QuFEM: Fast and Accurate Quantum Readout Calibration Using the Finite Element Method

Siwei Tan, Liqiang Lu*, Hanyu Zhang, Jia Yu, Congliang Lang, Yongheng Shang, Xinkui Zhao, Mingshuai Chen, Yun Liang, and Jianwei Yin*

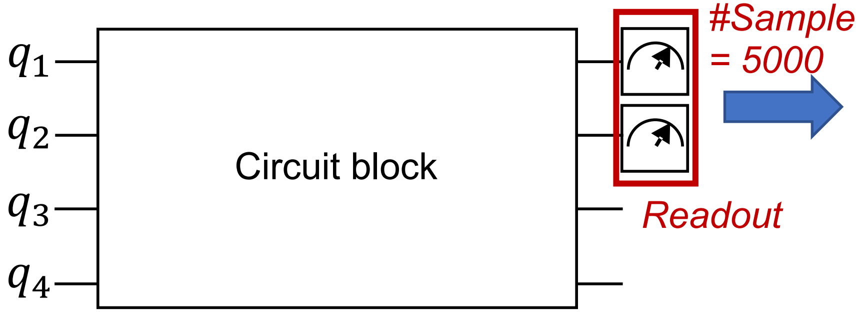
Reporter: Siwei Tan   Supervisor: Jianwei Yin, Liqiang Lu

College of Computer Science and Technology,
Zhejiang University

ASPLOS 2024

Background

Quantum readout is an operation to **read the information from quantum bits to classical bits.**

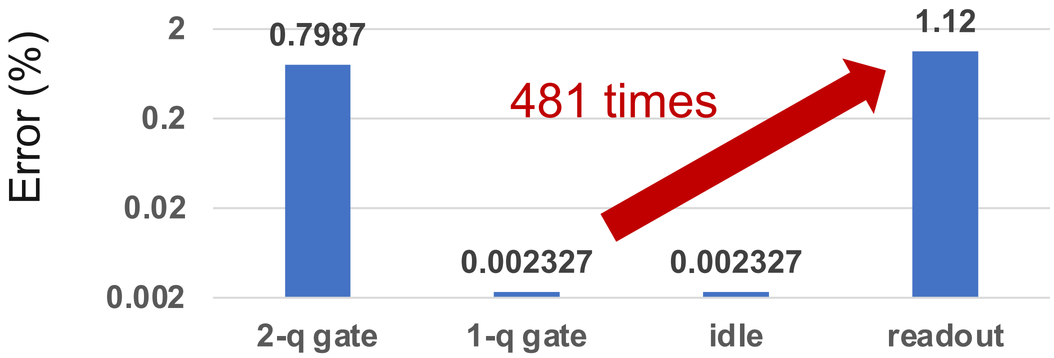


sample	1	2	3	4	4		4999	5000
q_1	0	1	1	1	0	…	1	0
q_2	1	0	1	1	1		1	1

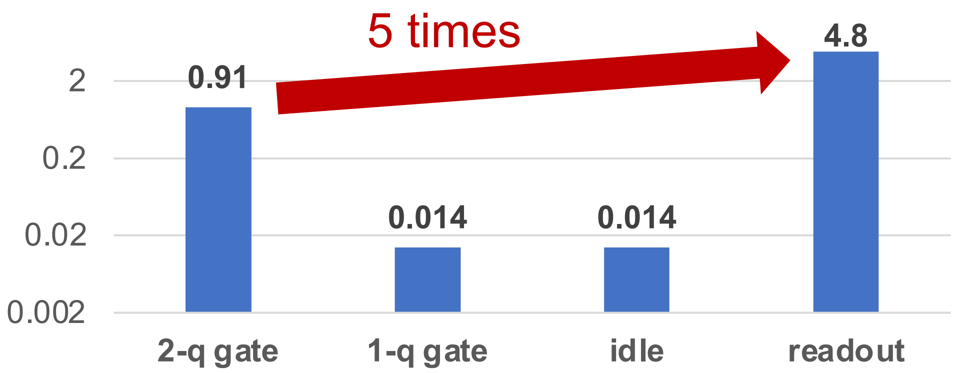
Measured bitstrings in different samples

Probability distribution

Readout error is significant on current quantum hardware.



Noise on 127-qubit IBM Sherbrooke quantum device

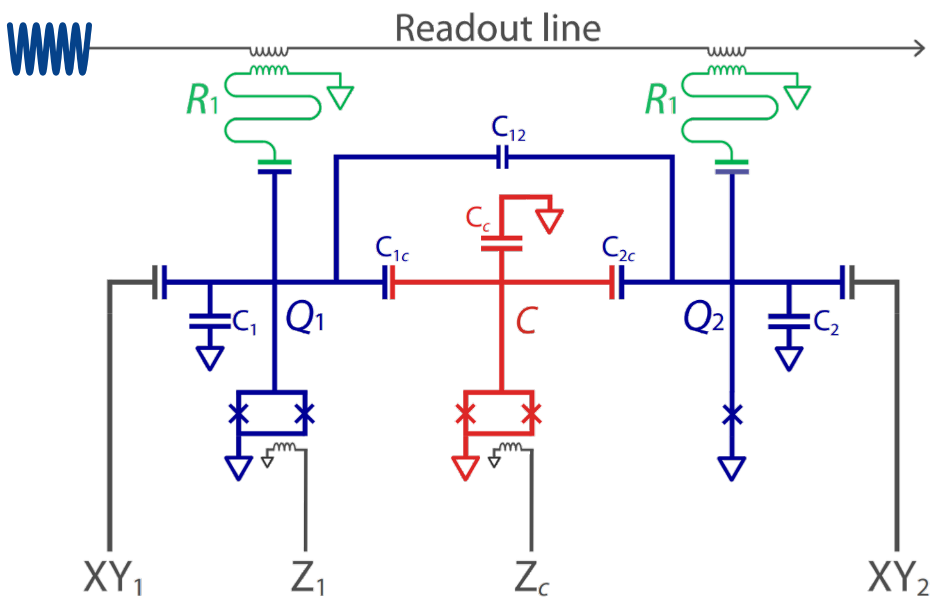


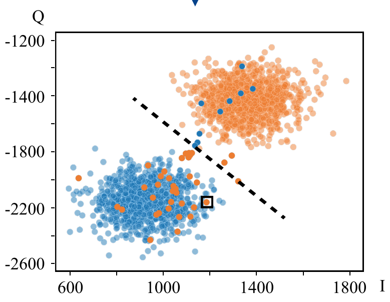
Noise on 10-qubit Tianmu quantum device

Background

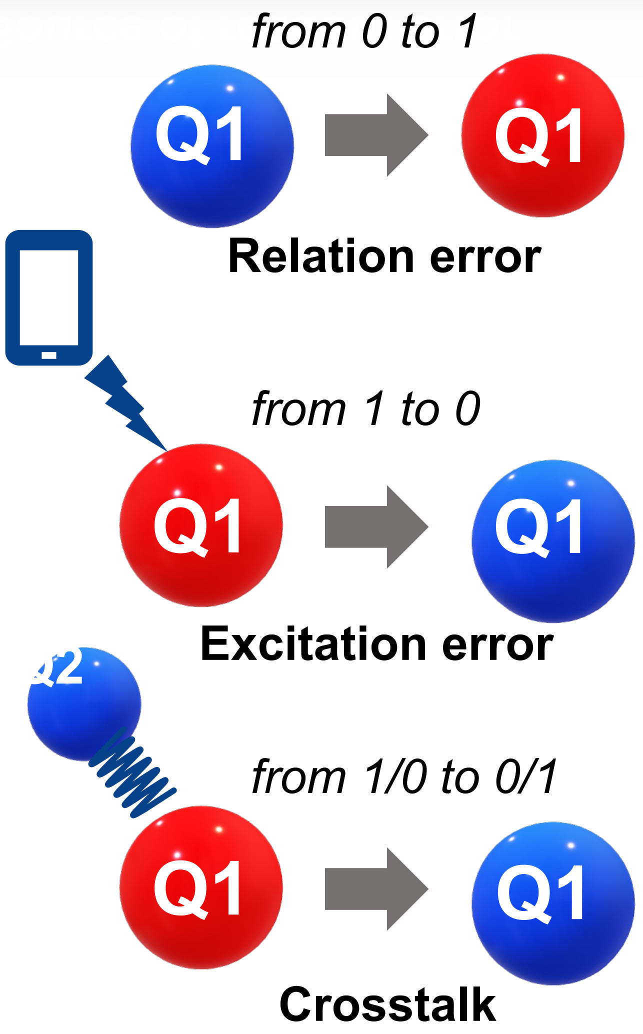
Implementation of readout on superconducting qubits

(1) Pulse in *(2) Pulse takes the information out.* *FFT*



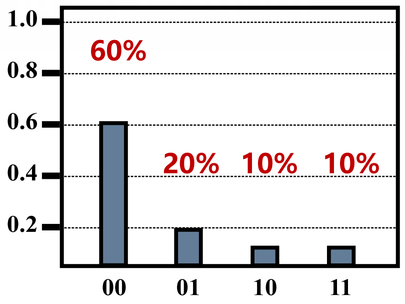


Source of readout error



Background

Ideal readout

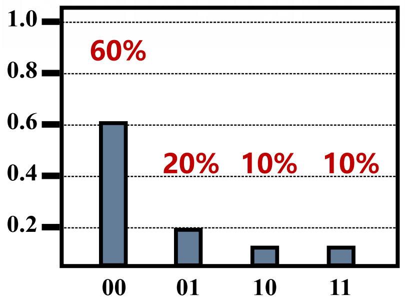


State vector

$$\begin{bmatrix} 0.6 \\ 0.2 \\ 0.1 \\ 0.1 \end{bmatrix}$$

Ideal distribution
(ideal program output)

Readout with noise

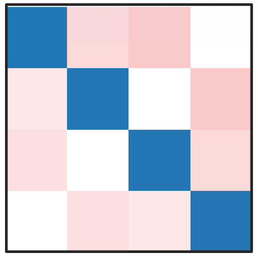


State vector

$$\begin{bmatrix} 0.4 \\ 0.1 \\ 0.09 \\ 0.32 \end{bmatrix}$$

Noisy distribution
(noisy program output)

Matrix-based readout error calibration

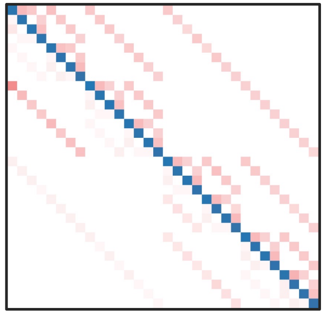


$$\cdot \begin{bmatrix} 0.4 \\ 0.1 \\ 0.09 \\ 0.32 \end{bmatrix} = \begin{bmatrix} 0.6 \\ 0.2 \\ 0.1 \\ 0.1 \end{bmatrix}$$

Calibration matrix

Noisy distribution

Calibrated distribution



Calibration matrix of a 5-qubit readout

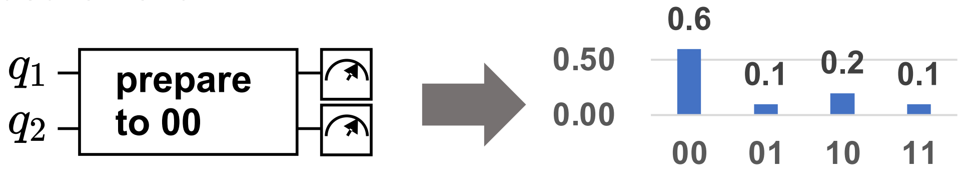
$2^5 \times 2^5$

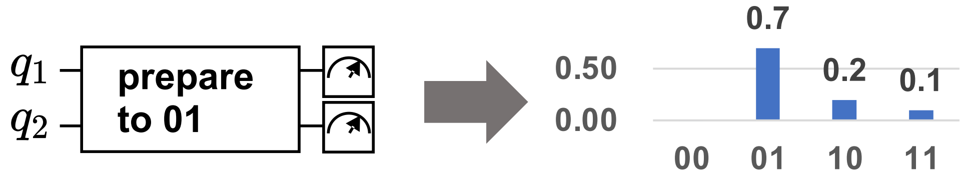
The size exponentially increases!

Basic Matrix-based readout calibration

Step 1. Matrix characterization

Prepares qubits to different basis states and apply measurement.



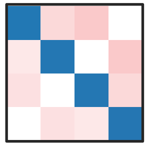


Fill in a noise matrix.

$$M = \begin{bmatrix} 0.6 & 0.1 & 0.2 & 0.1 \\ 0 & \boxed{0.7} & 0.2 & 0.1 \\ 0.2 & 0.2 & 0.6 & 0 \\ 0 & 0.1 & 0.1 & 0.8 \end{bmatrix}$$

Inverse the noise matrix

$$M^{-1} =$$



calibration matrix

Step 2. Calibration for any input

Represent the measured distribution as a vector.

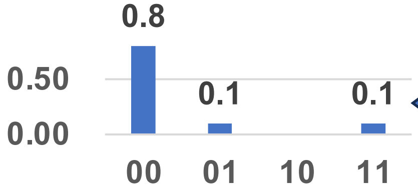
$$\begin{bmatrix} 0.6 \\ 0.1 \\ 0.2 \\ 0.1 \end{bmatrix}$$

数字算下

Apply matrix-vector multiplication.

$$\begin{bmatrix} 0.6 & 0.1 & 0.2 & 0.1 \\ 0 & 0.7 & 0.2 & 0.1 \\ 0.2 & 0.2 & 0.6 & 0 \\ 0 & 0.1 & 0.1 & 0.8 \end{bmatrix}^{-1} \cdot \begin{bmatrix} 0.6 \\ 0.1 \\ 0.2 \\ 0.1 \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0.1 \\ 0 \\ 0.1 \end{bmatrix}$$

$$M^{-1} \cdot P_{\text{read}} = P_{\text{calibrated}}$$



Basic Matrix-based readout calibration

Step 1. Matrix characterization

Prepares qubits to different basis states and apply measurement.

2^N circuits are executed to measure qubits on all basis states.

Fill in a noise matrix.

The size of the noise matrix is $2^N \times 2^N$.

Inverse the noise matrix

Calcaute the inverse has $\mathcal{O}(4^N)$ complexity.

Step 2. Calibration for any input

Represent the measured distribution as a vector.

Transformation has linear complexity.

Apply matrix-vector multiplication.

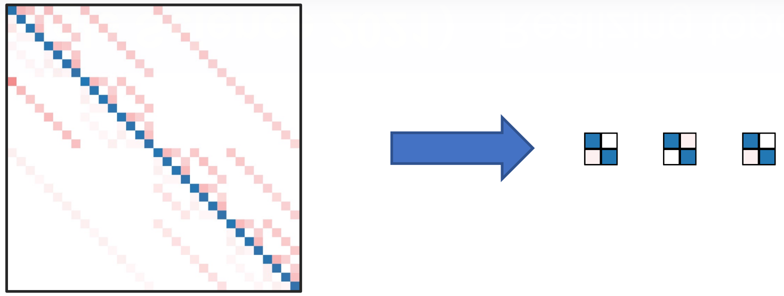
Multiplication has $\mathcal{O}(4^N)$ complexity.

8.8 TB and 2.3 min for a 18-qubit calibration

数字算下

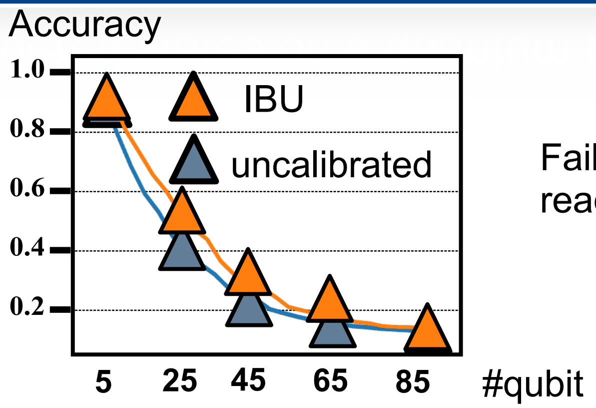
Limitations of Current Methods

IBU (Google Science 2021) Realizing topologically ordered states on a quantum processor.



Real calibration matrix   Single-qubit matrix

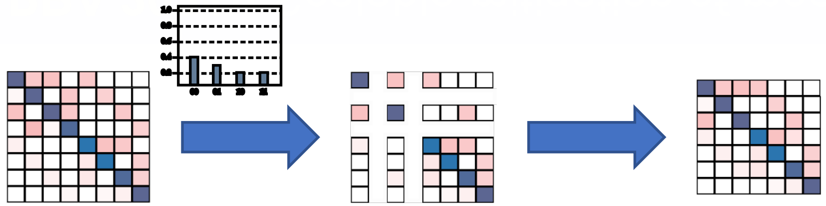
use tensor product of a series of single-qubit meta-matrices



Fail to calibrate on 80-qubit readout output

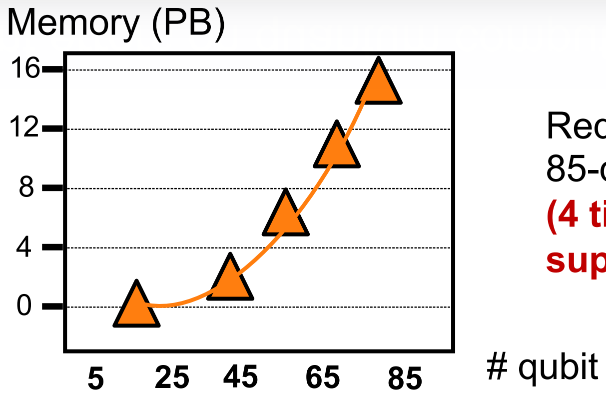
Fast but not accurate: ignore the qubit interactions.

M3 (IBM PRA 2021): Scalable mitigation of measurement errors on quantum computers



Before pruning   Pruning based on program output   After pruning

use a sparsity-aware method prune the matrix under a threshold of Hamming distance



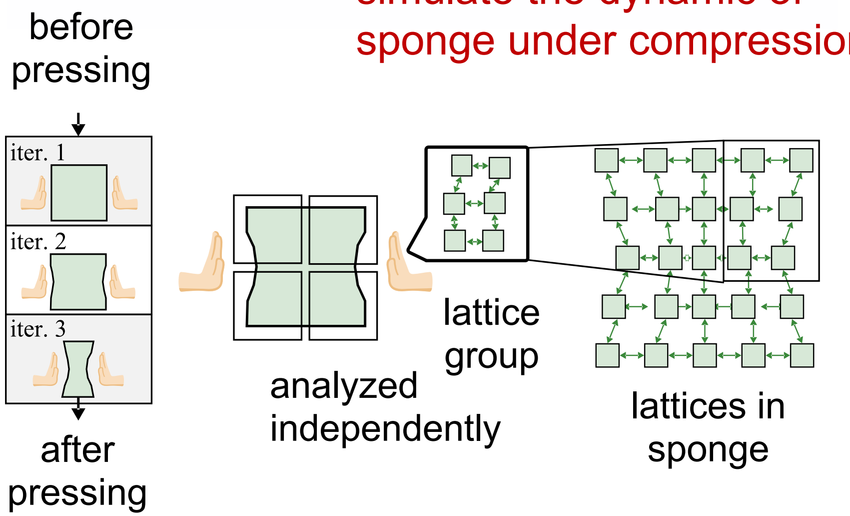
Require 16PB to calibrate a 85-qubit result.
(4 times the Fugaku supercomputer)

Accurate but not fast: many matrix elements cannot be ignored

Calibration based on Finite Element method

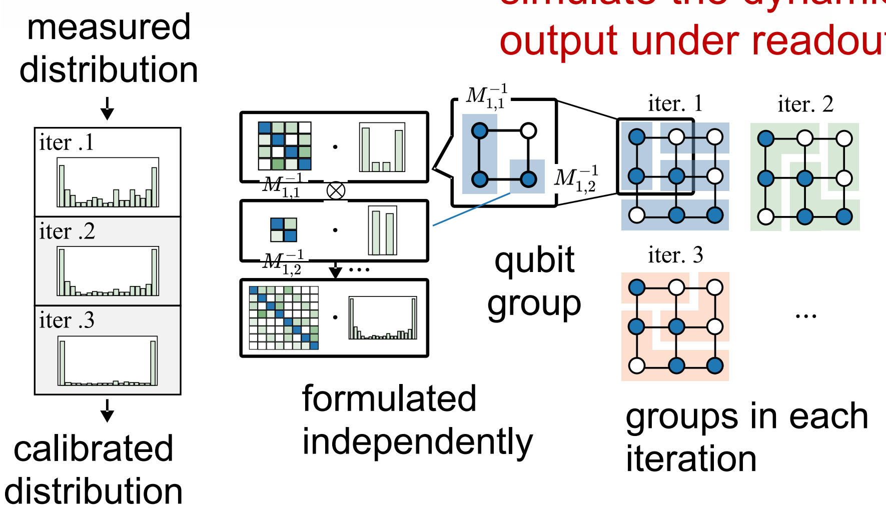
Classical Finite Element Method

simulate the dynamic of sponge under compression



Quantum Finite Element Method

simulate the dynamic of qubit output under readout noise



① partitions the sponge into lattices
② analyzes the state of each lattice independently
③ simulate the interaction
④ update the state of sponge

① partitions qubits into groups
② analyze the noise in each group independently
③ simulate the interaction
④ update the calibration result of qubits

A **divide-and-conquer strategy** to calibrate measured distribution

Calibration formulation

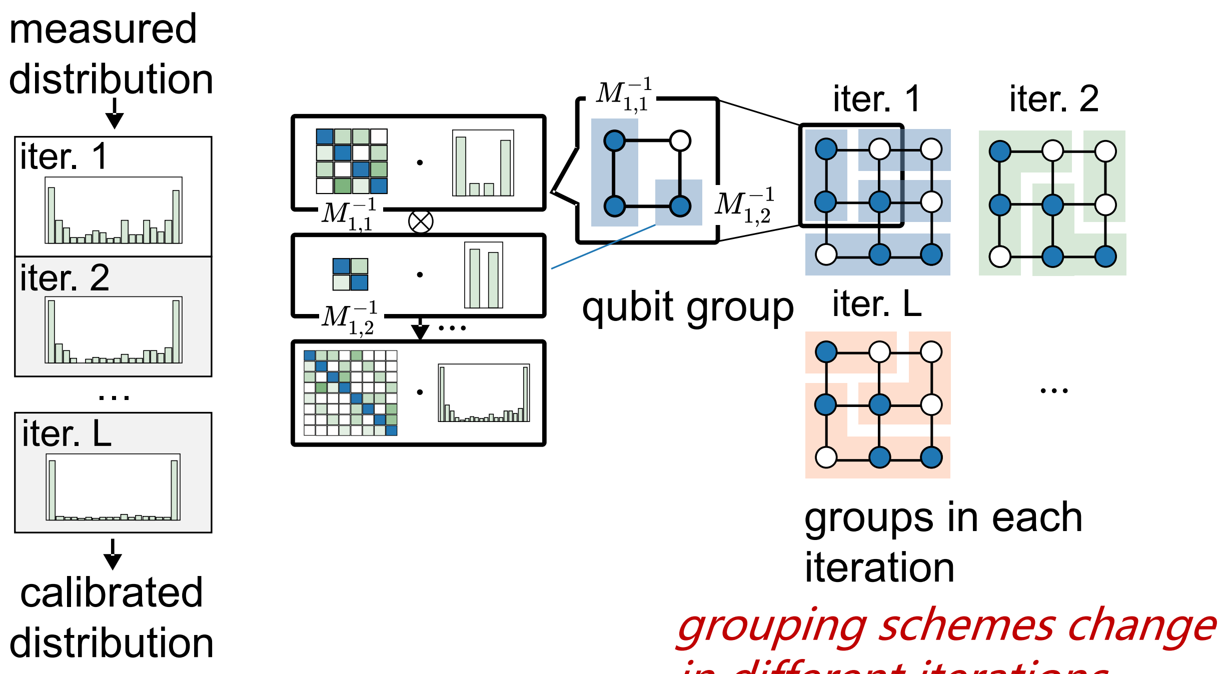
QuFEM reformulates the calibration as an iterative process with a series of sub-noise matrices.

Iter. 1: $P_2 = (\overbrace{M_{1,1} \otimes M_{1,2} \otimes \cdots M_{1,k}}^{\text{sub-noise matrices}})^{-1} \overbrace{P_1}^{\text{measured distribution}}$

Iter. 2: $P_3 = (M_{2,1} \otimes M_{2,2} \otimes \cdots M_{2,k})^{-1} P_2$

…

Iter. L: $\underbrace{P_{L+1}}_{\text{calibrated distribution}} = (M_{L,1} \otimes M_{L,2} \otimes \cdots M_{L,k})^{-1} P_L$

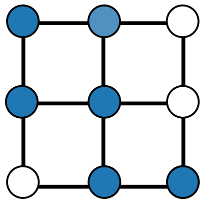


grouping schemes change in different iterations

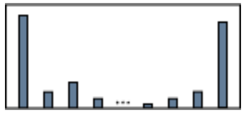
An example

Input :

measured qubits



measured distribution



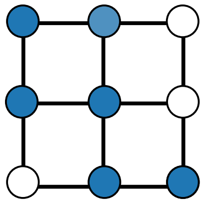
换成6比特的分布

直接写成拆开的形式

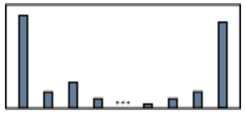
An example

Input :

measured qubits

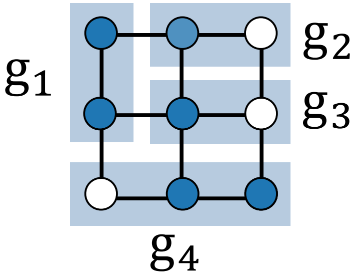


measured distribution

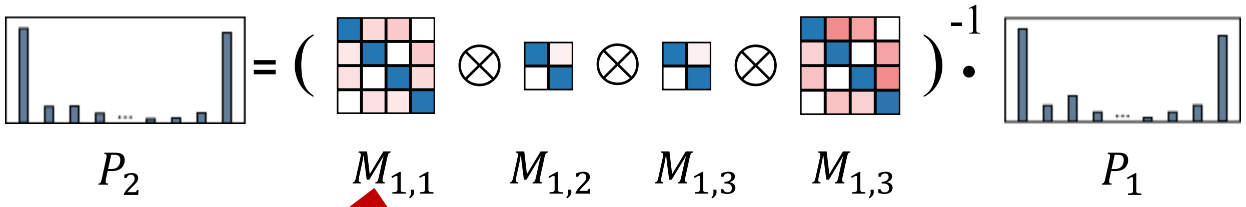


Iteration 1 :

grouping scheme



computation formulation

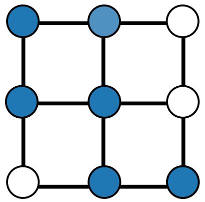


Matrices are generated according to the measured qubits.

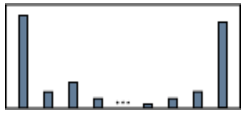
An example

Input :

measured qubits

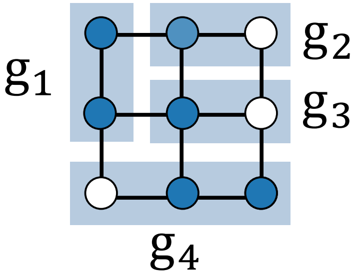


measured distribution

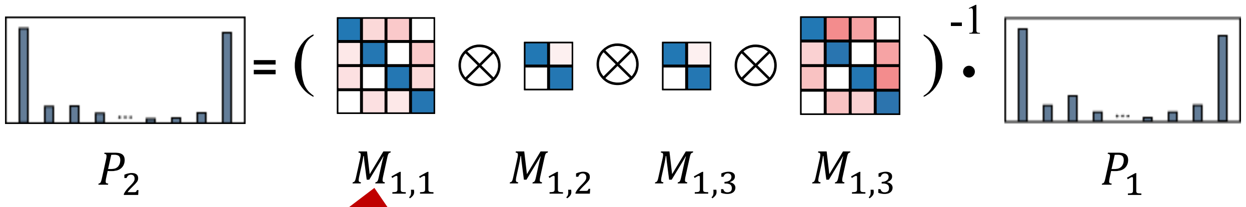


Iteration 1 :

grouping scheme



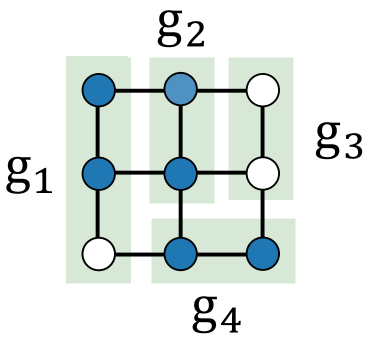
computation formulation



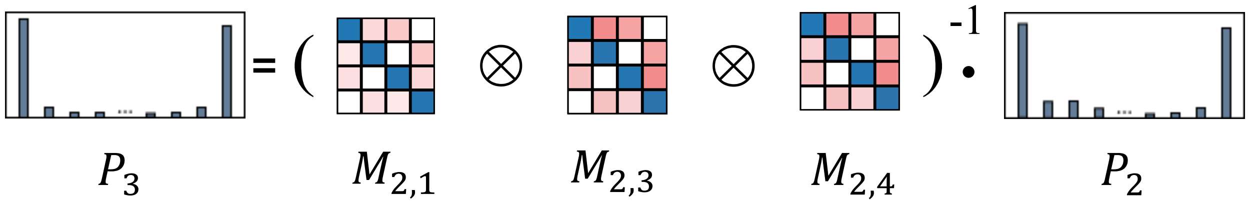
Matrices are generated according to the measured qubits.

Iteration 2 :

grouping scheme



computation formulation



Deployment on Specific Quantum Device

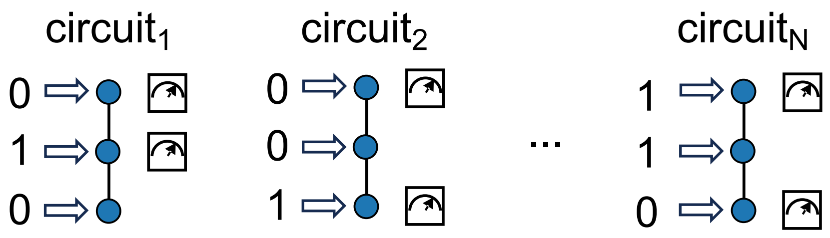
Characterize readout error in one iteration

Goals:

1. Collect the readout data under different qubit states.
2. Determine the grouping scheme in each iteration.

1. Data collection

Run benchmarking circuits.



Prepare qubits to different basis states and measure a subset of them.

2. Determine the grouping scheme

Possible states of a qubit in a benchmarking circuit:

- 1: qubit is set 0 and measured
- 2: qubit is set 1 and measured
- 3: qubit is set 0 or 1 and not measured

Characterize the **readout error** of a qubit under different states:

$$P(\underbrace{q_i.\mathrm{error} = 1}_{q_i \text{ is error}} \mid \underbrace{q_i.\mathrm{state} = x}_{q_j \text{ is set } x},\ \underbrace{q_j.\mathrm{state} = y}_{q_j \text{ is set } y})$$

$$y \in \{1,2,3\}\ ,\ x \in \{1,2,3\}$$

给个例子

要不直接简写，然后说key insight

直接说iteration

Deployment on Specific Quantum Device

Characterize readout error in one iteration

2. Determine the grouping scheme

Possible states of a qubit in a benchmarking circuit:

- 0: qubit is set 0 and measured
- 1: qubit is set 1 and measured
- 2: qubit is set 0 or 1 and not measured

Characterize the **readout error** of a qubit under different states:

$$P(\underbrace{q_i.\text{error} = 1}_{q_i \text{ is error}} \mid \underbrace{q_i.\text{state} = x}_{q_j \text{ is set x}},\ \underbrace{q_j.\text{state} = y}_{q_j \text{ is set y}})$$

$$y \in \{0,1,2\}\ ,\ x \in \{0,1\}$$

给个例子

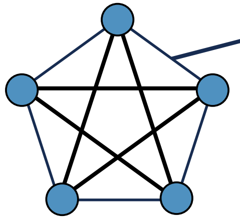
Characterize the **interaction** from one qubit to another qubit under different states:

$$\text{interact}(q_i.\text{state} = x \ \rightarrow\ q_j.\text{state} = x)$$

$$= \underbrace{P(q_j.\text{error} = 1 \mid C1, C2)}_{\text{error rate of } q_j \text{ under } C1, C2} - \underbrace{P(q_j.\text{error} = 1 \mid C2)}_{\text{average error rate of } qz_j}$$

$$C1:\ q_i.\text{state} = x, \qquad C2:\ q_j.\text{state} = y$$

Construct a **weighted qubit graph**:



labeled by the sum of interactions under all states

Deployment on Specific Quantum Device

Characterizing readout error in one iteration

2. Determine the grouping scheme

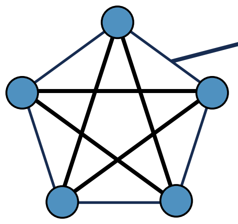
Characterize the **interaction** from one qubit to another qubit under different states:

$$\text{interact}(\mathrm{q_i.state} = x \rightarrow \mathrm{q_j.state} = x)$$

$$= \underbrace{\mathrm{P}(\mathrm{q_j.error} = 1 \mid C1, C2)}_{\text{error rate of } \mathrm{q_j} \text{ under } C1, C2} - \underbrace{\mathrm{P}(\mathrm{q}_j\mathrm{.error} = 1 \mid \mathrm{C2})}_{\text{average error rate of } \mathrm{qz_j}}$$

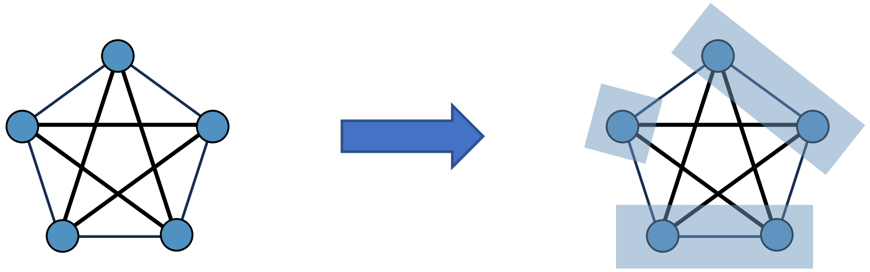
$\mathrm{C1:\ q_i.state = x}$, $\mathrm{C2:\ q_j.state = y}$

Construct a **weighted qubit graph**:



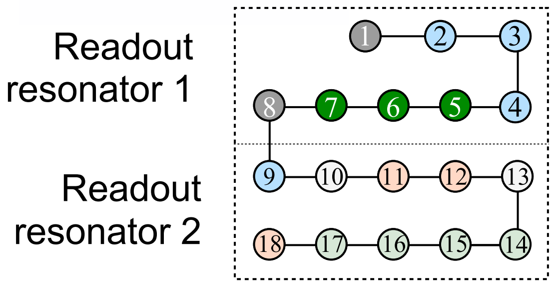
labeled by the sum of interactions under all states

Partitions with a **MAX-CUT solver**:



An Example

Prior knowledge of hardware helps grouping



18-qubit topology

group 1: 14 15 16 17
same readout resonator

group 2: 2 3 4 9
similar readout frequency

group 3: 1 8

overlapping frequency shift region

Deployment on Specific Quantum Device

Calibration in one iteration

Perform matrix-vector multiplication

$$\text{Iter. i: } P_{i+1} = (M_{i,1} \otimes M_{i,2} \otimes \cdots M_{i,k})^{-1} P_i$$

Matrix generation

Noise matrix formulates the transformation probability from the ideal state to measured state.

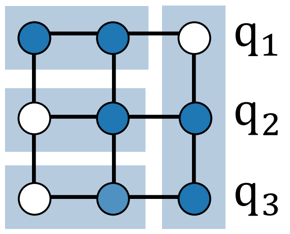
$$\begin{bmatrix} 0.6 & 0 & 0.1 & 0 \\ 0.1 & 0.7 & 0.2 & 0.1 \\ 0.2 & 0.2 & 0.6 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.8 \end{bmatrix} =$$

set state

read state

	00	01	10	11
00	0.6	0	0.1	0
01	0.1	0.7	0.2	0.1
01	0.2	0.2	0.6	0.1
11	0.1	0.1	0.1	0.8

Sub-noise matrices of QuFEM formulates the transformation probability of states inside the qubit groups.



set state of q_2, q_3

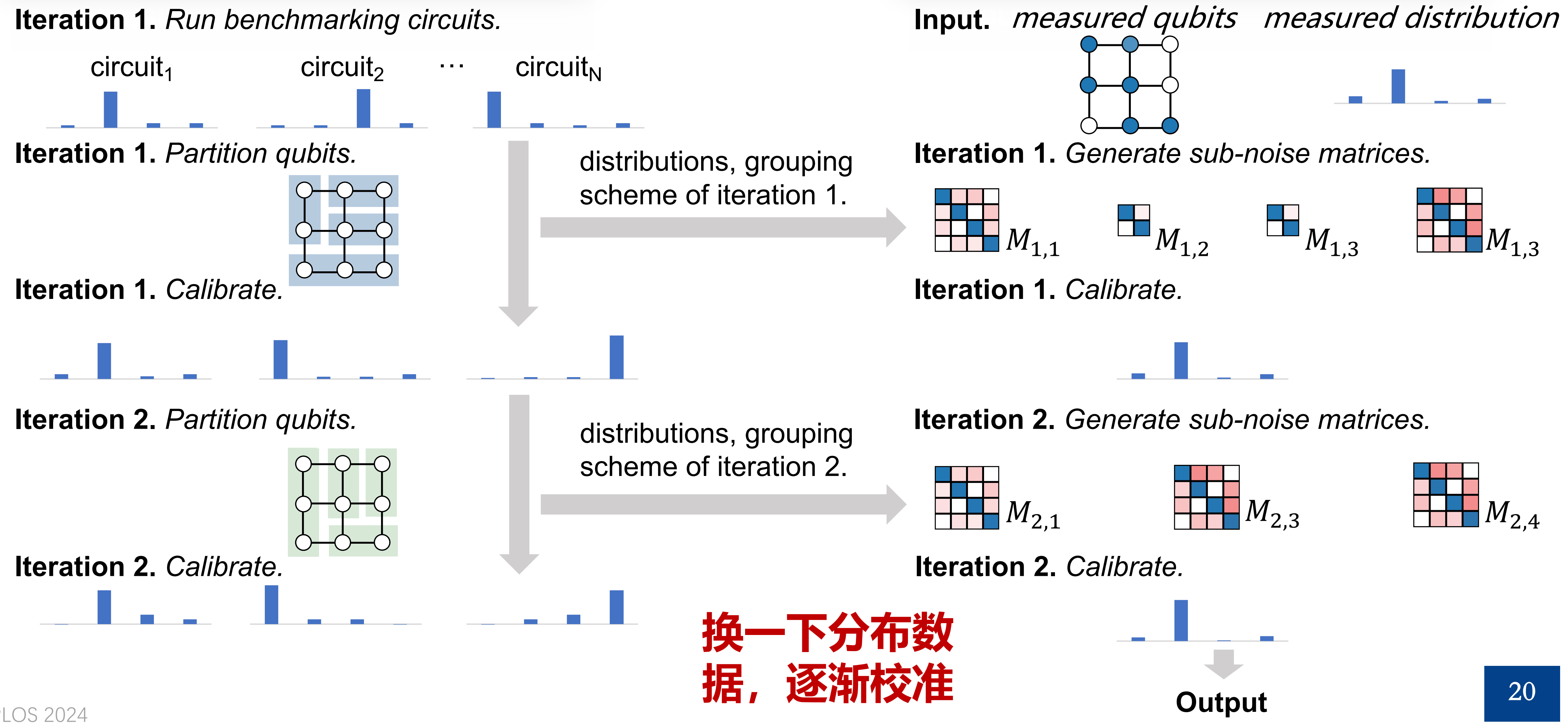
read state of q_2, q_3

	00	01	10	11
00	⋯	⋯		
01	⋯	⋯		
01	⋯	⋯		
11	⋯	⋯		

$$M[x][y] = P(\{q_2, q_3\}.\text{read} = x \mid \{q_2, q_3\}.\text{set} = y, \boxed{q_1 = 2})$$

Transformation probability when q_1 is not measured

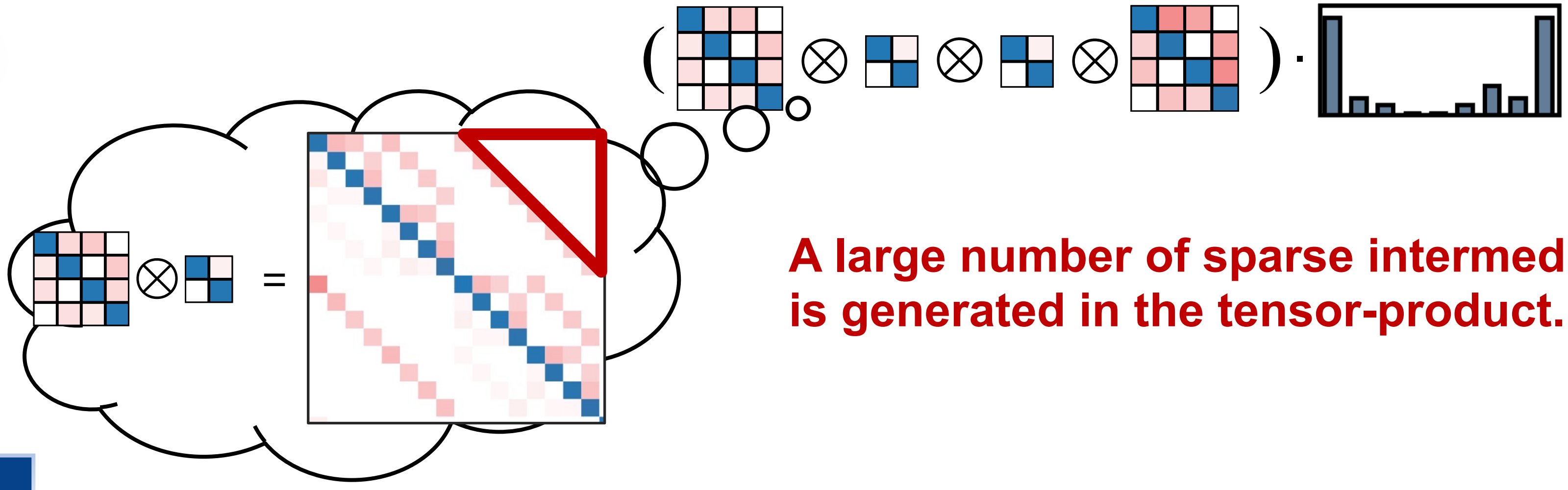
Put all together



换一下分布数据，逐渐校准

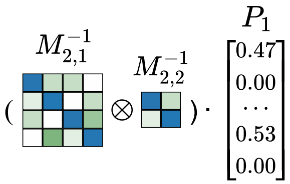
Sparse Tensor-Product Engine

Observation



A large number of sparse intermediate vectors is generated in the tensor-product.

Implementation



target equation

Use a key-value table to store sparse vector

x	*prob.*
$P_1(000)$	0.47
$P_1(011)$	0.53

① →

	value
00	0.50
01	-0.02
10	0.01
11	0.01

Calculate the tensor product

② ⊗

	value
0	0.99
1	0.01

· 0.47 =

	value
000	0.49
001	0.01
010	-0.01
100	0.01
~~101~~	~~10^{-4}~~
~~110~~	~~10^{-4}~~

...

③ $|value| < \beta$

Aggregate the tensor-product result

④ → $\sum$ →

x	*prob.*
$P_2(000)$	0.48
$P_2(001)$	6×10^{-3}
$P_2(010)$	6×10^{-3}
$P_2(011)$	0.50
$P_2(111)$	6×10^{-3}

Prune values < threshold (10^{-5})

For each basis states
① calculate the matrix-vector multiplication
② calculate the tensor-product
③ prune intermediate values
④ sum intermediate values to obtain output.

Compute the tensor-product of other basis states

Experiment

Setup

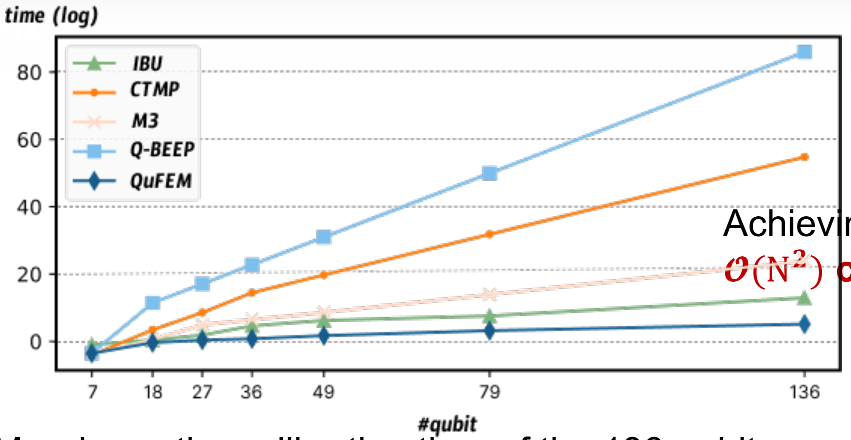
Platform	#Qubits	1-q fidelity	2-q fidelity	Instructions
Quafu	136	94.6±3.1%	94.6±3.0%	ID,RX,RY,RZ,H,CX
	18	95.9±1.3%	95.9±1.3%	ID,RX,RY,RZ,H,CX
Rigetti	79	99.5±1.1%	90.0±6.4%	CPHASE,XY
Self-developed	36	99.9±0.1%	98.7±0.8%	U3,CZ
IBMQ	7	99.9±0.1%	99.2±0.1%	CX,ID,RZ,SX,X

Evaluated hardware

IBU: KJ Satzinger, et al. Realizing topologically ordered states on a quantum processor. Science 2021
CTMP: Sergey, et al. Mitigating measurement errors in multiqubit experiments. PRA 2021.
M3: Paul D Nation , et al. Scalable mitigation of measurement errors on quantum computers. PRX Quantum 2021.
Q-BEEP: Nathan Wiebe, et al. Qbeep: Quantum Bayesian error mitigation employing Poisson modeling over the hamming spectrum. ISCA 2023.

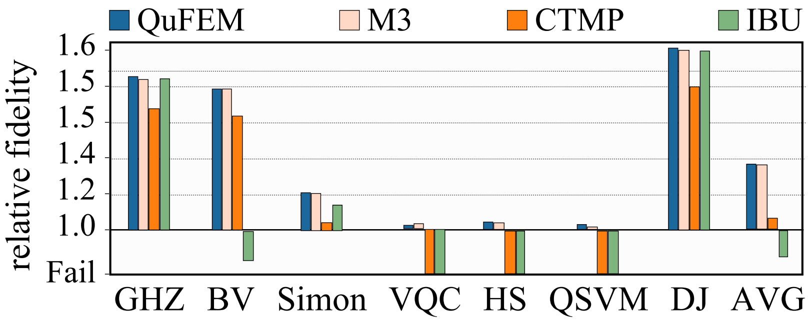
Baselines

Result



Achieving closed to $\mathcal{O}(N^2)$ **complexity**

QuFEM reduces the calibration time of the 136-qubit program output from **119.44 hours** (IBU) to **169.65 seconds** (**119.44 × reduction**).



QuFEM shows an average improvement in relative fidelity of **1.003×, 1.2×,** and **1.4×** compared to M3, CTMP, and IBU, respectively.

Thanks for listening

QuFEM: Fast and Accurate Quantum Readout Calibration Using the Finite Element Method

Siwei Tan, Liqiang Lu*, Hanyu Zhang, Jia Yu, Congliang Lang, Yongheng Shang, Xinkui Zhao, Mingshuai Chen, Yun Liang, and Jianwei Yin*