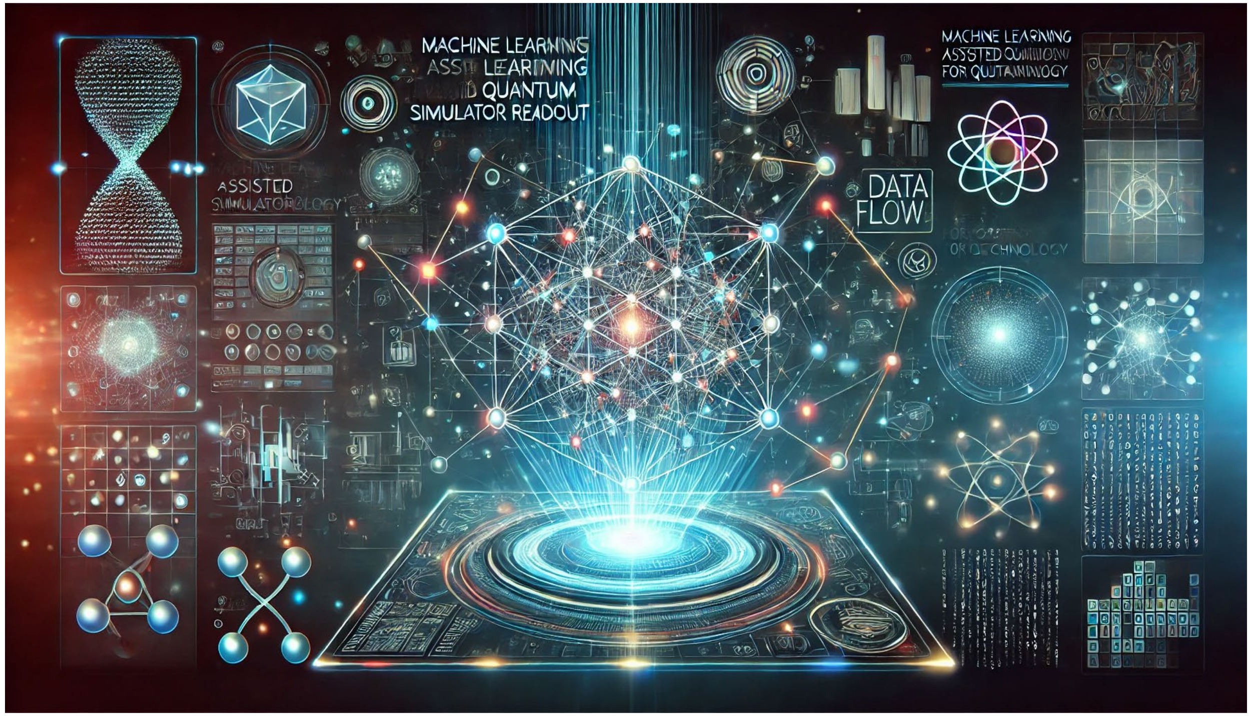
FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

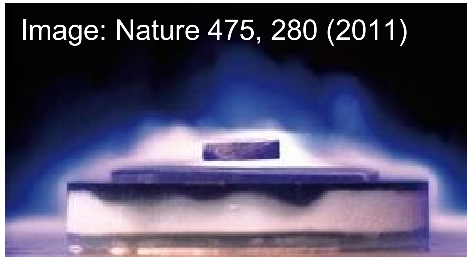


Prof. Dr. Martin Gärttner

Machine learning assisted quantum simulator readout

ML for QT workshop, Erlangen, 06.11.2024

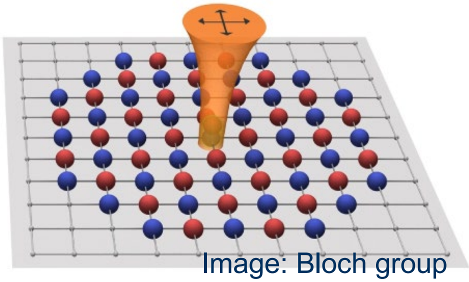
Quantum Simulation



abstraction of

$\hat{\mathcal{H}}$

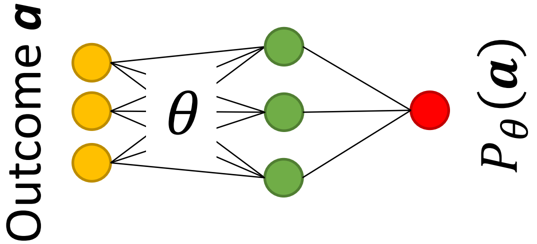
emulates



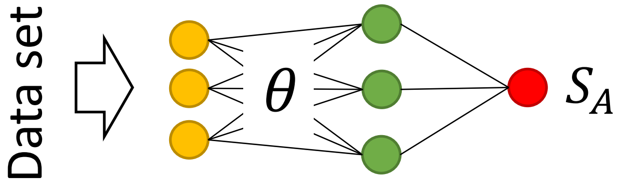
Physical system | Mathematical model | Quantum simulator

Readout challenge: Efficient extraction of observables from data

Idea: Exploit prior knowledge about data distribution



Generative modeling



Supervised learning

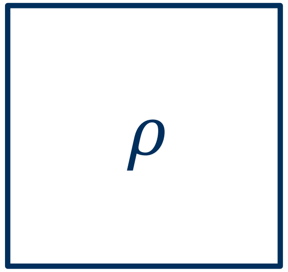
NQS tomography

Review article:

Torlai and Melko, Annual Review of Condensed Matter Physics 11:325-344 (2020)

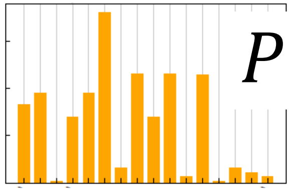
N qubits

2^N



2^N

Tomographically complete



$\geq 4^N$ outcomes

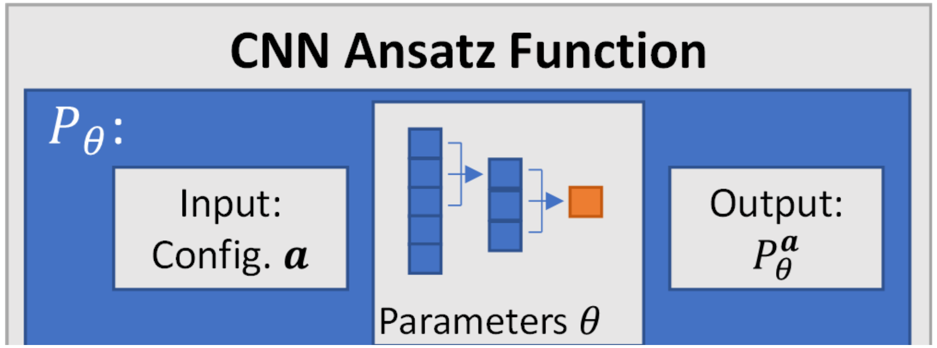
Full reconstruction hard! → Variational approach

e.g. N qubits:
Measure each qubit in X, Y, Z

$$\langle \mathcal{O} \rangle = \sum_{\boldsymbol{a}} Q_{\boldsymbol{a}}^{\mathcal{O}} P^{\boldsymbol{a}} = \langle Q_{\boldsymbol{a}}^{\mathcal{O}} \rangle_{a \sim P}$$

Moritz Reh Tobias Schmale

npj Quantum Information **8**, 115 (2022)



Data set: $(a_1^{(1)}, a_2^{(1)}, \dots a_N^{(1)})$

$(a_1^{(2)}, a_2^{(2)}, \dots a_N^{(2)})$

…

$(a_1^{(N_s)}, a_2^{(N_s)}, \dots a_N^{(N_s)})$

Training:
Maximize likelihood

Model → Data

NQS tomography

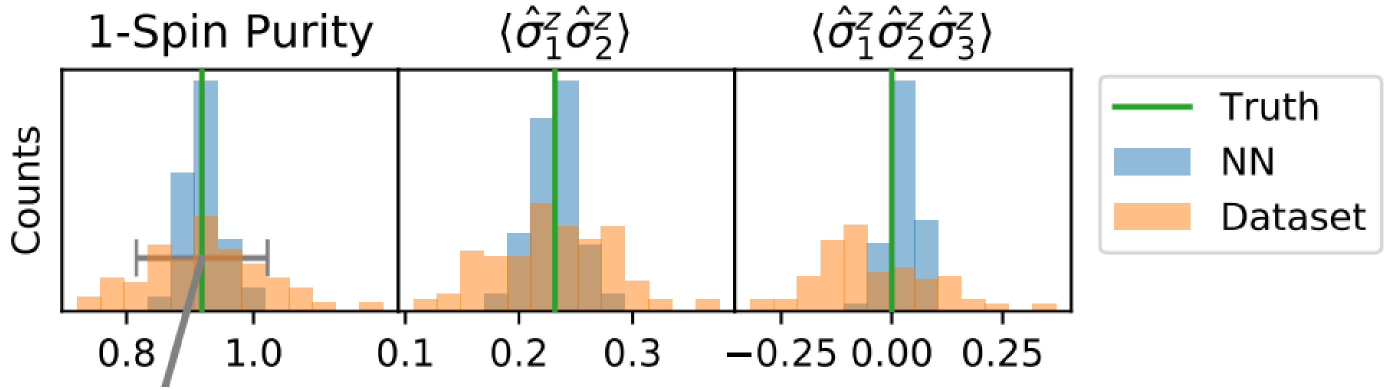
$$H = -J\sum_{\langle i,j\rangle} S_z^i S_z^j - B\sum_i S_x^i$$

Need to build trust to exclude bias!

Moritz Reh Tobias Schmale

npj Quantum Information **8**, 115 (2022)

Critical Ground state $J/B = 1$ 4x4 spins Parameter scaling $\sim N^3$

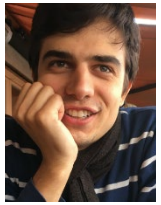


Variance reduction for local observables

Questions:
Controlling bias?
Physical properties of ansatz?

Spiking neural networks as NQS


Stefanie Czischek


Andreas Baumbach

SciPost Phys. **12**, 039 (2022)

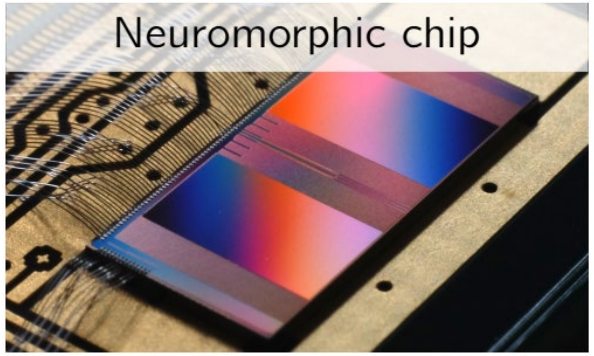
iScience **25**, 104707 (2022)

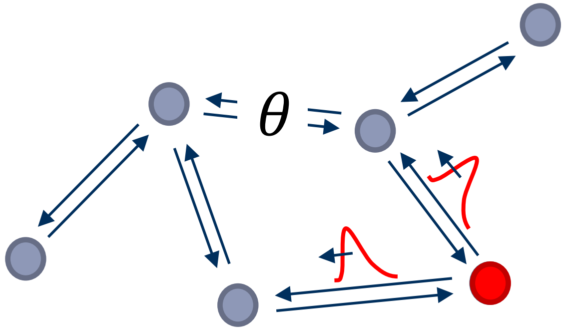
Problem: Sampling can be costly and correlated

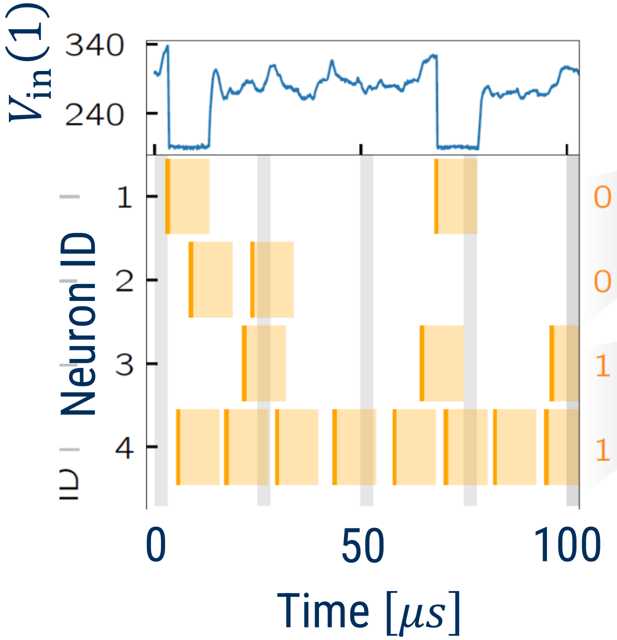
Idea: Use a *physical* neural network

BrainScaleS

BSS-2







→ *Constant* sample rate!

(indep. of network size)

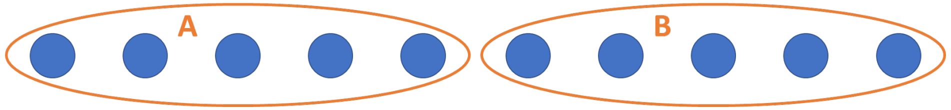
Minimize:

$$D_{\text{KL}}(P\|p) = \sum_{\mathbf{a}} P(\mathbf{a}) \ln \left[\frac{P(\mathbf{a})}{P_\theta(\mathbf{a})} \right]$$

Problem: Gradient determination inefficient

Question: Learning based on local quantities?

Supervised entropy prediction

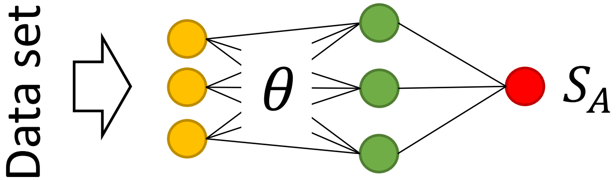


Moritz Reh   Maximilian Rieger

Phys. Rev. A **109**, 012403 (2024)

Renyi entanglement entropy:

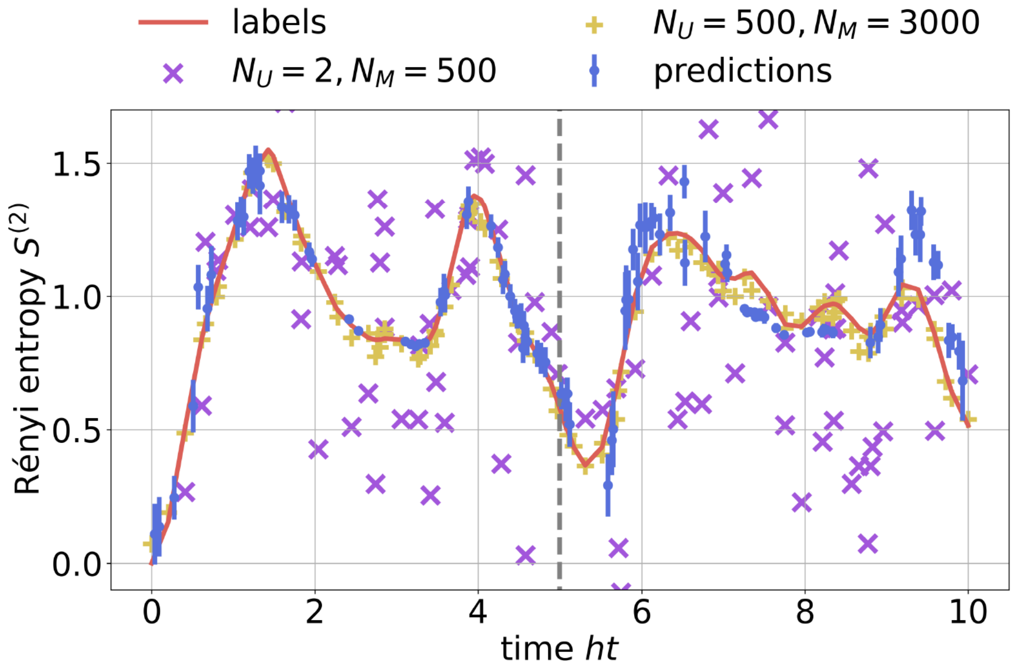
$S_A = -\log[\mathrm{Tr}(\rho_A^2)] \qquad \rho_A = \mathrm{Tr}_B(\rho)$



Input data structure: (N_S, N, d)

Training data: examples with labels S_A

Questions: More data? Extrapolation?



Transverse field Ising time evolution

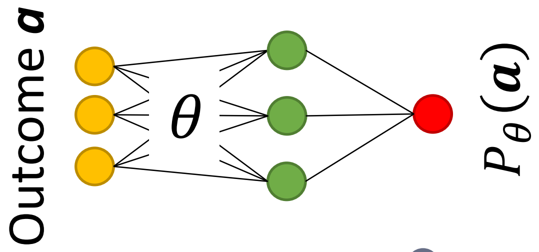
→ Interpolation works

→ Sample efficient

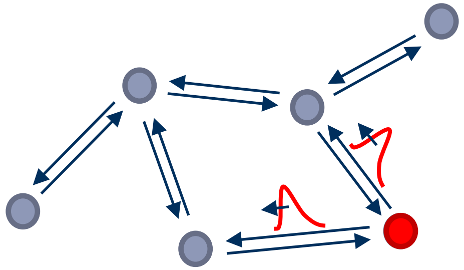
→ Extrapolation fails

Summary

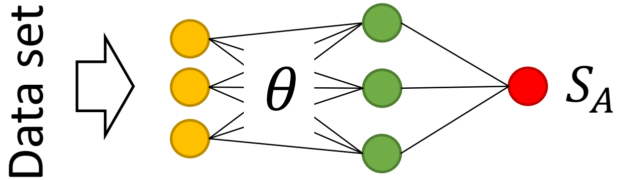
Variational tomography with neural networks



Neuromorphic hardware quantum states



Supervised learning for entropy estimation



Thank you!

Funding:

DFG
STRUCTURES CLUSTER OF EXCELLENCE
Baden-Württemberg Stiftung
WIR STIFTEN ZUKUNFT
isoQuant SFB1225
European Commission
NOA

Backup slides

Work on using ML for many-body physics

Quantum (and classical) dynamics

- Schrödinger equation — Czischek et al., PRB 98, 024311 (2018)
- Lindblad Master equation — Reh et al., PRL 127, 230501 (2021)
- Fokker Planck equation — Reh, Gärttner, MLST 3, 04LT02 (2022)
- Ground states — Reh, Schmitt, Gärttner, PRB 107, 195115 (2023)

Quantum state tomography

- Two-qubits, experimental — Neugebauer et al., PRA 102, 042604 (2020)
- **Scaling to many qubits, using CNNs** — **Schmale et al., npj QI 8, 115 (2022)**
- **Learning entropy prediction** — **Rieger et al., PRA 109, 012403 (2024)**

Neuromorphic hardware implementation

- **Learning entangled states** — **Czishek et al., SciPost Phys. 12, 39 (2022)**
- Learning ground states — Klassert et al., iScience 25, 104707 (2022)

NQS tomography

Quantum State Tomography

Quantum state tomography:

Reconstructing ρ from measurements: $2^N \times 2^N$ matrix (N qubits)

Tomographically complete measurement: 4^N possible independent outcomes

Example: Pauli measurements

Single qubit: $|\pm\rangle = [|\uparrow\rangle \pm |\downarrow\rangle]/\sqrt{2}$ $|R/L\rangle = [|\uparrow\rangle \pm i|\downarrow\rangle]/\sqrt{2}$

Measure in **z-basis** **x-basis** **y-basis**

$P_\uparrow = \mathrm{tr}(\rho|\uparrow\rangle\langle\uparrow|)$ 0 $P_+ = \mathrm{tr}(\rho|+\rangle\langle+|)$ 1 $P_R = \mathrm{tr}(\rho|R\rangle\langle R|)$ 2 $P_a = \mathrm{tr}(\rho M_a)$

$P_\downarrow = \mathrm{tr}(\rho|\downarrow\rangle\langle\downarrow|)$ $P_- = \mathrm{tr}(\rho|-\rangle\langle-|)$ $P_L = \mathrm{tr}(\rho|L\rangle\langle L|)$ 3 invertible

N qubits: $M_{\boldsymbol{a}} = M_{a_1} \otimes M_{a_2} \dots \otimes M_{a_N}$

Quantum state tomography schemes

Maximum likelihood estimation: parameterize ρ with set of 4^N parameters $\boldsymbol{t}$

$$\text{Minimize } \sum_{\boldsymbol{a}} \left(P(\boldsymbol{a}) - P_{\boldsymbol{t}}(\boldsymbol{a})\right)^2 \text{ wrt. } \boldsymbol{t}$$

Challenge:

Curse of dimensionality for both **sampling complexity** and **post-processing complexity**

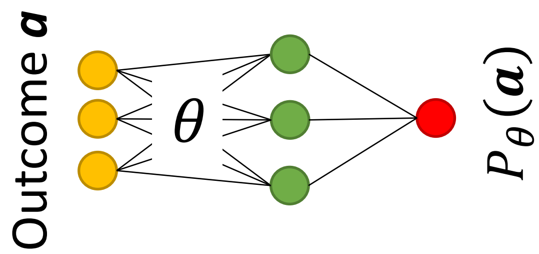
Random measurements, classical shadows

Variational QST: Variational ansatz for ρ or $P(\boldsymbol{a})$ with polynomially many parameters

→ restrict the set of states over which we optimize based on physical constraints

Examples:

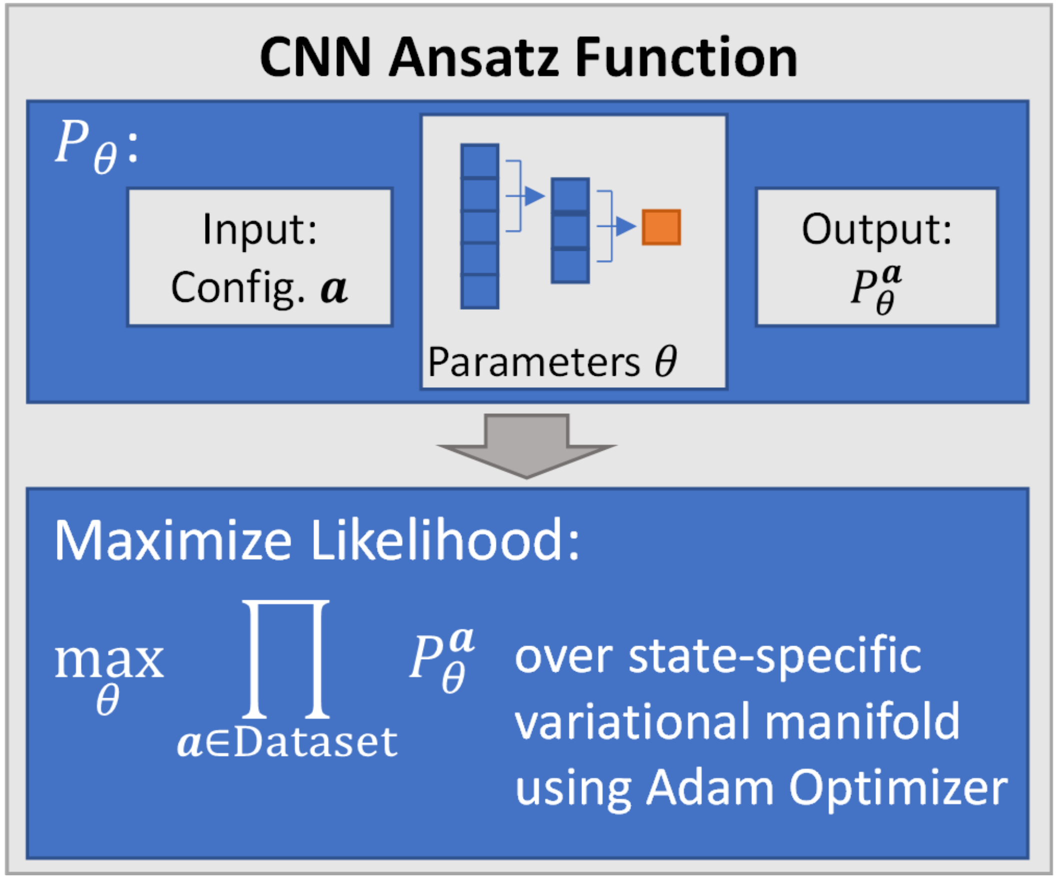
- Matrix product state tomography
- Low rank tomography
- … → any data compression method
- **Neural network quantum state tomography**



Review article:

Torlai and Melko, Annual Review of Condensed Matter Physics 11:325-344 (2020)

Neural network QST: Working principle



Data set:

$$(a_1^{(1)}, a_2^{(1)}, \dots a_N^{(1)})$$
$$(a_1^{(2)}, a_2^{(2)}, \dots a_N^{(2)})$$
$$\dots$$
$$(a_1^{(N_s)}, a_2^{(N_s)}, \dots a_N^{(N_s)})$$

$$a_i^{(j)} \in \{0,1,2,3\}$$

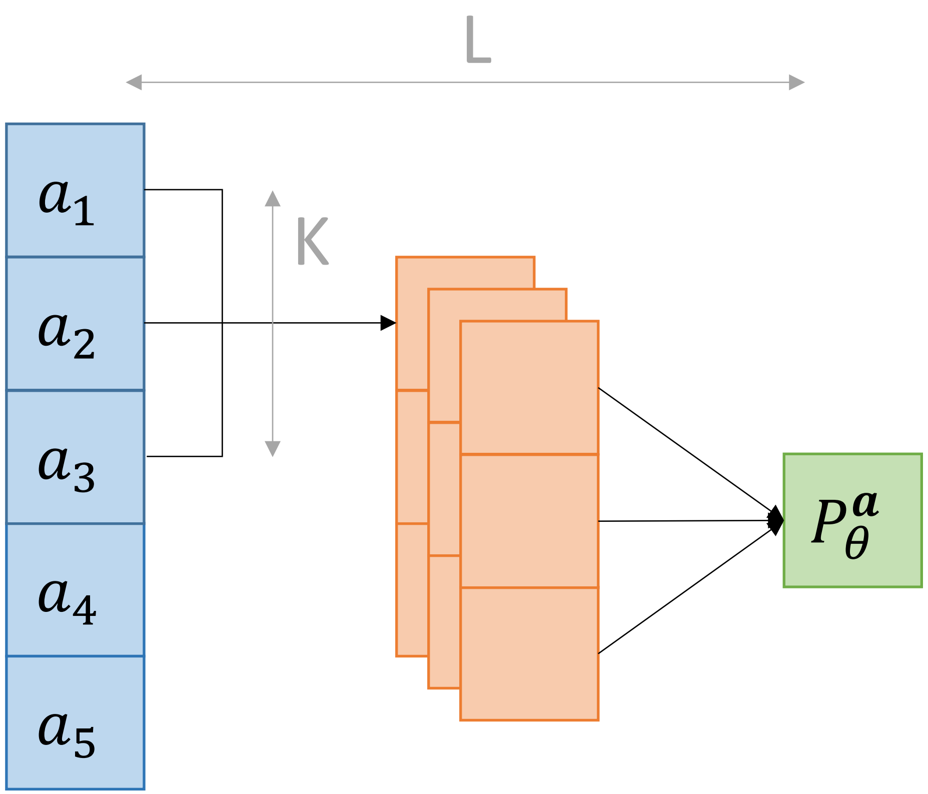
Efficiently evaluate expectation values

$$\langle \mathcal{O} \rangle = \sum_{\boldsymbol{a}} Q_{\boldsymbol{a}}^{\mathcal{O}} P_\theta^{\boldsymbol{a}} = \sum_{\boldsymbol{a}\in S} Q_{\boldsymbol{a}}^{\mathcal{O}}$$

Sample from model ↗ calculate ↖

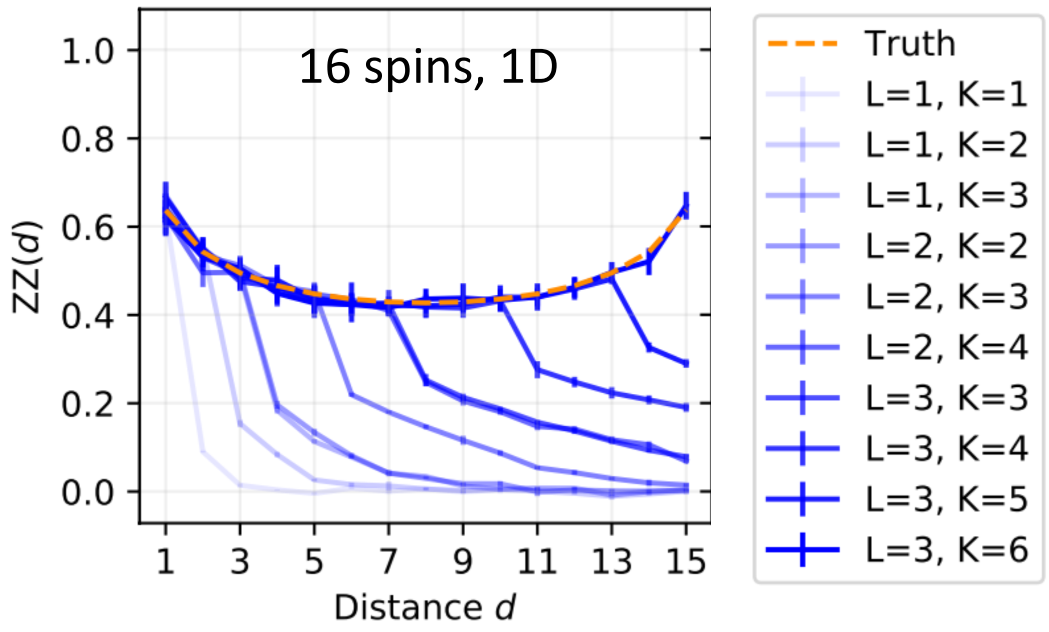
Neural network QST: Expressiveness of CNNs

Convolutional Neural Network (CNN):



$$H = -J\sum_{\langle i,j\rangle} S_z^i S_z^j - B\sum_i S_x^i$$

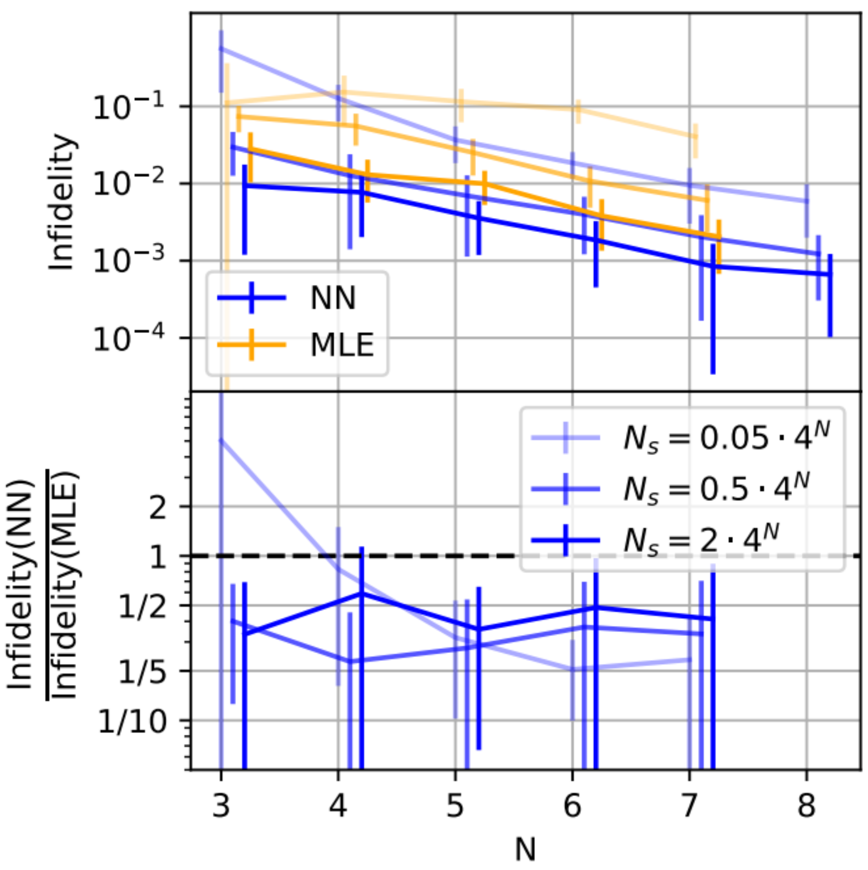
Critical Ground state $J/B = 1$



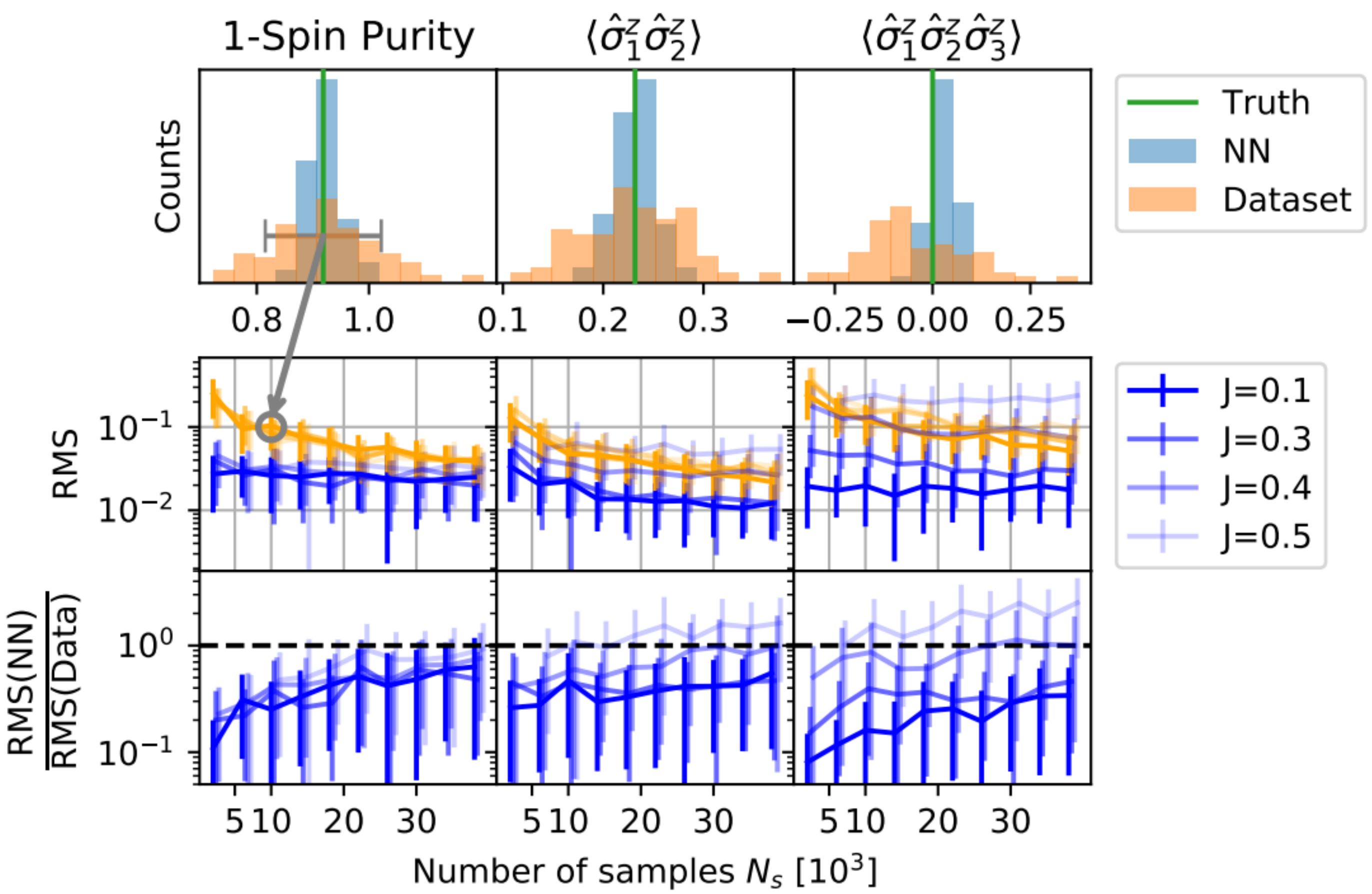
→ Correlation length controllable, polynomial scaling of parameters

Neural network QST: Generalization

Need to build trust to exclude bias!



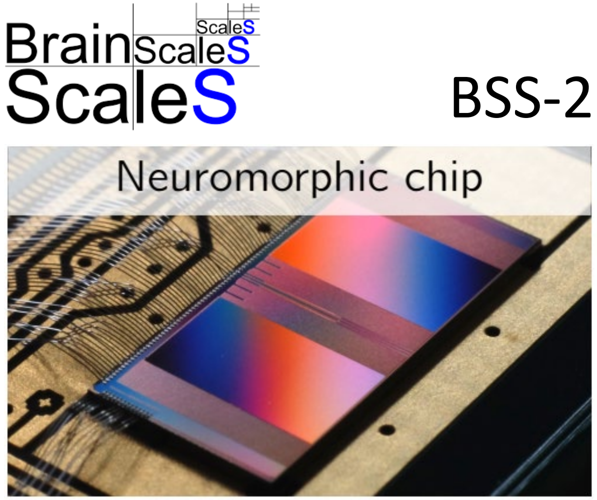
1D system: **Advantage in reconstruction fidelity** w.r.t. MLE

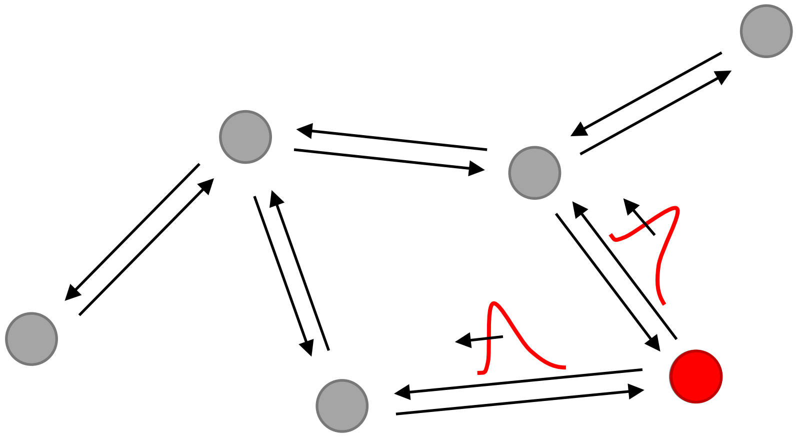


4x4 system: **Variance reduction** for local observables

Neuromorphic quantum states

Idea: Use a *physical* neural network

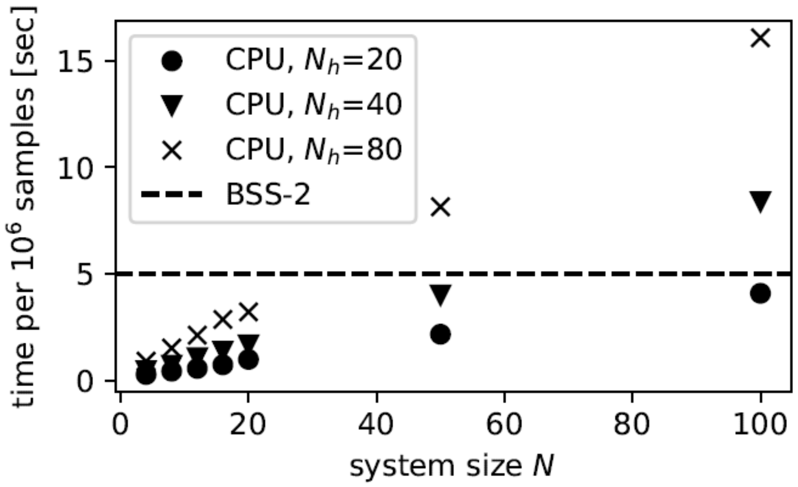




Spiking neural network

→ *Constant* sample rate!

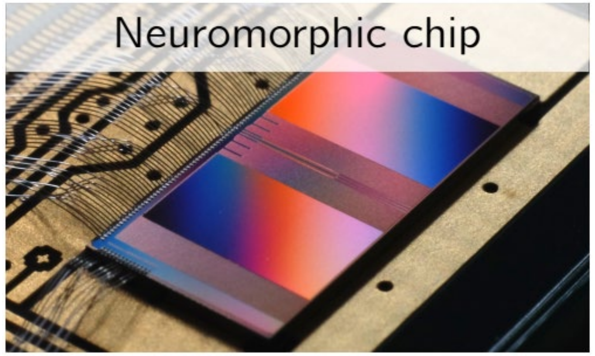
- Electronic circuits
- Analog core
- Emulate spiking dynamics (brain)
- ML applications
- Configurable network parameters

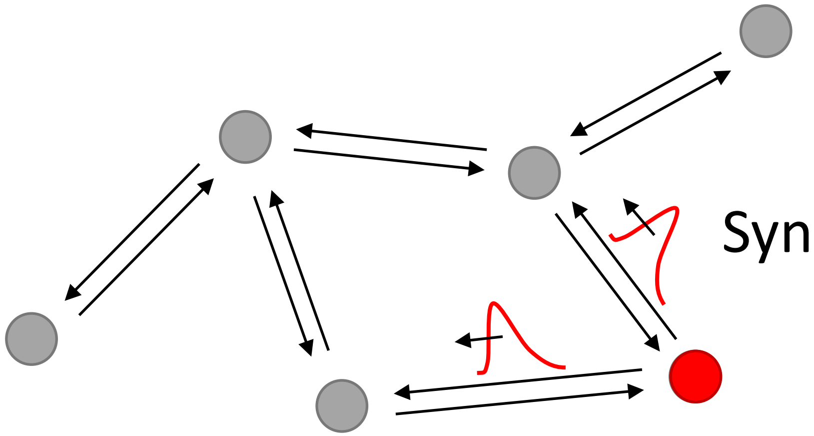


Block Gibbs sampling in a software RBM vs. BSS-2

Details on spiking dynamics

BrainScaleS BSS-2





Neuron ≈ capacitor

Bias b_i ≈ offset potential

Synapse ≈ conductor

Network weights W_{ij} ≈ conductivities

Neuron model: Leaky Integrate and Fire Neurons (LIF)

$$V_{\text{in}}(i) \propto \sum_j I_{\text{out}}(j) W_{i,j} + b_i$$

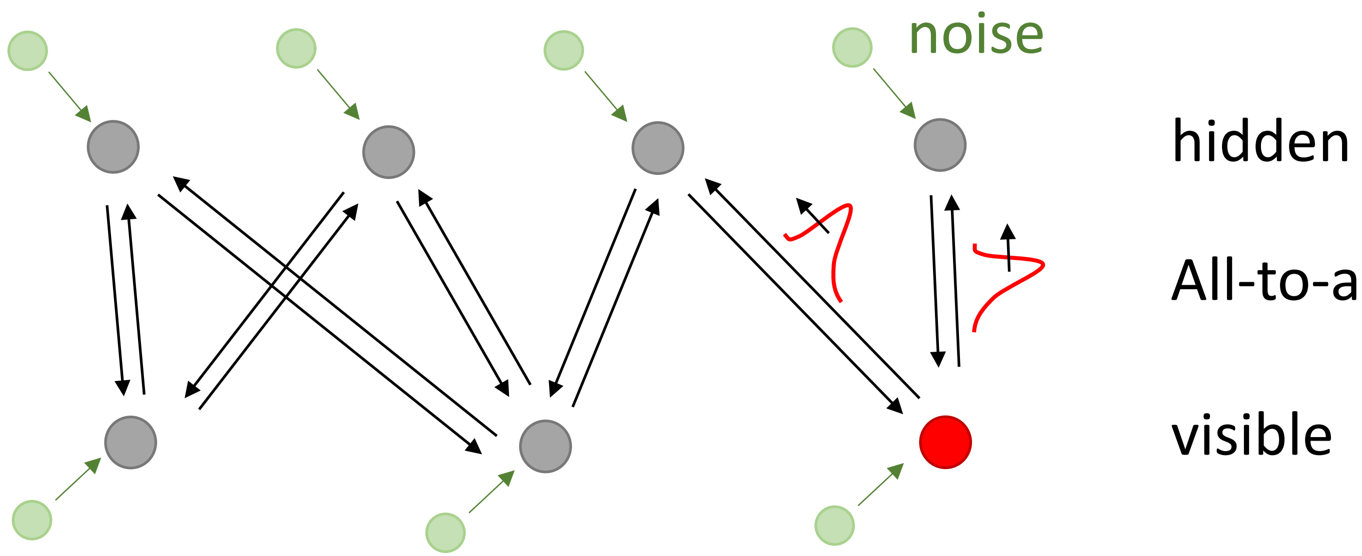
More detailed:

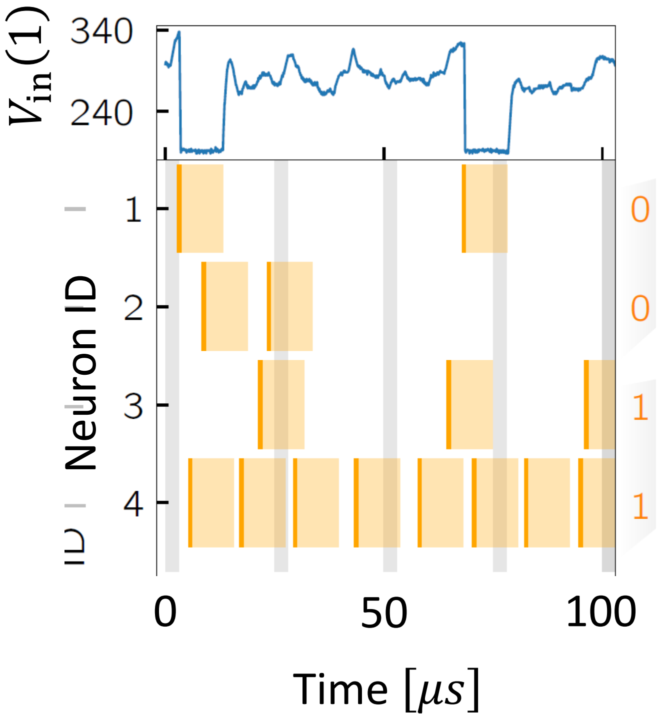
$$C_{\mathrm{m}} \frac{d}{dt} u_k = g_{\mathrm{l}}(E_{\mathrm{l}} - u_k) + I_k$$

(C_{m}: capacitance; u_k: Membrane potential; g_{l}: Leak conductance; E_{l}: Leak potential; I_k: Input current)

$$I_k = \underbrace{I_k^{\mathrm{rec}} + I_k^{\mathrm{noise}}}_{\sum_i g_{ki}\left(E_i^{\mathrm{rev}} - u_k\right)} + I_k^{\mathrm{ext}}$$

Sampling with spiking neurons





Engineer spiking network to sample from a Boltzmann distribution:

$$P_\theta(\mathbf{v},\mathbf{h}) = \exp\left[\sum_{i,j} W_{ij} \mathrm{v}_i \mathrm{h}_j + \sum_i a_i \mathrm{v}_i + \sum_j b_j \mathrm{h}_j\right]$$

Stochasticity through additional noise neurons

M. A. Petrovici, J. Bill, I. Bytschok, J. Schemmel, and K. Meier, PRE 94, 042312 (2016)

Encoding quantum states with probabilities

IC-POVM representation

$P_{\boldsymbol{a}} = \mathrm{tr}(\rho M_{\boldsymbol{a}})$ $\{M_{\boldsymbol{a}}\}$ informationally complete measurement

N spins:

$$M_{\boldsymbol{a}} = M_{a_1} \otimes M_{a_2} \dots \otimes M_{a_N} \qquad \{M_a\}_{a=0\dots3}$$

Parameterize $P_{\boldsymbol{a}}$ as

$$P_\theta(\boldsymbol{a}) = p_\theta(\mathbf{v}) = \sum_{\mathbf{h}} P_\theta(\mathbf{v}, \mathbf{h})$$

Each physical spin represented by **two** physical visible neurons.

Positive wave functions

"stoquastic" Hamiltonians

→ Ground state has real non-negative wave function coefficients

→ $c_{\mathbf{v}} = \sqrt{p_{\mathbf{v}}}$ where $p_{\mathbf{v}}$ is a probability distribution!

$$|\psi_\theta\rangle = \sum_{\mathrm{v}_1 \dots \mathrm{v}_N} \sqrt{p_\theta(\mathbf{v})} |\mathrm{v}_1 \dots \mathrm{v}_N\rangle$$

$$p_\theta(\mathbf{v}) = \sum_{\mathbf{h}} P_\theta(\mathbf{v}, \mathbf{h})$$

Learning schemes

Distribution learning

Minimize w.r.t. θ:

$$D_{\mathrm{KL}}(P\|p) = \sum_{\mathbf{v}} P(\mathbf{v}) \ln\left[\frac{P(\mathbf{v})}{p_\theta(\mathbf{v})}\right]$$

target → P; network encoded → p

$$\frac{\partial D_{\mathrm{KL}}(P\|p)}{\partial W_{i,j}} = \left\langle \left[1 - \frac{P(\mathbf{v})}{p_\theta(\mathbf{v})}\right] \mathrm{v}_i \mathrm{h}_j \right\rangle_{p_\theta(\mathbf{v},\mathbf{h})}$$

From network samples → $p_\theta(\mathbf{v})$

→ Wake-sleep learning

→ Use Adam optimizer

Ground state learning

Minimize w.r.t. θ:

$$E[\theta] = \langle \psi(\theta)|H|\psi(\theta)\rangle = \sum_{\mathbf{v},\mathbf{v}'} \sqrt{p_\theta(\mathbf{v}) p_\theta(\mathbf{v}')}\, H_{\mathbf{v}\mathbf{v}'}$$

Gradients (weights):

$$\frac{\partial E[\theta]}{\partial W_{i,j}} = \left\langle \left[\underbrace{\sum_{\mathbf{v}'} H_{\mathbf{v}\mathbf{v}'} \sqrt{\frac{p_\theta(\mathbf{v}')}{p_\theta(\mathbf{v})}}}_{\text{Local energy}} - E[\theta] \right] \mathrm{v}_i \mathrm{h}_j \right\rangle_{p_\theta(\mathbf{v},\mathbf{h})}$$

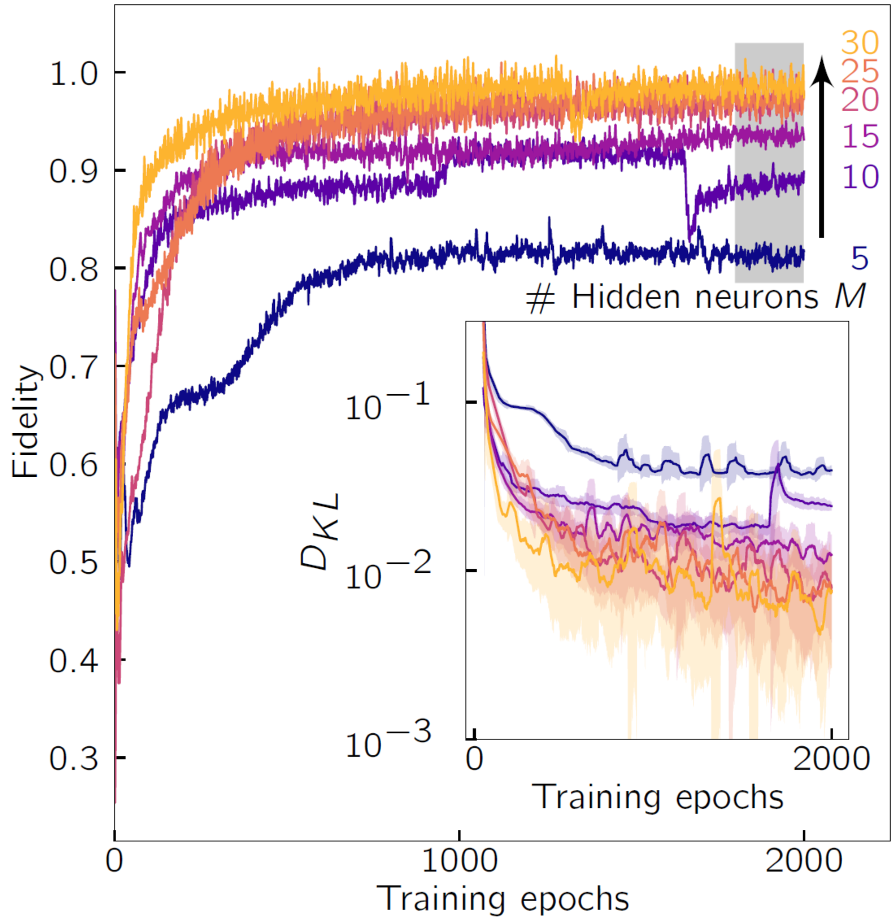
From network samples → $p_\theta(\mathbf{v})$

→ Use Adam optimizer, learning rate decay

POVM distributions

Czischek et al. (2022), SciPost Phys. 12, 039

Training performance



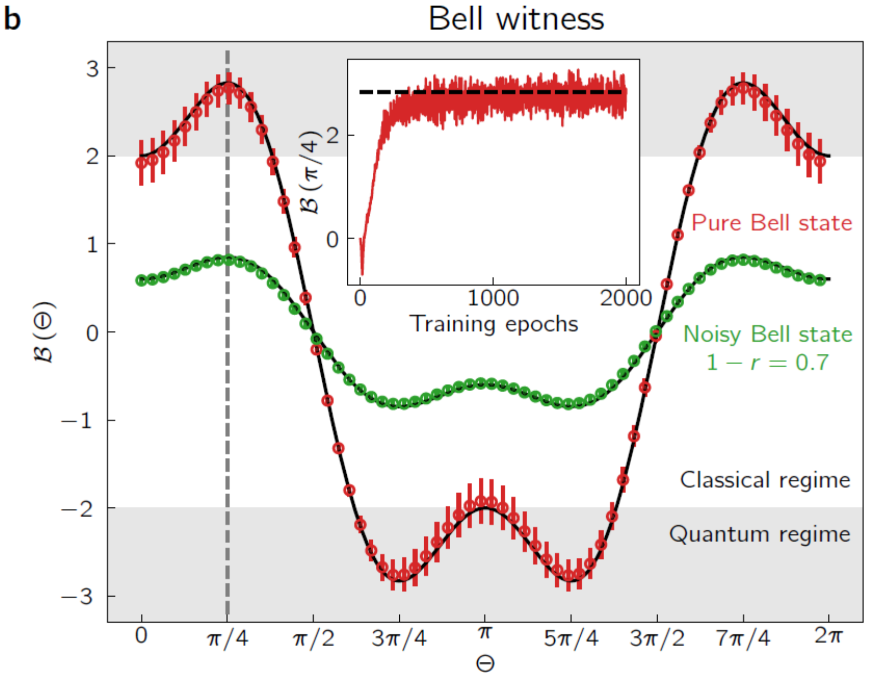
Learning (noisy) Bell states

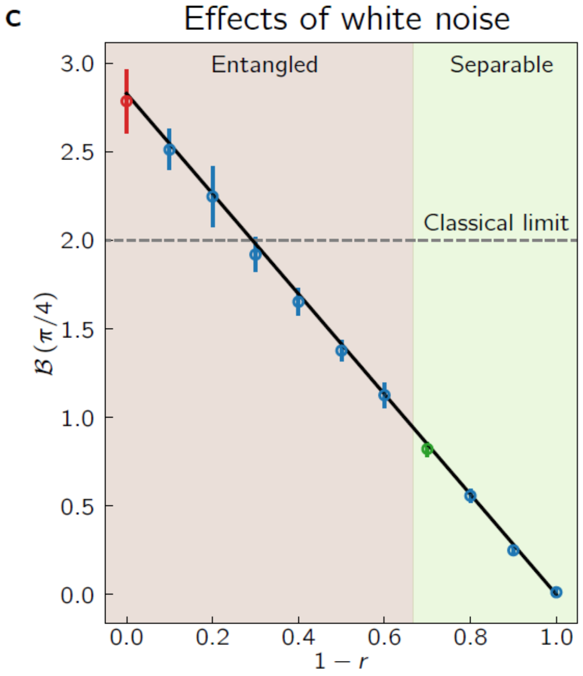
$$\rho_B = |\psi_+\rangle\langle\psi_+|$$

$$|\psi_+\rangle = \frac{1}{\sqrt{2}}[|\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle]$$

Werner state:

$$\rho_W = r\rho_B + \frac{1-r}{4}\mathbb{I}$$





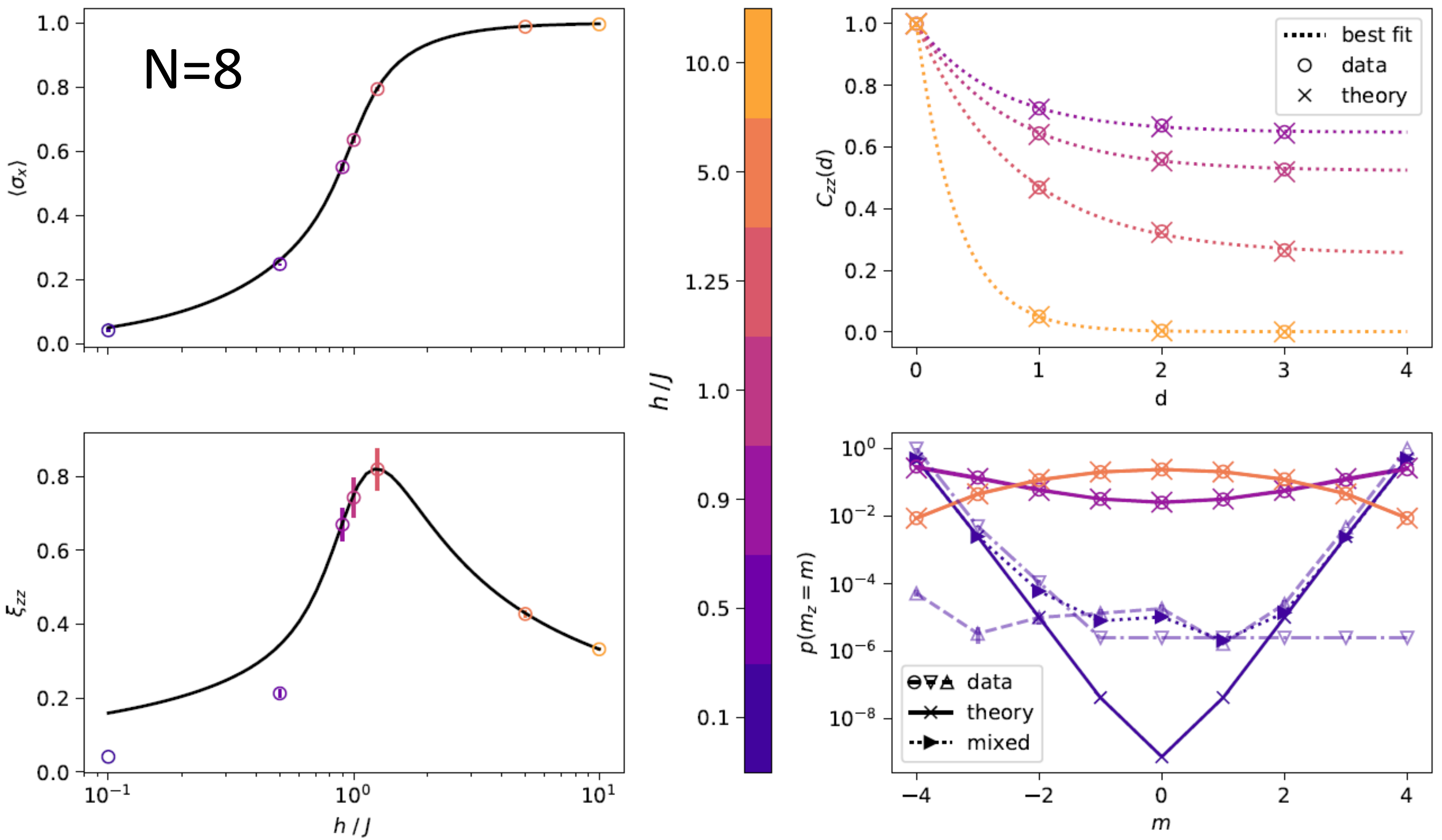
125000 samples in training and evaluation

20 hidden neurons

TFIM ground states

$$\hat{H} = -J\sum_i \hat{\sigma}_i^z \hat{\sigma}_{i+1}^z - h\sum_i \hat{\sigma}_i^x$$

Klassert, Baumbach, Petrovici, Gärttner
arXiv:2109.15169



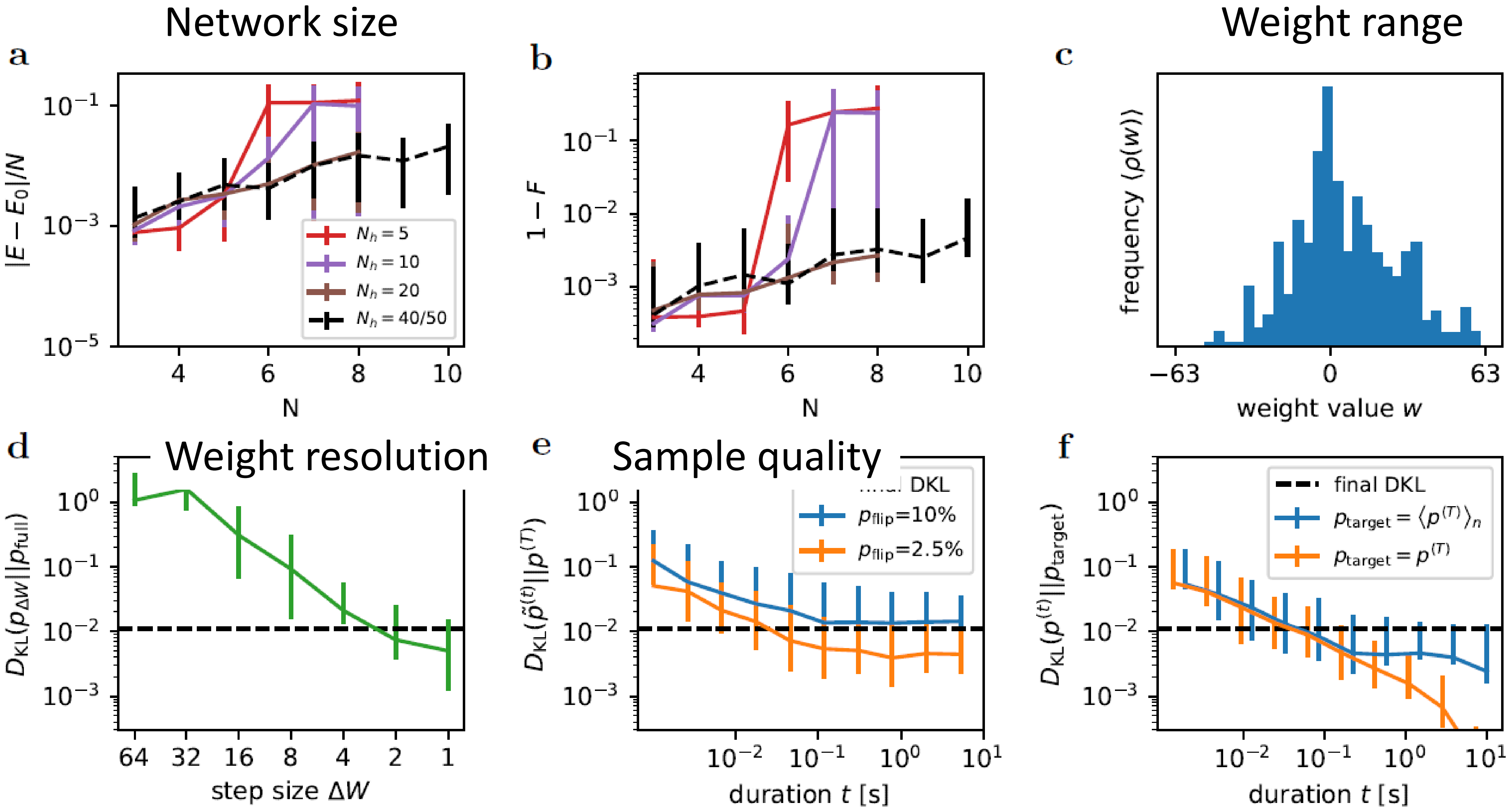
Quantum phase transition at h=J

Symmetry breaking for h>J, fix by averaging different initializations

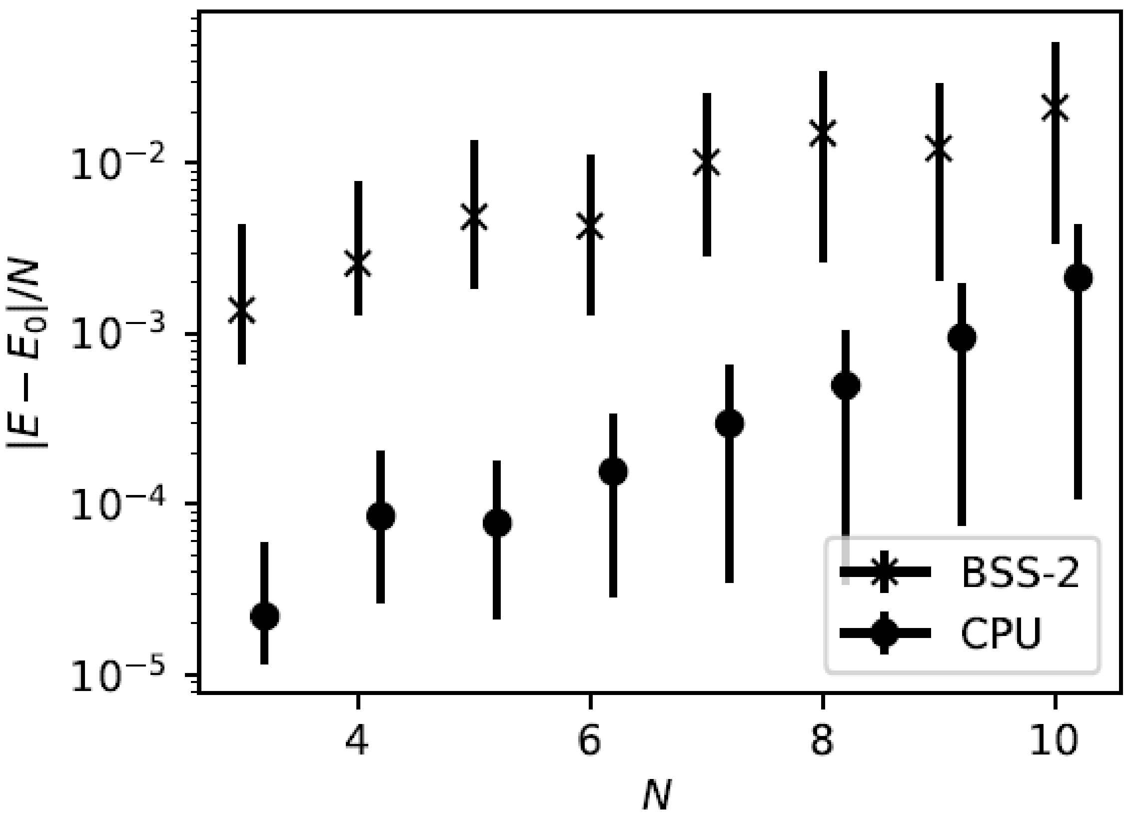
300k samples per epoch

40 hidden neurons

Scaling behavior and limitations



Scaling behavior and limitations



$$\frac{\partial E[\boldsymbol{\theta}]}{\partial W_{i,j}} = \left\langle \left[\sum_{\mathbf{v}'} H_{\mathbf{v}\mathbf{v}'} \sqrt{\frac{p_\theta(\mathbf{v}')}{p_\theta(\mathbf{v})}} - E[\boldsymbol{\theta}] \right] \mathrm{v}_i \mathrm{h}_j \right\rangle_{p_\theta(\mathbf{v},\mathbf{h})}$$

From network samples

No exact likelihoods available

→ Need to densely sample the network distribution

→ Algorithmic and hardware limitations!

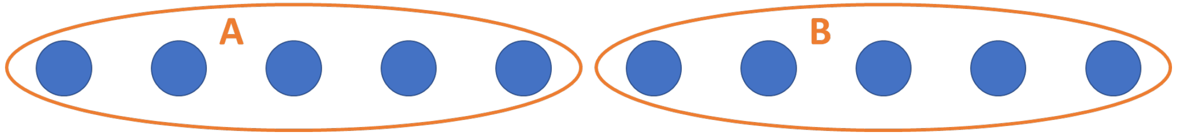
Ideas for overcoming limitations

$$E[\boldsymbol{\theta}] = \sum_{\mathbf{v},\mathbf{v}'} \sqrt{p_\theta(\mathbf{v})p_\theta(\mathbf{v}')}\, H_{\mathbf{v}\mathbf{v}'}$$

- Choose a problem where gradients can be efficiently calculated from samples (e.g. steady states of Lindblad master equation?)
- Use analogy to quantum variational algorithms
- Contrastive divergence learning (problem: need “clamp” neuron states)
- Efficient evaluation of local energies by suppression of terms with large Hamming distance (duplicate network) (Giuseppe idea)
- Simplified effective description of hardware dynamics → analytical formula for likelihoods
- Other types of neuromorphic devices (tradeoff between speed and controllability)
- Hardware improvements

Supervised learning

Supervised learning



N spin-1/2 particles, transverse field Ising model

Input data structure: (N_S, N, d)

Experimental shots → N_S; particles → N; Measurement outcomes, Here: Pauli-4 → d

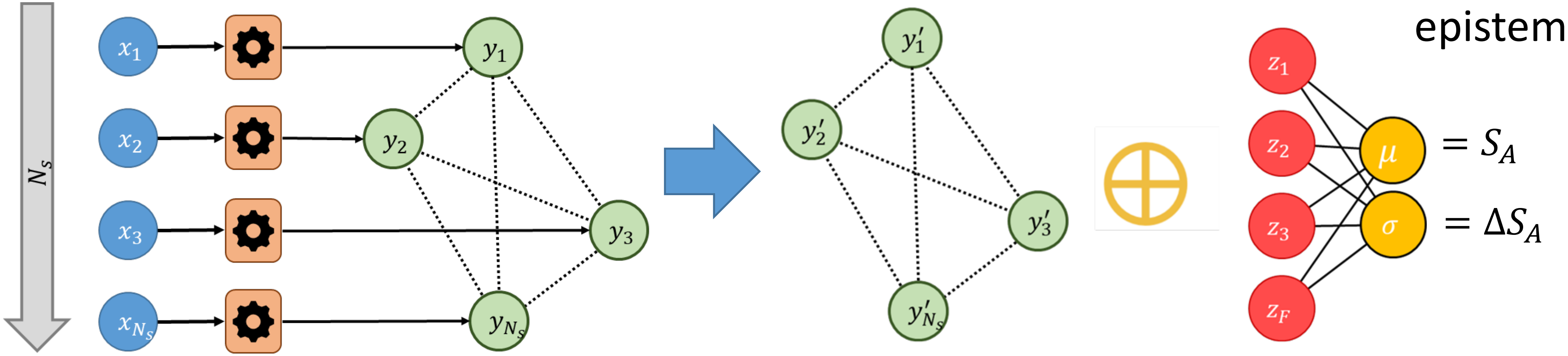
Training data: many such examples with labels S_A

Loss Function:

$$L = \frac{(\mu - \mu^*)^2}{2\sigma^2} + \frac{1}{2}\log(\sigma^2)$$

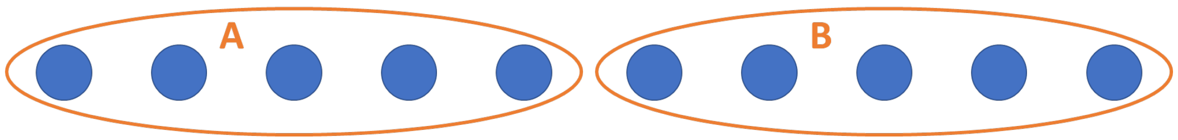
Network architecture: Graph neural net → Permutation invariance

Aleatoric vs. epistemic error



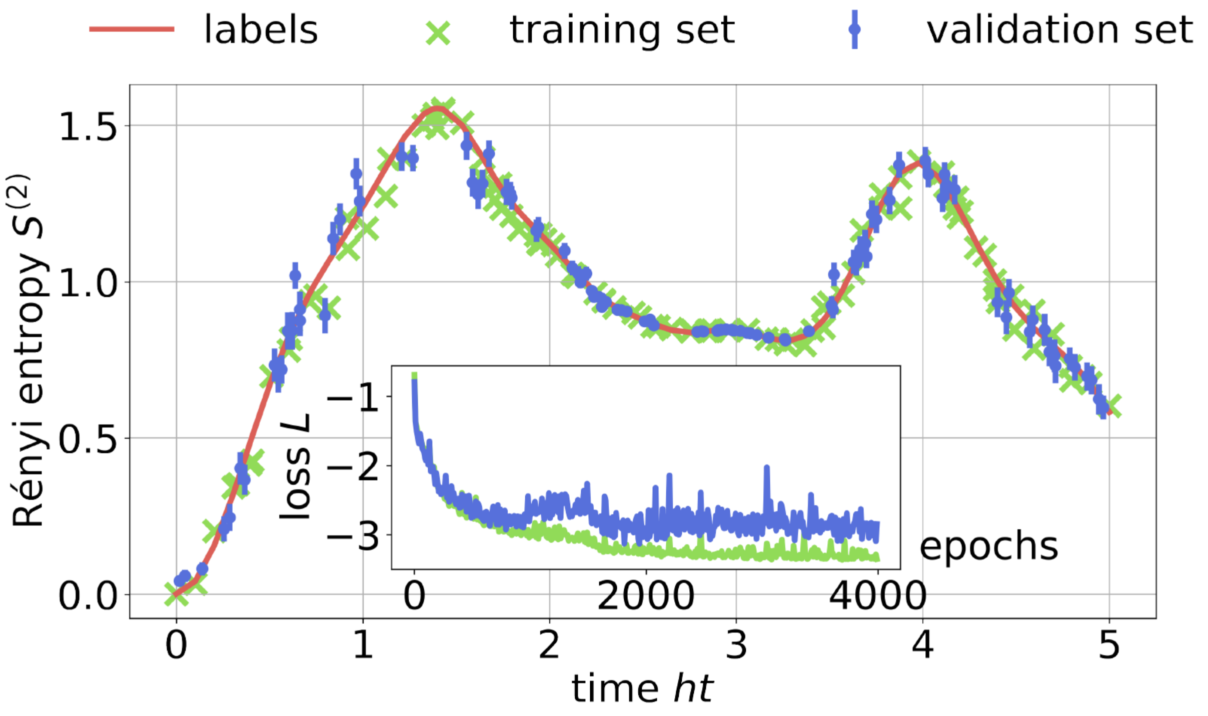
Encoding Layer **Graph Attention Mechanism** **Dense Output**

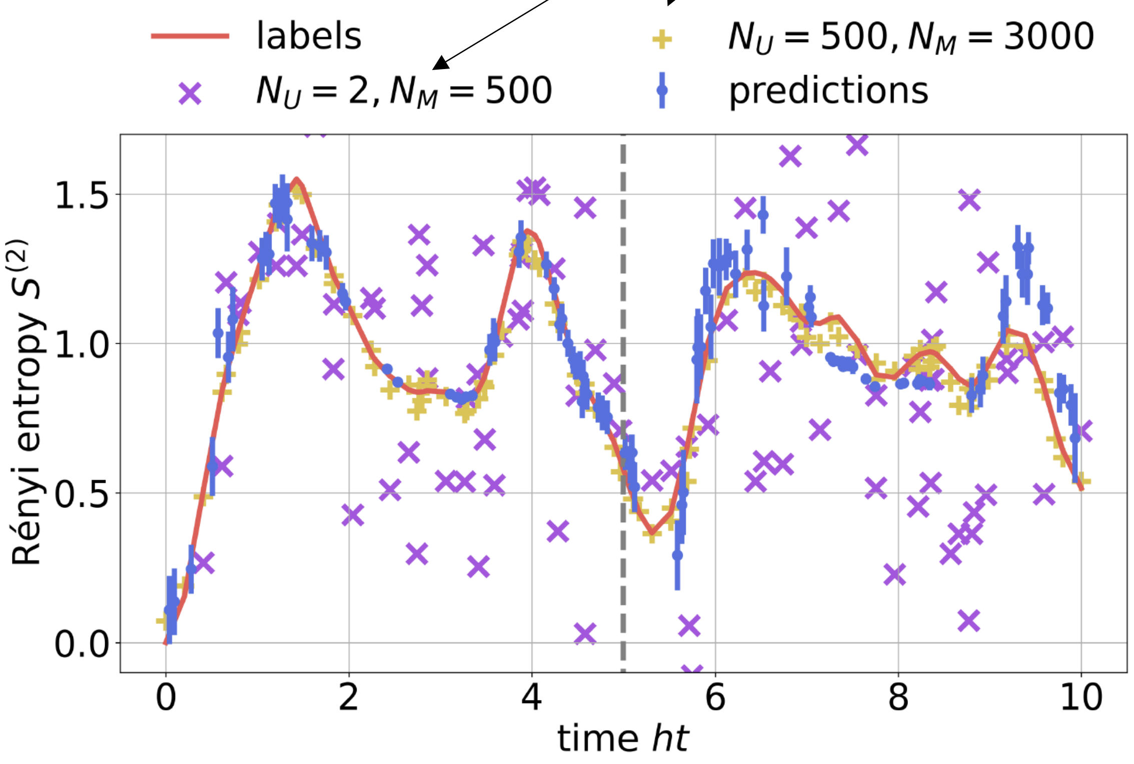
Unitary evolution



TFIM quench, $N = 10$, initial state x-polarized, $h/J = 1$, $N_S = 1000$

Random measurements



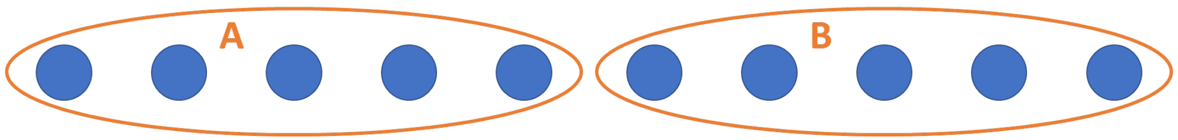


→ Interpolation works well

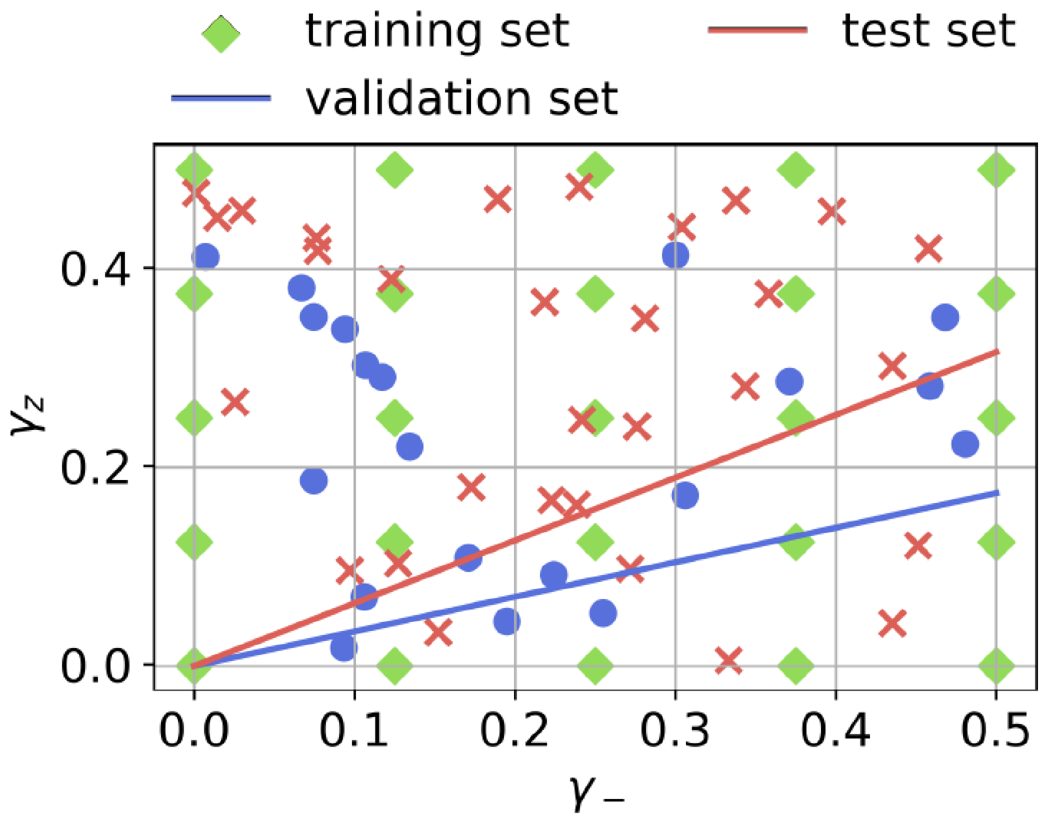
→ Sample efficient

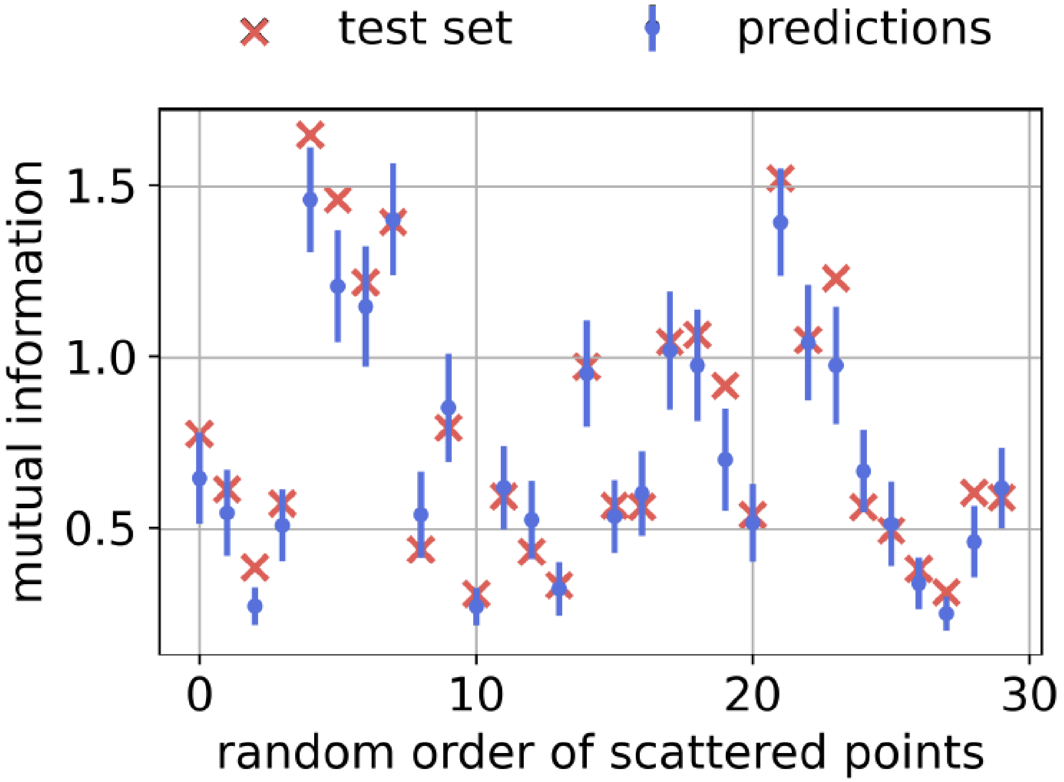
→ Extrapolation fails

Dissipative evolution



Decay γ_- Dephasing γ_z Quantum mutual information $I(A{:}B) = S_A + S_B - S_{AB}$





$N = 8$
$t = 0.75$