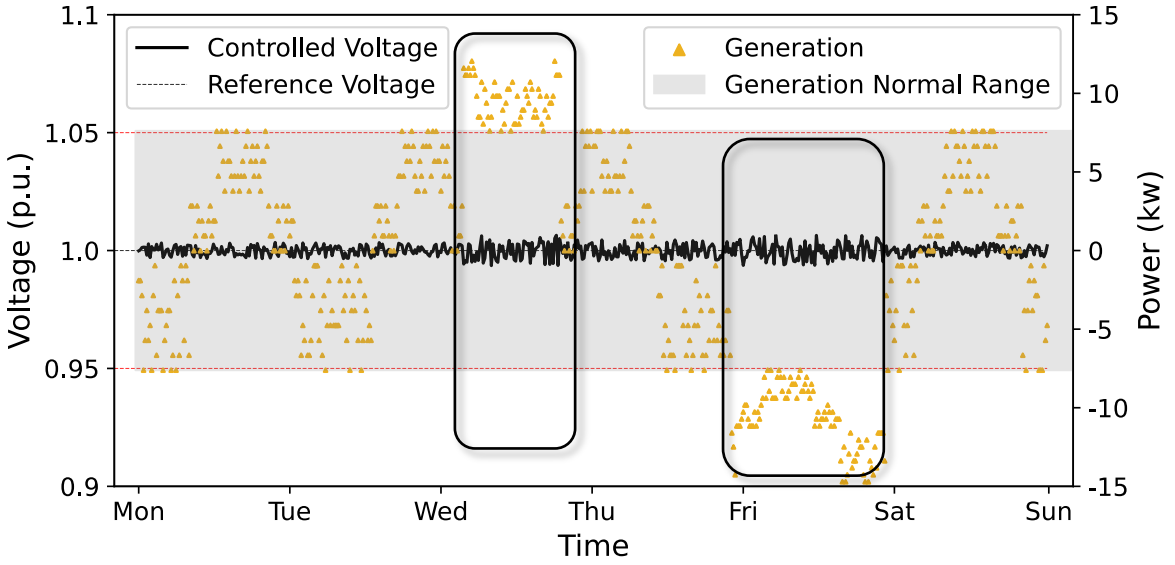
cyclic structure

Generalizability Successful: Physics Helps for the Future

Basic ICNN



GICNN



Better

Comparison

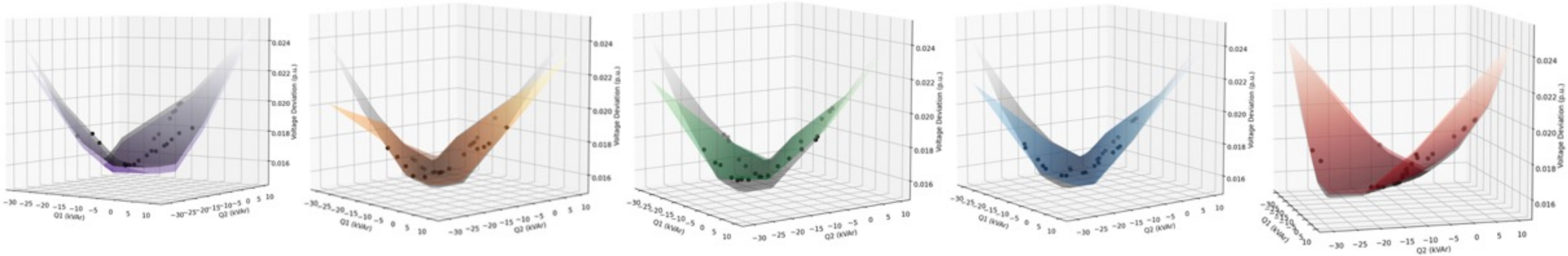
Convex relaxation with large errors

DNN may converge to a local optimal.

ICNN with (u, -u) guarantees a global optimal.

ICNN with (u)

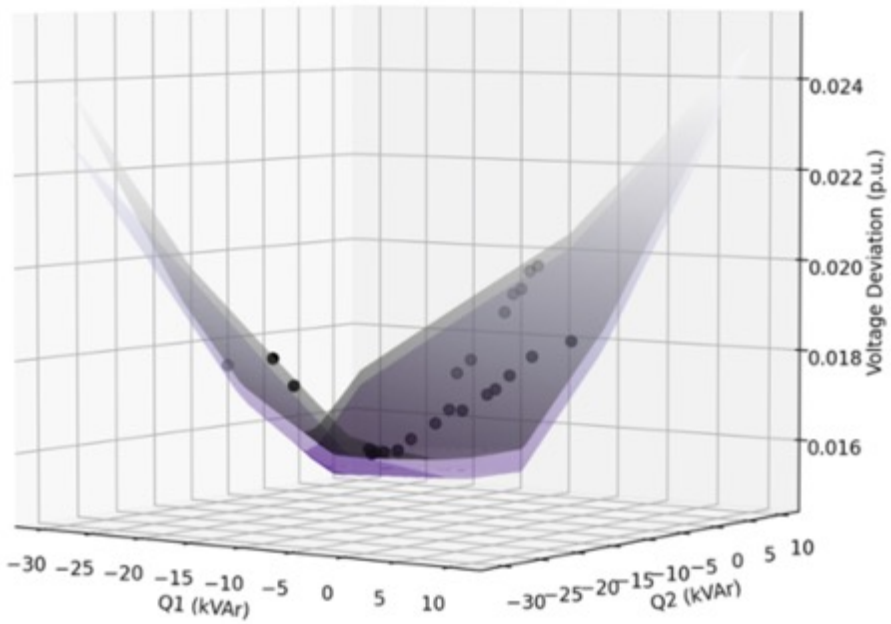
GICNN: Generalizability and the global optimal.



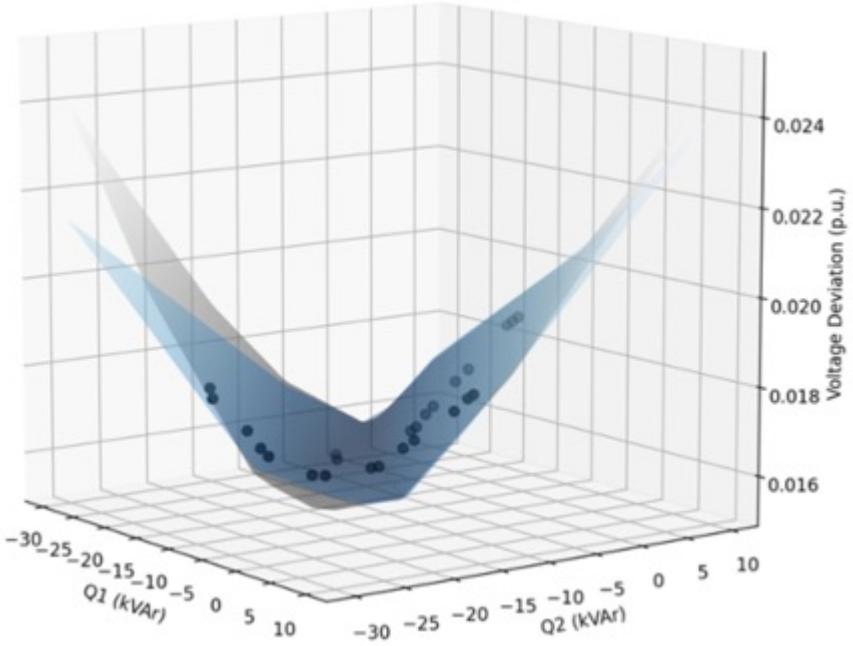
Grey: Variable relationship according to physics

Comparison

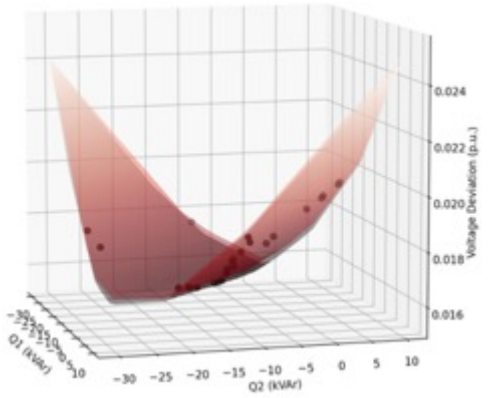
Convex relaxation
with large errors



ICNN



GICNN, Generalizability.
and global optimal.



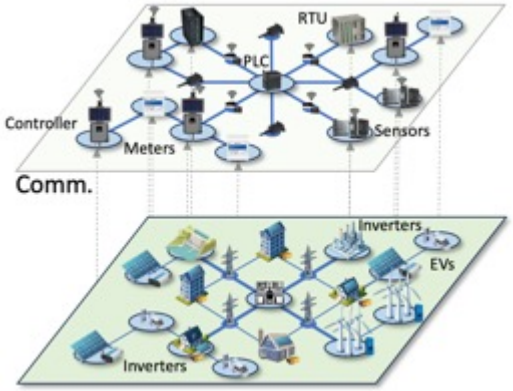
Grey: Variable relationship according to physics

Outline

Normal Conditions AI-Enhanced Optimal Control

Good: Now, Future

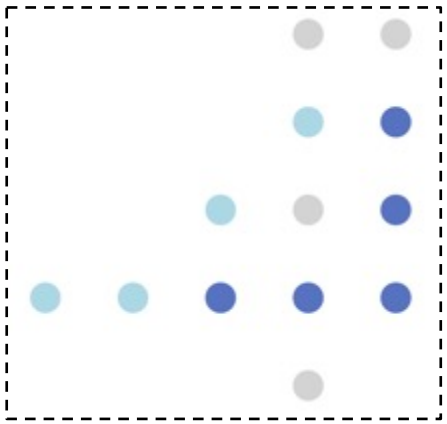
- Optimal Voltage Control
  - Now: Input Convex Neural Network for Optimality
  - Future: Generalizability Design
- Operate with Limited System Information
  - Deep Learning Twin for Physical Consistency
- Operate Through Compromise
  - Inject False Data without System Information
  - Defense Strategy and its Optimal Control
- Conclusion



Learn the Power Flow Equation in Distribution Grids

Why Today?

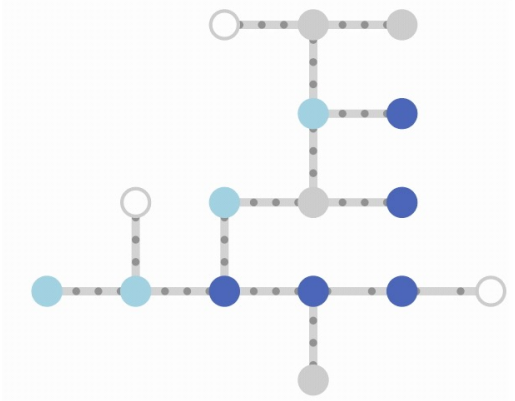
- Given: Sensor Data



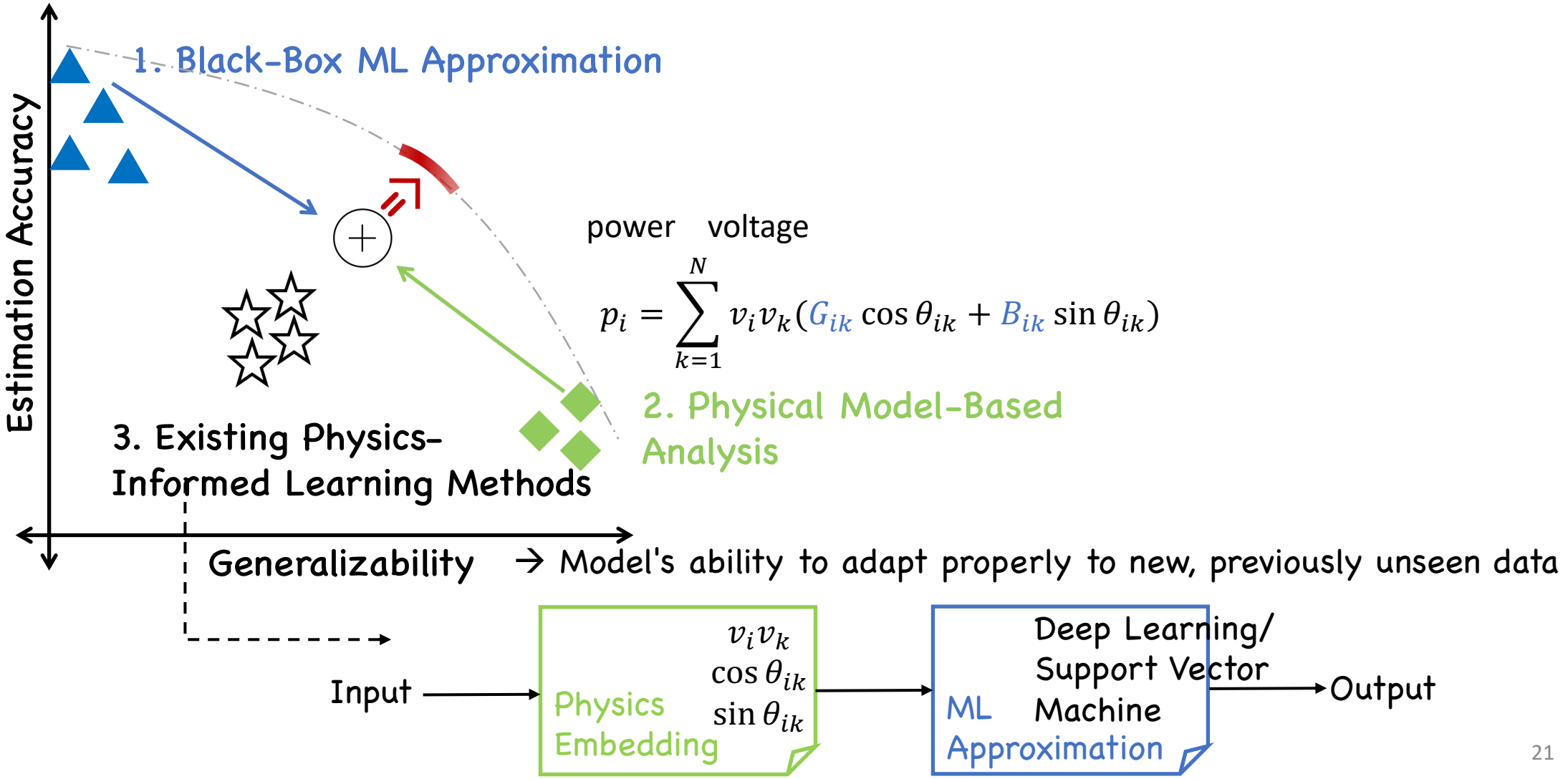
voltage power
(v, p)

- ✓, ✓
- ✓, X
- X, ✓
- X, X

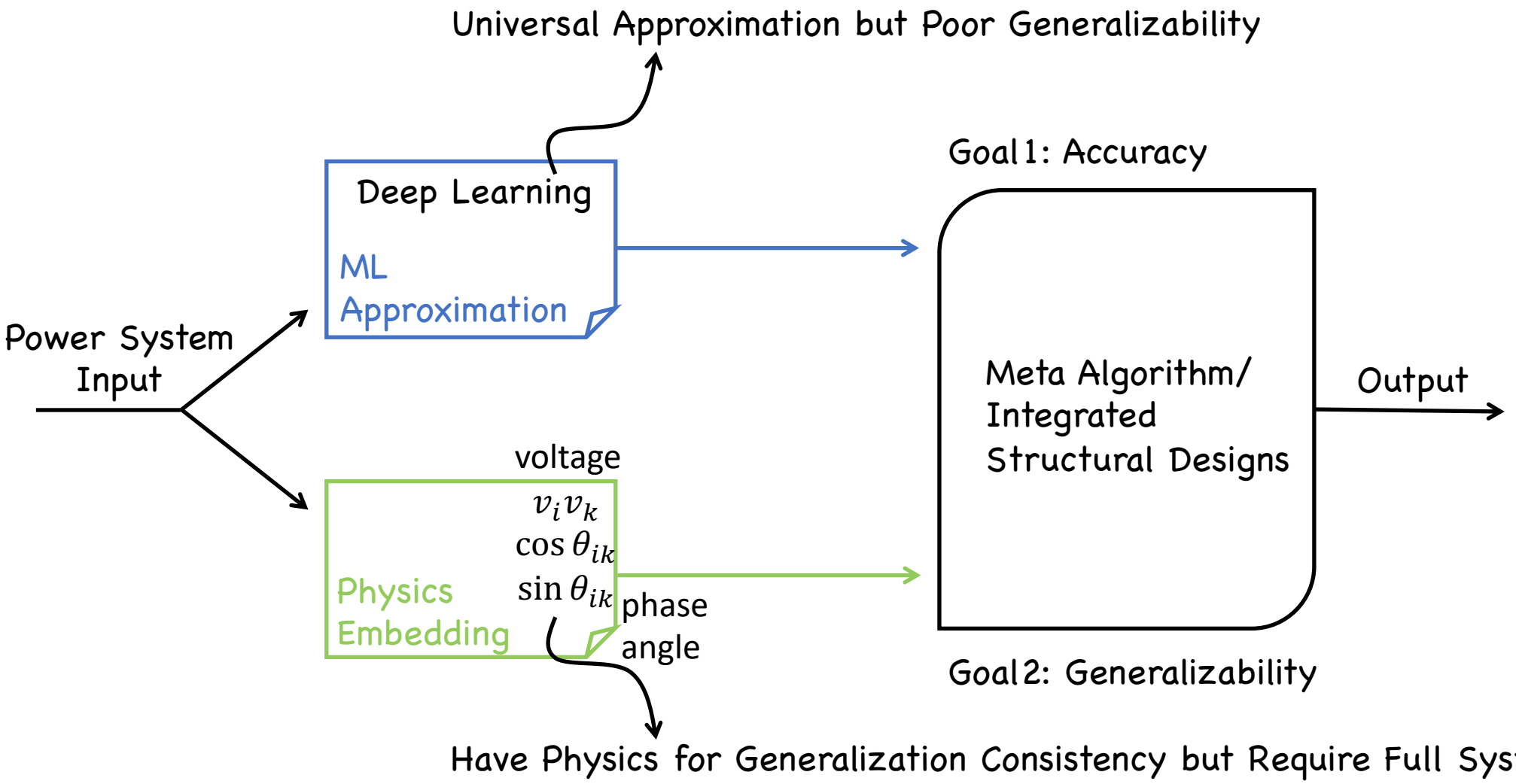
- Find: the Topology, Parameter, and Virtual Nodes for Power Flow Equation



Past Methods



New Idea: Series to Parallel → Enable Flexibility



Mathematical Modeling

Power Flow Equation

$$p_i = \sum_{k=1}^{N} v_i v_k (G_{ik} \cos\theta_{ik} + B_{ik} \sin\theta_{ik})$$

Physical Parameter

Data Data

$$\boldsymbol{p} = \begin{pmatrix} p_i^{(1)} \\ \cdots \\ p_i^{(T)} \end{pmatrix} = \begin{bmatrix} v_i^{(1)} v_1^{(1)} \cos\theta_{i1}^{(1)} & \cdots & v_i^{(1)} v_N^{(1)} \cos\theta_{iN}^{(1)} & v_i^{(1)} v_1^{(1)} \sin\theta_{i1}^{(1)} & \cdots & v_i^{(1)} v_N^{(1)} \sin\theta_{iN}^{(1)} \\ & & & \cdots & \cdots & \\ v_i^{(T)} v_1^{(T)} \cos\theta_{i1}^{(T)} & \cdots & v_i^{(T)} v_N^{(T)} \cos\theta_{iN}^{(T)} & v_i^{(T)} v_1^{(T)} \sin\theta_{i1}^{(T)} & \cdots & v_i^{(T)} v_N^{(T)} \sin\theta_{iN}^{(T)} \end{bmatrix} \begin{pmatrix} G_{i1} \\ \cdots \\ G_{iN} \\ B_{i1} \\ \cdots \\ B_{iN} \end{pmatrix}$$

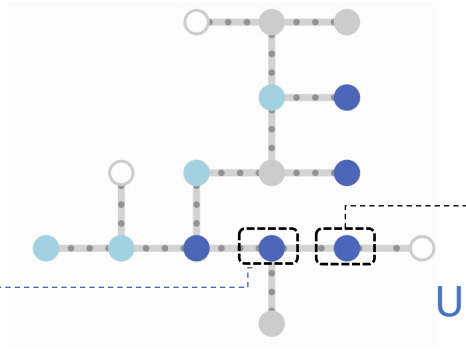
$\phi(\mathrm{V}, \Theta) \in \mathbb{R}^{T \times 2N}$

$\boldsymbol{\beta}^* \in \mathbb{R}^{2N}$

$\boldsymbol{p} = \phi\, \boldsymbol{\beta}^*$

$\boldsymbol{p} \in \mathbb{R}^T$: power injection at bus i up to time T, $\mathrm{V} = (v_k^{(t)}) \in \mathbb{R}^{T \times N}$: voltage magnitude at N buses up to time T,
$\Theta = (\theta_{ik}^{(t)}) \in \mathbb{R}^{T \times N}$: voltage angle difference between bus i and other buses up to time T, and $T \in \mathbb{N}$: No. of historical samples.

Separate the Physically Recoverable Part and the Virtual Parts



Observable bus set $\mathcal{O}$

Unknown set $\mathcal{U}$

Boundary set B

$$\boldsymbol{p} = \phi(\mathrm{V},\Theta)\boldsymbol{\beta}^* = \underbrace{(\phi(\mathrm{V}_{\mathcal{O}},\Theta_{\mathcal{O}}) \quad \phi(\mathrm{V}_{\mathcal{O}B},\Theta_{\mathcal{O}B})}_{(\mathrm{V},\Theta)} \quad \underbrace{\phi(\mathrm{V}_{B\mathcal{U}},\Theta_{B\mathcal{U}}) \quad \phi(\mathrm{V}_{\mathcal{U}},\Theta_{\mathcal{U}})}_{\text{Unknown}} \begin{pmatrix} \boldsymbol{\beta}_{\mathcal{O}\mathcal{O}} \\ \boldsymbol{\beta}_{\mathcal{O}B} \\ \boldsymbol{\beta}_{B\mathcal{U}} \\ \boldsymbol{\beta}_{\mathcal{U}\mathcal{U}} \end{pmatrix}$$

If Physically Learnable:

Physical Contribution

Borrow

$$\boldsymbol{p} = \boldsymbol{h}_p\big(\phi(\mathrm{V},\Theta)\big) \quad + \quad \boldsymbol{h}_v(\mathrm{V},\Theta)$$

Use Blackbox Learning Methods

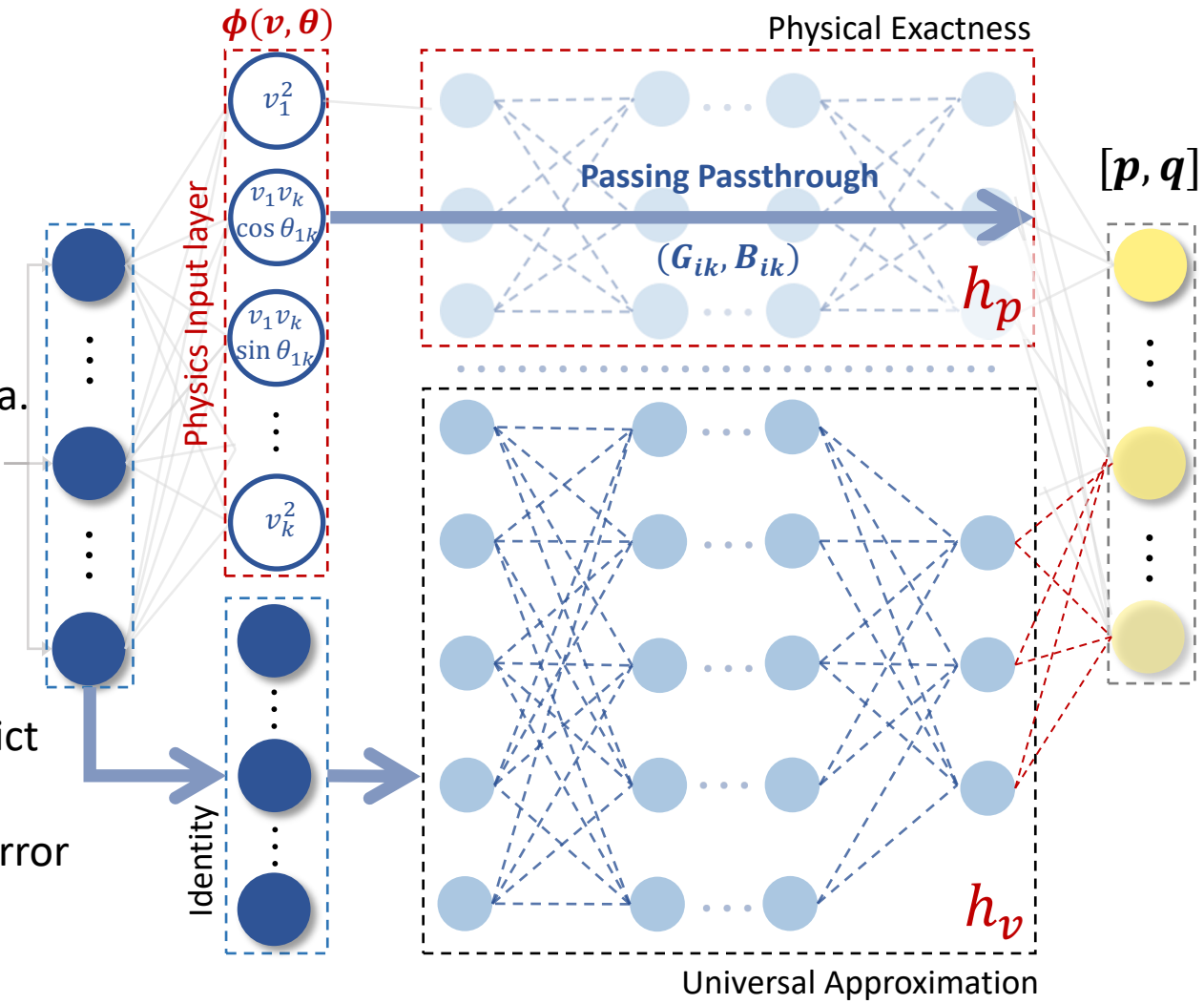
Twin Neural Network Model and the Objective

Optimization

$$\min \varepsilon_{\boldsymbol{p}}^2 = \min \frac{1}{T} ||\boldsymbol{p} - f(\mathrm{V}, \Theta)||_2^2$$

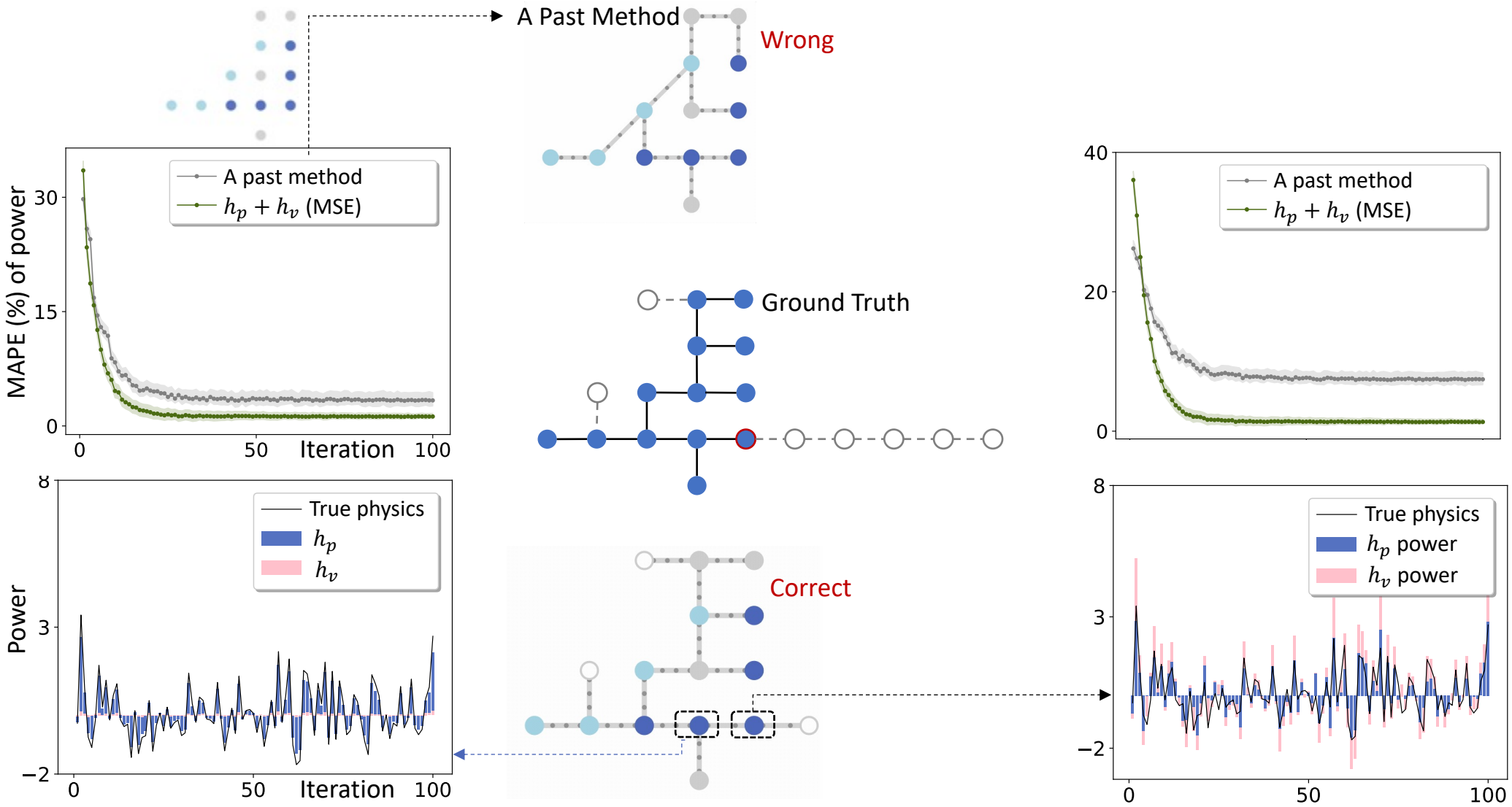
$$= \min_{\boldsymbol{\beta}, W_v} \frac{1}{T} \sum_{t=1}^{T} ||\boldsymbol{p}_t - h_p\big(\phi(\boldsymbol{v}_t, \boldsymbol{\theta}_t)\big) - h_v(\boldsymbol{v}_t, \boldsymbol{\theta}_t)||_2^2$$

Condition for the universal approximation:
there exists a function $f(.,.)$ to perfectly fit the data.



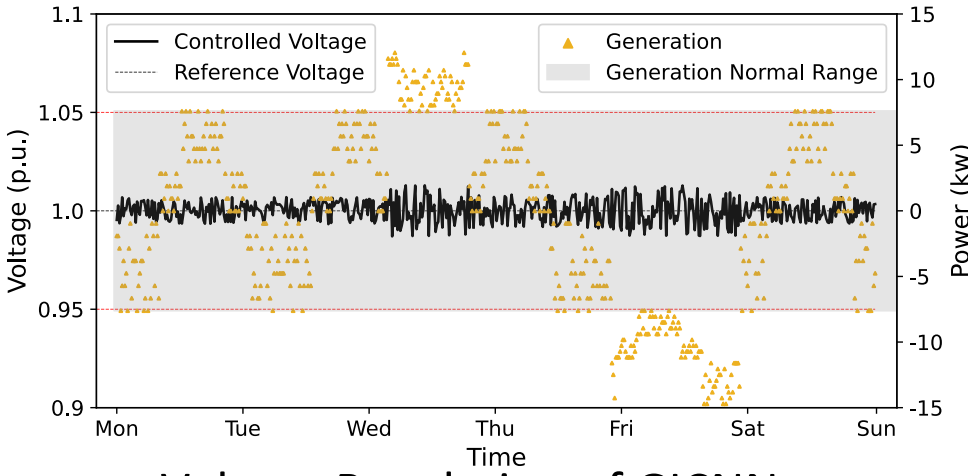
Theorem. Our proposed method can accurately recover the physical parameters ($\mathbb{E}[\varepsilon_\beta] \to 0$) and restrict the error of predicting power, given sufficient data samples and training capacity. The power estimation error is upper bounded as $\mathbb{E}\left[\varepsilon_p^2\right] \leq 2 \cdot [\mathrm{Var}(\boldsymbol{p}_v) - \mathrm{Var}[\mathrm{E}(\boldsymbol{p}_v|\boldsymbol{v})]$, where $\boldsymbol{p}_v$ denotes unobserved virtual power and $\boldsymbol{v}$ denotes observed voltage data.

Numerical Result: Proposed Collaboration is Better

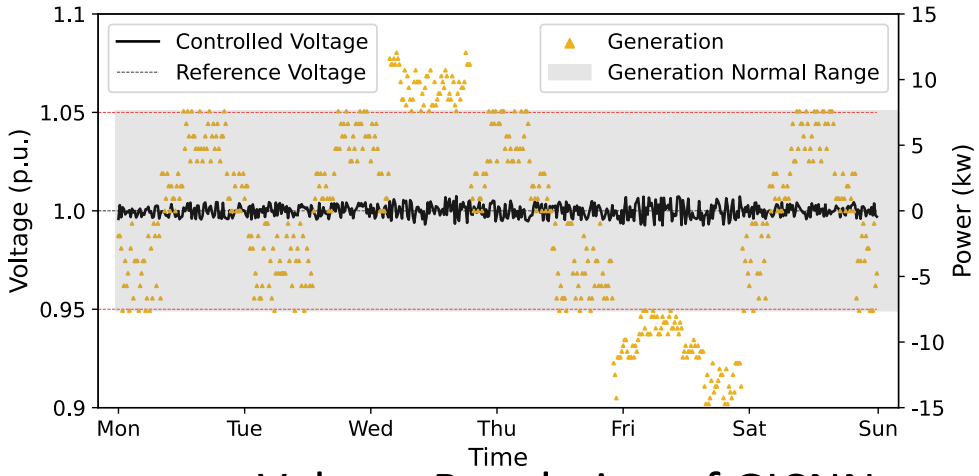


Numerical Results with Partial Observability

40% Unobservable → 80% Unobservable



Voltage Regulation of GICNN



Voltage Regulation of GICNN

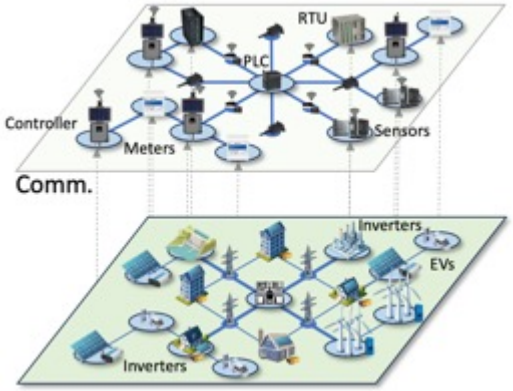
≈ Full observable performance

Outline

Normal Conditions AI-Enhanced Optimal Control

Good: Now, Future

- Optimal Voltage Control
  - Now: Input Convex Neural Network for Optimality
  - Future: Generalizability Design
- Operate with Limited System Information
  - Deep Learning Twin for Physical Consistency
- Operate Through Compromise
  - Inject False Data without System Information
  - Defense Strategy and its Optimal Control
- Conclusion

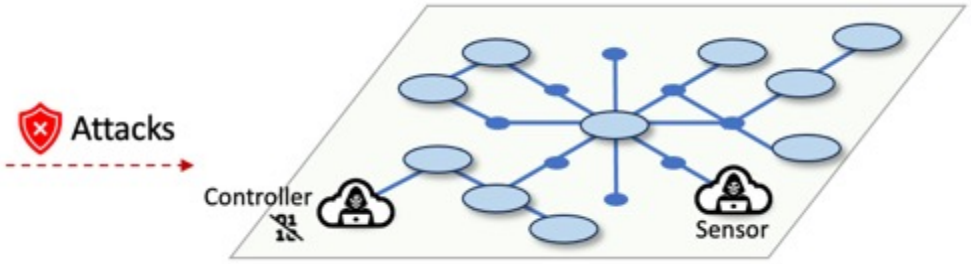


Outline

Normal Conditions AI-Enhanced Optimal Control

Good: Now, Future

- Optimal Voltage Control
  - Now: Input Convex Neural Network for Optimality
  - Future: Generalizability Design
- Operate with Limited System Information
  - Deep Learning Twin for Physical Consistency
- Operate Through Compromise
  - Inject False Data without System Information
  - Defense Strategy and its Optimal Control
- Conclusion



Past Work on False Data Injection Attack (FIDA)

Sensors
Controller

False Measurement

$z \rightarrow z_{attack}$

SCADA

Impact

Economic Attack [Xie 11, Choi 13, Jia 14, Rahman 14]

Load Redistribution Attack [Yuan 11, Yuan 12]

Energy Deceiving Attack [Lin 12]

State Estimation

$$\hat{x} = (H^{\top}WH)^{-1}H^{\top}Wz$$

Defense

Bad Data Detector

Largest Normalized Residual [Liu 11]

PMU-based Protection [Khare 21, Khan 23]

ML-based detection [Khan 22, Kumar 21]

New Types of Attacks: Constraints

AC Power Flow Model [Jin 17, Jiao 21]

Access to Limited Sensors [Sreeram 19]

Incomplete System Info. [Li 18, Pan 22]

False Topology [Sun 15, Lan 17]

Reduce Info for Security?

Attacks without System Information

Let $\boldsymbol{p}$ represents all the measurements

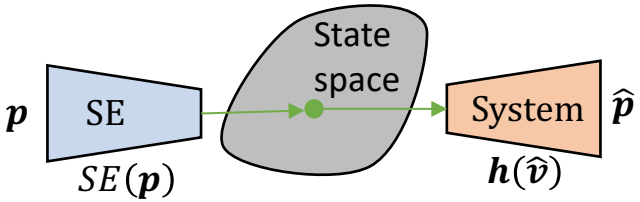
Chi-Square Test

$$J(\boldsymbol{p}) = \|\boldsymbol{p} - \boldsymbol{h}(SE(\boldsymbol{p}))\|^2 \geq \tau$$

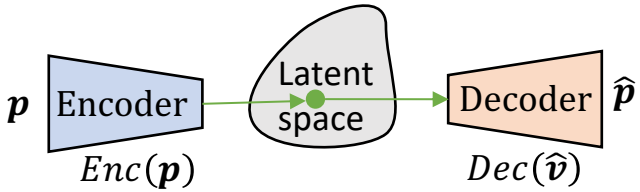
Observations: $SE(\cdot)$: $\boldsymbol{p} \rightarrow \hat{\boldsymbol{v}}$, $\boldsymbol{h}(\cdot)$: $\hat{\boldsymbol{v}} \rightarrow \hat{\boldsymbol{p}}$

Unknown

Minimize?



$$\|\boldsymbol{p} - \boldsymbol{h}(SE(\boldsymbol{p}))\|^2$$



$$\|\boldsymbol{p} - Dec(Enc(\boldsymbol{p}))\|^2$$

Minimize Reconstruction Error

Autoencoder → Similar

Autoencoder → Chi-Square Test (Physics)

Let $\boldsymbol{p}$ represents all the measurements

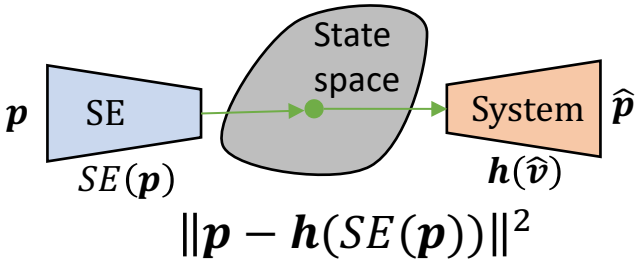
Chi-Square Test

$$J(\boldsymbol{p}) = \|\boldsymbol{p} - \boldsymbol{h}(SE(\boldsymbol{p}))\|^2 \geq \tau$$

Observations: $SE(\cdot)$: $\boldsymbol{p} \rightarrow \hat{\boldsymbol{v}}$, $\boldsymbol{h}(\cdot)$: $\hat{\boldsymbol{v}} \rightarrow \hat{\boldsymbol{p}}$

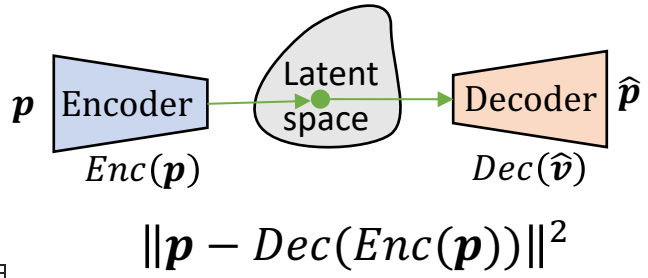
Unknown

Minimize?

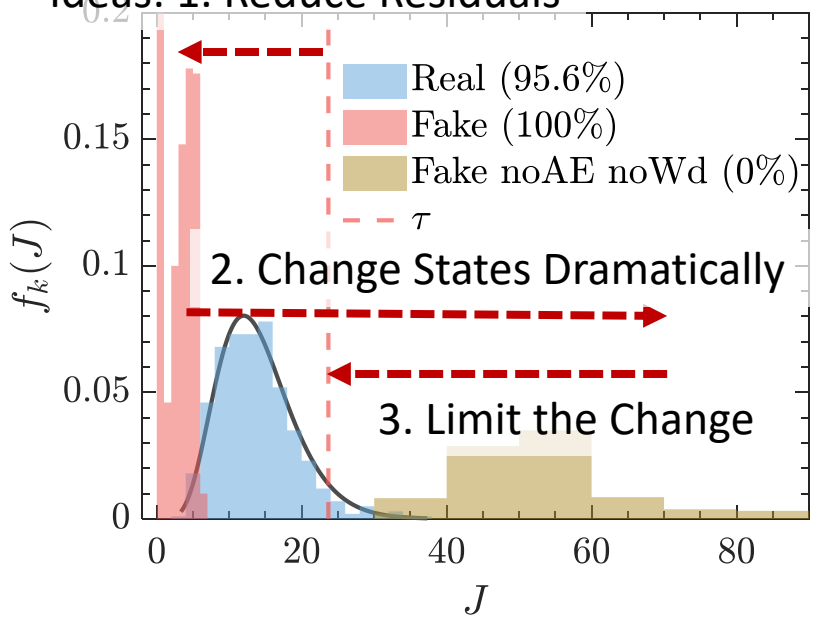


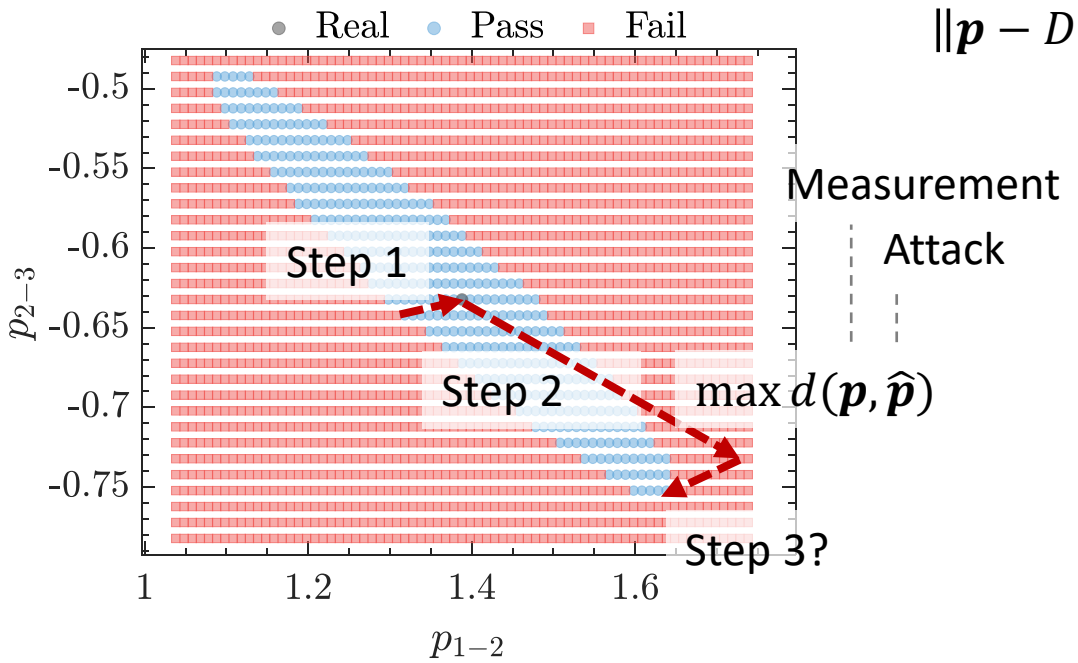
Autoencoder → Similar

Minimize Reconstruction Error



Ideas: 1. Reduce Residuals



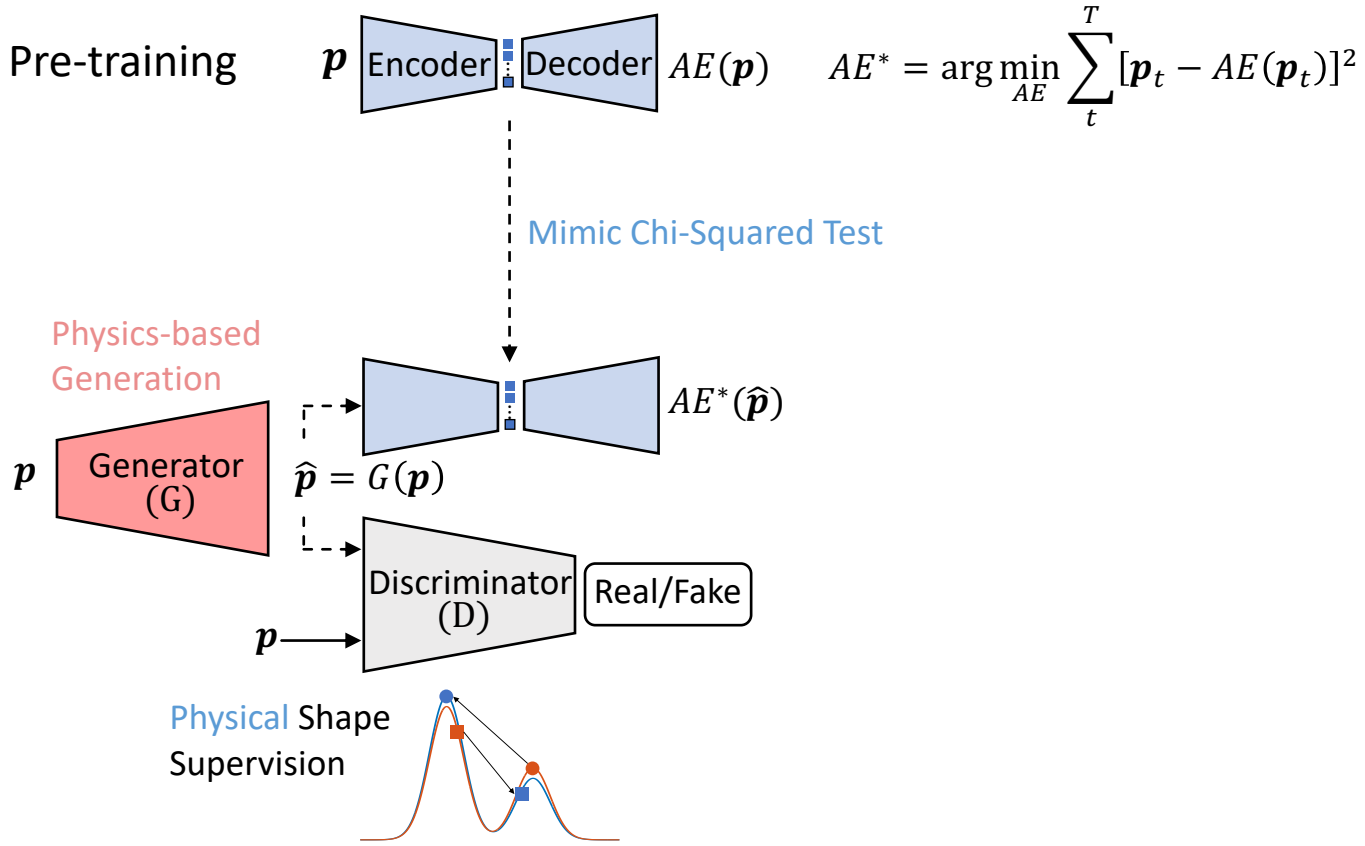


General Framework to Generate False Data

Step 1. [Create a test measure] Train a "virtual" Chi-square test using an Autoencoder and historical measurements $\{\boldsymbol{p}_t\}_t^T$.

Step 2. [Sample high-quality fake data] Train a GAN to approximate the distribution of historical data $\{\boldsymbol{p}_t\}_t^T$.

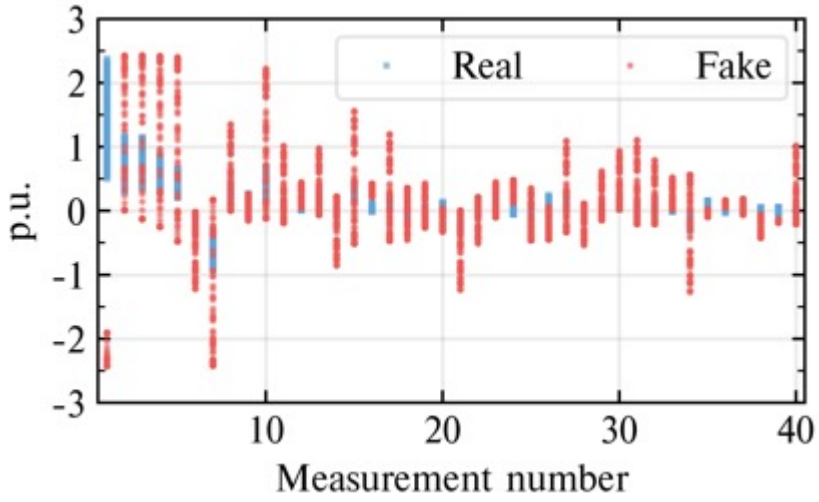
Step 3. [Create false data] Create false data based on the sampled data from GAN. The false data should pass AE.



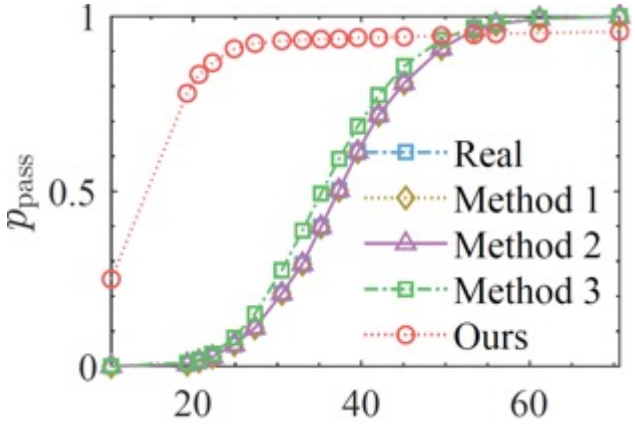
$\mathbb{P}_r$: distribution of real power data, $\mathbb{P}_g$: distribution of generated power data, w: penalty term, $d(\cdot,\cdot)$: distance measure.

Numerical Results and Defense Strategies

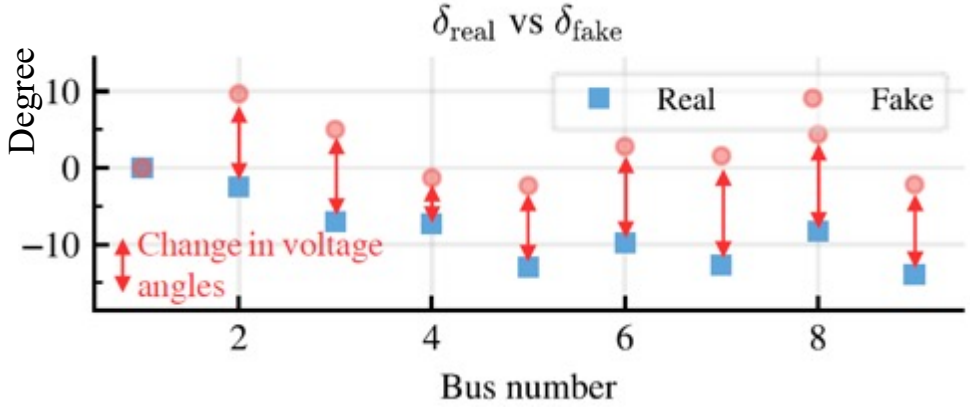
Change Measurements Dramatically
No Alarms



Pass the Tests



Change States Significantly



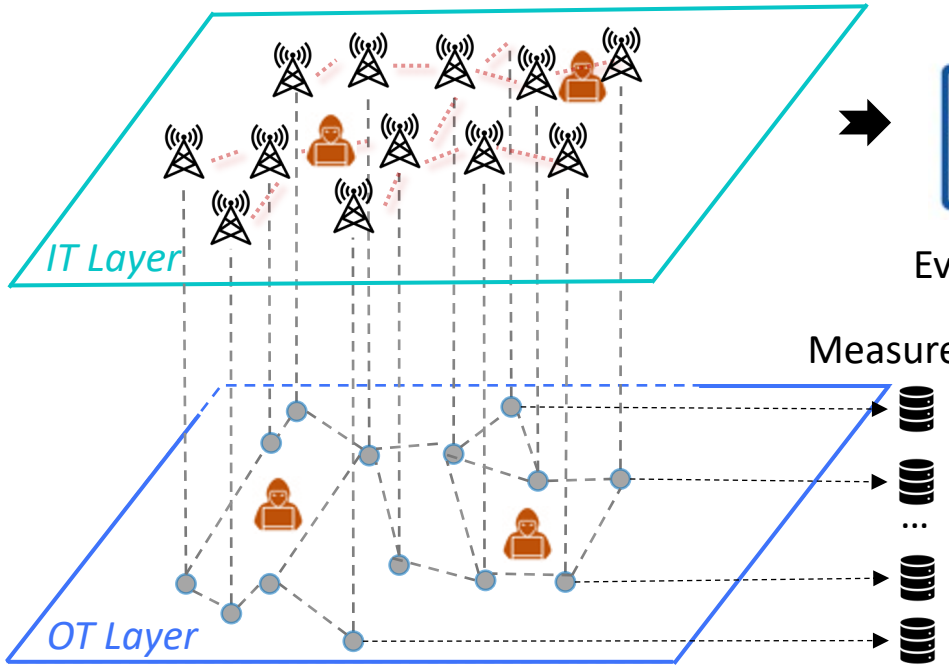
Attack Ideas

1. Learn the Physics
2. Learn the Distribution
3. Close to the Boundary
…

How to Defend?

- Segment Data with Topology Changes
- Use Inertia
- Inject Noise without Changing the States
- …

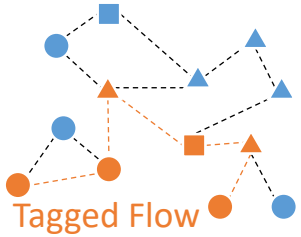
IT/OT Detection Algorithms



Measurements

1. Dynamic Information Flow Tracking [Sahabandu et al 19]: Graph Search → Malicious Behavior

Event logs

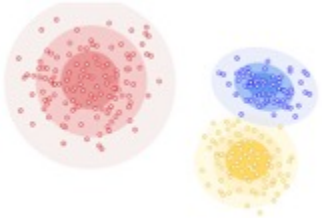


Information Flow Graph

Detect Anomalous Behaviors

2. Statistical Test → Power System Behavior [Xu et al 17]

3. AI-based Classification → Feature Pattern [Wang et al 20]



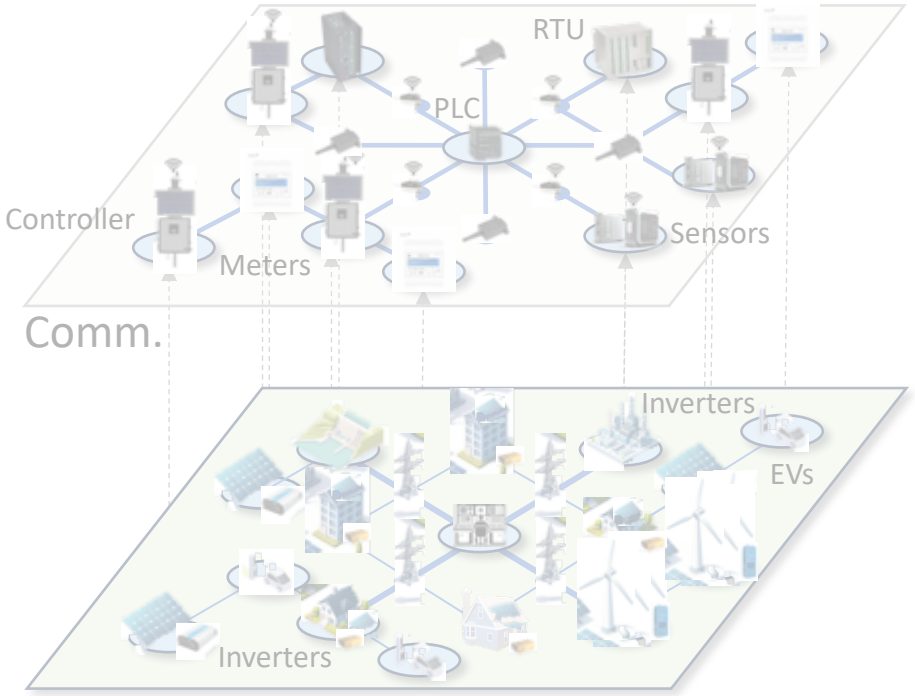
Collaborative Inverter-based Control when Some Fails

Moving from Normal Operation to Operation through Compromise

1. Normal Conditions

AI-Enhanced Optimal Control

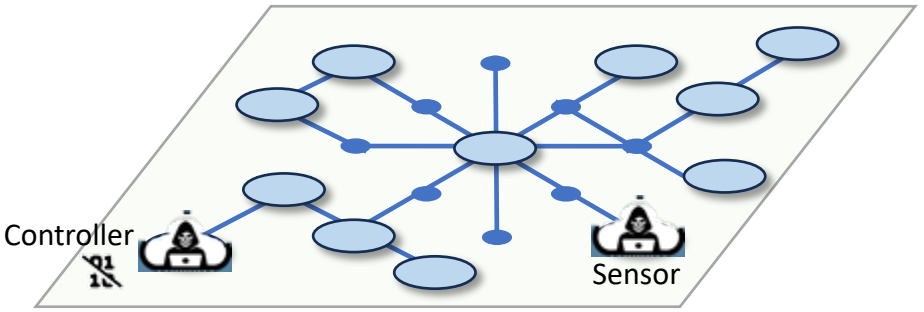
Good: Now
Future



Attacks

2. Under Attack

Operate Through Compromise

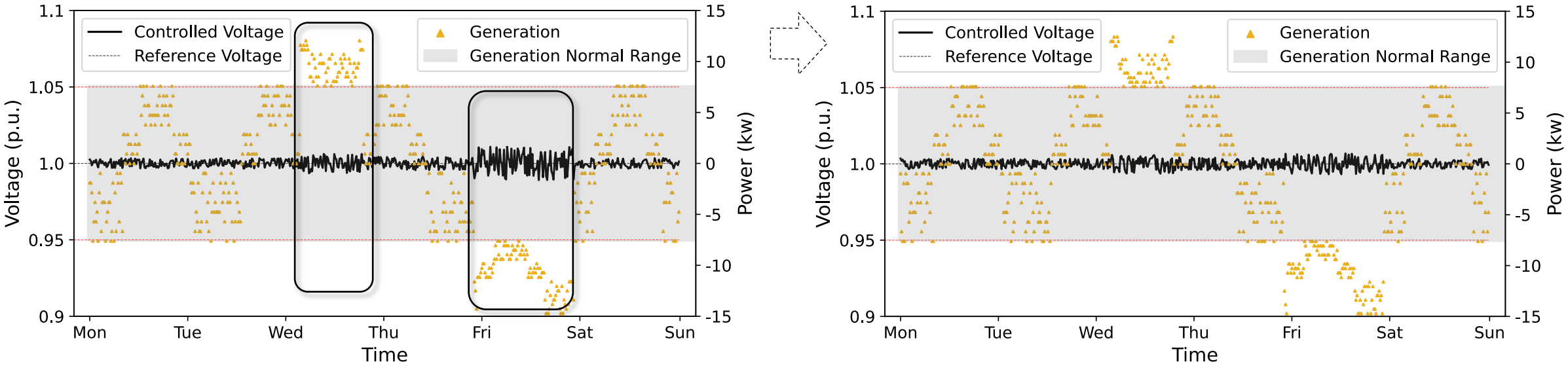


Locate and Remove

Locate and Remove

Improvement by Using GICNN without Attacked Sensors & Controllers

Voltage Regulation with GICNN



Conclusion

1. Machine Guarantees Learning for DER Integration

2. Distributed Control Algorithms with Performance Guarantees

3. Guarantees for Power System Security