From algorithms to applications: potential roles of quantum computing

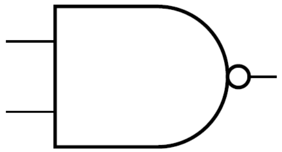
Kosuke Mitarai
Osaka University

@ High Energy Physics in the Quantum Era 2024

Quantum computing

Conventional computers

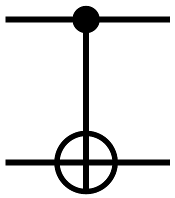
{NAND} is a universal gateset



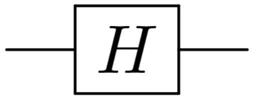
Input	Output
00	1
01	1
10	1
11	0

Quantum computers

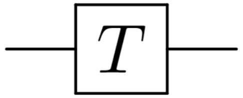
{CNOT, T, H} is a universal gateset



Input	Output
$\|00\rangle$	$\|00\rangle$
$\|01\rangle$	$\|01\rangle$
$\|10\rangle$	$\|11\rangle$
$\|11\rangle$	$\|10\rangle$



$\|0\rangle$	$(\|0\rangle + \|1\rangle)/\sqrt{2}$
$\|1\rangle$	$(\|0\rangle - \|1\rangle)/\sqrt{2}$



$\|0\rangle$	$\|0\rangle$
$\|1\rangle$	$\exp(i\pi/4)\,\|1\rangle$

Quantum computers works in a similar manner to classical ones, but with different gates

Power of quantum computers

- Simulating dynamics of interacting n ½-spins [S. Lloyd, Science, **273**, 1073-1078 (1996)]

$$O(2^n) \rightarrow \text{poly}\, n$$

- Factoring of n bit integers [P. W. Shor, Proceedings 35th Annual Symposium on Foundations of Computer Science, 124-134 (1994)]

$$O\left(e^{1.9 n^{1/3} (\log n)^{2/3}}\right) \rightarrow O(n^2 \log n \log \log n)$$

- Searching among N possibilities [L. K. Grover, Proceedings, 28th Annual ACM Symposium on the Theory of Computing, 212-219, (1996)]

$$O(N) \rightarrow O(\sqrt{N})$$

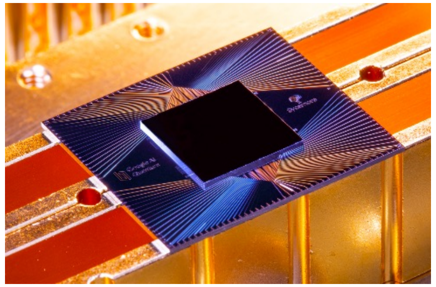
- Inversion of sparse $N \times N$ matrix (sparseness s, condition number κ, precision $1/\epsilon$) [A. Harrow et al., PRL, **103**, 150502 (2009)]

$$O(Ns\sqrt{\kappa} \log 1/\epsilon) \rightarrow \tilde{O}(\log N\, s^2 \kappa^2/\epsilon)$$

Many applications, but needs hardware for realizing them

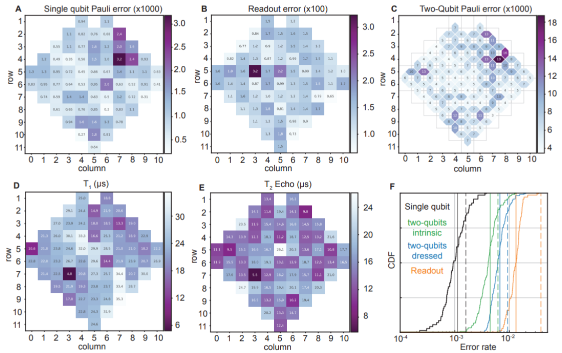
Current quantum hardware as of 2024

2019 53 qubit (transmon qubits)



F. Arute et al., Nature 2019

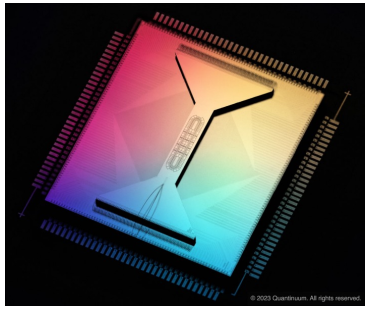
2023 70 qubit (transmon qubits)



Google Quantum AI., arXiv: 2304:11119

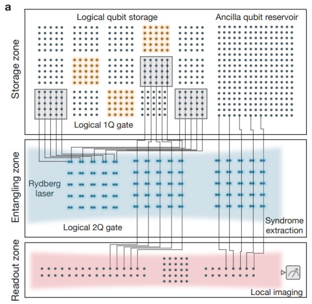
Quantum supremacy demonstrated:
classical supercomputer could not simulate dynamics of a programmable, gate-based, quantum device.

2023 56 qubit (ion trap)



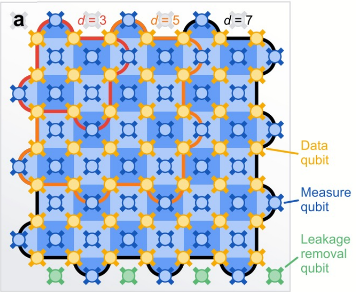
S. A. Moses et al., PRX 2023

2023 280 qubit (neutral atoms)



D. Bluvstein et al., Nature 2023

2024 105 qubit (transmon qubits)



Google Quantum AI., arXiv: 2408.13687

Quantum error correction demonstrated:
error correction seems to be possible in real world, for the first time.

Our ultimate goal: fault-tolerant quantum computing

Current error rate of qubits ~ **0.1%** [Arute et. al., Nature (2019)]

⇕

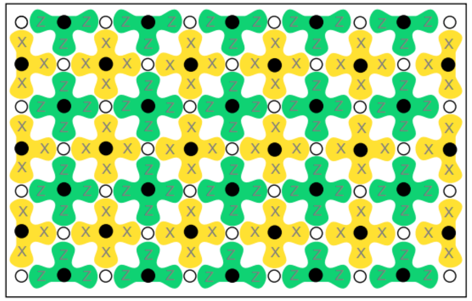
Error rate of classical bits ~ **10^{-17} %** [Oliveira et al, SC17 (2017)]

* Converting FIT to error rate from the number of clocks

Error correction is essential for “normal" calculations

Repetition code 000 → (Error) 010 → (Majority vote) 000

Surface code



Make clean 1 qubit with ~1000 qubits

[Phys. Rev. A **86**, 032324 (2012)]

My talk today

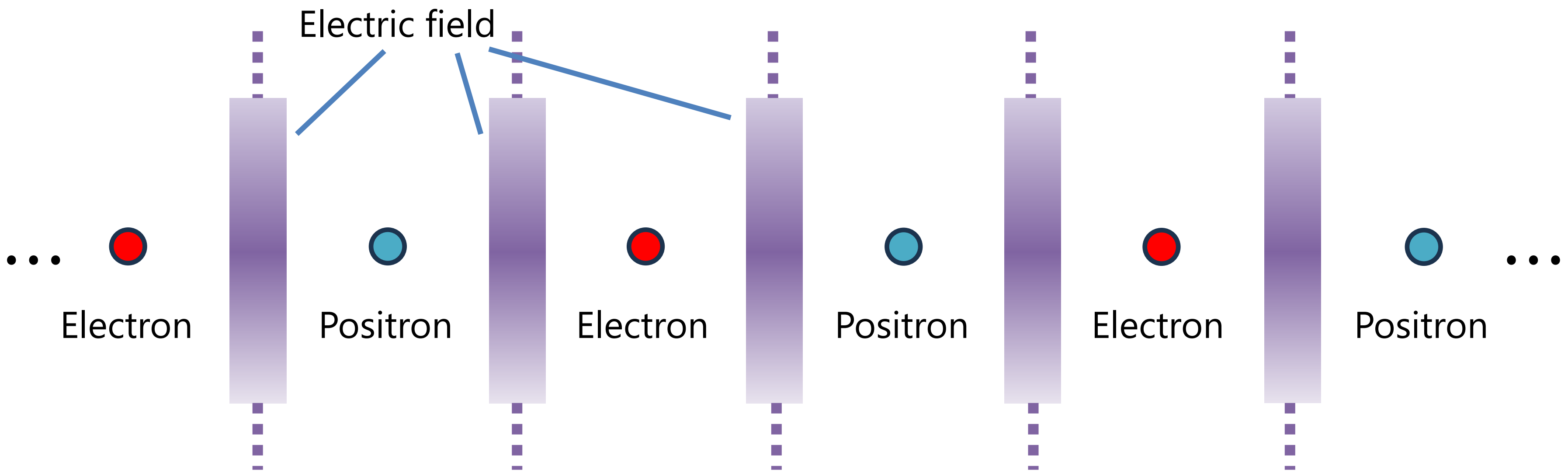
- Developing more efficient algorithms and frameworks are very important to harness the power of quantum.
- First part: efficient simulation algorithm for Schwinger model and its applications.
- Second part: a novel quantum machine learning framework

First part: quantum algorithm for Schwinger model

K. Sakamoto, Hayata Morisaki, Junichi Haruna, Etsuko Itou, Keisuke Fujii, Kosuke Mitarai, "End-to-end complexity for simulating the Schwinger model on quantum computers", arXiv:2311.17388

Schwinger model

One of the simplest yet non-trivial gauge theories



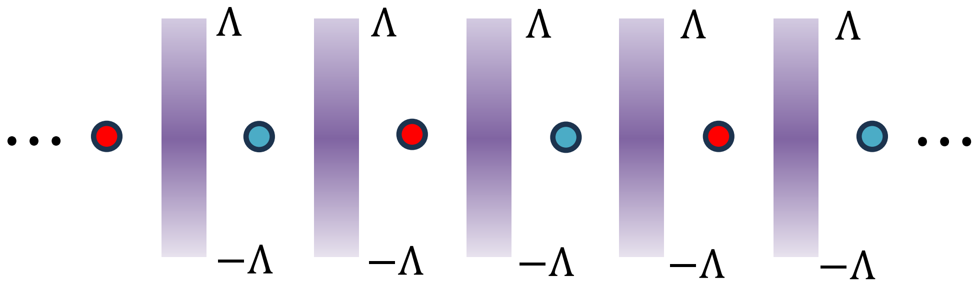
Truncate the electric field at Λ

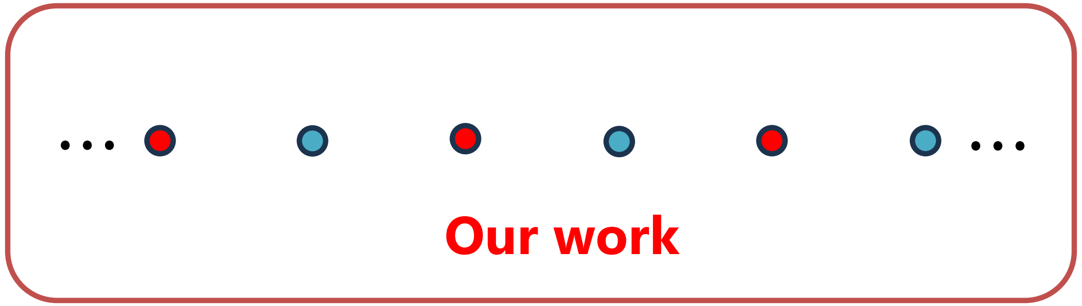
J. Kogut, et al, Phys. Rev. D (1975).

Two types of Hamiltonian formulation

Remove the electric field with Gauss's law

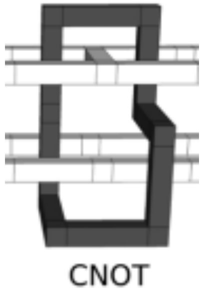
T. Banks, et al, Phys. Rev. D (1976).

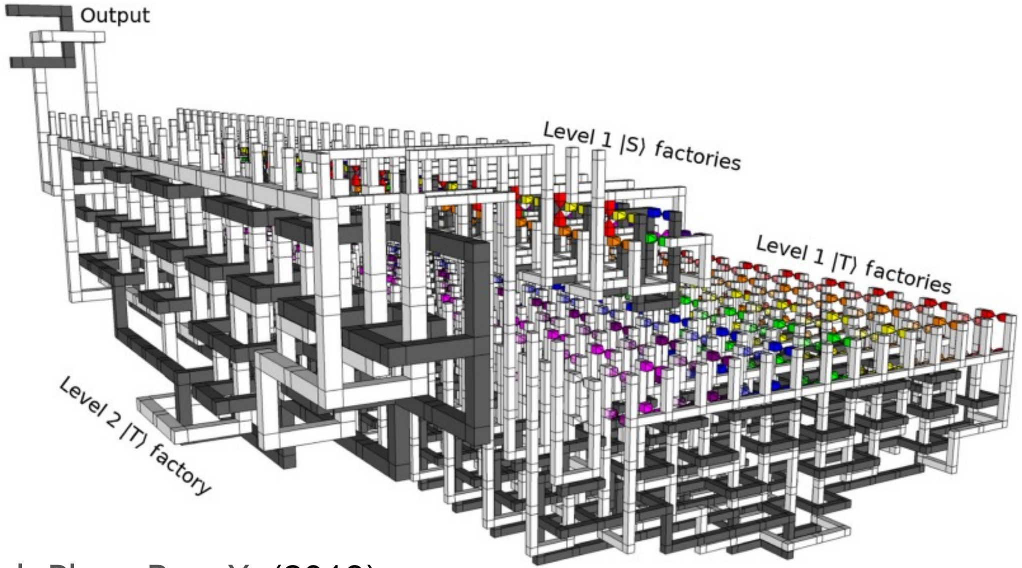




The cost is estimated via number of T gates

- **In FTQC setting, T gates are the most costly.**
  - FTQC usually allows {H, CNOT, T} gates, which are universal.
  - H and CNOT gates are very easy, but T gates need large space-time cost.
  - It is because of the structure of error-correction codes defined via commuting Pauli operators.





R. Babbush, et al, Phys. Rev. X, (2018).

- *Note added: recent works (Itogawa et al., arXiv: 2403.03991, Gidney et al., arXiv:2409.17595) might change the situation. Number of T gates, however, still roughly represents how many gates we need.*

Previous works on Schwinger model for e^{-iHt}

System size : N
Precision : ε
Evolution time : t

- The Hamiltonian formulation which does not have electric field

  E. A. Martinez, et al, Nature 534, 516 (2016).
  N. H. Nguyen, et al, PRX Quantum 3, 020324 (2022). .... $O(N^{4.5}t^{1.5}/\varepsilon^{0.5})$

  - Based on Trotter formula

  Our work .... $\tilde{O}\left(N^4 t + \log(1/\varepsilon)\right)$

- The Hamiltonian formulation which has electric field

  A. F. Shaw, et al, Quantum 4, 306 (2020). .... $\tilde{O}(N^{2.5}t^{1.5}/\varepsilon^{0.5}))$

  - Based on Trotter formula
  - Provides rigorous cost analysis

  Y. Tong, et al, Quantum 6, 816 (2022). .... $\tilde{O}(Nt\ \mathrm{polylog}(1/\varepsilon))$

  - The smallest query complexity at present
  - Probably needs a huge number of qubits

- **Our work improves in every factor from the previous Trotter-based one.**
- **Compared to ones with electric field, our algorithm needs smaller number of qubits.**

Block-encoding

See also: John M. Martyn et al., "Grand Unification of Quantum Algorithms", PRX Quantum **2**, 040203 (2021)

- Block-encoding of a Hamiltonian H is defined as,

$$U = |0^b\rangle\langle 0^b| \otimes H + \cdots = \begin{pmatrix} H & \cdot \\ \cdot & \cdot \end{pmatrix}$$

  We will see how to implement such U on the next page.

- Here we assume $U^2 = I$. This holds for popular block-encoding implementations.
- Let $R = 2|0^b\rangle\langle 0^b| \otimes I - I \otimes I$; R adds phase -1 when first b qubits are not $|0\rangle$.
- Surprisingly, the following holds:

$$(RU)^n = RU \cdots RURU = \begin{pmatrix} T_n(H) & \cdot \\ \cdot & \cdot \end{pmatrix}$$

  where $T_n(H)$ is the Chebyshev polynomial.

- **Advantages:**
  - **Block-encoding U of H with error ϵ only requires $O(\log(1/\epsilon))$ gates in most cases.** (Trotter expansion needs $\mathrm{poly}(1/\epsilon)$ gates.)
  - **We can get any information about H with $T_n(H)$; Most functions can be efficiently approximated by linear combination of $T_n(x)$.**

Block-encoding of Pauli-sum Hamiltonians

- Assume Hamiltonian is decomposed as sum of Pauli operators $P \in \pm\{I, X, Y, Z\}^{\otimes n}$:

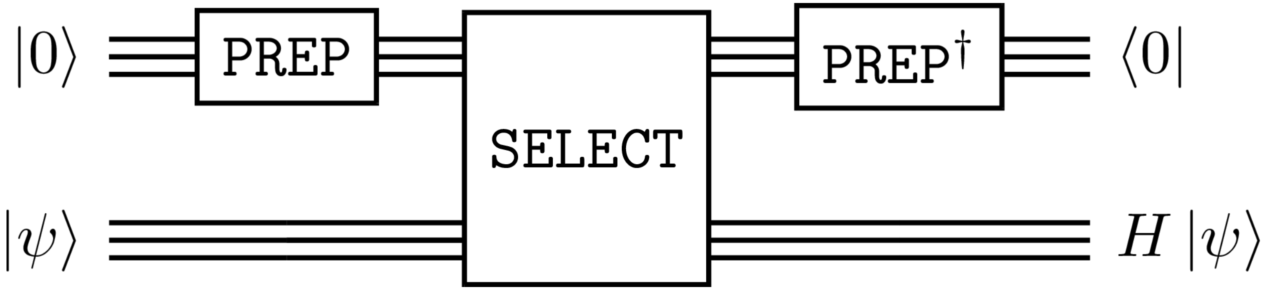
$$H = \sum_{l=0}^{L-1} a_l P_l$$

  Furthermore, assume $a_i > 0$ and it is normalized such that $\sum_i a_i = 1$.

- Let the PREPARE operator PREP and SELECT operator SELECT be ones that satisfies:

$$\mathrm{PREP}|0^b\rangle = \sum_{l=0}^{L-1} \sqrt{a_l}\,, \mathrm{SELECT}(|l\rangle|\psi\rangle) = |l\rangle \otimes (P_l|\psi\rangle)$$

- The following gives a block-encoding:



- P and V can be implemented $O(L + \log 1/\epsilon)$ gates (using ancillary qubits). R. Babbush, et al, Phys. Rev. X, (2018)
- This technique is called the **linear combination of unitaries (LCU)**.

Our idea to efficiently implement block-encoding

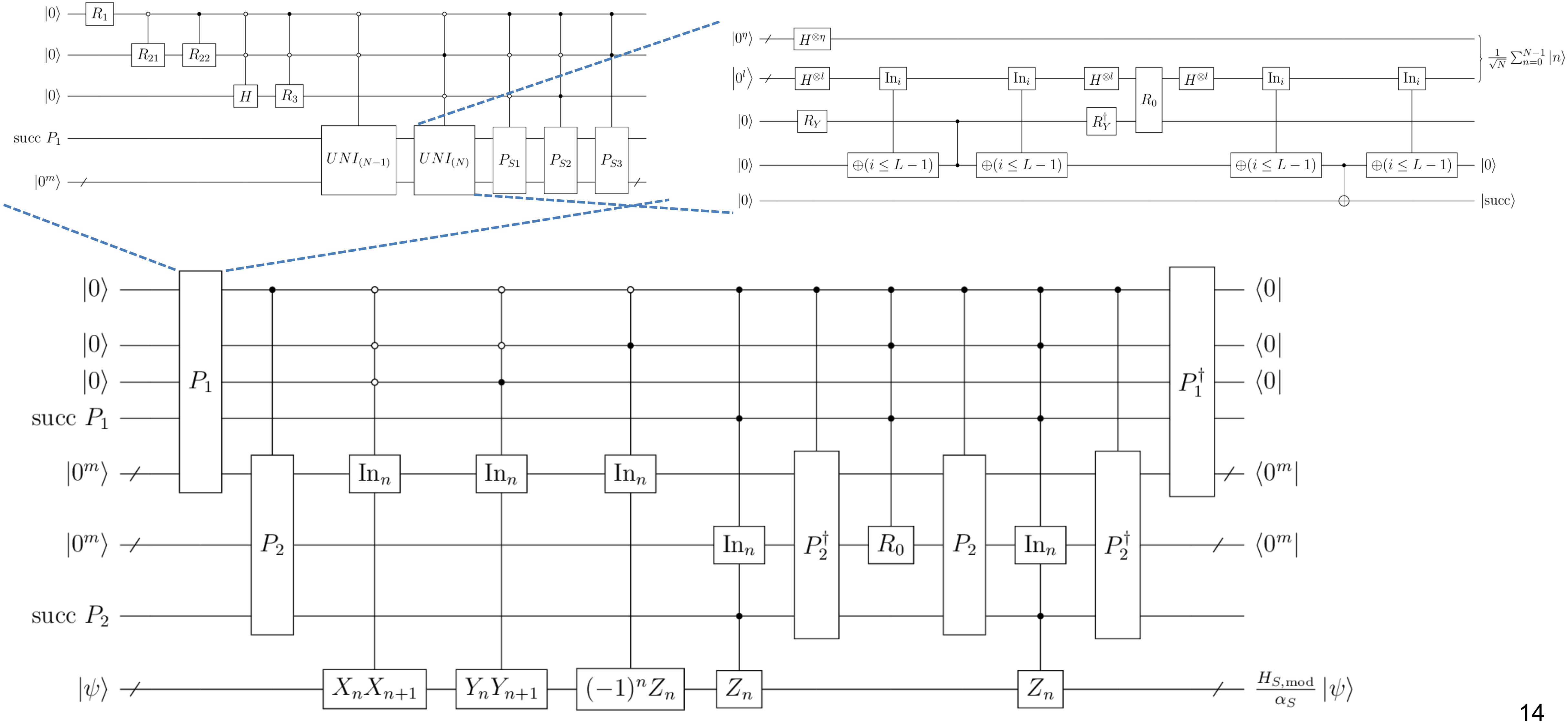
- The Schwinger model Hamiltonian after Jordan-Wigner transformation looks like:

$$H_S = J\sum_{n=0}^{N-2}\left(\sum_{i=0}^{n}\frac{Z_i+(-1)^i}{2}+\frac{\theta_0}{2\pi}\right)^2+\frac{w}{2}\sum_{n=0}^{N-2}(X_nX_{n+1}+Y_nY_{n+1})+\frac{m}{2}\sum_{n=0}^{N-1}(-1)^nZ_n$$

- It has $O(N^2)$ terms, we naively need $O(N^2)$ gates to block-encode H_S.
- **Our strategy to realize it with $O(N)$ gates:**
  - Uniform superposition states $\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}|i\rangle$ can be prepared efficiently with $O(\log N)$ T gates.
    Y. R. Sanders, et al, PRX Quantum (2020)
  - We can take a linear combination of the block-encodings via LCU.
  - Noting the above, group the terms as follows:

$$H_S = \boxed{\frac{J}{4}\sum_{n=1}^{N-1}\left(\sum_{i=0}^{n-1}Z_i\right)^2}+\boxed{J\frac{\theta}{2\pi}\sum_{\substack{n=1\\ n:\text{even}}}^{N-1}\sum_{i=0}^{n-1}Z_i}+\boxed{J\left(\frac{1}{2}+\frac{\theta}{2\pi}\right)\sum_{\substack{n=1\\ n:\text{odd}}}^{N-1}\sum_{i=0}^{n-1}Z_i}+\boxed{\frac{w}{2}\sum_{n=0}^{N-2}X_nX_{n+1}}+\boxed{\frac{w}{2}\sum_{n=0}^{N-2}Y_nY_{n+1}}+\boxed{\frac{m}{2}\sum_{n=0}^{N-1}(-1)^nZ_n}$$

Quantum circuit for block-encoding looks like...



Resource estimates for computing $\langle \mathrm{vac}|e^{-iHt}|\mathrm{vac}\rangle$

- $|\mathrm{vac}\rangle = |1010\cdots\rangle$ is the ground state of H_S for $J = \theta_0 = w = 0, m = m_0$, representing vacuum without any particle.
- $\langle \mathrm{vac}|e^{-iHt}|\mathrm{vac}\rangle$ is *vacuum persistent amplitude*, representing the creation and annihilation of electron-positron pairs

J. Schwinger, Phys. Rev. (1951)

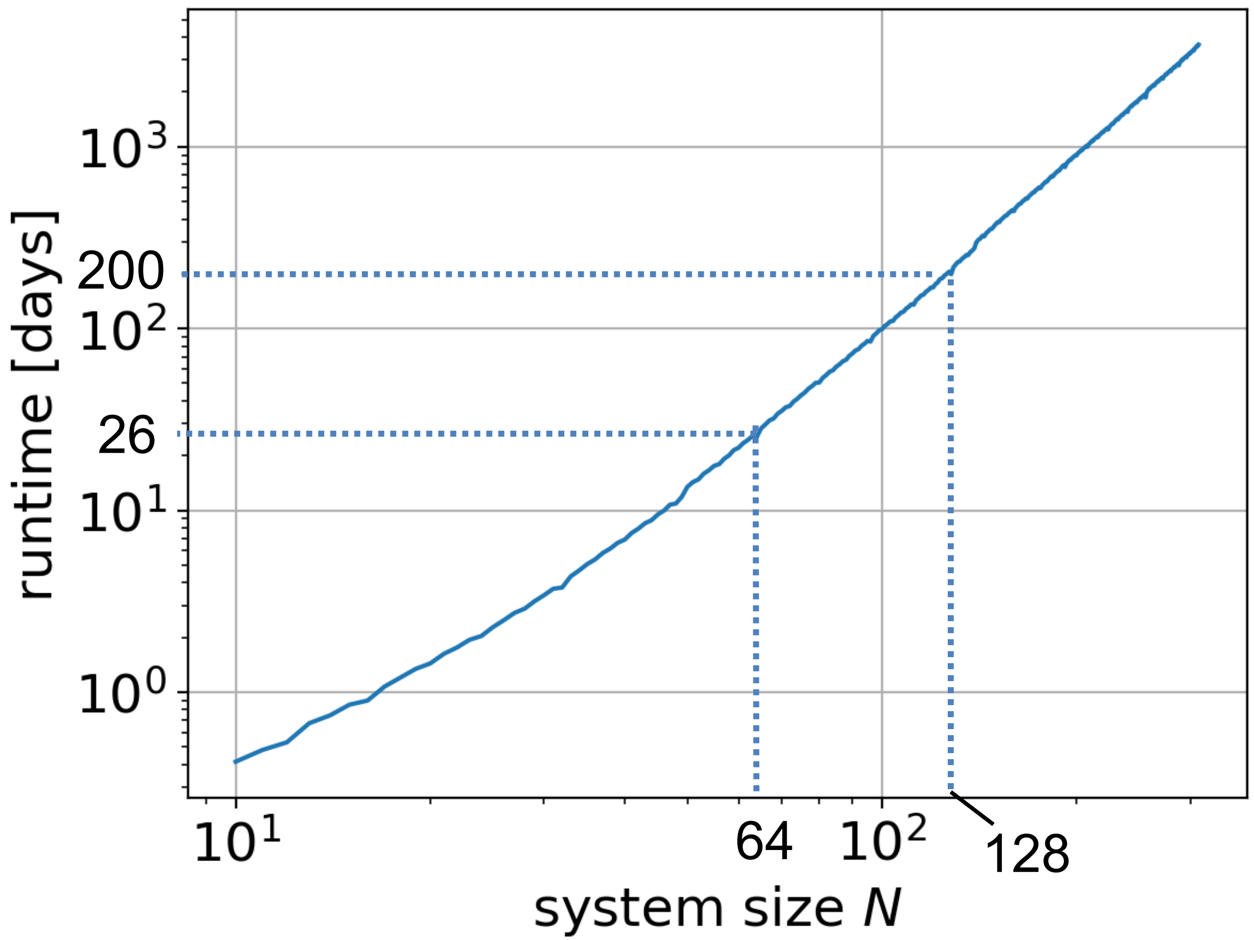
- Based on our block-encoding, how long does it take to compute it with FTQC?

Resource estimate result - runtime

- Parameters
  - Precision (additive error) : $\varepsilon = 0.01$
  - Evolution time : $t = 4$
  - T gate consumption rate : 1MHz
  - Lattice spacing : $a = 0.2$
  - electron mass : $m = 0.1$
  - $w = \frac{1}{2a} = 2.5$
  - $J = \frac{g^2 a}{2} = 0.1, (g = 1)$
  - $\theta_0 = \pi$

Examples

System size	Runtime [days]
64	26
128	200



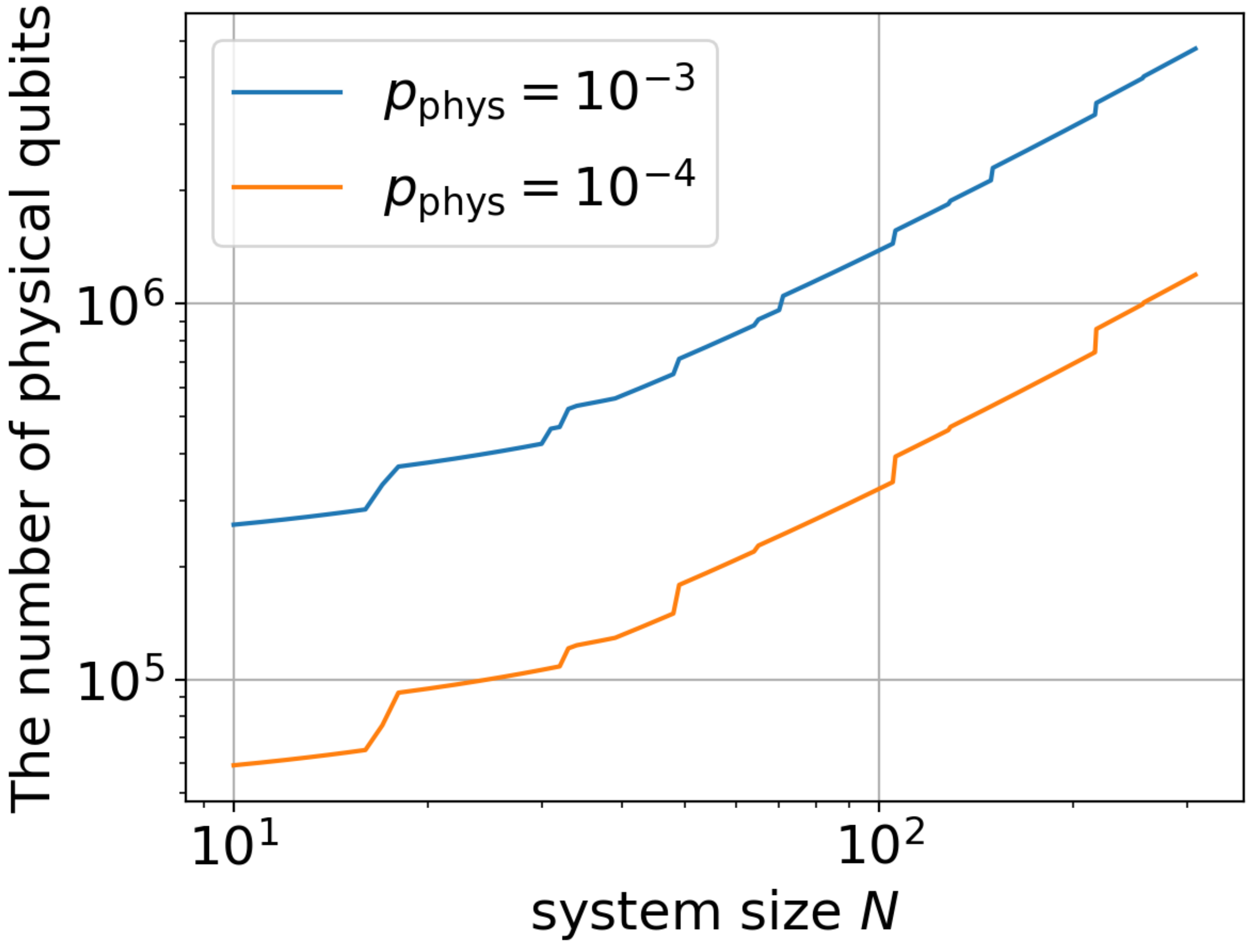
Runtime for calculating the vacuum persistence amplitude.

Resource estimate result: qubit requirements

- Parameters
  - Precision (additive error) : $\varepsilon = 0.01$
  - Evolution time : $t = 4$
  - Lattice spacing : $a = 0.2$
  - electron mass : $m = 0.1$
  - $w = \frac{1}{2a} = 2.5$
  - $J = \frac{g^2 a}{2} = 0.1, (g = 1)$
  - $\theta_0 = \pi$

Examples ($N = 64$)

Physical error rate	Physical pubits
10^{-3}	9×10^5
10^{-4}	2×10^5



The number of physical qubits
for calculating the vacuum persistence amplitude.

Summary

Comparing resource to other applications

Condensed matter physics (e.g. Hubbard model)	$<$	Schwinger model	$\approx$	Quantum chemistry (electronic Hamiltonian)
N. Yoshioka, et al, arXiv:2210.14109, (2022)				J. Lee, et al, PRX Quantum (2021)
T count : $\sim 10^8$		$\sim 10^{12}$		$\sim 10^{12}$

Technical contributions:

- An efficient block-encoding of the Schwinger model Hamiltonian
  - Decompose the Hamiltonian into several parts.
  - Use $O(\log^2 N)$ T gates for P, $O(N)$ T gates for V, with a normalization factor of $O(N^3)$.
- End-to-end complexity for the Schwinger model

Future challenges:

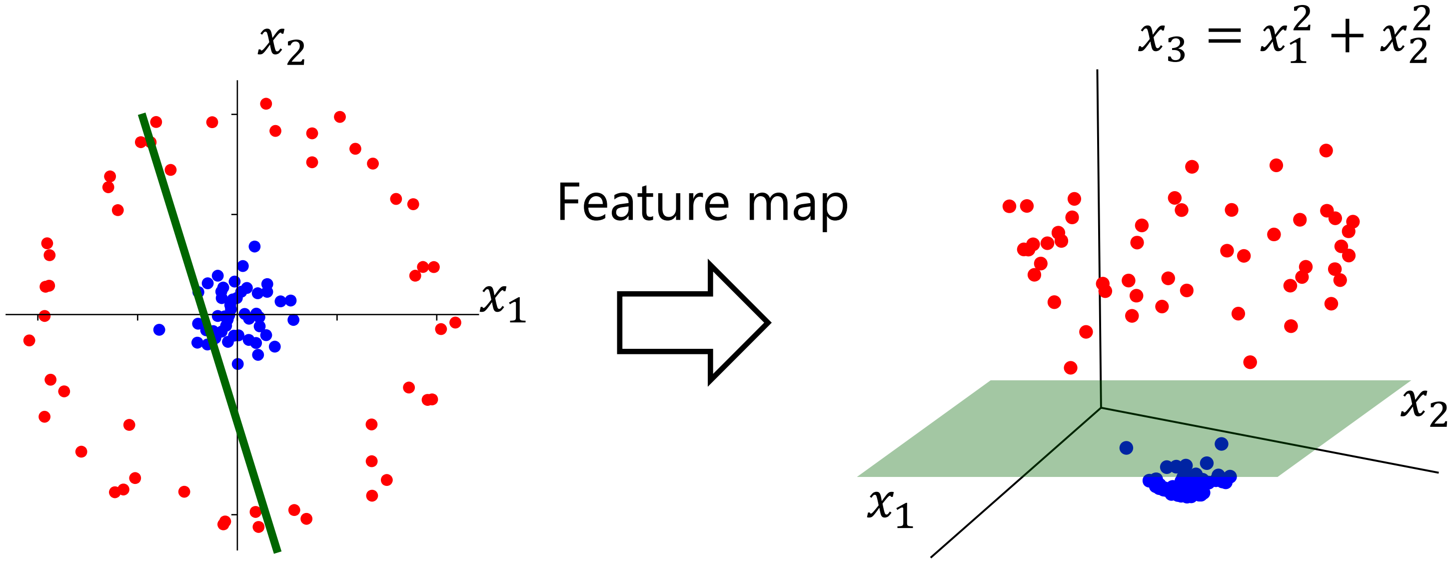
- More precise resource estimates. Maybe using libraries such as qualtran or Qiskit, which have implementations of reversible arithmetics.

A new quantum machine learning framework: Explicit quantum surrogate

Akimoto Nakayama, Hayata Morisaki, Kosuke Mitarai, Hiroshi Ueda, Keisuke Fujii, "Explicit quantum surrogates for quantum kernel models", arXiv:2408.03000

Feature map

Transform data $\boldsymbol{x}$ to $\boldsymbol{\phi}(\boldsymbol{x})$ to extract "pattern" in the data.

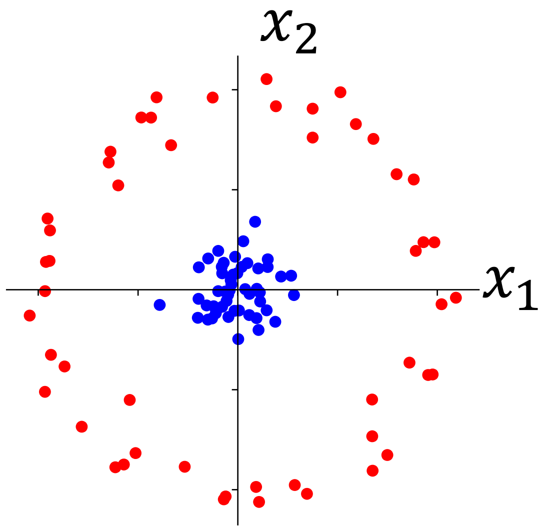


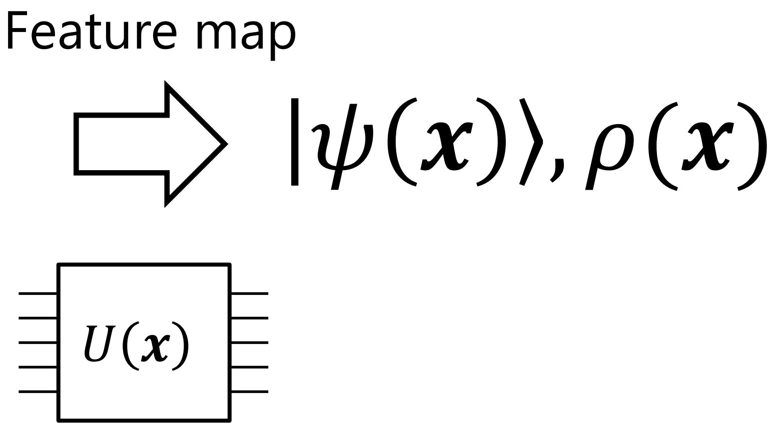
Linear model fails

Linear model succeeds

Quantum feature

- Map a data $\boldsymbol{x}$ to a quantum state $|\psi(\boldsymbol{x})\rangle$

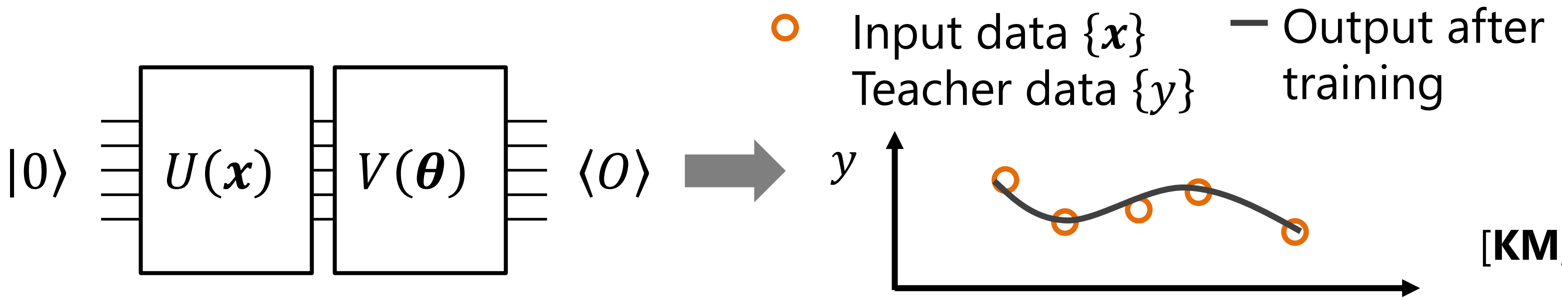




- Difficult to classically simulate $|\psi(\boldsymbol{x})\rangle$
  → **We can construct a model which cannot be treated with classical computers**
- Note!: it doesn't mean there is practical advantage.
- Models based on quantum feature can be categorized into two major class:
  explicit models and implicit models

Explicit quantum models

- **Apply parameterized unitary $V(\boldsymbol{\theta})$ and use some expectation value as a prediction.**



[**KM**, et al, Phys. Rev. A **98**, 032309 (2018)]

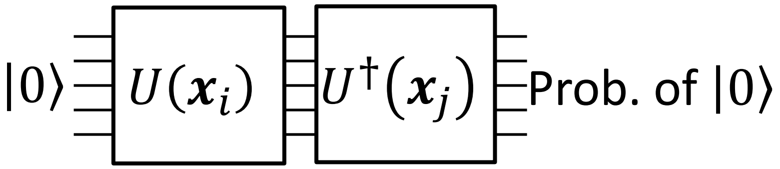
- Training is performed by tuning $\boldsymbol{\theta}$ via e.g. gradient decent.
- Advantage:
  - Can use optimizers from neural networks, such as Adam.
  - One training iteration needs only $O(N)$ resource for N data.
- Disadvantage:
  - Theoretical performance not guaranteed.
  - Difficult optimization, *barren plateau* (gradient vanishes when using random initialization.)

McClean et al., Nat. Comm. **9**, 4612 (2018)

Implicit quantum models

- **Using $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = |\langle\psi(\boldsymbol{x}_i)|\psi(\boldsymbol{x}_j)\rangle|^2$ as a kernel function, rely on classical kernel techniques for constructing a model. Model in this case is represented by**

$$y = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{i=1}^{N} \alpha_i$$



- Training is performed by tuning $\boldsymbol{\alpha}$ via, e.g., solving linear system of equations.
- Advantage:
  - *Convergence to global optimum guaranteed*.
  - Friendly to experiments, >20 qubits experiments are possible.
- Disadvantage:
  - Needs at least $O(N^2)$ cost for training, $\boldsymbol{O}(\boldsymbol{N})$ **cost for prediction.**

So we thought, can we train a model *implicitly* and then convert it to *explicit* ones?

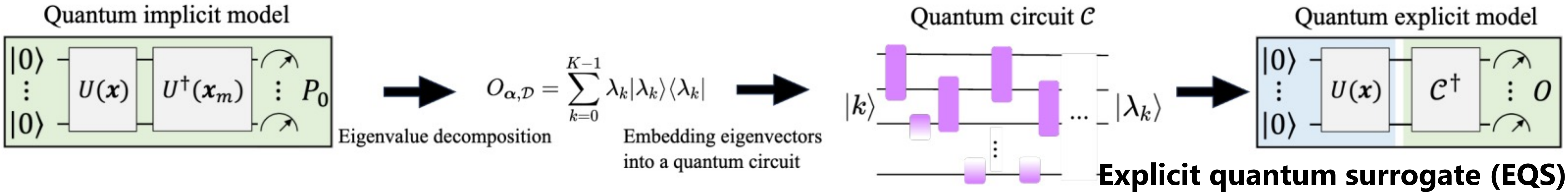
Implicit to explicit conversion

Algorithm

1. Train a quantum implicit model with your choice of quantum feature map $U(\boldsymbol{x})$
2. Diagonalize $O = \sum_i \alpha_i |\psi(\boldsymbol{x}_i)\rangle\langle\psi(\boldsymbol{x}_i)|$ and identify K important eigenvectors $|\lambda_k\rangle$.
3. Construct a circuit C that satisfies $C|k\rangle \approx |\lambda_k\rangle$ using AQCE algorithm [Shirakawa et al., Phys. Rev. Research 6, 043008 (2024)].

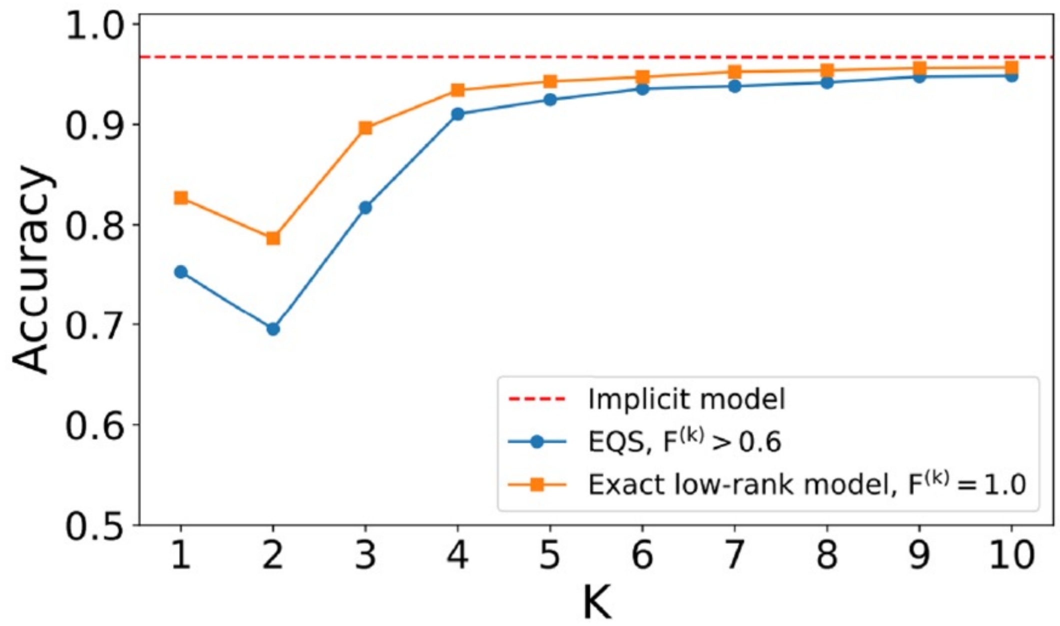
Notes

- Step 2 can be done efficiently on a classical computer when given $\langle\psi(\boldsymbol{x}_i)|\psi(\boldsymbol{x}_j)\rangle$ from quantum, because $\mathrm{rank}\, O \leq N$
- AQCE algorithm brute-forcely searches possible circuits using ideas from tensor network.



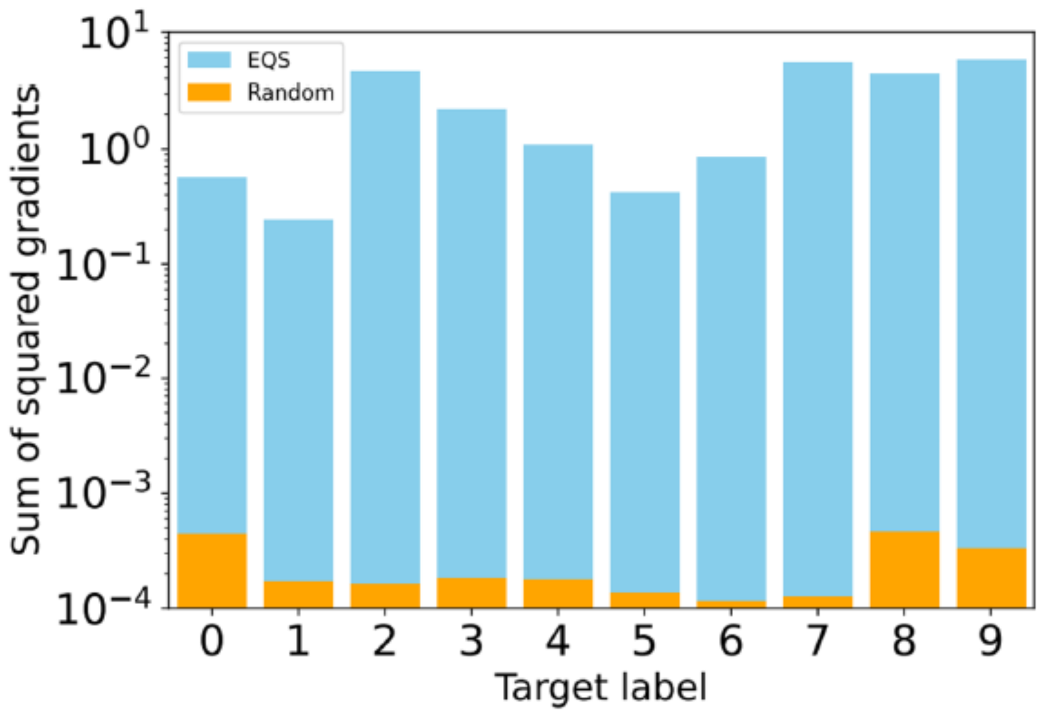
Explicit quantum surrogate (EQS)

Numerics: Accuracy for MNISQ-MNIST dataset



- ✓ MNISQ dataset [Placidi et al., arXiv:2306.16627] is a dataset developed by us, which consists of quantum circuits that approximately encodes MNIST handwritten digits.
- ✓ The conversion needs relatively small K and low fidelity (>0.6).
- ✓ **Numerics demonstrate that we can suppress the prediction cost by this approach.**

Numerics: EQS as an initialization strategy



- ✓ Motivation Random circuit initialization leads to barren plateau, but EQS constrction is not random. **Can our approach mitigate the barren plateau?**
- ✓ Numerics show that magnitude of gradients are (EQS) $\gg$ (Random initialization)
- ✓ EQS might open a way to mitigate the barren plateau.

My talk today

- Developing more efficient algorithms and frameworks are very important to harness the power of quantum.
- First part: efficient simulation algorithm for Schwinger model and its applications.
  - We constructed algorithm based on block-encoding framework with detailed resource estimates.
  - We need around 1 million physical qubits and for 10^{12} gates for ~100 site Schwinger model.
- Second part: a novel quantum machine learning framework
  - Converting trained implicit models to explicit models has various benefits: Shorter prediction time, potential to mitigate barren plateau, etc.

Some ads

Qcoder: competitive quantum programming

- An IPA MITOU project (for which I am doing technical adviser)
- Competitive programming using qiskit.

An example problem from the latest contest:

Problem Statement

You are given an integer n. Implement the operation of preparing the state $|\psi\rangle$ from the zero state on a quantum circuit qc with n qubits.

The state $|\psi\rangle$ is defined as

$$|\psi\rangle = \frac{1}{\sqrt{n}}(|10...0\rangle_n + |010...0\rangle_n + \cdots + |0...01\rangle_n).$$

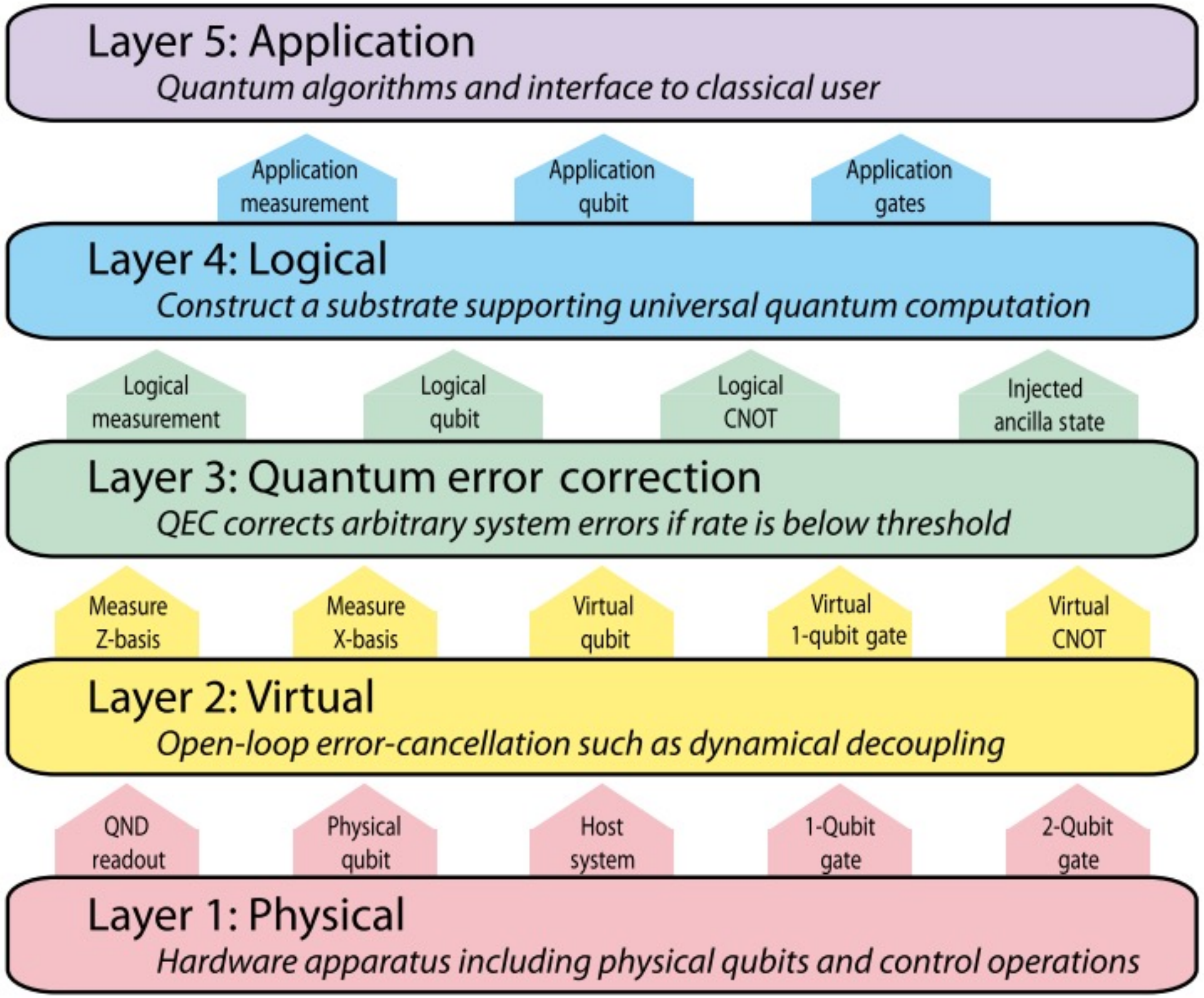
Constraints

- $2 \leq n \leq 15$
- **The circuit depth must not exceed** 10.
- Global phase is ignored in judge.

Access **qcoder.jp**

We are conducting full-stack research

Layers for practical quantum computing



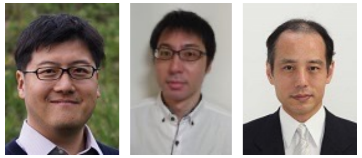
Phys. Rev. X **2**, 031007 (2016)

Chemistry: Prof. Mizukami
Many-body/cond-mat: Prof. Ueda
Machine learning: Mitarai

Mitarai, Prof. Fujii

Prof. Fujii

Prof. Negoro, Miyoshi, Ogawa with QuEL, quantum middleware startup



Superconducting qubits: Prof. Negoro
Ion trap: Prof. Toyoda

Current quantum computer system @ OU

User

↓ OpenQASM (quantum assembly)

AWS (authentication, data storage, etc.)

↓ OpenQASM (quantum assembly)

Edge server (transpiling/compiling)

↓ Transpiled QASM

Control system (middleware)

↓ Microwave pulses

Qubit chip

```
OPENQASM 2.0;
include "qelib1.inc";
qreg q[4];
cx q[0],q[1];
rz(-0.0960973223923264З) q[1];
cx q[0],q[1];
cx q[1],q[2];
rz(-0.06088586113564654) q[2];
cx q[1],q[2];
...
```

We are open to collaborate!

Hosting/sending students/researchers is always welcome.

Contact me at mitarai.kosuke.es@osaka-u.ac.jp