From theory to real implementation in the NISQ era

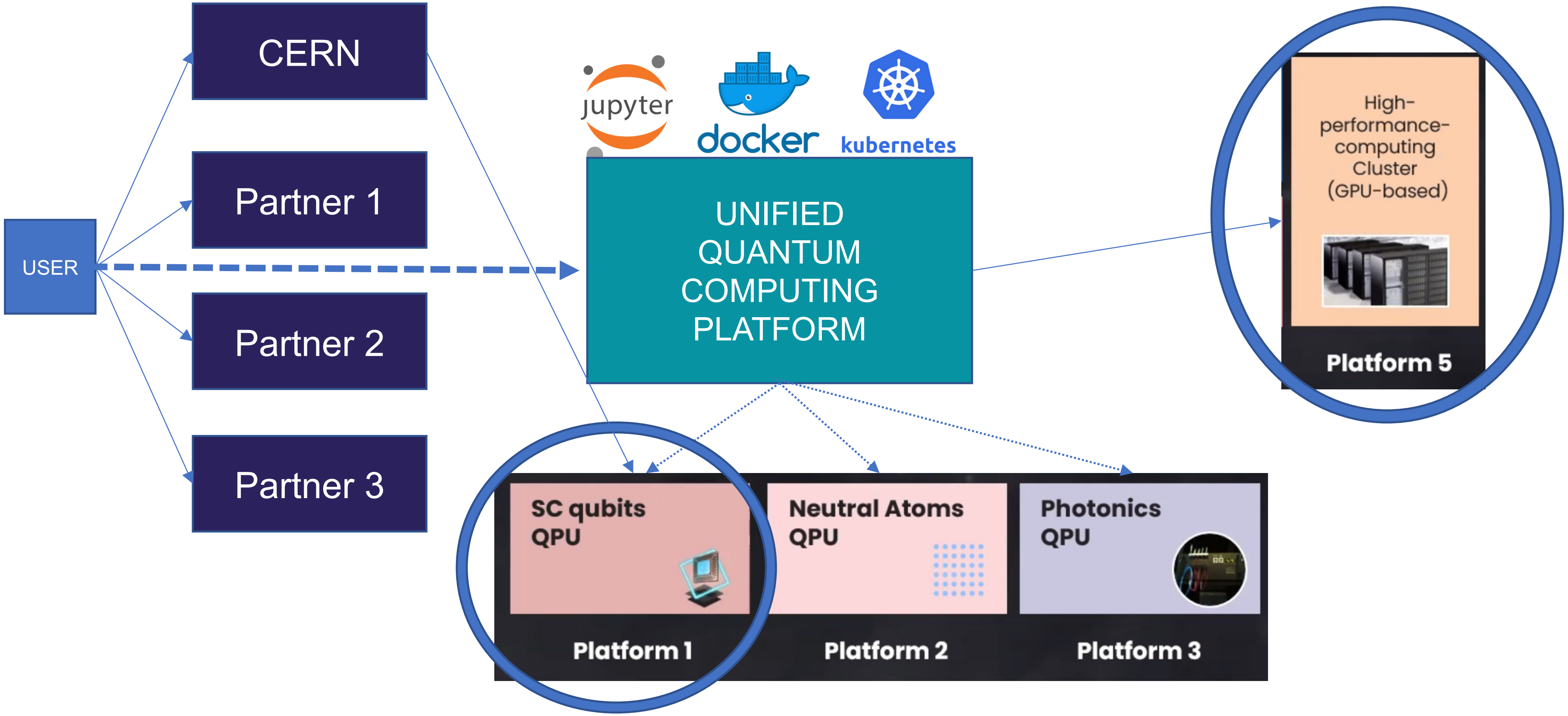
CERN | QUANTUM TECHNOLOGY INITIATIVE

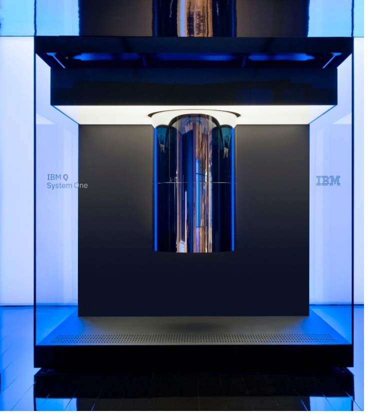
Michele Grossi, PhD

CERN QTI
Quantum Computing Sen. Fellow

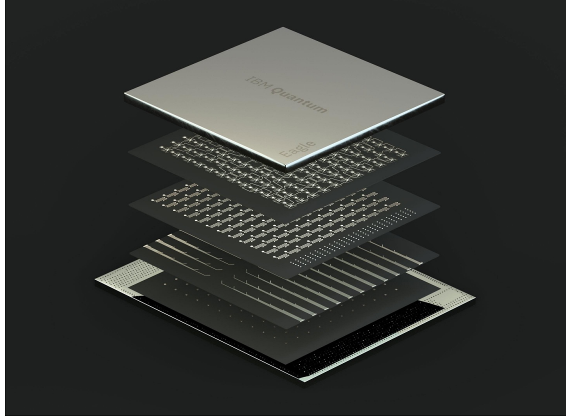
- **Quantum Computing and Simulation**
- **Some Examples**

Quantum Computing Platform





QUANTUM TECHNOLOGIES



Near-term

Noisy Intermediate-Scale Quantum (NISQ) devices

Available now, up to 433 qubits (IBM)

Smaller devices, tipically noisier

Suffer from gate and other types of errors

Current algorithms are desegned to deal with noise and errors: low-depth circuits

Fault-tolerant

Available (maybe) in the future

Many qubits, enabling complex computation

Error-free

Used for theoretical, high-depth algorithms

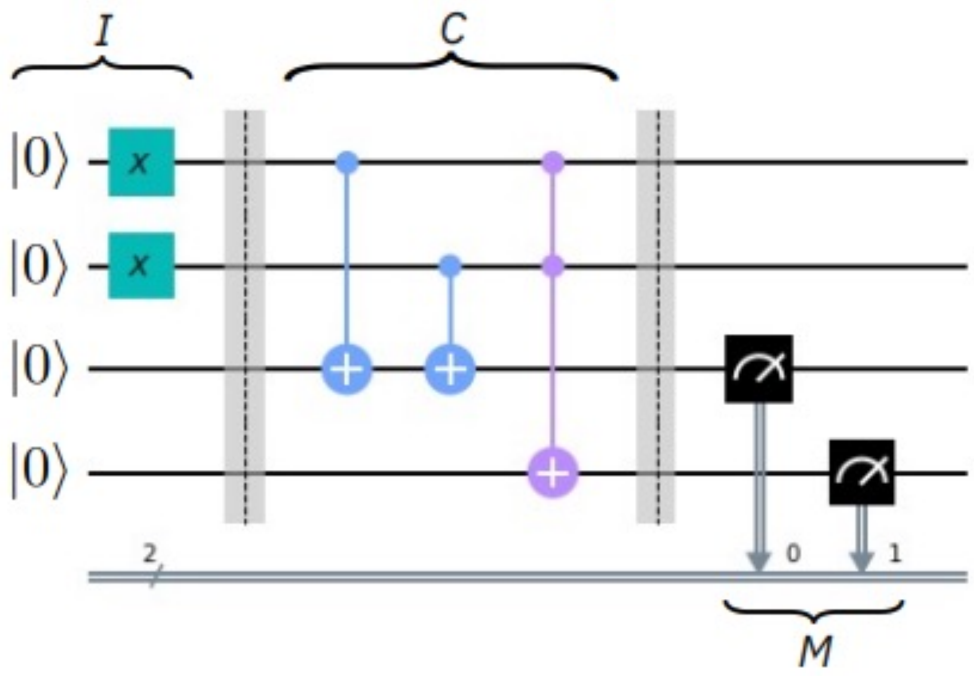
Challenges of Quantum Programming

- Qubits are analog
- Quantum programs result in probabilistic outputs
- You can’t read the entire exact state of a quantum program
- Each device (and qubit!) has characteristics a programmer has to be aware of, such as noise and connectivity
- Qubits have a short coherence time (or lifetime)

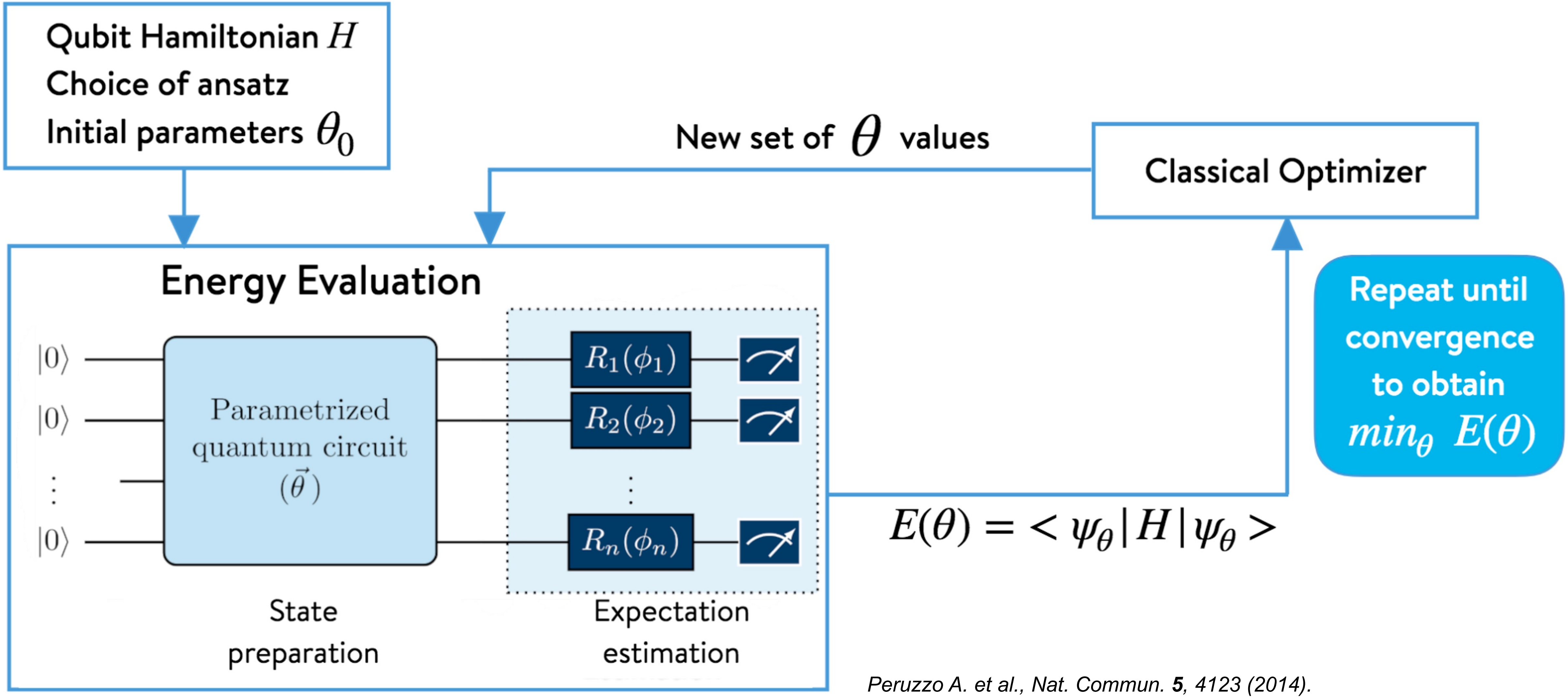
Simulating a Quantum Computer

Why and when to use simulator

- Prototype/understand quantum algorithms before running on a real device
- Simulation of large quantum systems is, in the general case, an exponentially hard computational task for a classical computer, *memory requirements double* for every qubit added
- *This in fact one of the main motivations for building a quantum computer in the first place!*
- For now, it's generally "cheaper" and "easier" to run an experiment, at least a small-scale one, on a simulator as compared to hardware
- In simulation we can understand noise properties of real devices and how noise affects performance (of e.g. algorithms)



Going Hybrid: Variational Quantum Eigensolver (VQE)



Peruzzo A. et al., Nat. Commun. **5**, *4123 (2014).*

- Quantum Computing and Simulation
- **Some Examples**

Finite-size criticality in the Lipkin-Meshkov-Glick (LMG) model

$$H = -\frac{1}{N}\sum_{i<j}^{N} \sigma_x^i \sigma_x^j + \gamma \sigma_y^i \sigma_y^j - B \sum_{i=1}^{N} \sigma_z^i.$$

System of N spins with **anisotropic interaction** in the x-y plane, in a external **transverse magnetic** field.

- Study the finite size criticality (exponential number of degree of freedom).
- Interpolation between fully connected Ising model and Lipkin-Dicke model.

Quantum phase transitions ($N \to \infty$): $E_{gs} = E_{1st}$
Non-analycity in the ground state energy (crossing)

Grossi, Kiss et al, Finite-size criticality in fully connected spin models on superconducting quantum hardware - PhysRevE.107.024113

VQE for excited states

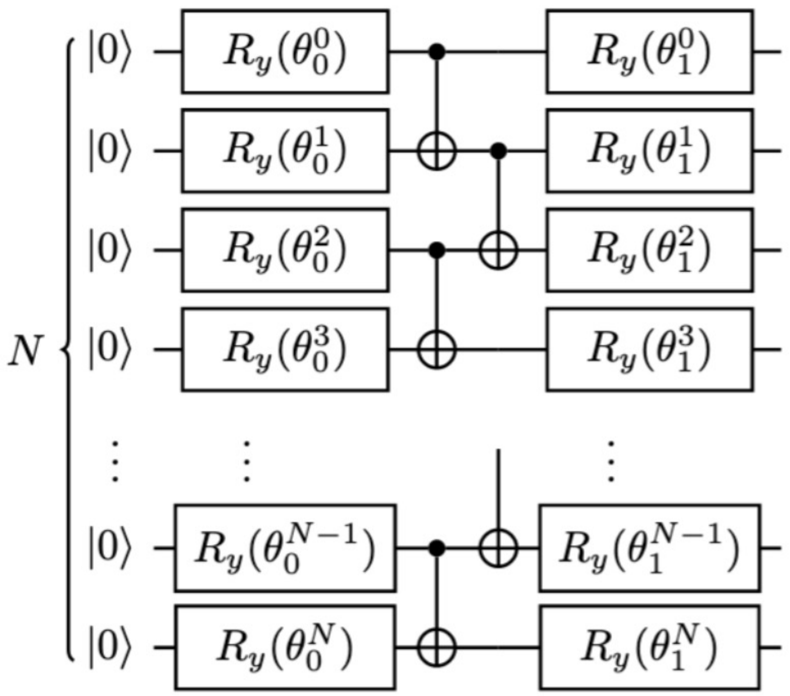
Higgot et al, Quantum **3**, 156 (2019)

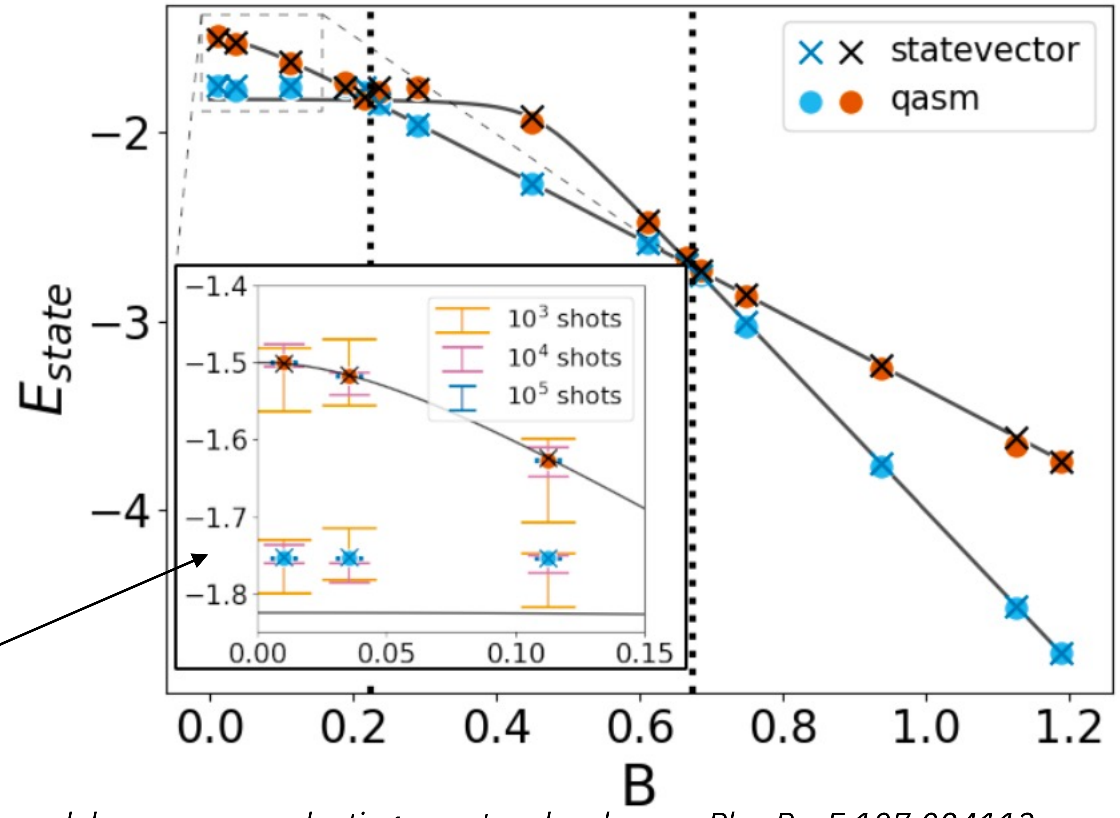
Iterative eigenstates preparation

$$H' = H + \beta_0 \left|\psi_{GS}\right\rangle \left\langle\psi_{GS}\right| .$$

Energy of the gs ans 1st es for N=5, $\gamma = 0.81$.

Hardware efficient ansatz





Shot noise effect

Grossi, Kiss et al, Finite-size criticality in fully connected spin models on superconducting quantum hardware - PhysRevE.107.024113

Runs on real hardware (IBMQ)

Blue: raw (starting from classical optimized parameters with fine tuning on noise).

Red: calibrated with inverse of the error matrices and zero noise extrapolation with exponential fit.

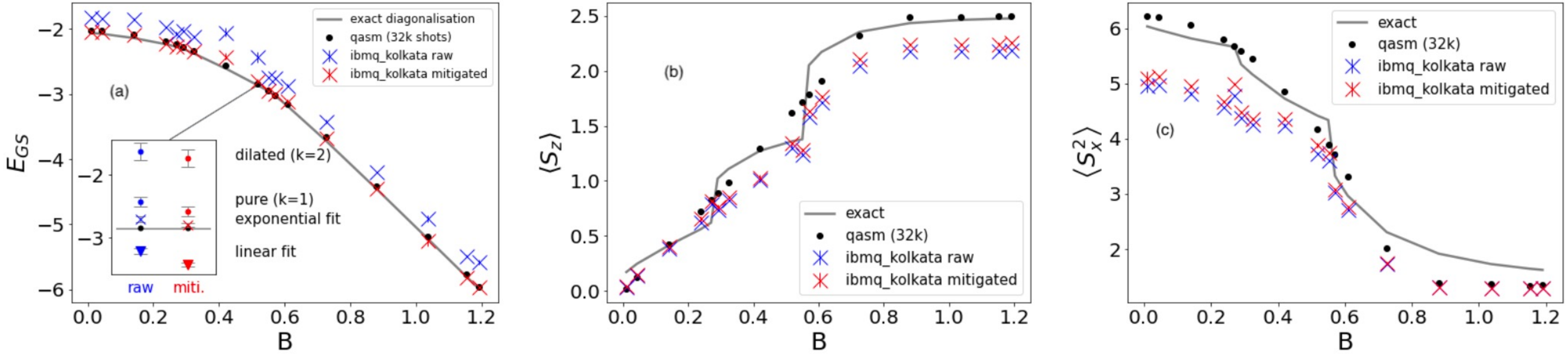
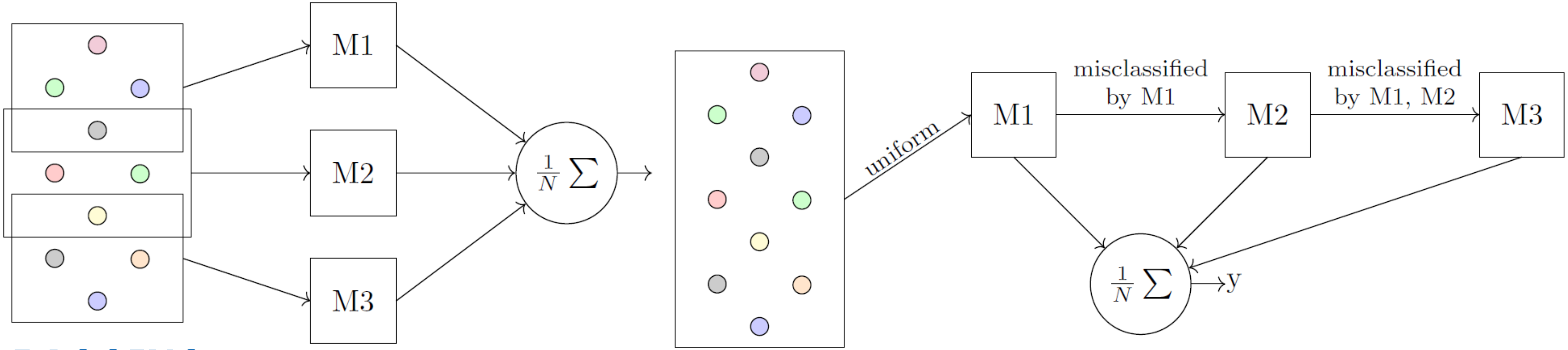


Figure 7. Ground state energy (GS) (a), and magnetization $\langle S_x^2 \rangle$ (b), and $\langle S_z \rangle$ (c), for $N = 5$ spins at $\gamma = 0.49$ values. Points are obtained on the superconducting device ibmq_kolkata with (red) and without (blue) error mitigation compared to noiseless simulation (black) and exact values (line). The inset shows the extrapolation to the zero noise regime, both with an exponential and linear fit.

Grossi, Kiss et al, Finite-size criticality in fully connected spin models on superconducting quantum hardware - PhysRevE.107.024113

Ensemble Techniques

- Evaluate ensemble schemes (**BAGGING** and **BOOSTING**) into *quantum neural networks (QNN)*
- *Layer-wise analysis of quantum neural network performance in the ensemble setting*
- Investigate the potential advantages of bagging techniques in mitigating the effects of noise



BAGGING

- any instance of QNN is trained on a bootstrapped subset of the data

*- analogy with **quantum circuit splitting** / divide-and-conquer techniques (e.g. Lowe et at. 22, Yoshikawa at al. 22))*

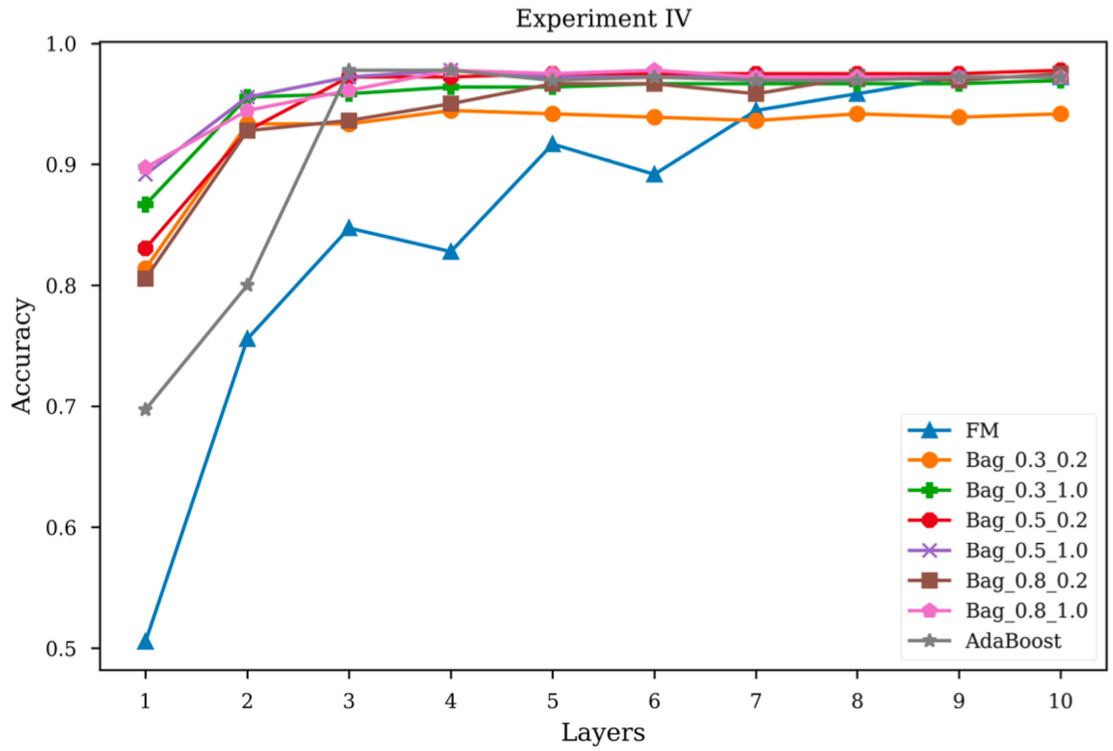
BOOSTING

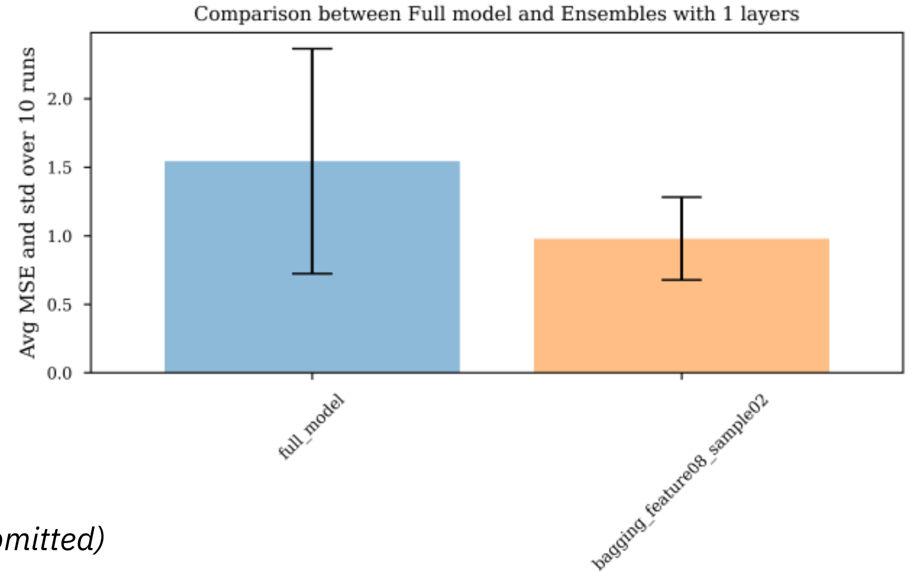
training a sequence of predictors

Incudini, Grossi , Vallecorsa et al – Resource Saving via Ensemble Techniques for Quantum Neural Networks (Submitted)

Ensemble: Settings & Results

- Tested the **LINEAR DATASET** on the IBM Lagos QPU (7 qubits)
- Ensemble technique: **BAGGING**
  80% features 20% samples
- **-1/3** of the MSE on average

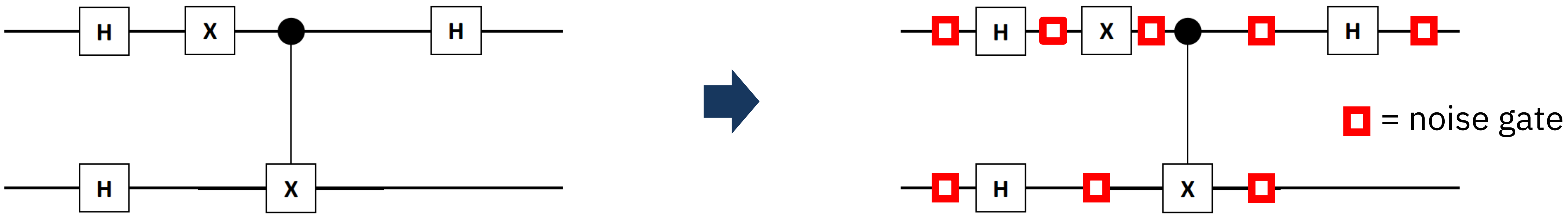




Incudini, Grossi, Vallecorsa et al – Resource Saving via Ensemble Techniques for Quantum Neural Networks (Submitted)

Noise Models

Noises (like gates) formally act instantly: Lindblad → $\text{Kraus}_\rho \;\rightarrow\; \sum_i K_i \rho K_i^\dagger \qquad \sum_i K_i^\dagger K_i = 1$



Standard noise simulation (e.g. in Qiskit)

- Gates and noise are formally **decoupled** (a sort of Trotterizzation), because time scales are small (IBM: gate time ~ 10^{-8} s, decoherence times ~ 10^{-4} s) ?
- Use the **quantum-jump-like approach** to replace the density matrix with (stochastic) state vector → stochastic (unravelling) dynamics √

Di Bartolomeo, Vischi, Wixinger, Grossi et al., A novel approach to noisy gates for simulating quantum computers - https://arxiv.org/abs/2301.04173

Comparison of the algorithms

SEE QTI lectures:
https://indico.cern.ch/event/1247873/

Kraus map for density matrix $\mathcal{E}(\rho) = \sum_i \mathrm{K}_i \rho \mathrm{K}_i^\dagger, \quad \sum_i \mathrm{K}_i^\dagger \mathrm{K}_i = \mathbb{1}.$

Unravelling for the state vector $|\psi'\rangle = \frac{1}{\sqrt{p_j}} \mathrm{K}_j |\psi\rangle, \quad p_j = |\langle\psi| \mathrm{K}_j^\dagger \mathrm{K}_j |\psi\rangle|^2.$

State vector simulations with standard method

Vischi, Wixinger, Grossi et al., A novel approach to noisy gates for simulating quantum computers - https://arxiv.org/abs/2301.04173

Algorithm 1 QISKIT SIMULATION

Input: Initial state $|\psi_0\rangle$, a noiseless circuit $\mathrm{C} = \{\mathrm{U}^{(1)}, ..., \mathrm{U}^{(n_g)}\}$ composed by n_g gates $\mathrm{U}^{(i)}$ and number of samples N_s

for $0 \leq k \leq N_s$ **do**
- **while** $1 \leq i \leq n_g$ **do**
  - compute $|\psi_k\rangle^{(i)} = \mathrm{U}^{(i)} |\psi_k\rangle^{(i-1)}$
  - compute $p_j = |\langle\psi_k|^{(i)} \mathrm{K}_j^\dagger \mathrm{K}_j |\psi_k\rangle^{(i)}|^2$
  - sample K_j operator from $\{p_j\}$
  - update the state to $|\psi_k\rangle^{(i)} = \frac{1}{\sqrt{p_j}} \mathrm{K}_j |\psi_k\rangle^{(i)}$
- **end**
- compute $\rho_k = |\psi_k\rangle^{(n_g)} \langle\psi_k|^{(n_g)}$

end

Output: $\rho_f = \frac{1}{N_s} \sum_{k=1}^{N_s} \rho_k$

Algorithm 2 NOISY GATES SIMULATION

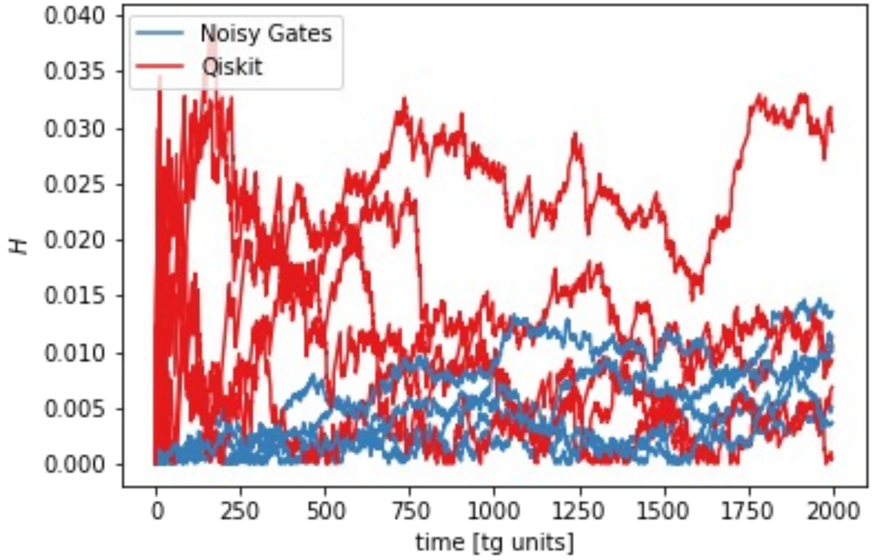
Input: Initial state $|\psi_0\rangle$, a noiseless circuit $\mathrm{C} = \{\mathrm{U}^{(1)}, ..., \mathrm{U}^{(n_g)}\}$ composed by n_g gates $\mathrm{U}^{(i)}$ and number of samples N_s

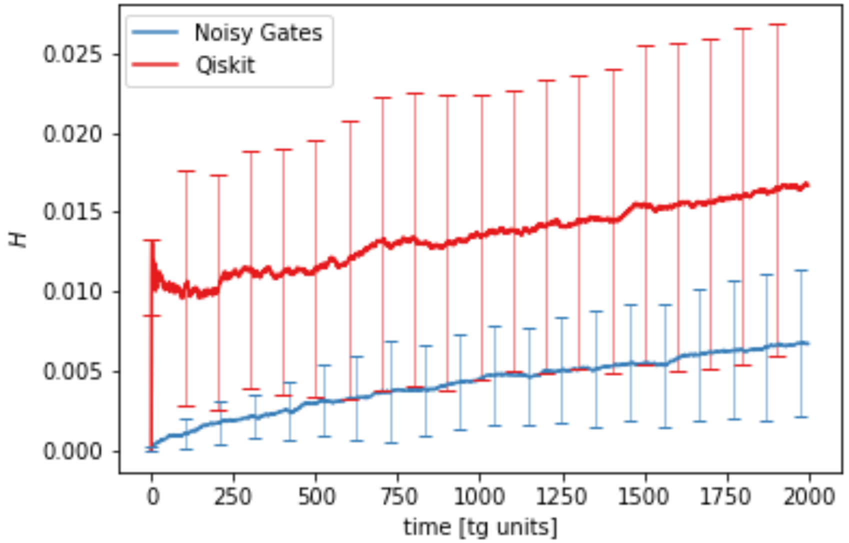
for $0 \leq k \leq N_s$ **do**
- map a noisy circuit $\tilde{\mathrm{C}} = \{\mathrm{N}^{(1)}, ..., \mathrm{N}^{(n_g)}\}$ on C
- sample stochastic processes ξ inside noisy gates $\mathrm{N}^{(i)}$
- compute $|\psi_k\rangle = \mathrm{N}^{(n_g)} \ldots \mathrm{N}^{(1)} |\psi_0\rangle$
- compute $\rho_k = |\psi_k\rangle \langle\psi_k|$

end

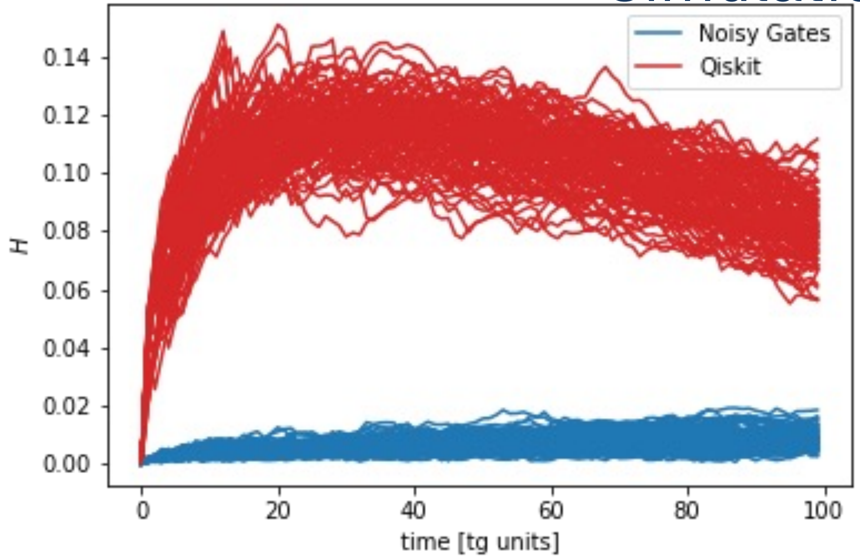
Output: $\rho_f = \frac{1}{N_s} \sum_{k=1}^{N_s} \rho_k$

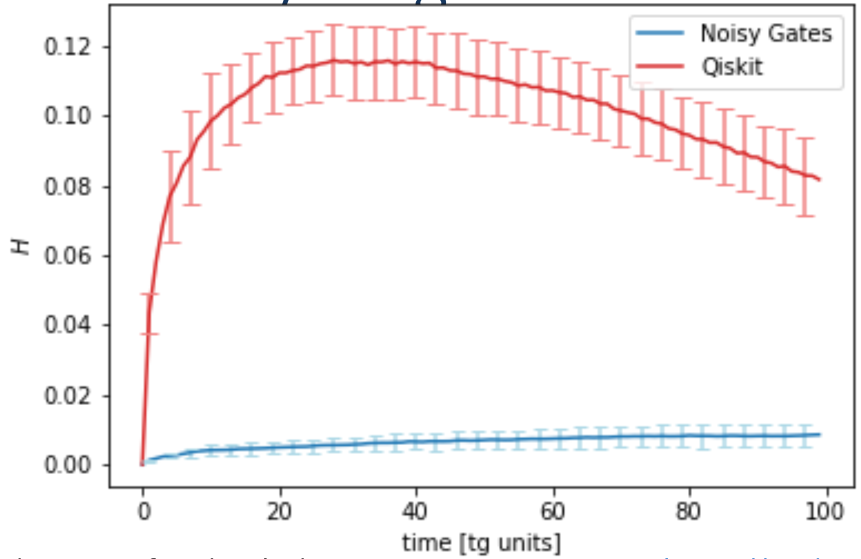
Simulation of the noisy X gate





Simulation of the noisy CR gate





Hellinger Distance

Lindblad and noisy gates

Lindblad and Qiskit simulator

$$H(P,Q) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{n}(\sqrt{p_i} - \sqrt{q_i})^2}$$

Di Bartolomeo, Vischi, Wixinger, Grossi et al., A novel approach to noisy gates for simulating quantum computers - https://arxiv.org/abs/2301.04173

Search projects

Help Sponsors Log in Register

quantum-gates 1.0.2

```
pip install quantum-gates
```

Latest version

Released: Feb 2, 2023

Quantum Noisy Gates Simulation with Python

Navigation

- Project description
- Release history
- Download files

Project description

Noisy Quantum Gates

University of Trieste | Bassi Group
CERN | CERN openlab
CERN | Open Source
CERN | QTI

Implementation of the Noisy Quantum Gates model, which is soon to be published. It is a novel method to simulate the noisy behaviour of quantum devices by incorporating the noise directly in the gates, which become stochastic matrices.

CERN 20-24 November 2023

Annual international conference focusing on the interdisciplinary field of quantum technology and machine learning

QTML
Quantum Techniques in Machine Learning

CERN QTI

https://quantum.cern/

CERN

QUANTUM
TECHNOLOGY
INITIATIVE