Quantum Convolutional Neural Networks

Iris Cong, Harvard University
Collaboration with Soonwon Choi and Mikhail D. Lukin

Nature Physics (2019), doi:10.1038/s41567-019-0648-8

Q2B Conference, December 2019

Outline

- Quantum Machine Learning: Our Motivations
- Classical and Quantum CNN Architecture
- Application 1: Quantum Phase Recognition
- Application 2: Optimizing Quantum Error Correction
- Conclusions

Outline

- **Quantum Machine Learning: Our Motivations**
- Classical and Quantum CNN Architecture
- Application 1: Quantum Phase Recognition
- Application 2: Optimizing Quantum Error Correction
- Conclusions

Machine learning: interpret and process large amounts of data

Unnamed Vehicle

Genomics

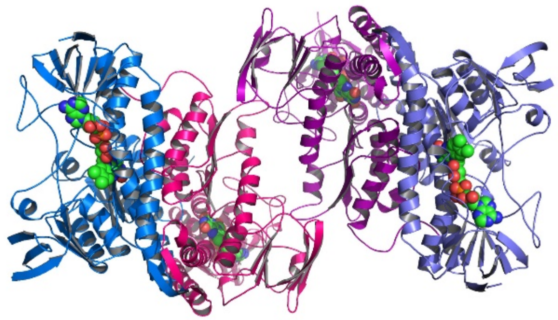


Image recognition

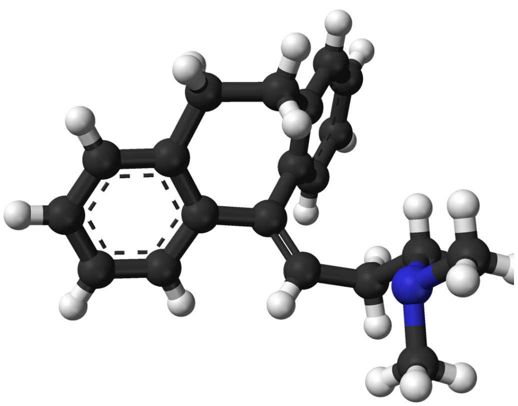
= Cat

= Dog

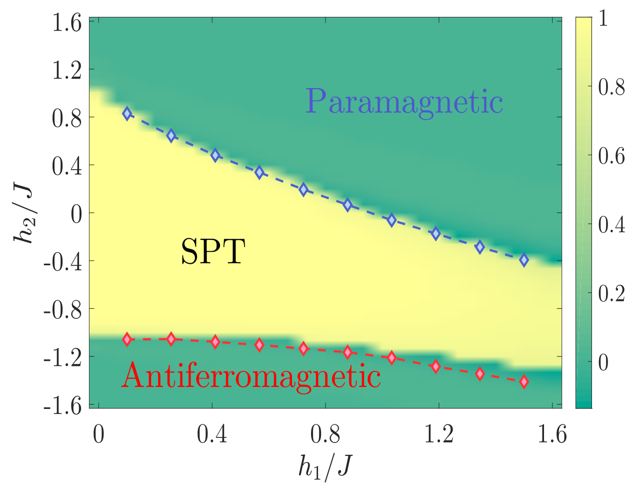
Quantum physics: many-body interactions → extremely large complexity

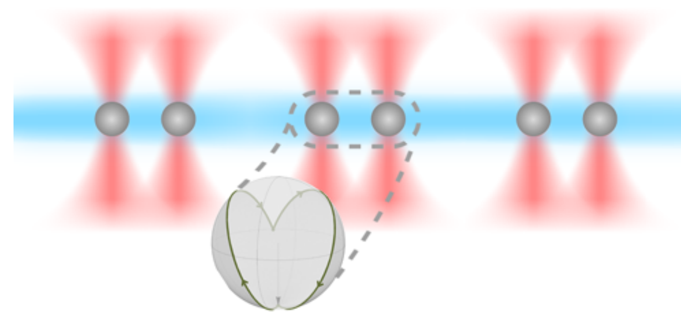
Near-term quantum computers/simulators

Quantum chemistry



Quantum phases of matter





H. Levine et al, *PRL* (2019)

Machine learning + Quantum many-body physics = Exciting!

Quantum Machine Learning

- Using quantum computers for machine learning tasks
  - Solve important problems in quantum many-body physics, quantum simulation/information
- Many open questions:
  - Why/how does quantum machine learning work?
  - Concrete models for near-term implementation?
  - Relationship to quantum many-body physics and quantum information theory?

→ Goal of our work

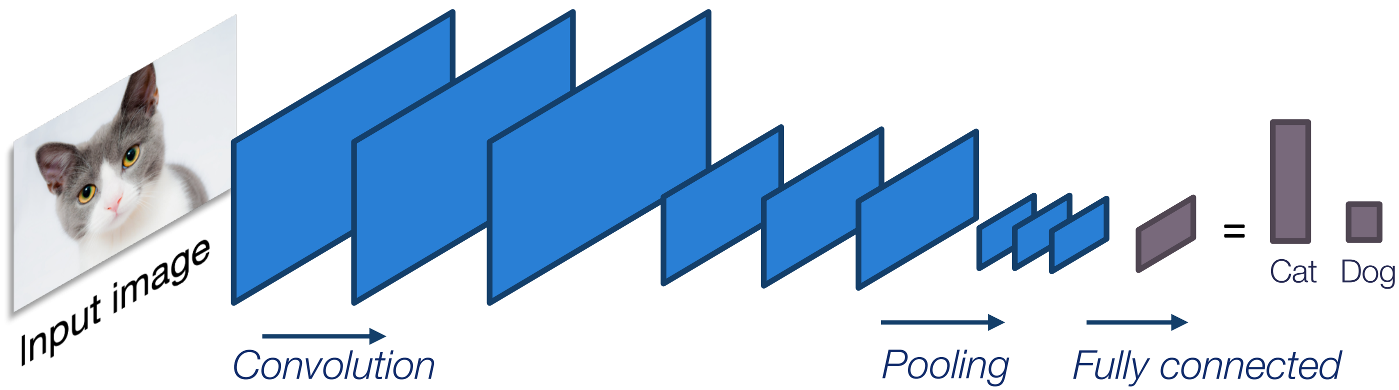
Outline

- Quantum Machine Learning: Our Motivations
- **Classical and Quantum CNN Architecture**
- Application 1: Quantum Phase Recognition
- Application 2: Optimizing Quantum Error Correction
- Conclusions

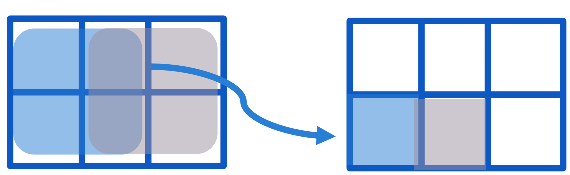
(Classical) CNN

- Structured neural network: multiple layers of image processing

Example (simplified):

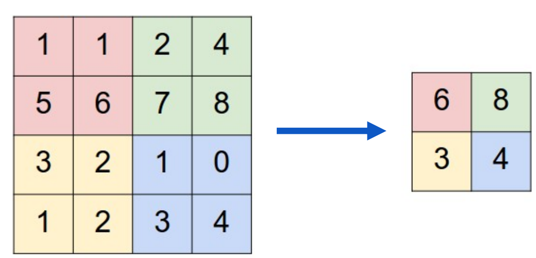


Convolution

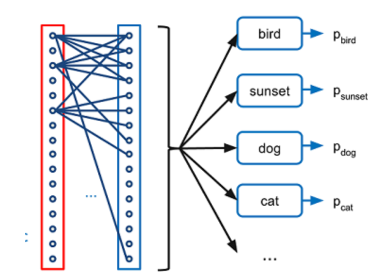


$$y_{i,j} = w_0 x_{i,j} + w_1 x_{i+,j} + w_2 x_{i,j+1} + w_3 x_{i+1,j+1}$$

Pooling



Fully connected

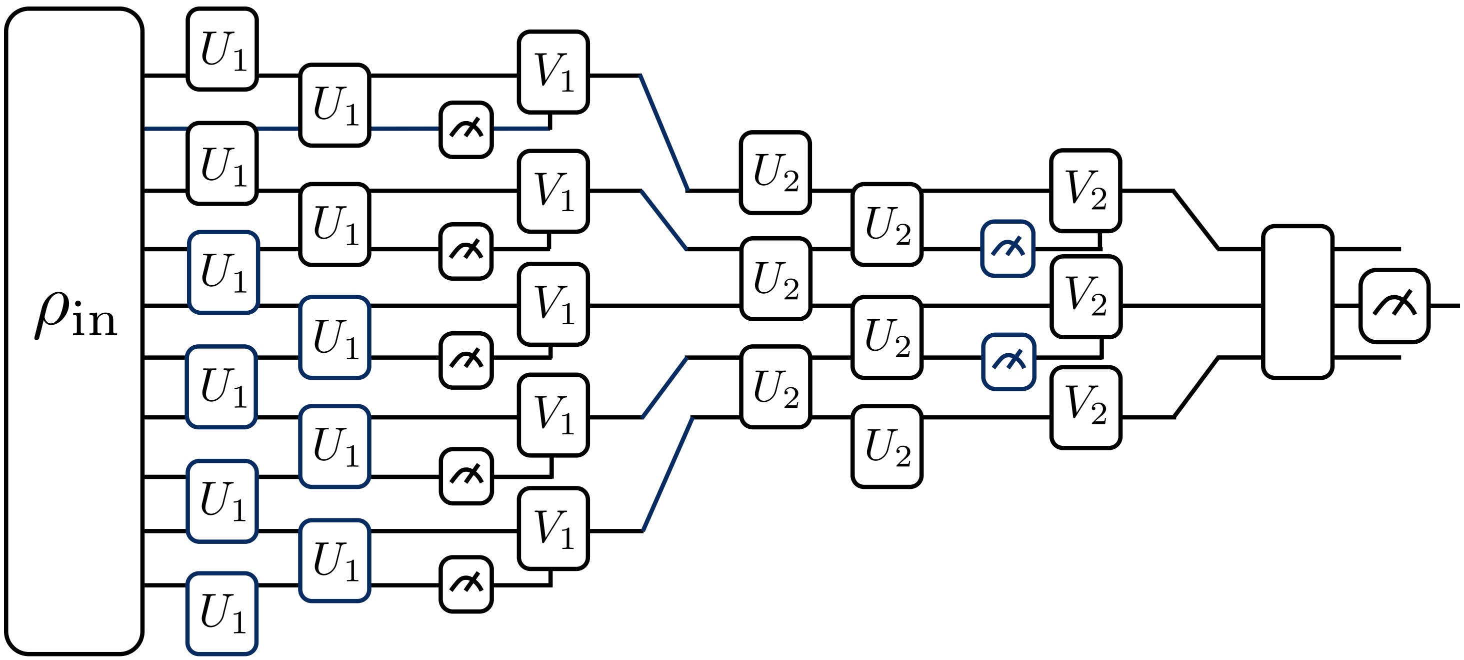


References: Fukushima (1980), Krizhevsky et al. (2012)

Quantum CNN Architecture

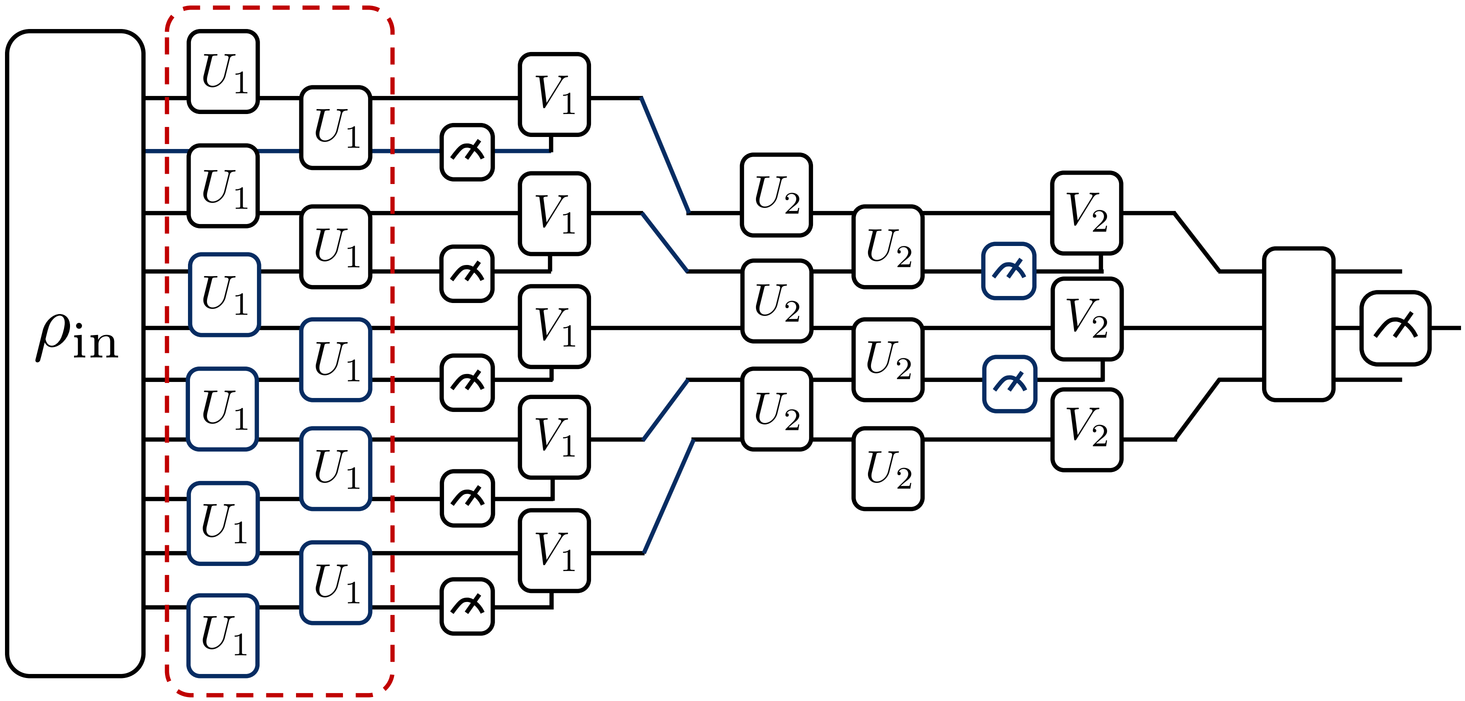
- Same types of layers:

Input state:
Pure or mixed, classical or quantum state (1D, 2D, 3D, …)



Quantum CNN Architecture

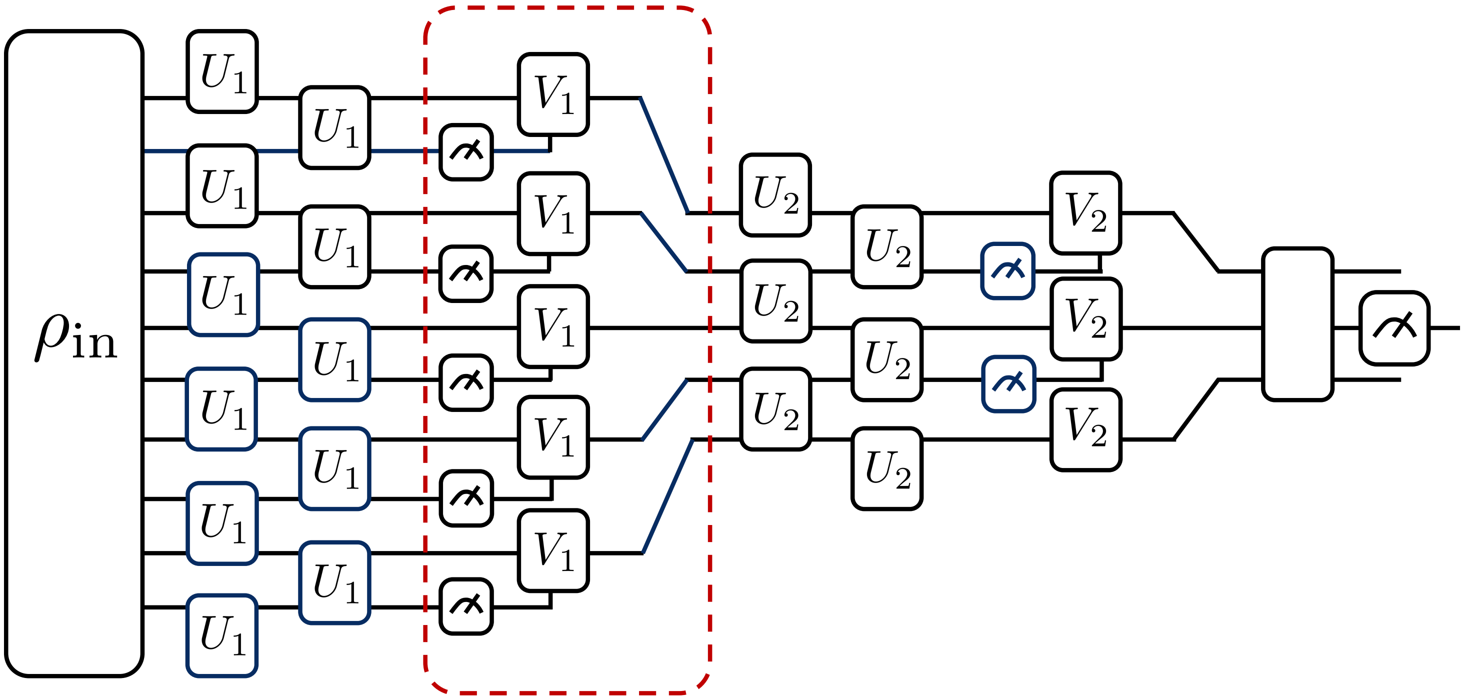
- Same types of layers:



1. Convolution:
Translationally invariant
quasilocal unitaries

Quantum CNN Architecture

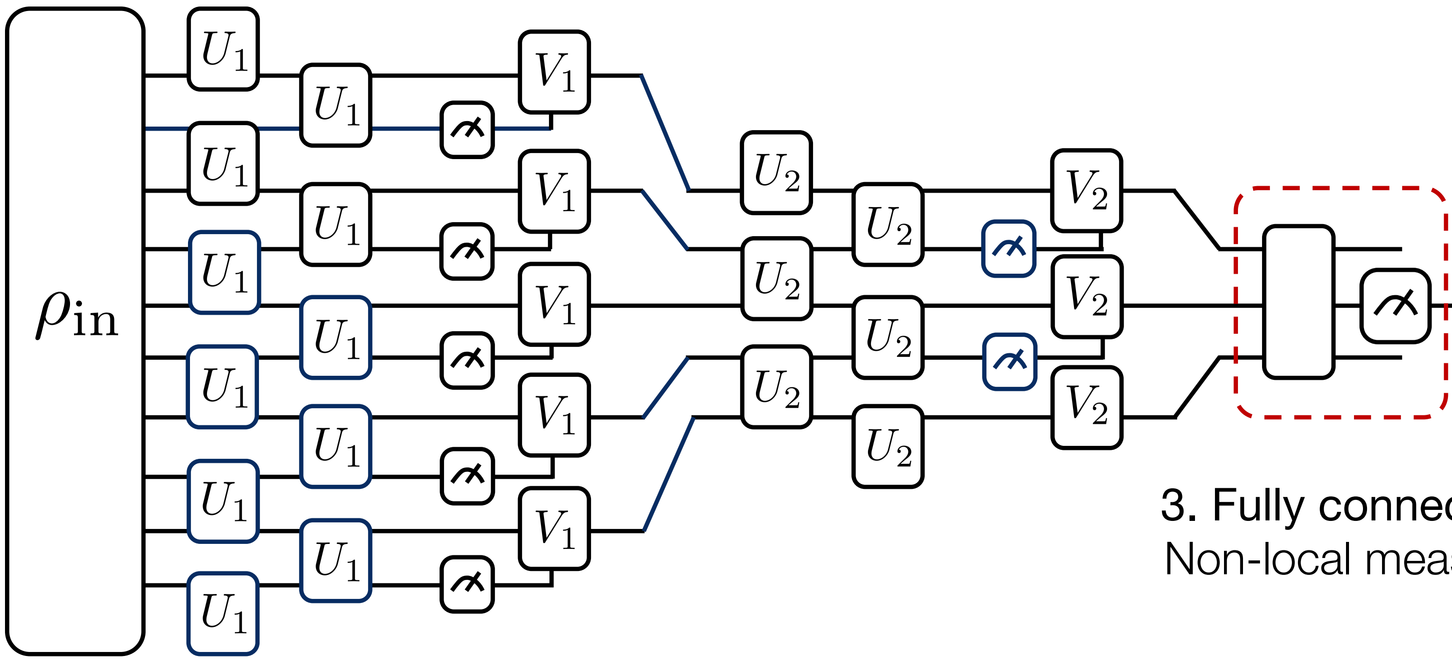
- Same types of layers:



2. Pooling:
Reduce system size: measure a fraction of the qubits, and apply unitaries to remaining qubits depending on outcome

Quantum CNN Architecture

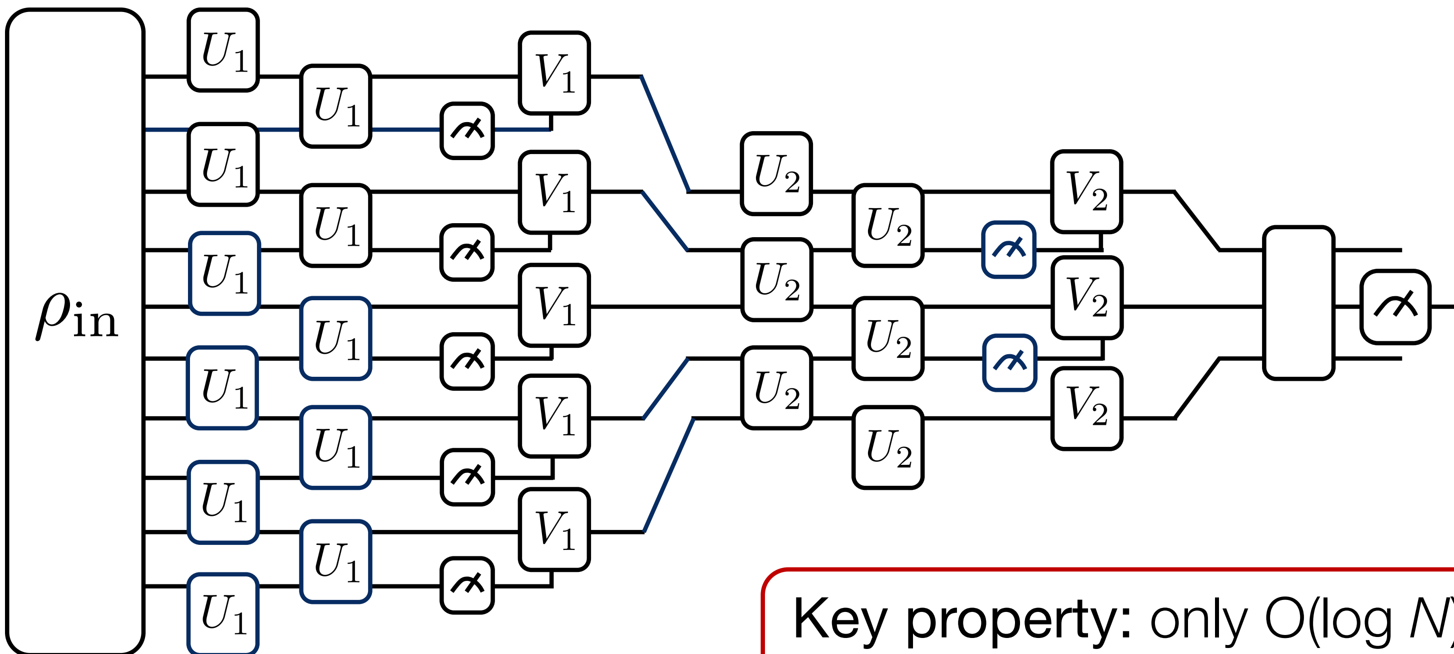
- Same types of layers:



3. Fully connected:
Non-local measurement

Quantum CNN Architecture

- Same types of layers:



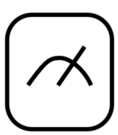
Key property: only O(log *N*) parameters → efficient implementation and training

Outline

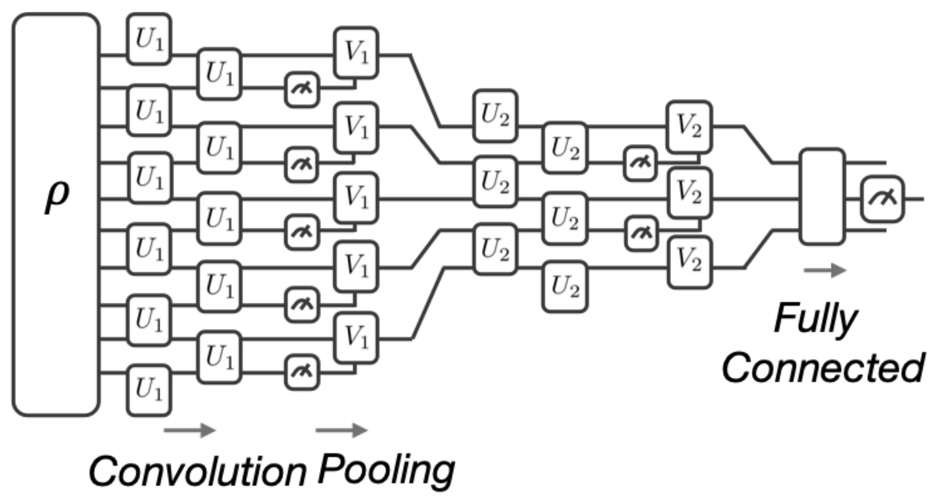
- Quantum Machine Learning: Our Motivations
- Classical and Quantum CNN Architecture
- **Application 1: Quantum Phase Recognition**
- Application 2: Optimizing Quantum Error Correction
- Conclusions

Quantum Phase Recognition

- **Problem:** Given a quantum many-body system in (unknown) state ρ, does ρ belong to a particular quantum phase $\mathcal{P}$?
  - Analogous to image classification, but *intrinsically quantum* problem
- Claim: QCNN can be very efficient in quantum phase recognition (QPR)



$$= \begin{cases} 1 & \text{if } \rho \in \mathcal{P} \\ 0 & \text{if } \rho \notin \mathcal{P} \end{cases}$$



Quantum Phase Recognition

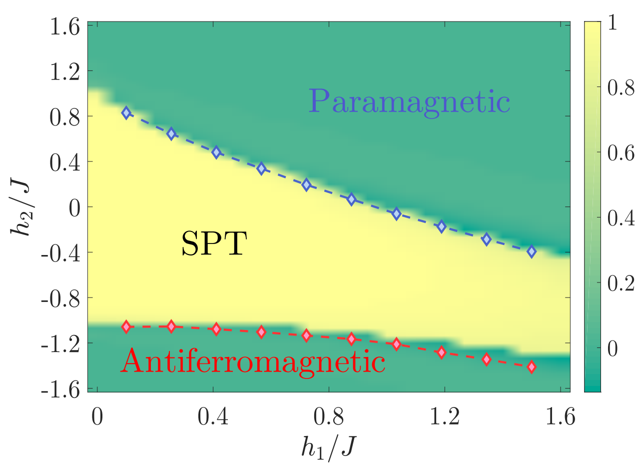
- Example: 1D ZXZ Model
  - Symmetry-protected topological (SPT) phase, protected by $G = \mathbb{Z}_2 \times \mathbb{Z}_2$
  - Cannot be detected by local order parameter
  - Same phase as $S = 1$ Haldane chain
- $H = -J \sum_i Z_i X_{i+1} Z_{i+2} - h_1 \sum_i X_i - h_2 \sum_i X_i X_{i+1}$

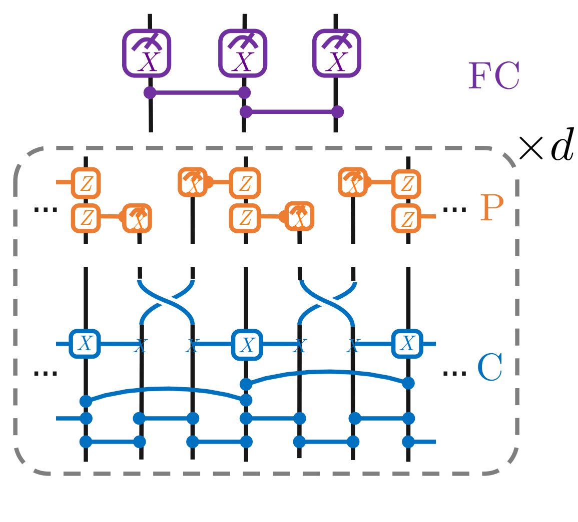
Reference: Haldane (1983)

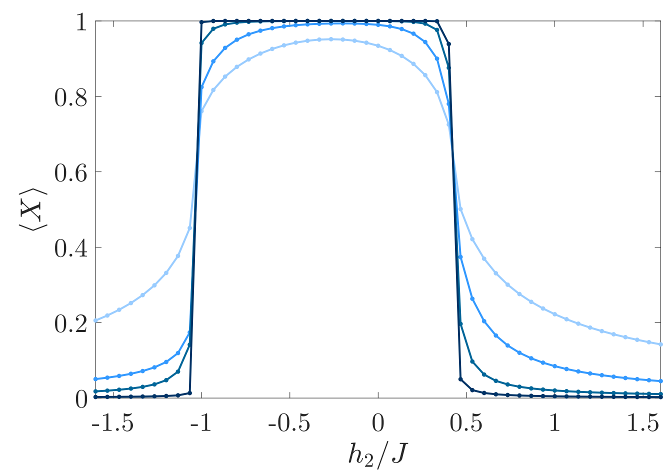
Quantum Phase Recognition

- Phase diagram from infinite system-size density matrix renormalization group (DMRG) vs. QCNN:

$$H = -J \sum_i Z_i X_{i+1} Z_{i+2} - h_1 \sum_i X_i - h_2 \sum_i X_i X_{i+1}$$







Phase boundaries (diamonds) are obtained from iDMRG with bond dimension 150.
QCNN input states are obtained from DMRG with system sizes 45, 135, bond dimension 130.
Circuit is performed using matrix-product state update.

Quantum Phase Recognition

- Quantitative advantage over traditional methods:
    - String order parameters (SOP): $\rho \in \mathcal{P}$ if $\langle S \rangle > 0$, $\rho \notin \mathcal{P}$ if $\langle S \rangle = 0$, $S = \pm 1$
    - Near phase boundaries: $\langle S \rangle \rightarrow 0$

50% H / 50% T

51% H / 49% T

Quantum Phase Recognition

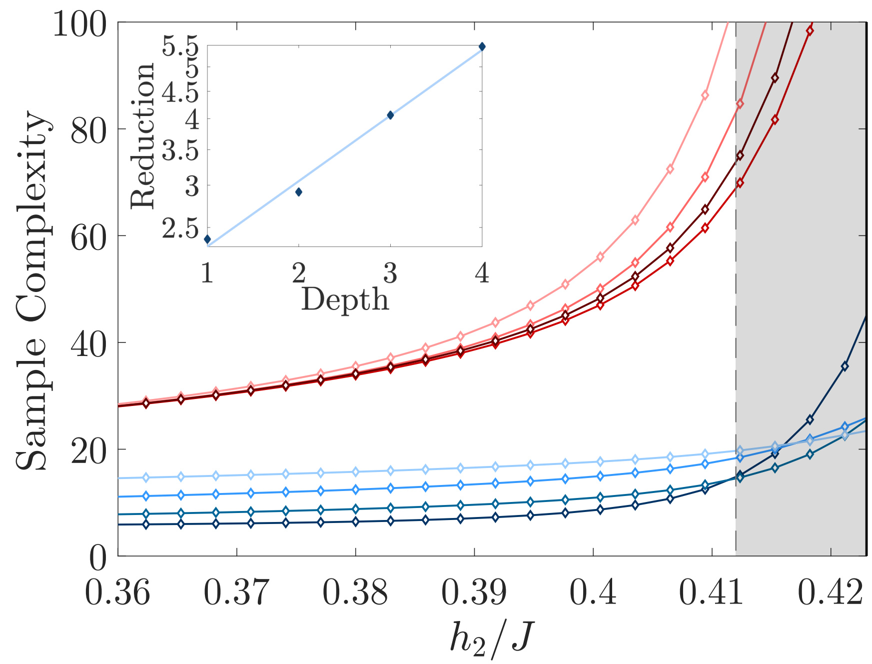
- Quantitative advantage over traditional methods:
  - String order parameters (SOP): $\rho \in \mathcal{P}$ if $\langle S \rangle > 0$, $\rho \notin \mathcal{P}$ if $\langle S \rangle = 0$, $S = \pm 1$
  - Near phase boundaries: $\langle S \rangle \to 0$
  - Sample complexity:

    \# repetitions to determine

    $\rho \in \mathcal{P}$ with 95% confidence
  - QCNN (blue) significantly outperforms SOP (red)

50% H / 50% T

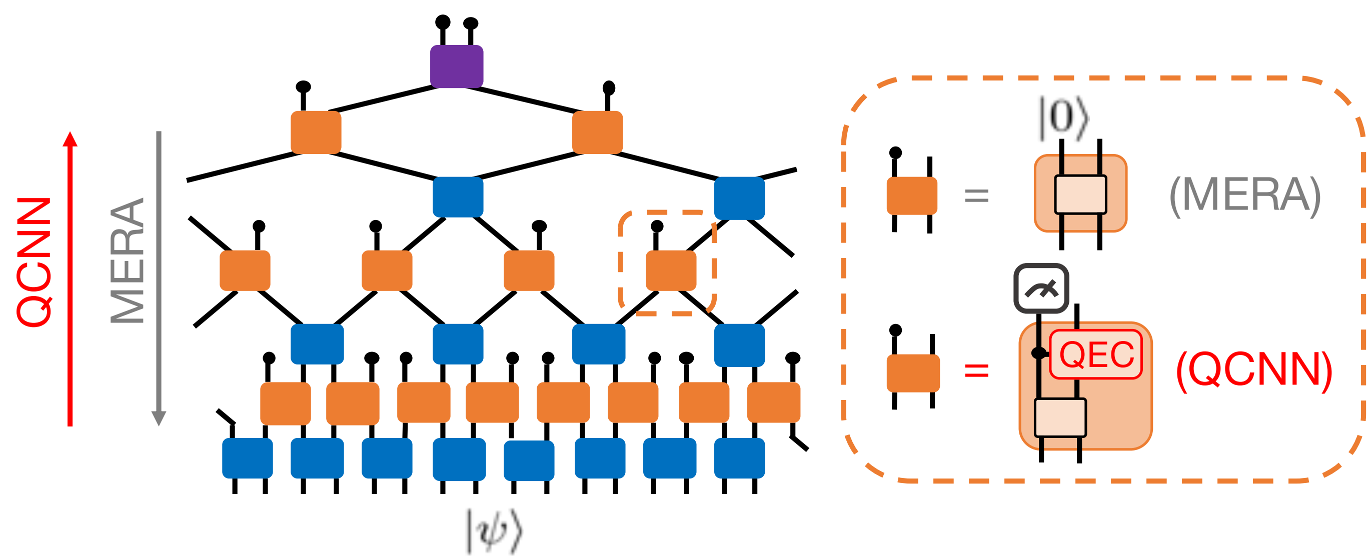
51% H / 49% T



Quantum Phase Recognition

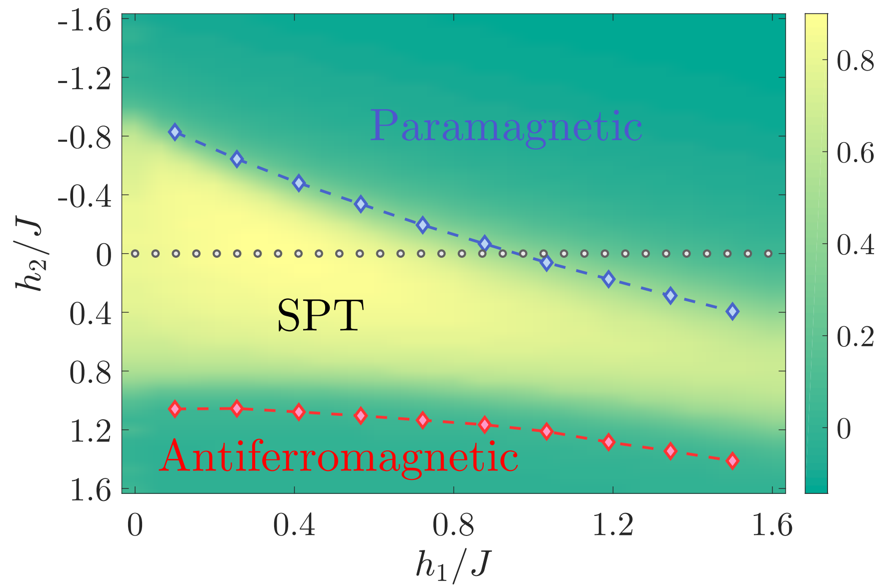
- Theoretical understanding:

QCNN = Multiscale Entanglement Renormalization Ansatz (MERA) + Quantum Error Correction (QEC)



Quantum Phase Recognition

- Training: Example
  - N = 15 spins (depth 1) for simulations
  - Initialize all unitaries to random values, perform gradient descent
  - Train along $h_2 = 0$ (solvable)
  - Observation: training along 1D, solvable set can still reproduce 2D phase diagram!
  - QCNN structure avoids overfitting



Outline

- Quantum Machine Learning: Our Motivations
- Classical and Quantum CNN Architecture
- Application 1: Quantum Phase Recognition
- **Application 2: Optimizing Quantum Error Correction**
- Conclusions

Optimizing Quantum Error Correction

- Quantum computing can be powerful:
  - Shor's algorithm, Grover, machine learning, optimization, …
- However, qubit and gate errors → need error correction for scalable computations

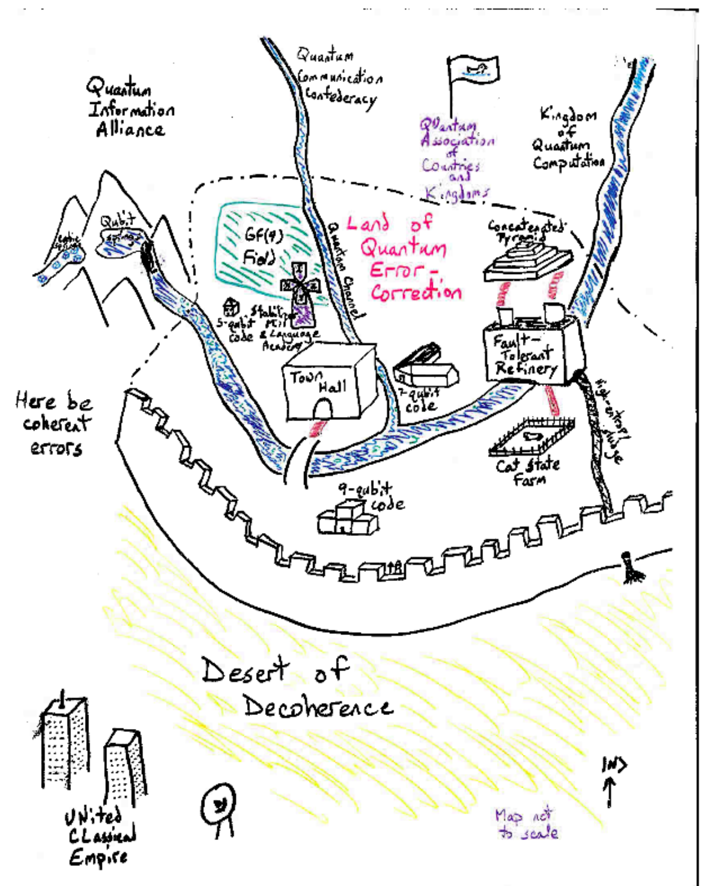
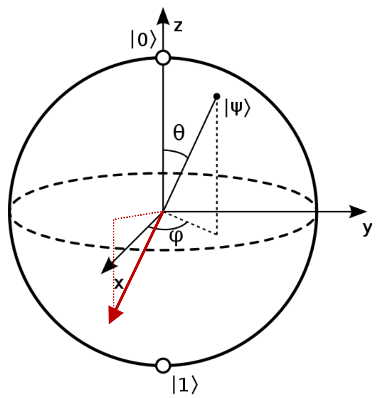


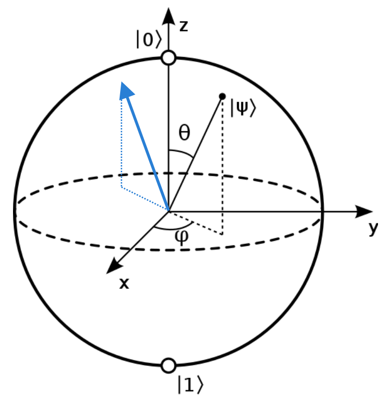
Figure credit: D. Gottesman

Optimizing Quantum Error Correction

- Quantum errors: $\rho \mapsto \sum_\alpha M_\alpha \rho M_\alpha^\dagger$ where $\sum_\alpha M_\alpha^\dagger M_\alpha = \mathbf{1}$
- Single-qubit errors generated by $M = X, Z$ → general-purpose QEC



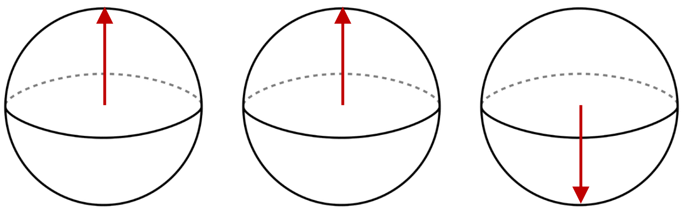
X error (bit-flip)



Z error (phase-flip)

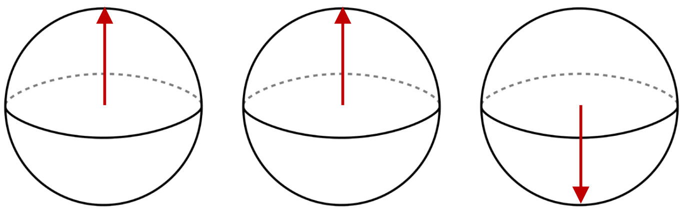
Optimizing Quantum Error Correction

- (Classical) error correction: repetition code
  - $0 \rightarrow 000,\ 1 \rightarrow 111$; majority vote



Optimizing Quantum Error Correction

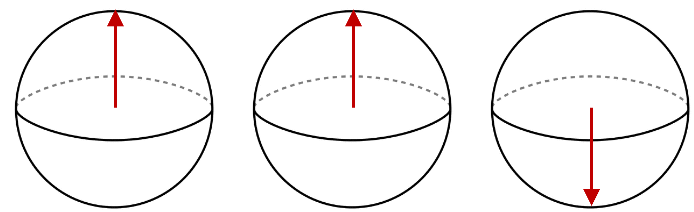
- (Classical) error correction: repetition code
  - $0 \rightarrow 000,\ 1 \rightarrow 111$; majority vote



- Quantum error correction (QEC):
  - No-cloning theorem; wavefunction collapse; X and Z errors → measure stabilizers (correlation function)
  - e.g. Shor's 9-qubit code: $|\pm\rangle_L = \frac{1}{2\sqrt{2}}(|000\rangle_L \pm |111\rangle_L)^{\otimes 3}$
  - Stabilizers: $Z_1Z_2,\ Z_2Z_3,\ Z_4Z_5,\ \ldots,\ X_1X_2X_3X_4X_5X_6,\ X_4X_5X_6X_7X_8X_9$

Optimizing Quantum Error Correction

- (Classical) error correction: repetition code
  - $0 \rightarrow 000,\ 1 \rightarrow 111$; majority vote



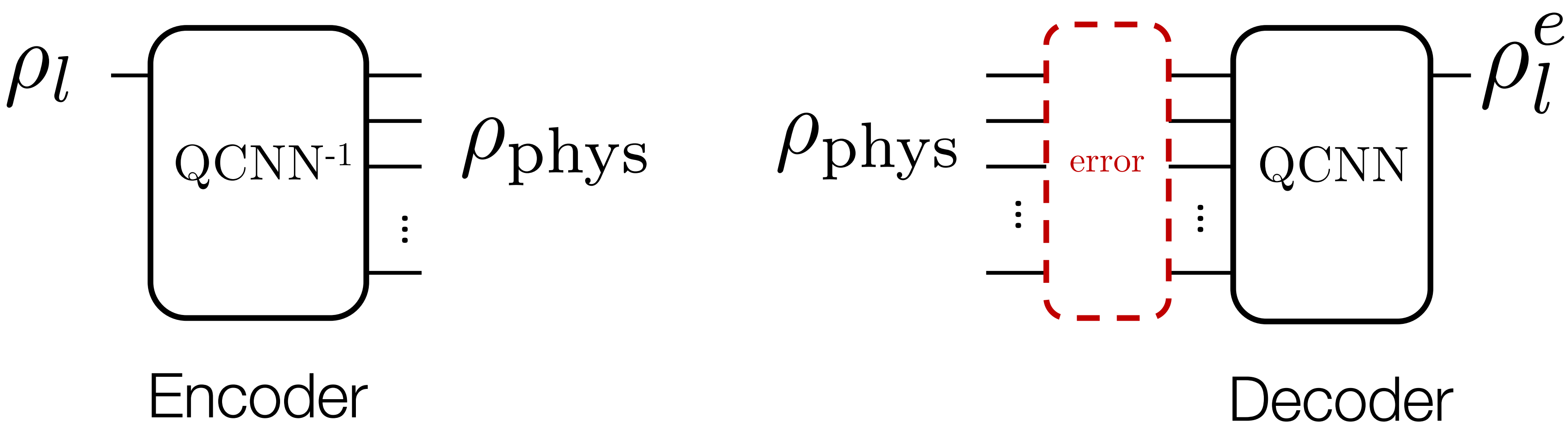
- Quantum error correction (QEC):
  - No-cloning theorem; wavefunction collapse; X and Z errors → measure stabilizers (correlation function)
  - e.g. Shor's 9-qubit code: $|\pm\rangle_L = \frac{1}{2\sqrt{2}}(|000\rangle_L \pm |111\rangle_L)^{\otimes 3}$
  - Stabilizers: $Z_1Z_2,\ Z_2Z_3,\ Z_4Z_5,\ \ldots,\ X_1X_2X_3X_4X_5X_6,\ X_4X_5X_6X_7X_8X_9$

If $Z_1Z_2 = +1,\ Z_2Z_3 =$ -1, then qubit 3 has X error

- Challenge: real experiments may have different errors!
  - e.g. correlated errors XX

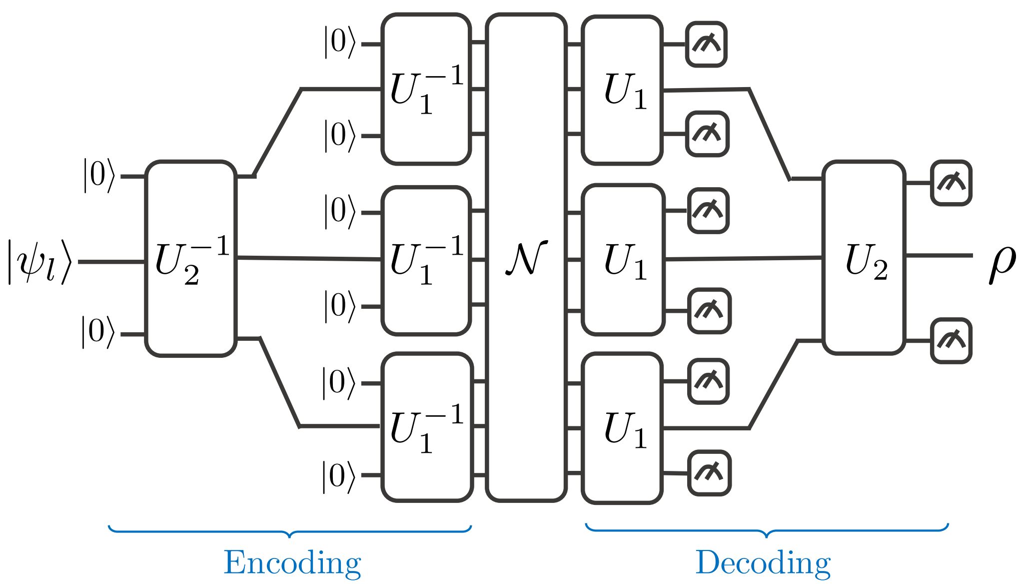
Optimizing Quantum Error Correction

- **Problem:** Given a realistic but unknown error model, find a resource-efficient quantum error correction (QEC) code to protect against these errors.
  - QCNN structure resembles nested QEC, and can be used to simultaneously optimize encoder and decoder

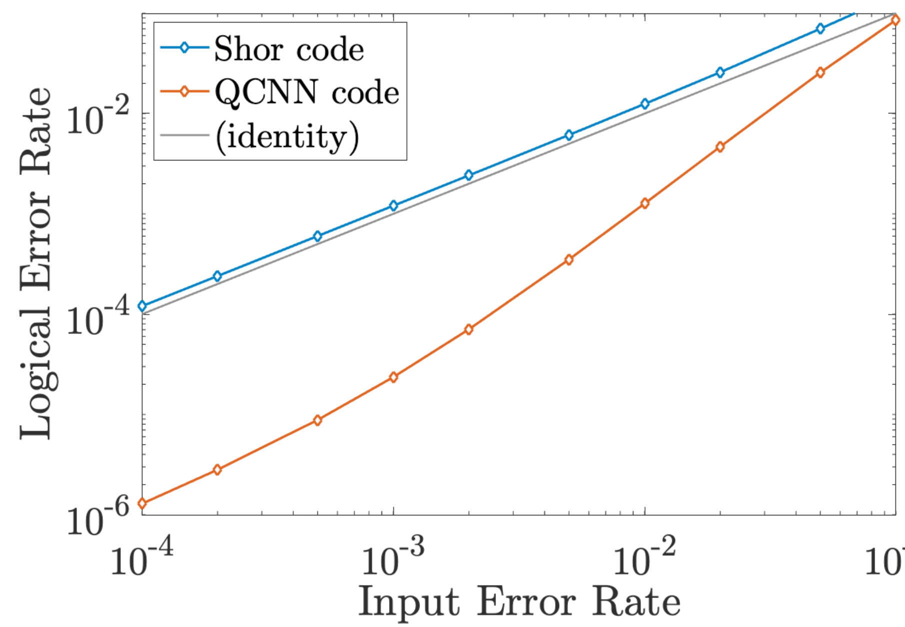


Optimizing Quantum Error Correction

Circuit structure:



Results for correlated error:

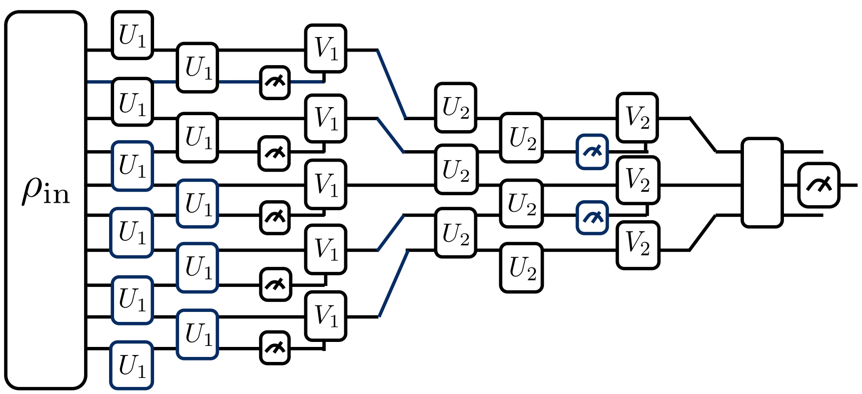


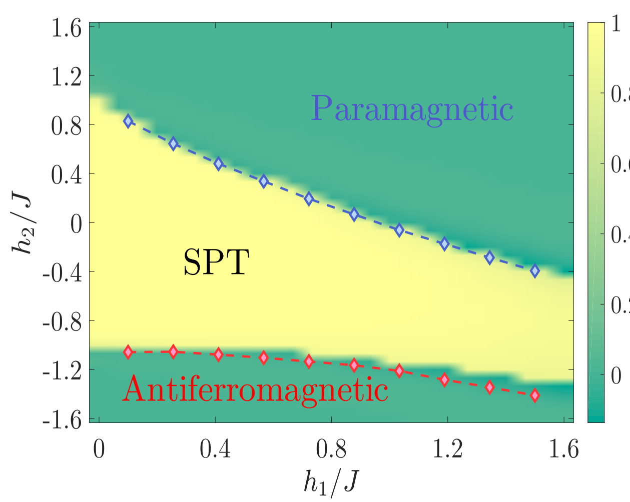
Outline

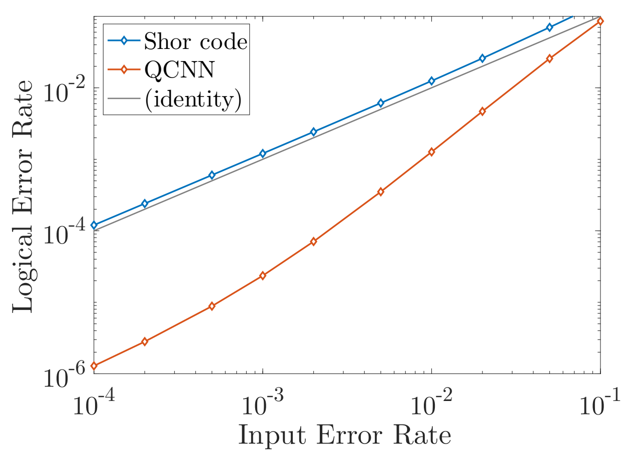
- Quantum Machine Learning: Our Motivations
- Classical and Quantum CNN Architecture
- Application 1: Quantum Phase Recognition
- Application 2: Optimizing Quantum Error Correction
- Conclusions

Conclusions

- Concrete circuit model for near-term quantum machine learning
- Application: Quantum Phase Recognition
- Application: Optimizing Quantum Error Correction







Thanks!

- Manuscript link: *Nature Physics* (2019), doi:10.1038/s41567-019-0648-8, arXiv:1810.03787v2
- Acknowledgements: