

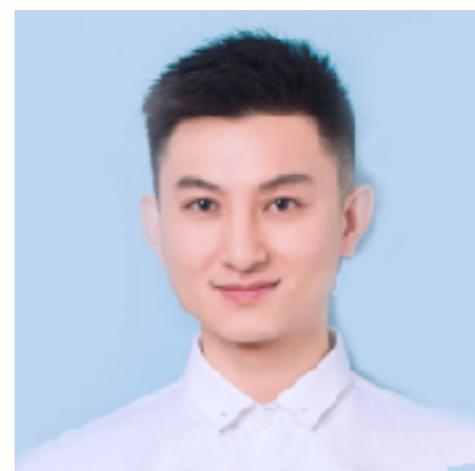


Simulating Noisy Variational Quantum Algorithms

Yuguo Shao (Tsinghua)

Joint work with Fuchuan Wei, Song Cheng and Zhengwei Liu

2023.9.20 HKUST



What is quantum algorithms?

Algorithms of quantum computers - a new computing paradigm based on the essential laws of quantum mechanics

Shor's Algorithm

- 1994 , Peter Shor of Bell Labs discovered the Shor's algorithm that can quickly complete the decomposition of large numbers: Knowing N , and it is known that it is the product of two large prime numbers p and q , what are p and q ?
- The optimal classical algorithm requires an exponential cost
- The intractability of this problem is the cornerstone of many commercial cryptosystems, such as RSA.

with a classical computer

# bits	1024
factoring in 2006	10^5 years
factoring in 2024	38 years
factoring in 2042	3 days

2048	4096
5×10^{15} years	3×10^{29} years
10^{12} years	7×10^{25} years
3×10^8 years	2×10^{22} years

with potential quantum computer
(e.g., clock speed 100 MHz)

# bits	1024	2048	4096
# qubits	5124	10244	20484
# gates	3×10^9	2×10^{11}	$X 10^{12}$
factoring time	4.5 min	36 min	4.8 hours

Physical Realization of Shor's Algorithms are extremely difficult

- 29 years since the Shor's algorithm was proposed in 1994, the implementation of a large-scale quantum computing of Shor's algorithm has not yet been achieved.
- In order to protect the internal quantum information, quantum computers should be isolated from environment as much as possible.
- In order to manipulate and read quantum information, quantum computers need to strongly connected with environment.
- The current quantum hardware has poor scalability, and it is difficult to quickly increase the scale of calculations

Shor's Algorithm

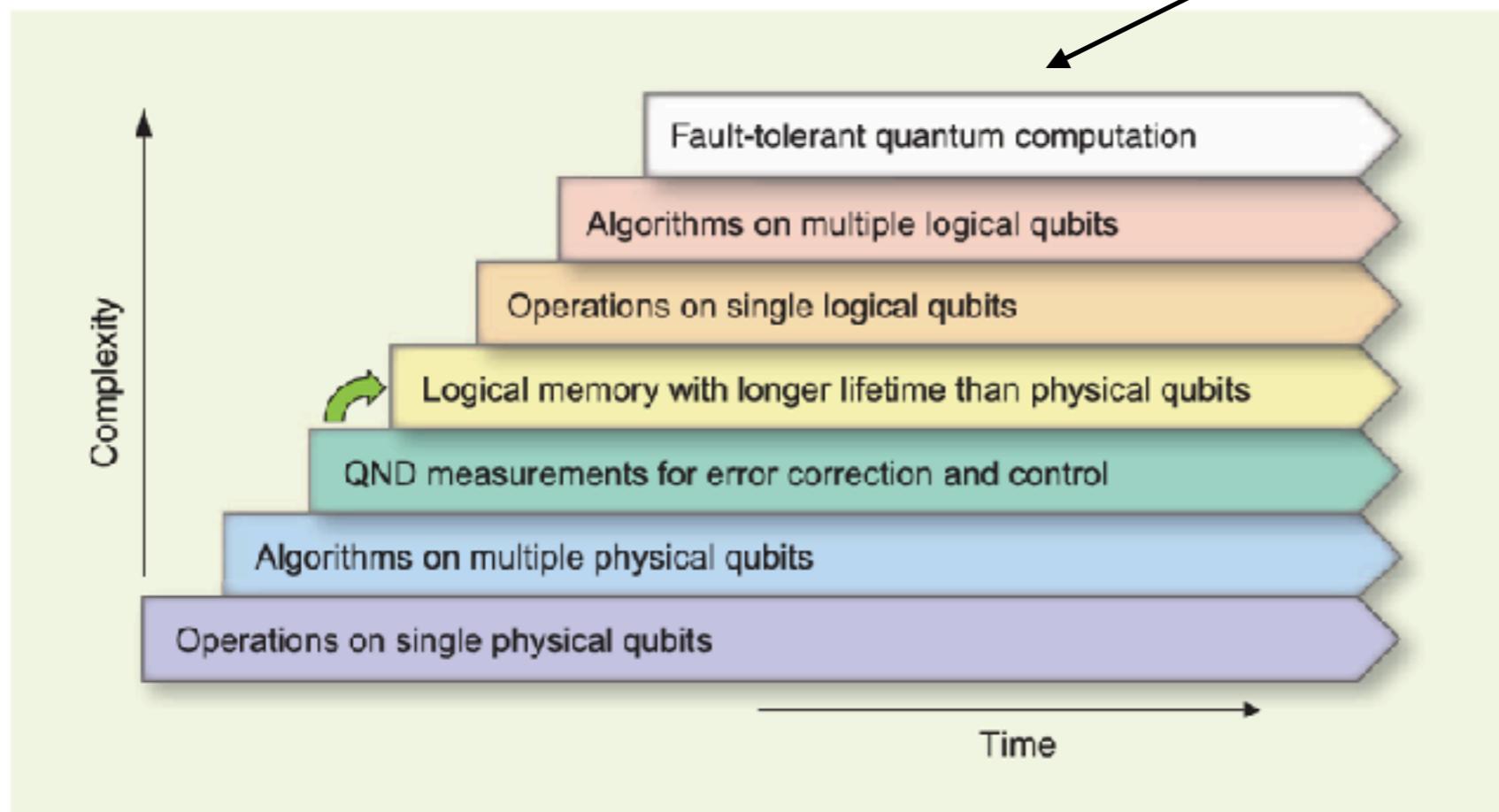


Fig. 1. Seven stages in the development of quantum information processing. Each advancement requires mastery of the preceding stages, but each also represents a continuing task that must be perfected in parallel with the others. Superconducting qubits are the only solid-state implementation at the third stage, and they now aim at reaching the fourth stage (green arrow). In the domain of atomic physics and quantum optics, the third stage had been previously attained by trapped ions and by Rydberg atoms. No implementation has yet reached the fourth stage, where a logical qubit can be stored, via error correction, for a time substantially longer than the decoherence time of its physical qubit components.

- Quantum information is extremely fragile

Weak noise is able to cause decoherence – information lost quickly

- Error correction is expensive

- Unknown quantum information cannot be cloned
- Measurements disturb unknown quantum information

Noisy Intermediate-Scale Quantum (NISQ) era

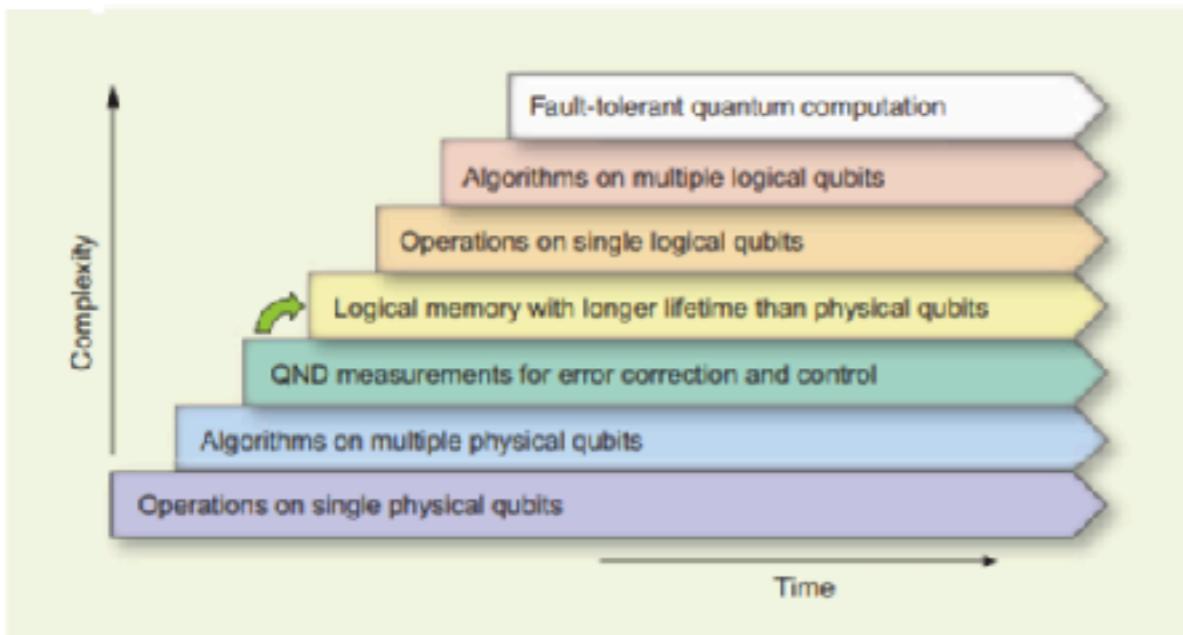
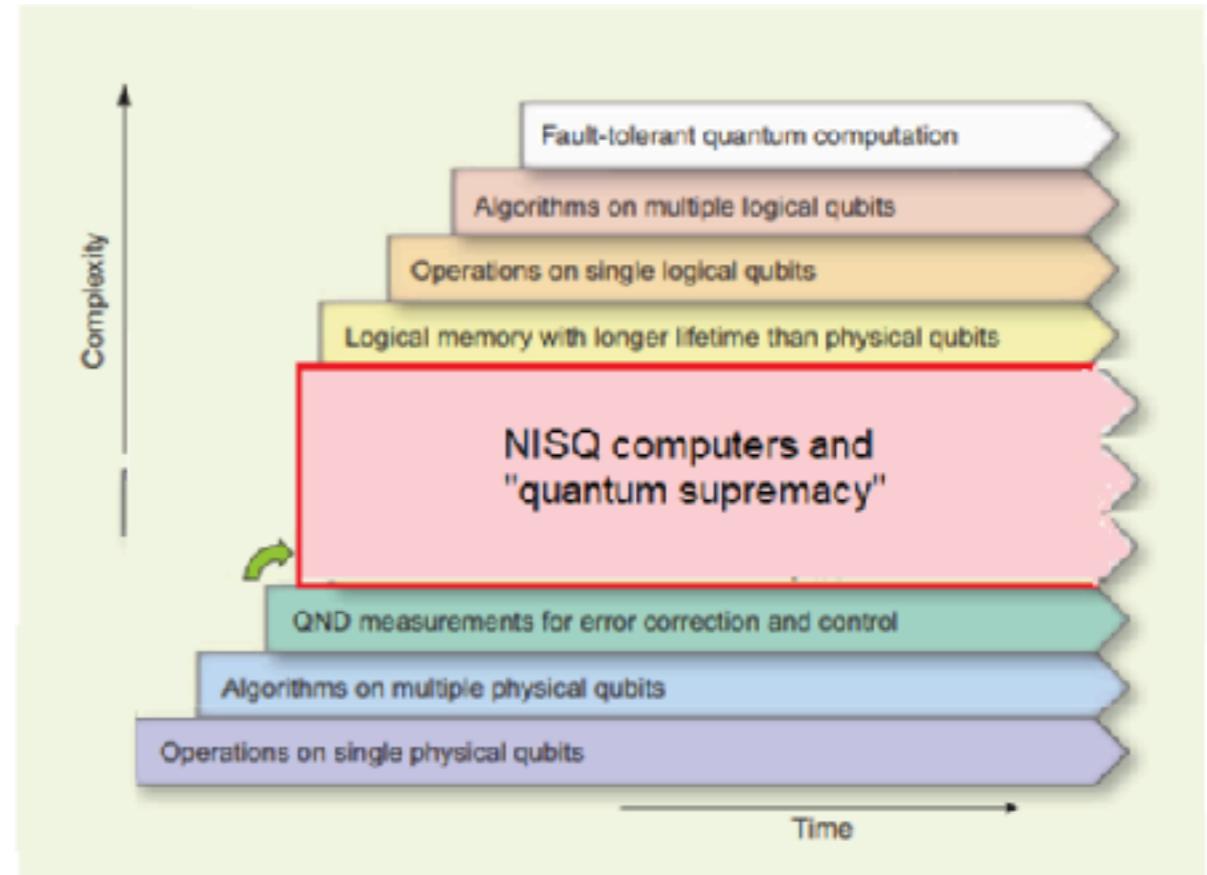


Fig. 1. Seven stages in the development of quantum information processing. Each advancement requires mastery of the preceding stages, but each also represents a continuing task that must be perfected in parallel with the others. Superconducting qubits are the only solid-state implementation at the third stage, and they now aim at reaching the fourth stage (green arrow). In the domain of atomic physics and quantum optics, the third stage had been previously attained by trapped ions and by Rydberg atoms. No implementation has yet reached the fourth stage, where a logical qubit can be stored, via error correction, for a time substantially longer than the decoherence time of its physical qubit components.

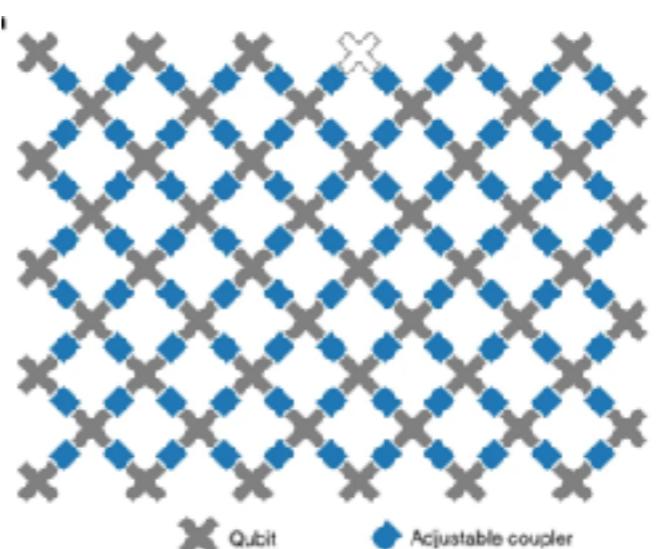


Quantum computation worldview 2010

Typical NISQ hardware

- ~100 qubits
- Imperfect **noisy** gate
- **local** two-qubit gate
- **shallow** circuit, considered of the decoherence time and error rate.

Quantum computation worldview 2020



**Consider showing quantum advantage on
easier tasks**

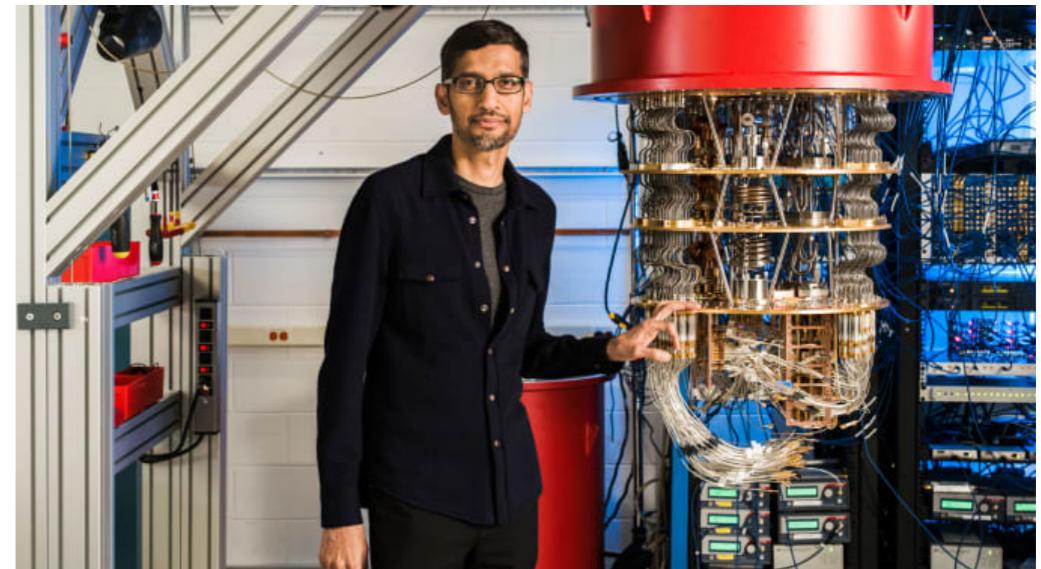
Quantum Superiority of Sampling Random Quantum State

- Demonstrating quantum computer could surpass the capabilities of classical computer, regardless of whether it's useful or not.

Quantum supremacy using a programmable superconducting processor

Frank Arute, Kunal Arya, [...] John M. Martinis [✉](#)

Nature 574, 505–510(2019) | [Cite this article](#)



In October 2019 , Google was the first to announce the realization of quantum superiority

- Sampling a random quantum circuit
- 3 minutes vs 10,000 years

Quantum supremacy & Classical simulators of (noiseless) quantum circuits

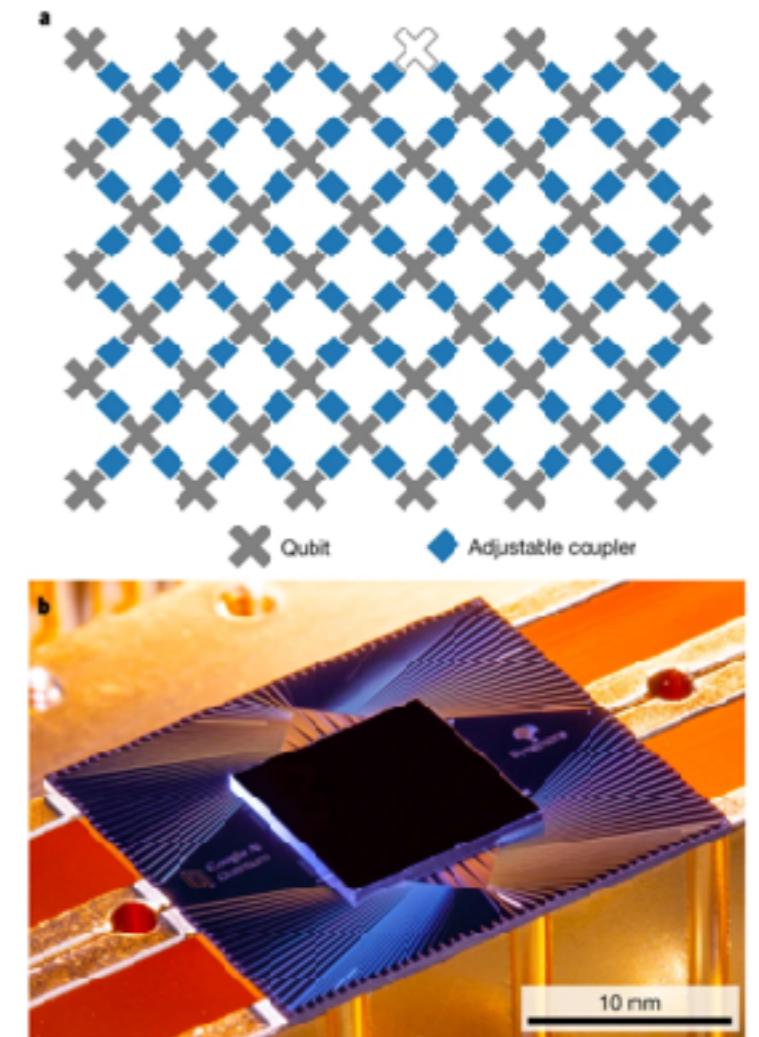
Quantum supremacy using a programmable superconducting processor

Frank Arute, Kunal Arya, [...] John M. Martinis 

Nature 574, 505–510(2019) | Cite this article

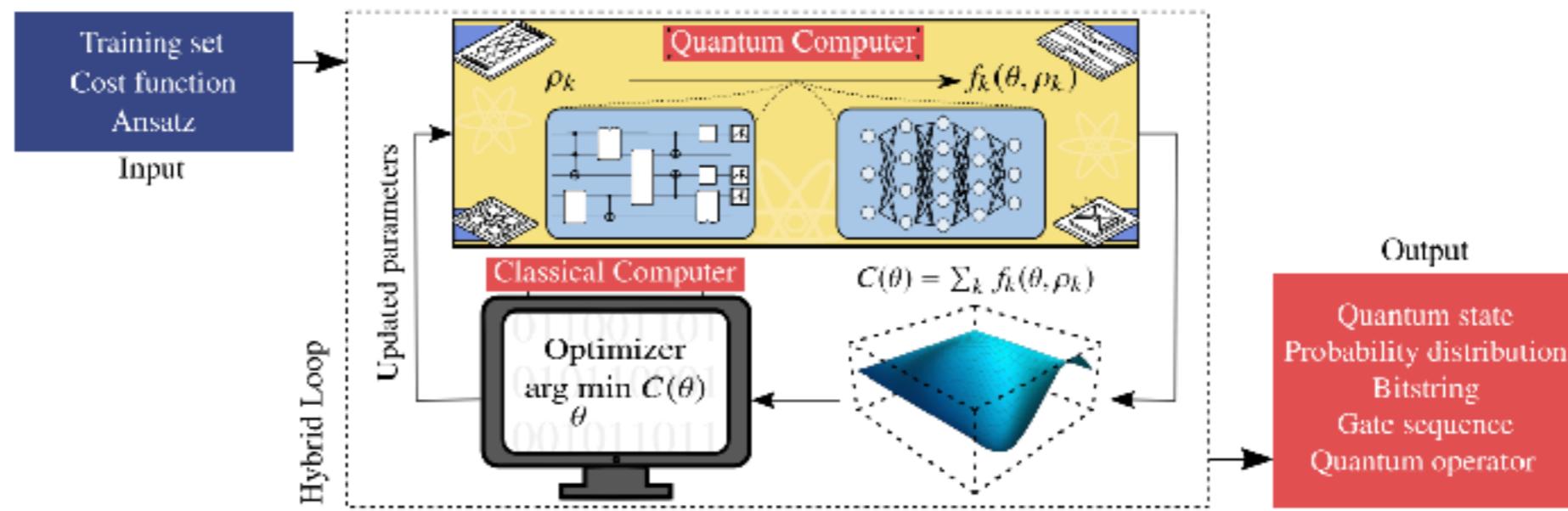
Our Sycamore processor takes about **200 seconds** to sample one instance of a quantum circuit a million times the equivalent task for a state-of-the-art classical supercomputer would take approximately **10,000 years**.

- IBM [Pednault et al 2019.12]:
250PB memory, 2.5 days for Summit
- Cotengra [Gray/kourtis 2020.03]:
3000 years / amplitude in one Quadro P2000 GPU
- Alibaba [Huang et al 2020.06]:
267 days / 64 amplitudes in one V100 GPU
- ITP CAS [Pan et al 2021.03]
149 days / 2^{21} amplitudes in one A100 GPU



Quantum Advantage in Noisy VQA

Variational Quantum Algorithms (VQA) framework



- Variational Ansatz (parametrized quantum circuits $U(\theta)$)
- Cost function for a set of functions $\{f_k\}$ and (possibly) a set of training data ρ_k : $C(\theta) = \sum f_k (\text{Tr}[O_k U(\theta) \rho_k U^\dagger(\theta)])$
- Quantum computer estimate the cost function, Classical computer update parameters to minimize $C(\theta)$.

What's the “specialty” for noisy VQA?

What's the “specialty” for noisy VQA?

- Still NISQ circuits
- Only care about expectation value, instead of state.
- Dominated by Pauli rotation gates.
- ...

Our contributions

- Introduce a faster and more accurate classical algorithm: OBPPP (observable's back-propagation on Pauli paths).
- Detailed theoretical analysis of OBPPP.
- Support firstly for deriving noisy outcomes from noise-free results, enabling us to accurately reproduce IBM's raw experimental observations.

Main Results in Theory

Given a **fixed error rate λ** of depolarizing noise, Pauli word observable H and input state $|0\rangle^{\otimes n}$. For arbitrary truncation error ε , there exists a polynomial-scale classical algorithm to approximate the observed value of quantum device, with a probability of at least $1 - \delta$ over all possible parameters θ .

The time complexity is **Poly($n, L, 1/\varepsilon, 1/\sqrt{\delta}, \|H\|_\infty$)**, where n is the number of qubits, and L is depth of noisy parameterized quantum circuit $U(\theta)$.

Prerequisite

- In typical VQAs, Quantum devices are used to estimate the mean $\langle H \rangle = \text{Tr}\{\langle 0| U(\theta)^\dagger H U(\theta) |0\rangle\}$.
- We consider circuit $U(\theta)$ consists of Pauli rotation gates $\exp -\frac{\theta}{2}\sigma$ where $\sigma \in P_n$ and Clifford gates in {H,S,CNOT}.
- For simplicity, we consider observable is a Pauli operator $H \in P_n$, and initial state is $|0\rangle$.

Remark

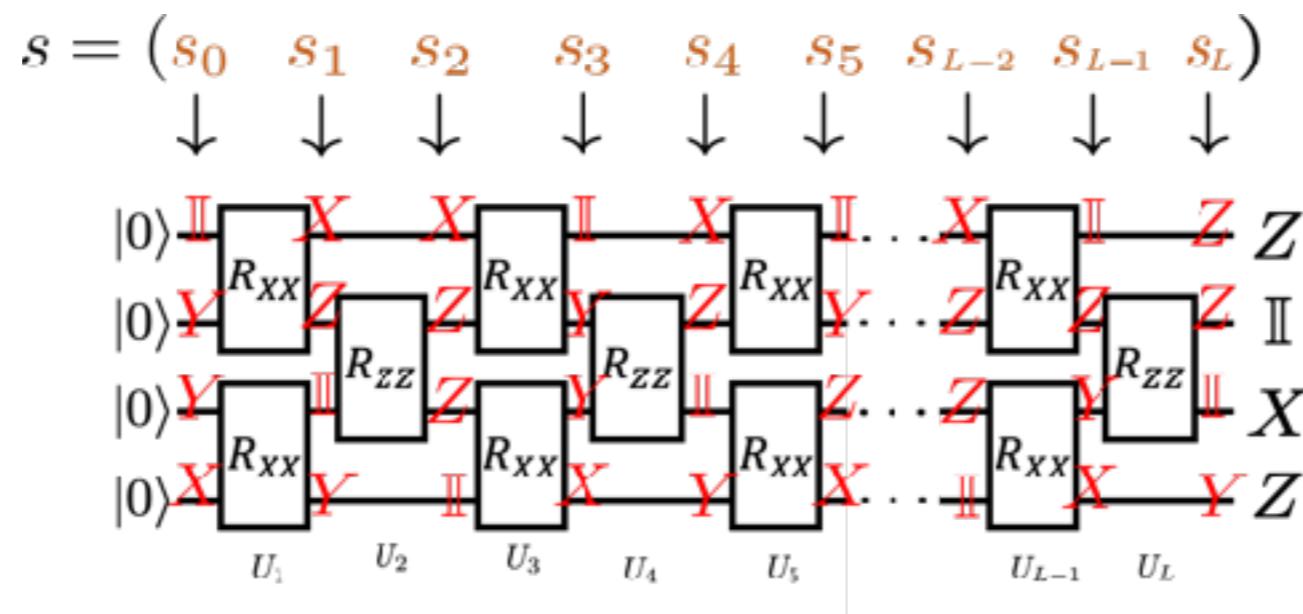
We discussed the following promotions in our work:

- Initial state is mixed state ρ , which has Poly(n) size non-zero elements.
- Observable H is linear combinations of Poly(n) size Pauli operators.

Pauli path integral

Map the calculation of expected values into computing path integrals in the Pauli basis.

Pauli path s for a L -depth circuit is $s = (s_0, \dots, s_L) \in P_n^{L+1}$.



$$\langle H \rangle = \text{Tr}\{\langle 0| U(\theta)^\dagger H U(\theta) |0\rangle\}$$

$$\begin{aligned}
 &= \overbrace{H \rightarrow U \rightarrow |0\rangle\langle 0| \rightarrow U^\dagger}^{\text{outer loop}} \\
 &= \overbrace{H \rightarrow \overline{U}_L \rightarrow \dots \rightarrow \overline{U}_1 \rightarrow |0\rangle\langle 0|}^{\text{inner loop}}
 \end{aligned}$$

Represented in Pauli path integrals

$$\begin{aligned}
 \langle H \rangle &= \langle 0|0\rangle = \overbrace{\dots}^{\bar{U}_L} \overbrace{\dots}^{\bar{U}_1} |0\rangle\langle 0| \\
 &= \sum_{s_0, \dots, s_L \in P_n} \langle 0|0\rangle = \sum_{s_0, \dots, s_L \in P_n} \underbrace{\dots}_{\substack{s_0 \\ s_1 \\ \vdots \\ s_L}} \overbrace{\dots}^{\bar{U}_L} \overbrace{\dots}^{\bar{U}_1} |0\rangle\langle 0| \\
 &= \sum_{s_0, \dots, s_L \in P_n} \underbrace{\langle H|s_L\rangle\langle s_L|U_L|s_{L-1}\rangle\dots\langle s_1|U_1|s_0\rangle\langle s_0|0\rangle}_{f(\theta, s, H)} \\
 &= \sum_{s \in P_n^{L+1}} f(\theta, s, H)
 \end{aligned}$$

- Each $f(\theta, s, H)$ can be calculated in time $O(nL)$.

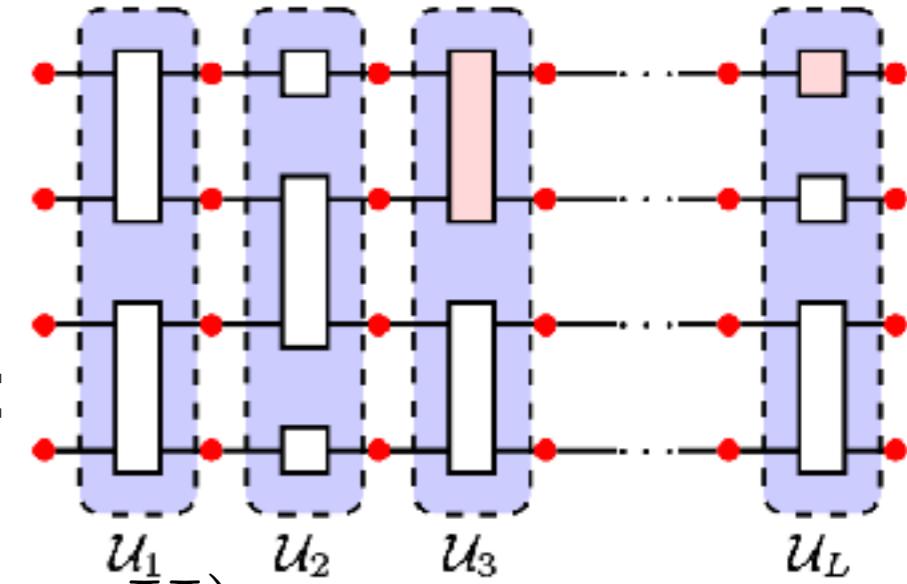
Single-qubit depolarizing noise

The noise channel can be modeled as : $\mathcal{N}(\phi) = (1 - \lambda)\phi + \lambda \frac{\text{Tr}(\phi)}{2} \mathbb{I}$

The noisy contribution \hat{f} can be characterized:

$$\hat{f}(\theta, s, H) = (1 - \lambda)^{|s|} f(\theta, s, H),$$

where $|s| = \sum_i |s_i|$ and $|s_i|$ denotes the Hamming weight (the number of non-identity elements) of s_i .

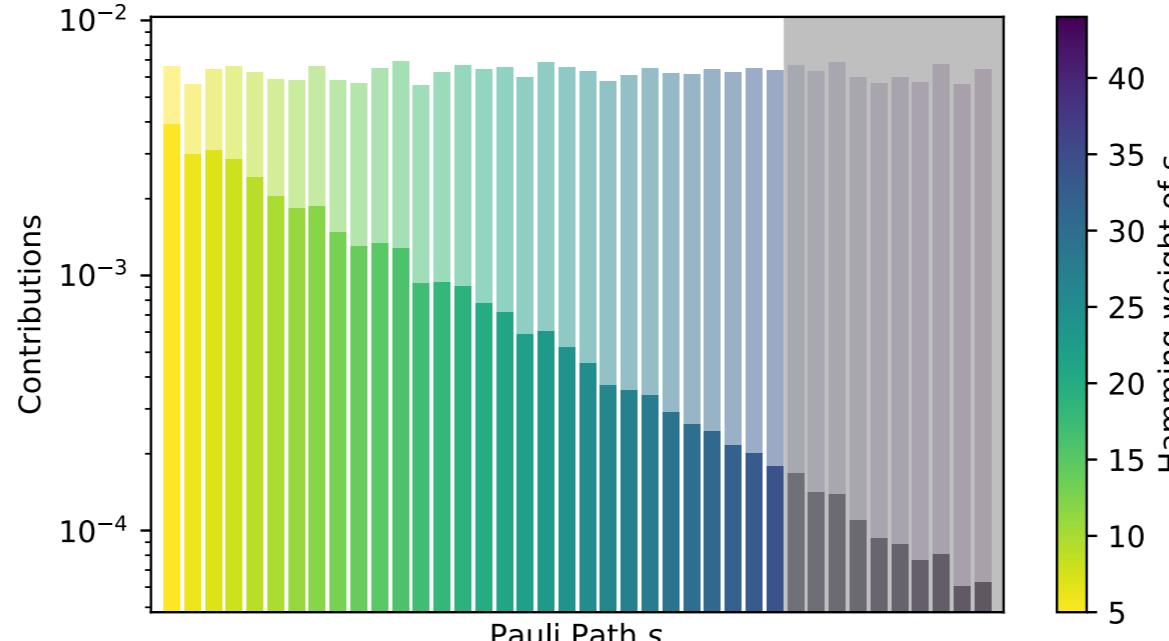


The noisy contributions of Pauli paths are suppressed exponentially with their Hamming weight.

- Our algorithm calculates the contributions of the Pauli path with Hamming weight $|s| \leq M$ to provide an approximation of $\langle H \rangle$, as

$$\sum_{|s| \leq M} (1 - \lambda)^{|s|} f(\theta, s, H).$$

- However, there are a total of $4^{n(L+1)}$ Pauli paths, fortunately **the vast majority of them are zero-contributing**. We use the OBPPP method to efficiently enumerate all Pauli paths with non-zero contributions and $|s| \leq M$.



Candidates for s_{i-1} with

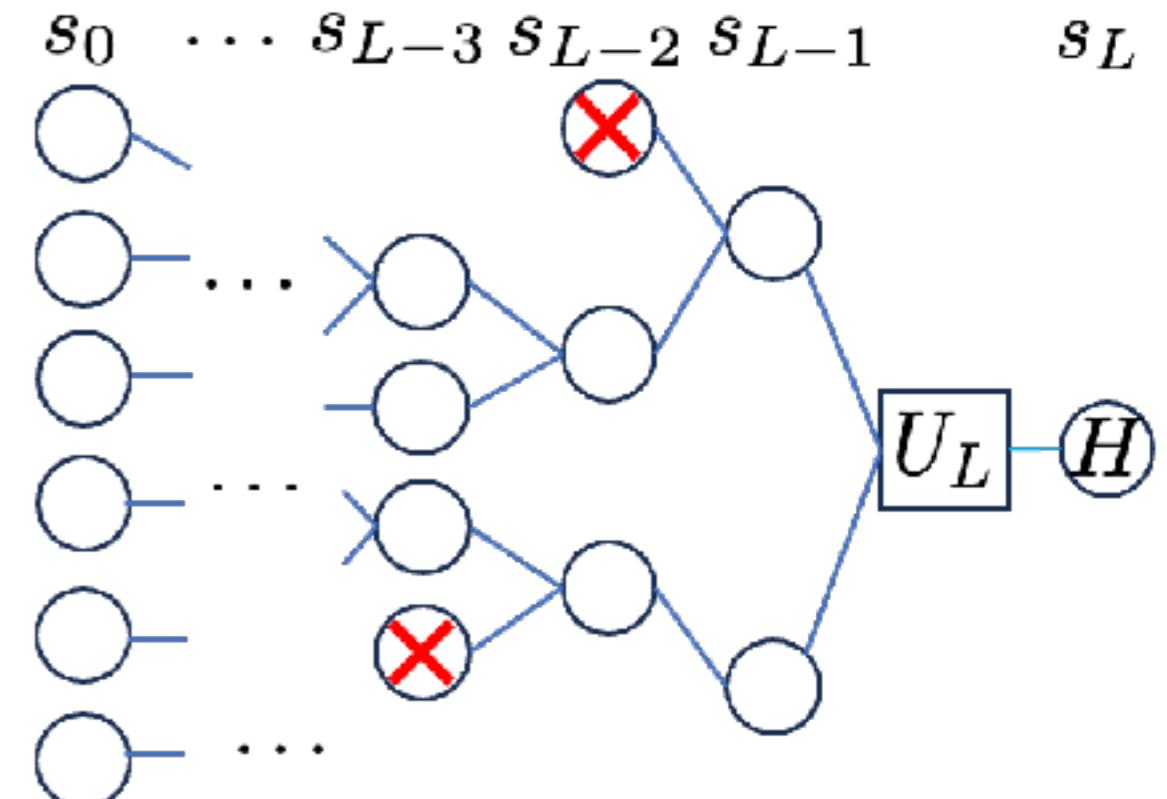
$$\langle\langle s_i | U_i | s_{i-1} \rangle\rangle \neq 0$$

- For qubits with I_i trivially applied, there is $s_i|_{I_i} = s_{i-1}|_{I_i}$, otherwise $f = 0$.
- For qubits with Clifford unitary V applied, there is $s_i|_V = (Vs_{i-1}V^\dagger)|_V$ up to phase, otherwise $f = 0$.
- For qubits with Pauli rotation $\exp(-i\frac{\theta}{2}\sigma)$ applied, we can divide into two cases depend on whether s_i commute or anti-commute with σ .
 - For the case of **commute**, $s_{i-1}|_\sigma = s_i|_\sigma$, otherwise $f = 0$.
 - For the case of **anti-commute**, we noted $\exp(-i\frac{\theta}{2}\sigma) = \cos\frac{\theta}{2}\mathbb{I} - i\sin\frac{\theta}{2}\sigma$. Thus, $s_{i-1}|_\sigma = s_i|_\sigma$ or $s_{i-1}|_\sigma = i\sigma s_i|_\sigma$ (up to phase) with factor $\cos\frac{\theta}{2}$ or $\sin\frac{\theta}{2}$, respectively.

There at most $|s_i|$ Pauli rotation anti-commute with s_i , results in $2^{|s_i|}$ candidates

Idea of OBPPP method

- Firstly, set $s_L = H$, otherwise $\langle\langle H | s_L \rangle\rangle = 0$ leads to $f = 0$.
- Then enumerate s_{L-1} that satisfy $\langle\langle s_L | U_L | s_{L-1} \rangle\rangle \neq 0$, and eliminate the cases $|s_L| + |s_{L-1}| > M$. There is at most $2^{|s_L|}$ candidates for s_{L-1} .
- Repeat this process until s_0 .



Algorithm's Complexity

- Our algorithm calculates the contributions of the Pauli path with $|s| \leq M$.
- For each Pauli path s , it is possible to determine $f(\theta, s, H)$ with a time complexity of $O(nL)$.
- By OBPPP method, we can enumerate all these Pauli paths s (at most 2^M), which make non-zero contributions while satisfying $|s| \leq M$, with time complexity $O(nL)2^M$.
- Consequently, the overall time required for determining the approximate noisy observed value is within $O(nL)2^M$.

Bound the Error

- **Lemma** For $\forall \nu > 0$, if $M \geq \frac{1}{2\lambda} \ln \frac{1}{\nu}$, the mean-square error $E_\theta |\widetilde{\langle H \rangle} - \langle H \rangle|^2$ is upper bounded by ν .
- **Corollary** For $\forall \nu > 0$ and a **fixed error rate** λ , the time complexity of obtaining $\widetilde{\langle H \rangle}$ with mean-square error $E_\theta |\widetilde{\langle H \rangle} - \langle H \rangle|^2 \leq \nu$ is $O(nL)(\frac{1}{\sqrt{\nu}})^{\frac{1}{\lambda}} = \text{Poly}(n, L, \frac{1}{\sqrt{\nu}})$.
- **Theorem** Given a **fixed error rate** λ , for Pauli word observable H and initial state $|0\rangle$, for arbitrary truncation error ε , there exists a polynomial-scale classical algorithm to determine the approximated noisy observed value $\widetilde{\langle H \rangle}$, which satisfies $|\widetilde{\langle H \rangle} - \langle H \rangle| \leq \varepsilon$ with a probability of at least $1 - \delta$ over all possible parameters θ . The time complexity is within $\text{Poly}\left(n, L, \frac{1}{\varepsilon}, \frac{1}{\sqrt{\delta}}\right)$.

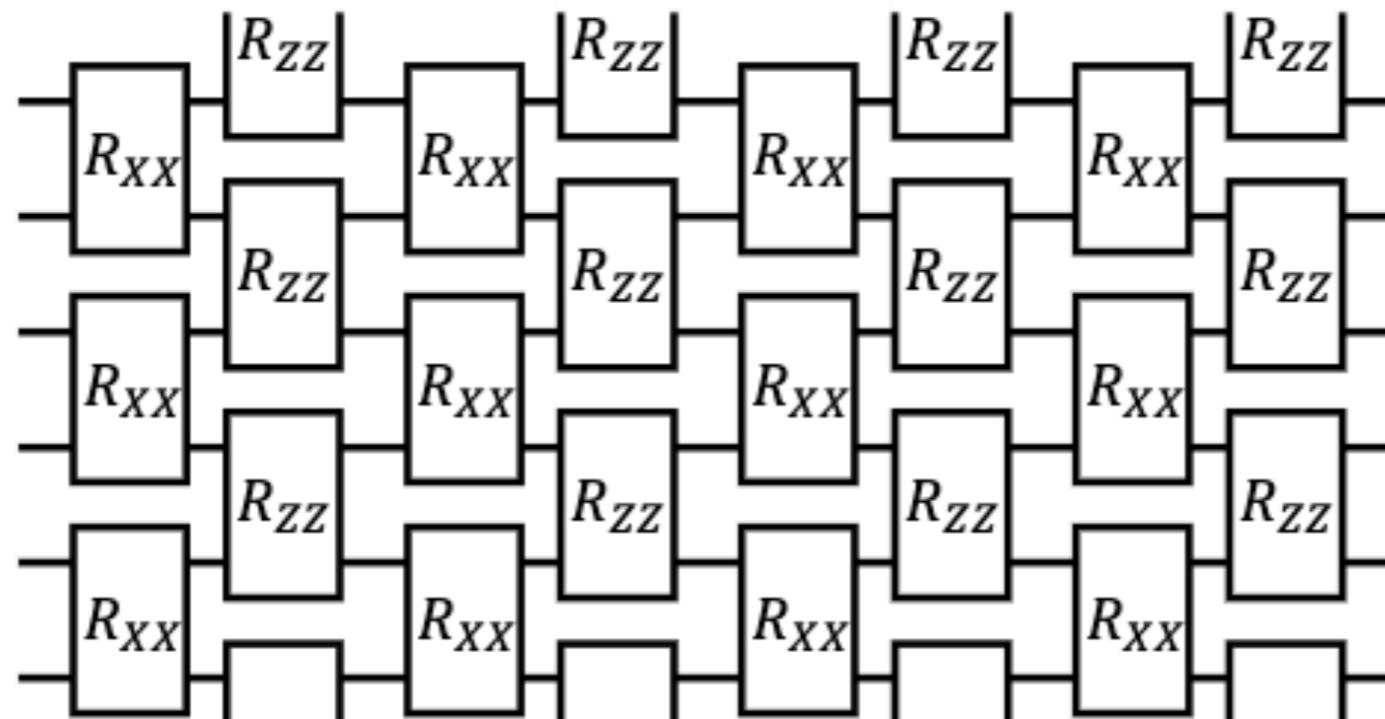
The influence of variable noise rate λ

Proposition To estimate the approximate noisy observed value with the mean-square error $E_\theta |\widetilde{\langle H \rangle} - \langle H \rangle|^2$ less than a sufficiently small constant, we have

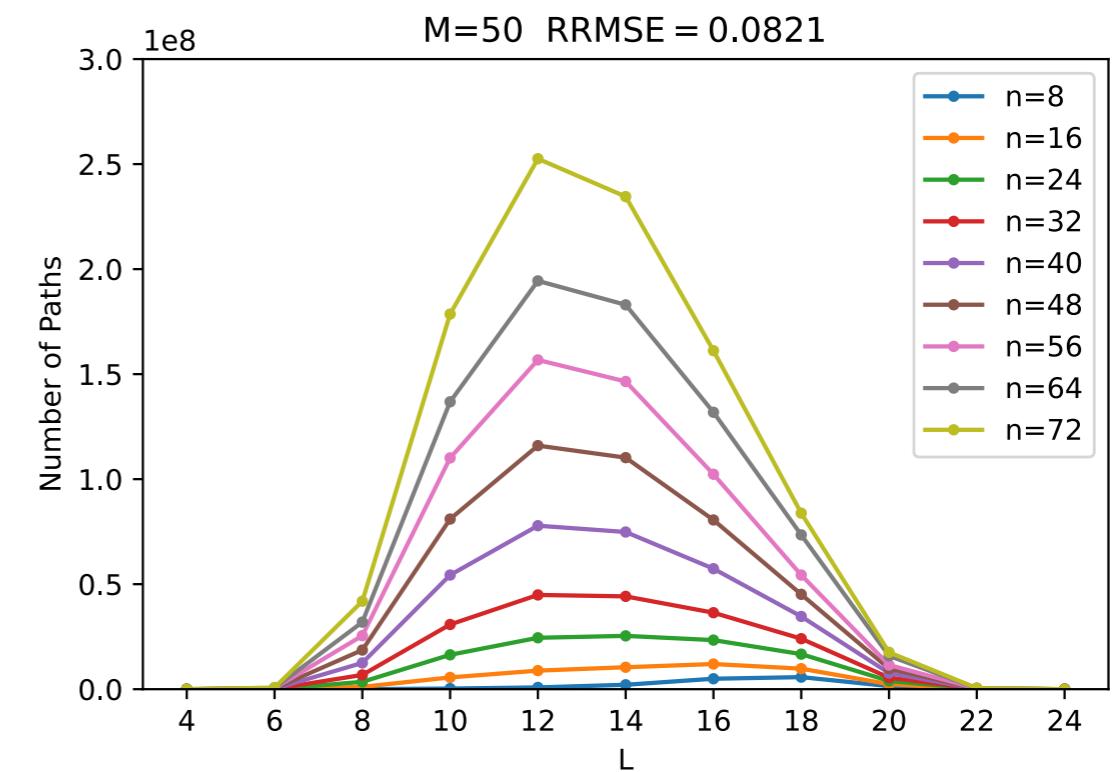
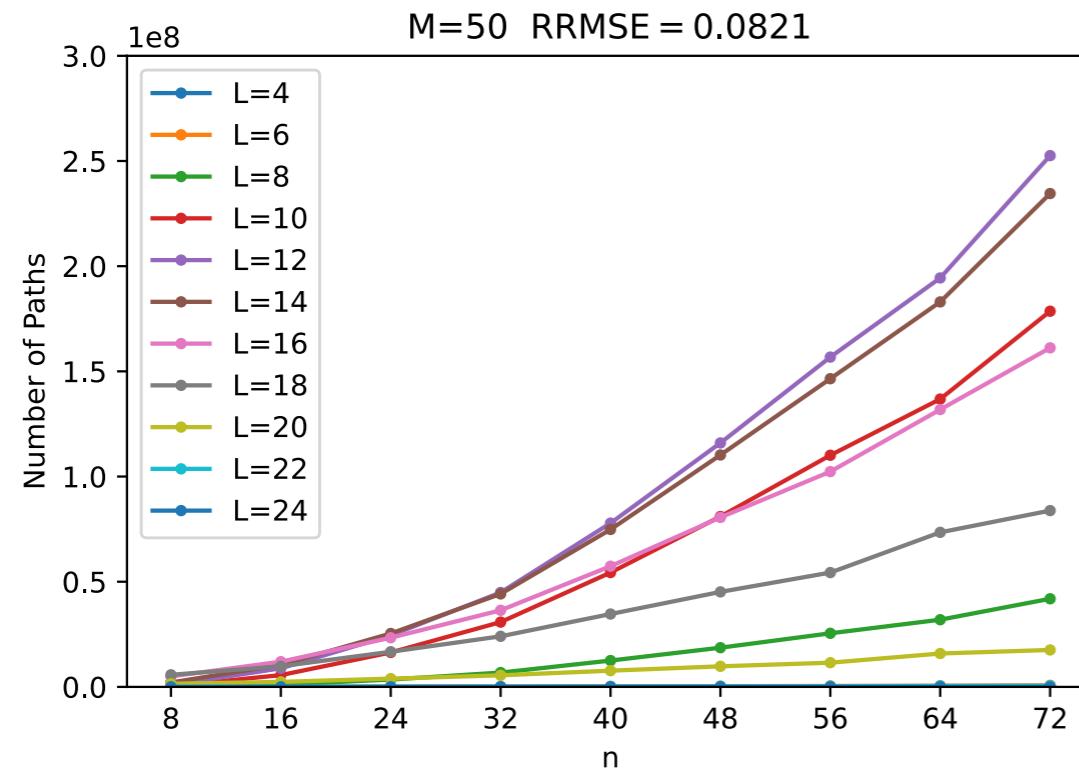
1. If $\lambda = \Omega(\frac{1}{\log L})$, there exists a classical algorithm that can complete the computation in time $\text{Poly}(n, L)$.
2. If $\lambda = O(\frac{1}{L})$, there exists a situation where our method exhibits **exponential** time complexity with respect to L .

Numerical results

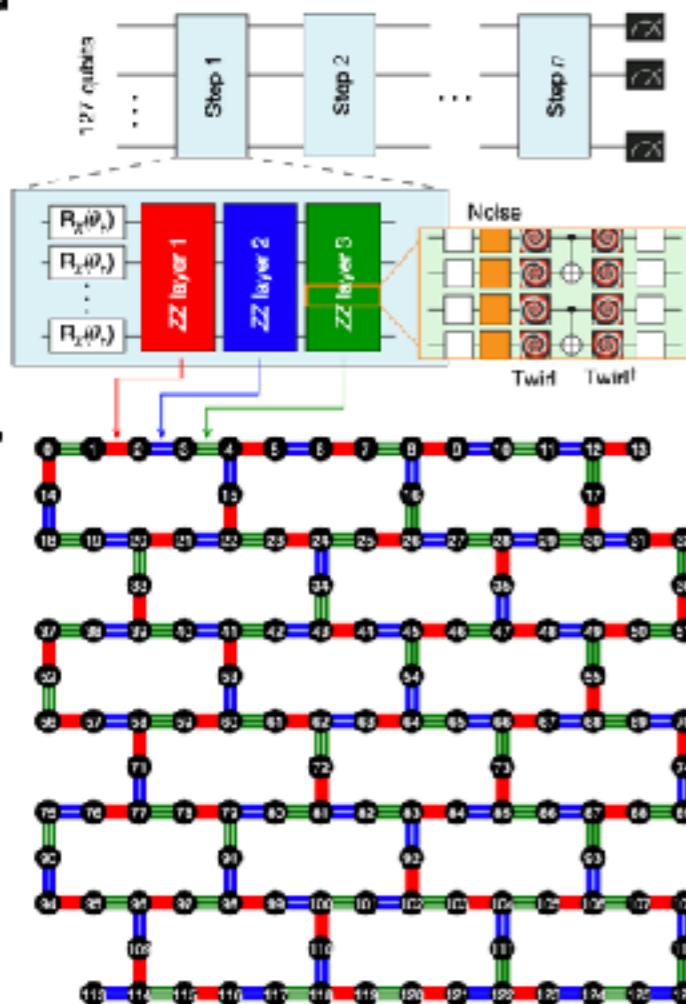
- In the theoretical analysis, the complexity of our algorithm has factor 2^M that comes from **#Pauli path with non-zero contribution** and far exceeds the real situation.
- We use the following ansatz as an example to explore the real #Path in the QAOA algorithm.



Numerical results



- From this, we reveal #Path is significantly smaller than the upper bound $2^M \approx 1e15$, as suggested by the theoretical analysis.
- Setting $\lambda = 0.2$, for a given NRMSE bound, #Path first increases and then decreases with the increase of L, which aligns with the findings in Ref.



Evidence for the utility of quantum computing before fault tolerance

<https://doi.org/10.1038/s41586-023-06096-3>

Received: 24 February 2023

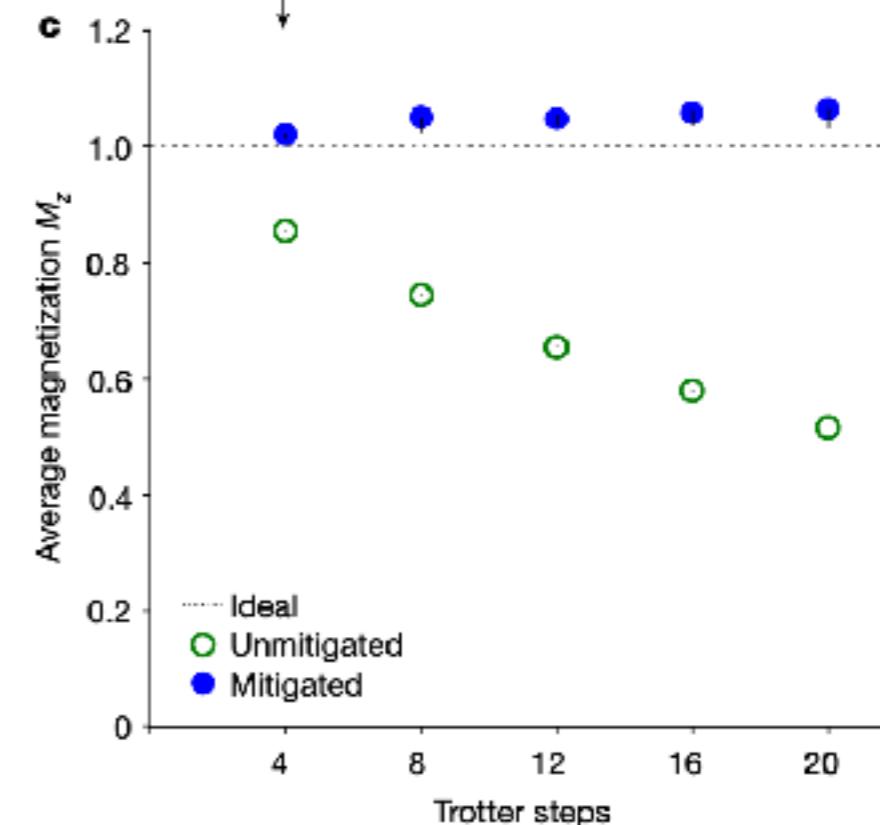
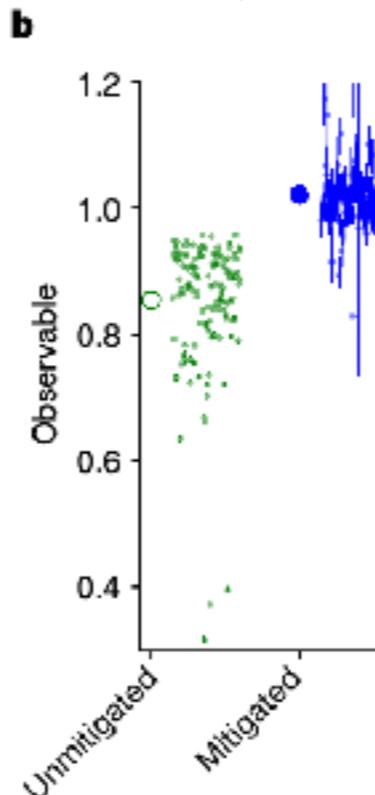
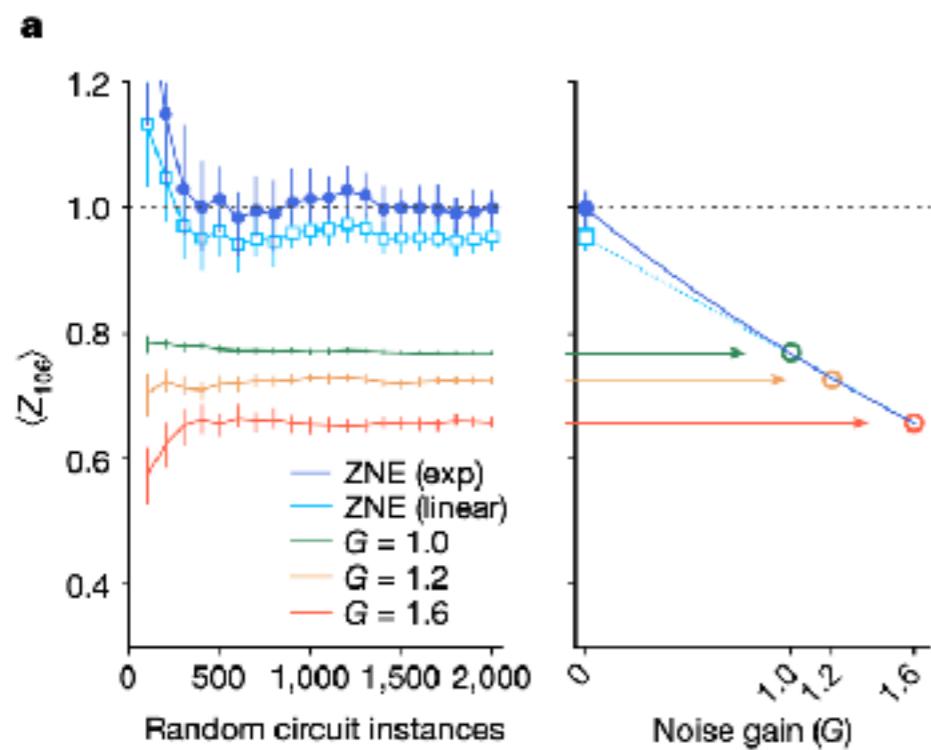
Youngseok Kim^{1,6,7}, Andrew Eddins^{2,6,8}, Sajant Anand³, Ken Xuan Wei¹, Ewout van den Berg¹, Sami Roserblatt¹, Hasan Nayfeh¹, Yantac Wu^{3,4}, Michael Zaletel^{3,5}, Kristan Temme¹ & Abhinav Kandala^{1,2*}

Accepted: 18 April 2023

The quantum computer provides correct results for which leading classical approximations such as pure-state- based 1D (matrix product states, MPS) and 2D (isometric tensor network states, isoTNS) tensor network methods break down.

Zero-noise extrapolation:

- Evaluate on noisy quantum computer,
- Mitigate error in classical computer



Circuit Structure

IBM has reported experiments on a 127-qubit Eagle processor, whose benchmark circuit were defined by the Trotterized time evolution of a 2D transverse-field Ising model:

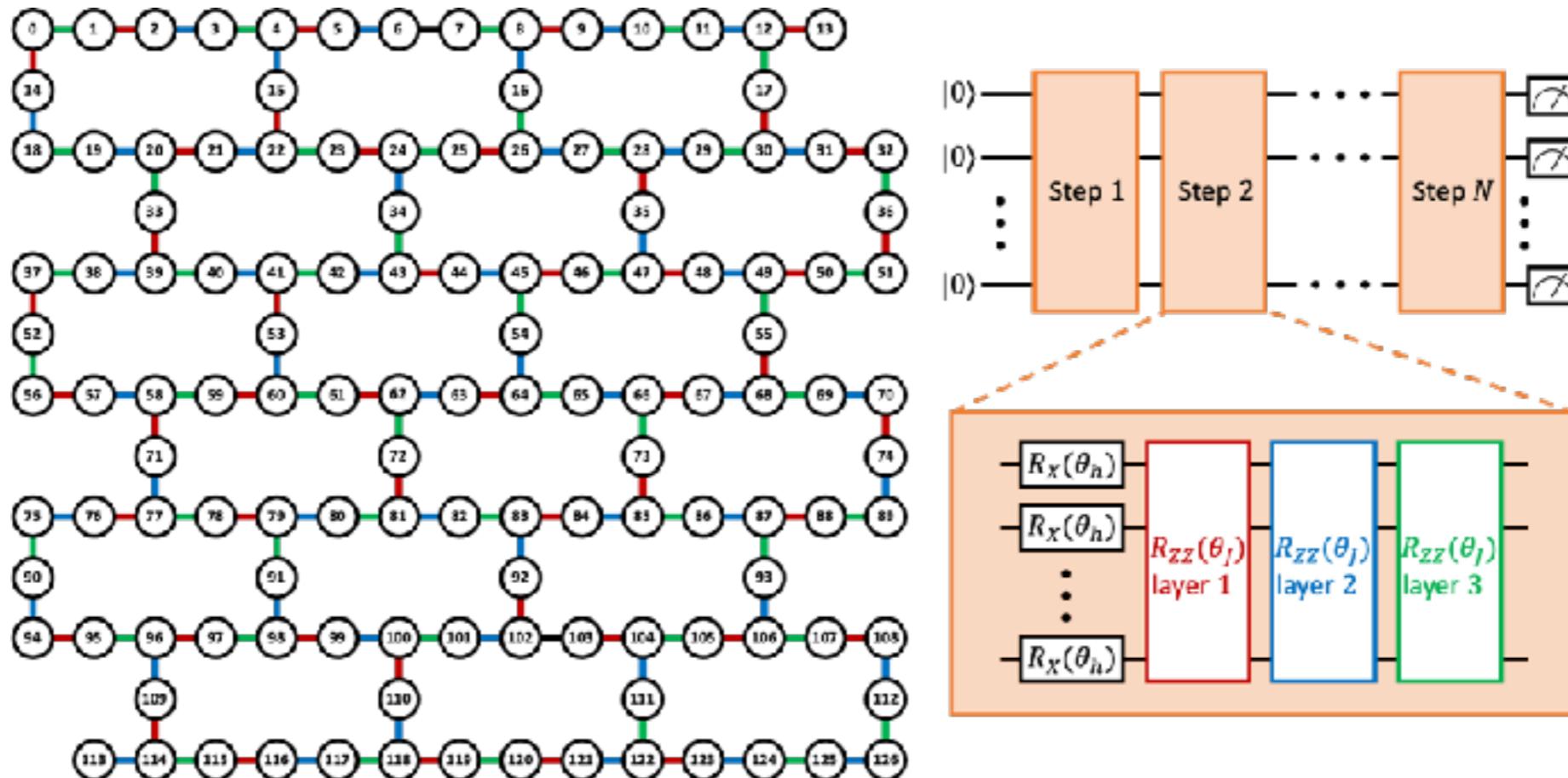
$$H = -J \sum_{\langle i,j \rangle} Z_i Z_j + h \sum_i X_i,$$

The first-order Trotterized time evolution of the Hamiltonian, which is given by

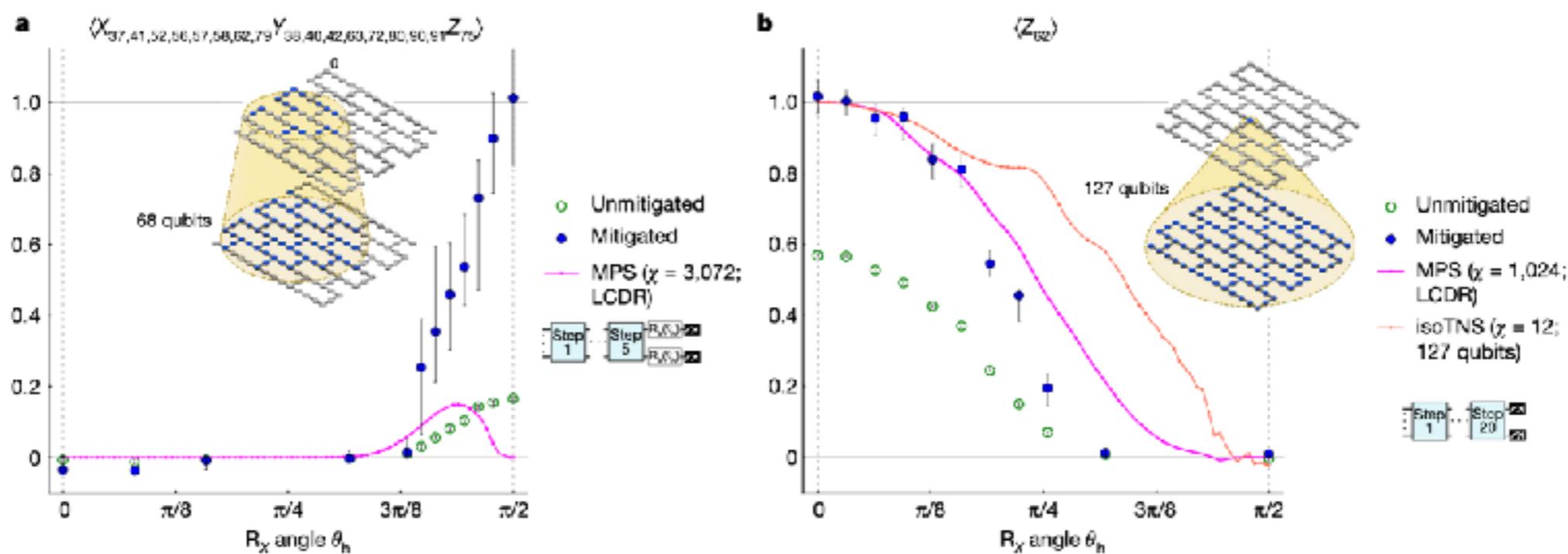
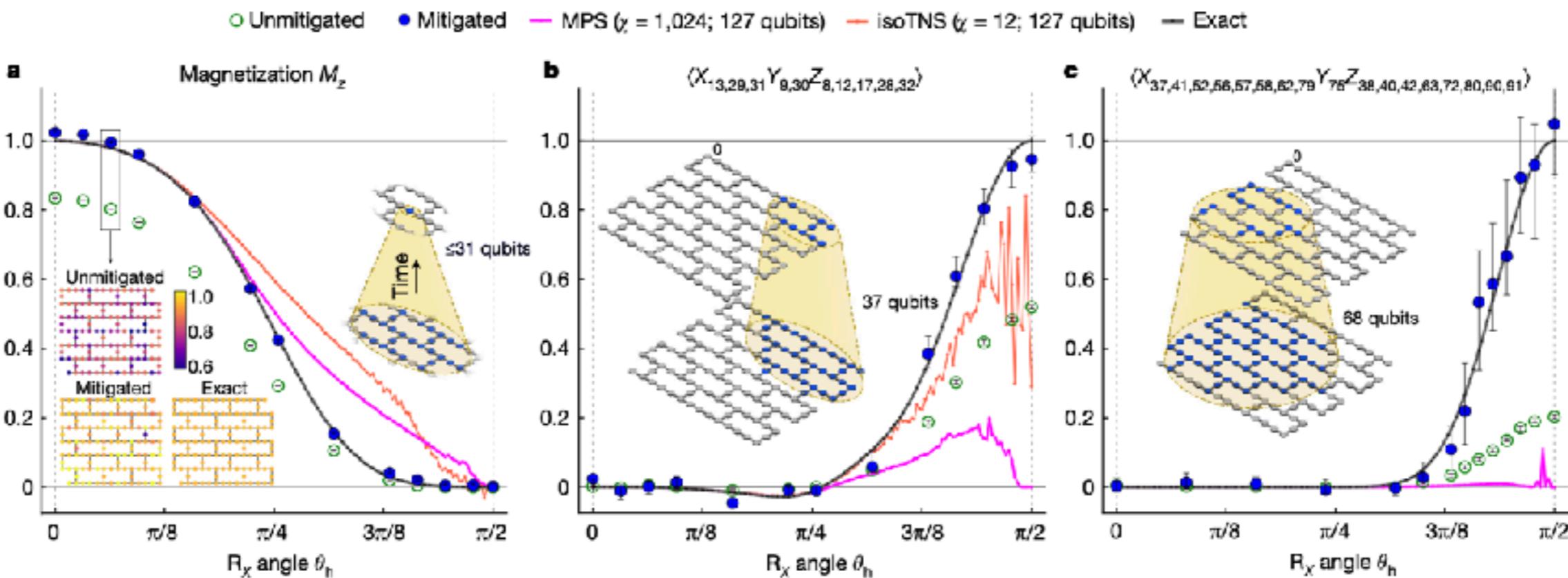
$$U(\tau) = \exp\{-i\tau H\} = \prod_{\langle i,j \rangle} \exp\{i\tau J Z_i Z_j\} \prod_i \exp\{-i\tau h X_i\} + \mathcal{O}(\tau^2) = \prod_{\langle i,j \rangle} R_{Z_i Z_j}(-2J\tau) \prod_i R_X(2h\tau) + \mathcal{O}(\tau^2),$$

in which the evolution time T is discretized into N Trotter steps, with a single step evolution time of $\tau = \frac{T}{N}$.

The time evolution is implemented by

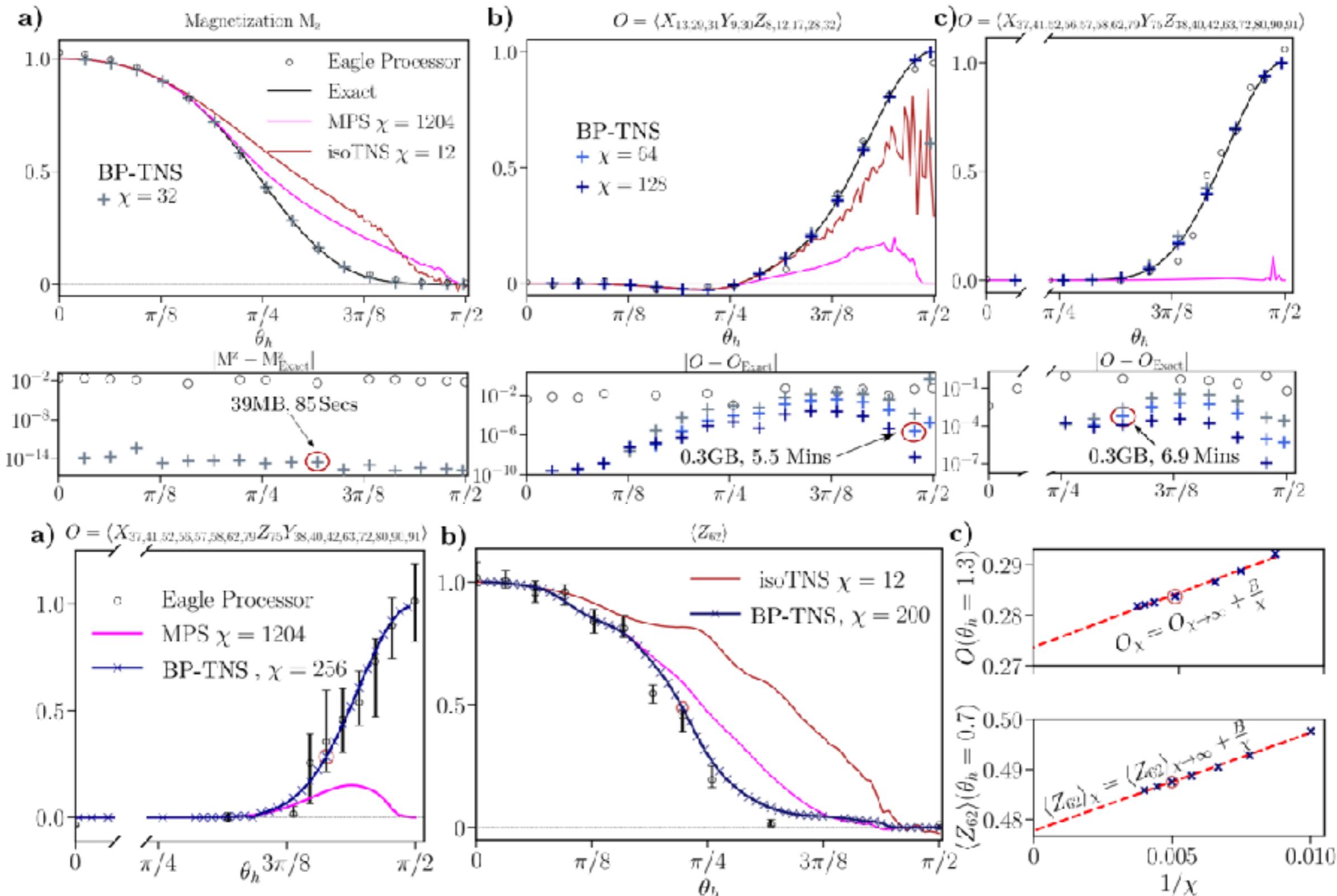


- A single Trotter step is composed of one layer of R_X gates and three layers of R_{ZZ} gates. For simplicity, IBM chooses $\theta_J = -2J\tau = -\frac{\pi}{2}$ and considers $\theta_h = 2h\tau$ to be in the range $[0, \frac{\pi}{2}]$.



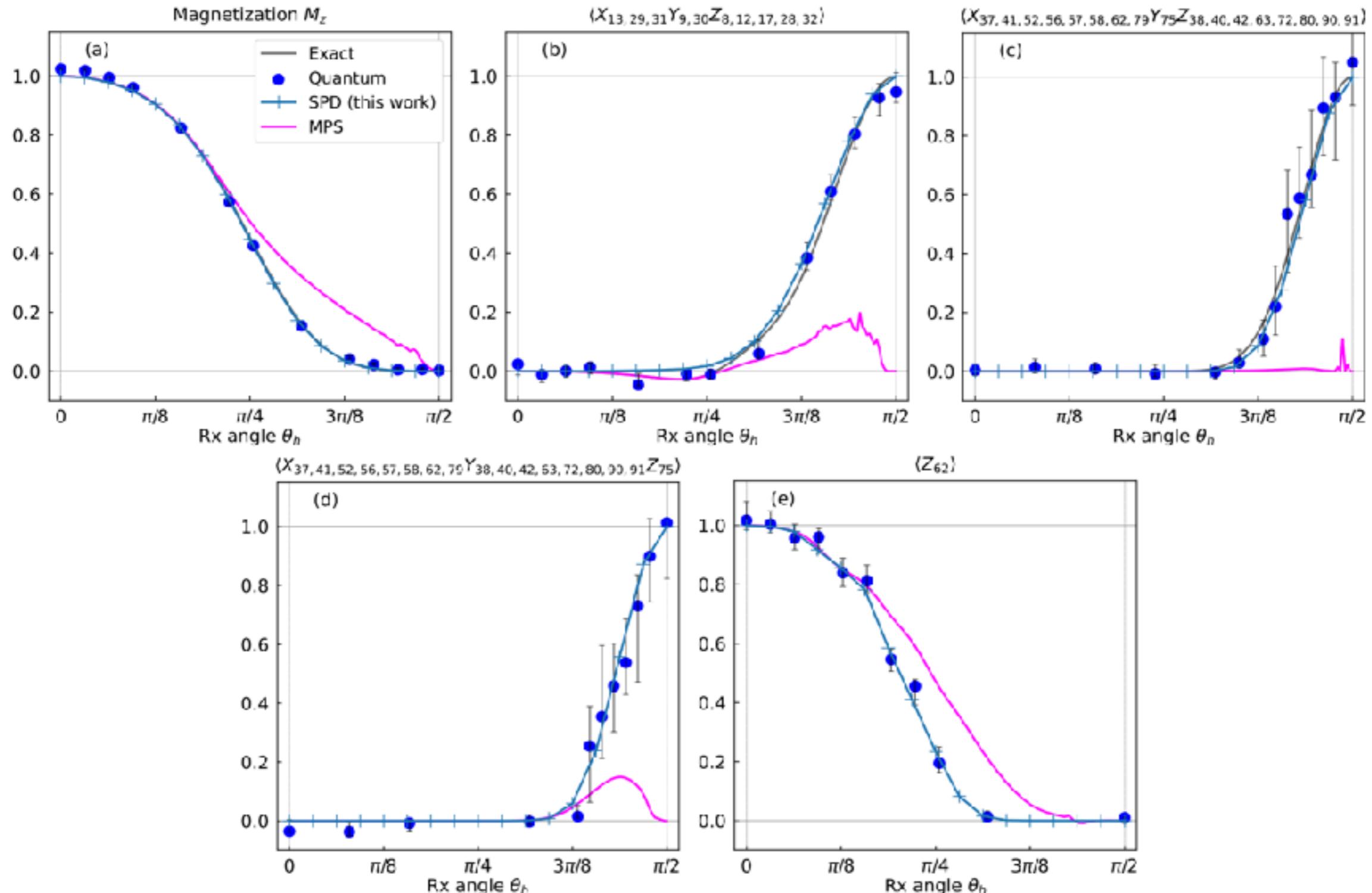
Efficient tensor network simulation of IBM's Eagle kicked Ising experiment

Joseph Tindall,¹ Matthew Fishman,¹ E. Miles Stoudenmire,¹ and Dries Sels^{1,2}



Fast classical simulation of evidence for the utility of quantum computing before fault tolerance

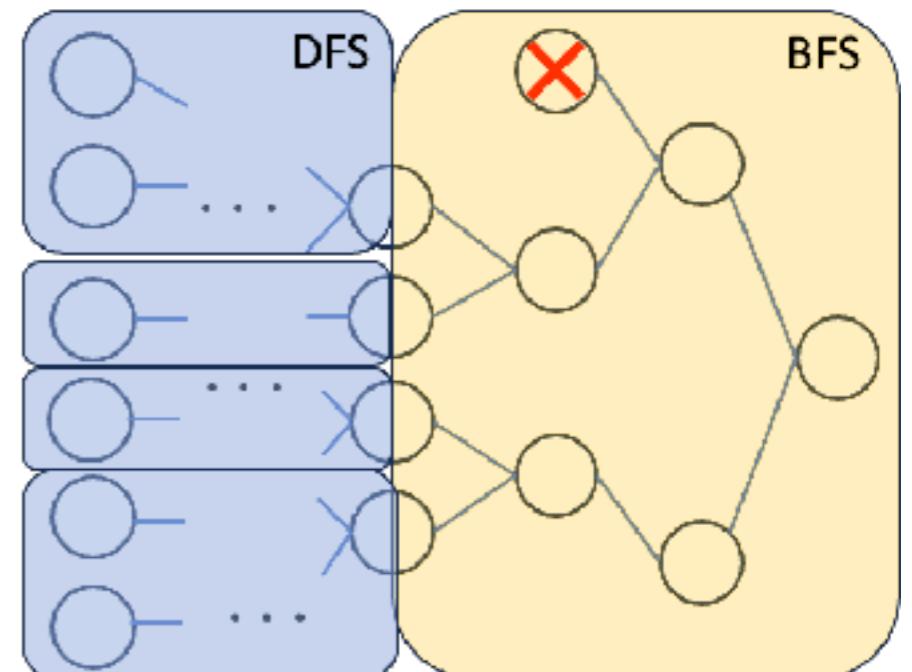
Tomislav Begušić and Garnet Kin-Lic Chan*



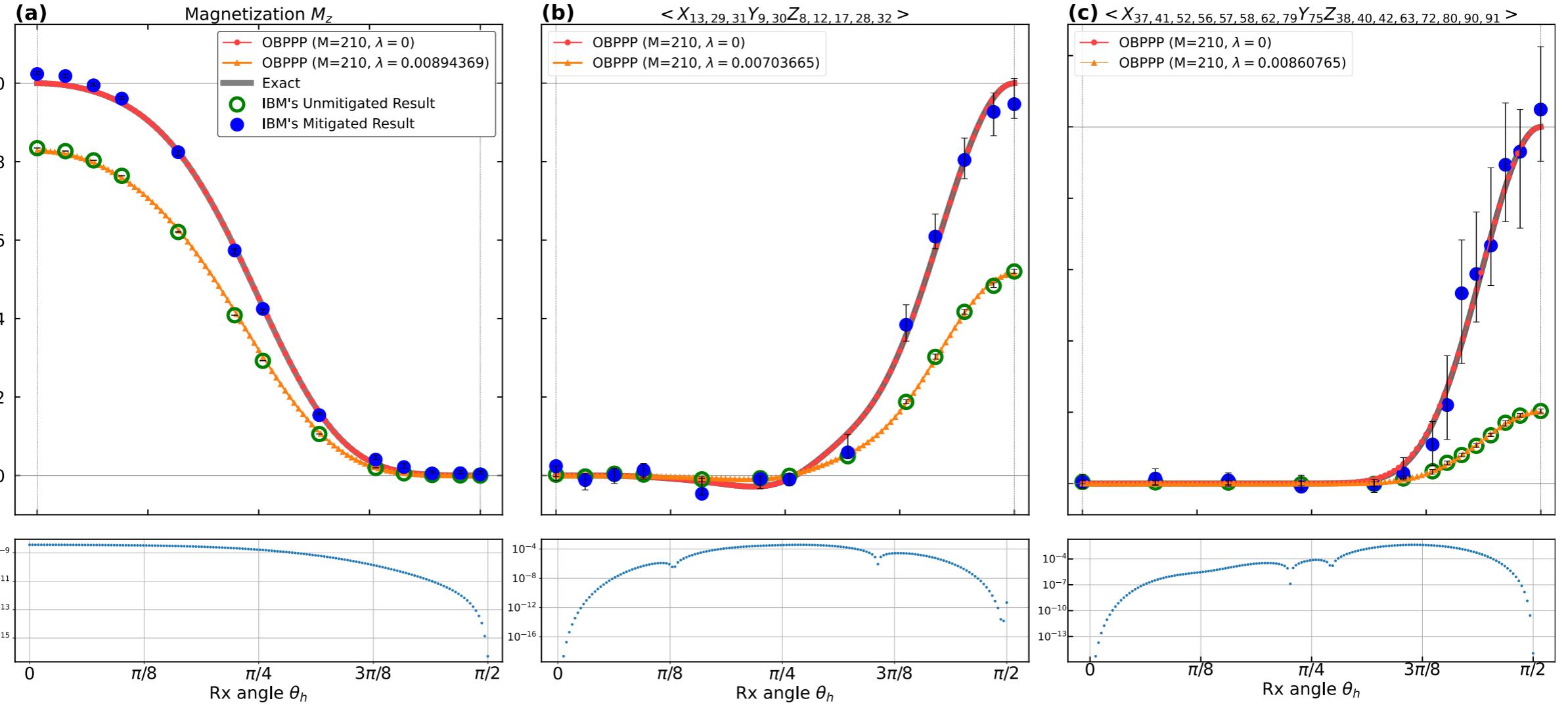
- Sparse Pauli dynamics(SPD) method
- 1-2 minitues for each point in single core.

Code implement

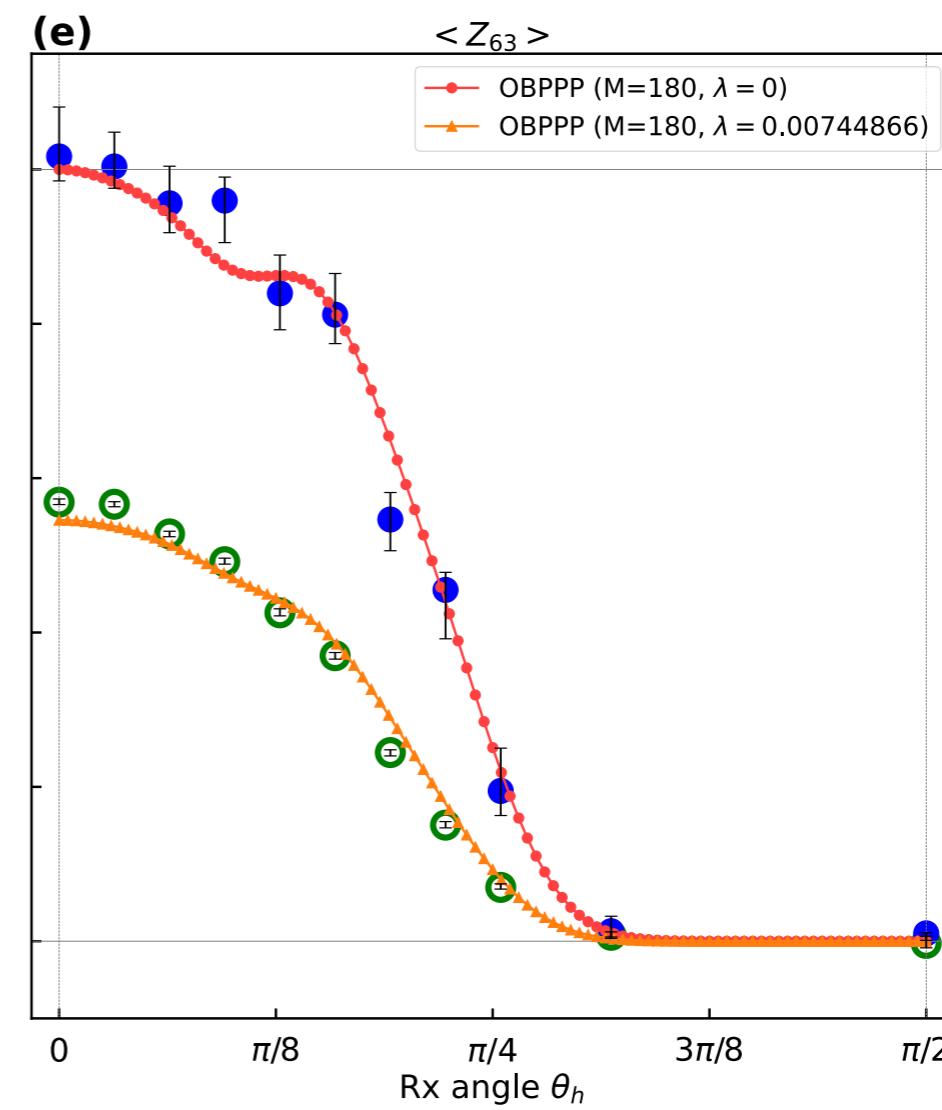
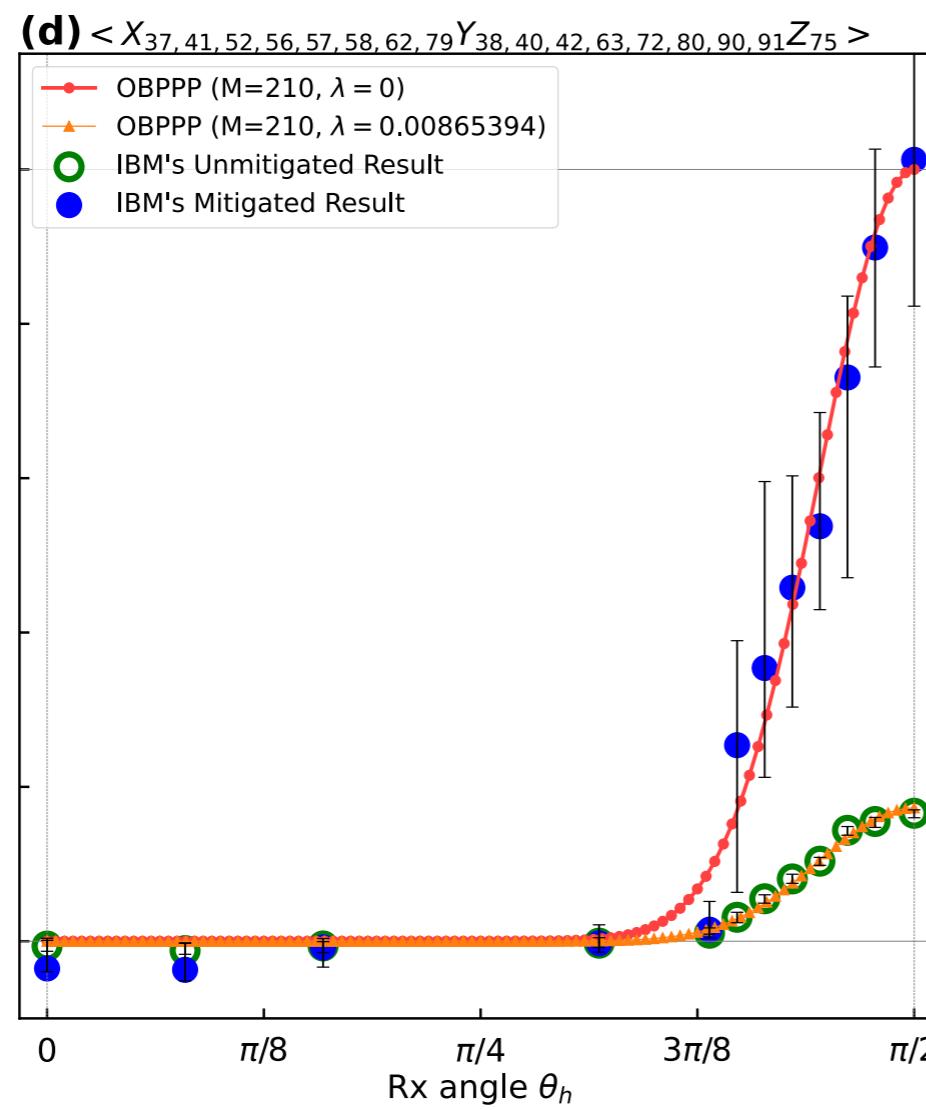
- First propagate several layers of circuit using **breadth-first search** to obtain some Path branches for parallelization.
- Then, we allocate these Path branches to 56 CPU cores, who performs **depth-first search** for Pauli paths that meet the criteria.
- After enumerating paths, we actually **get the analytical expressions of $\langle H \rangle$** , enabling us to obtain all corresponding results for θ_h across the entire continuous interval in a single computation.



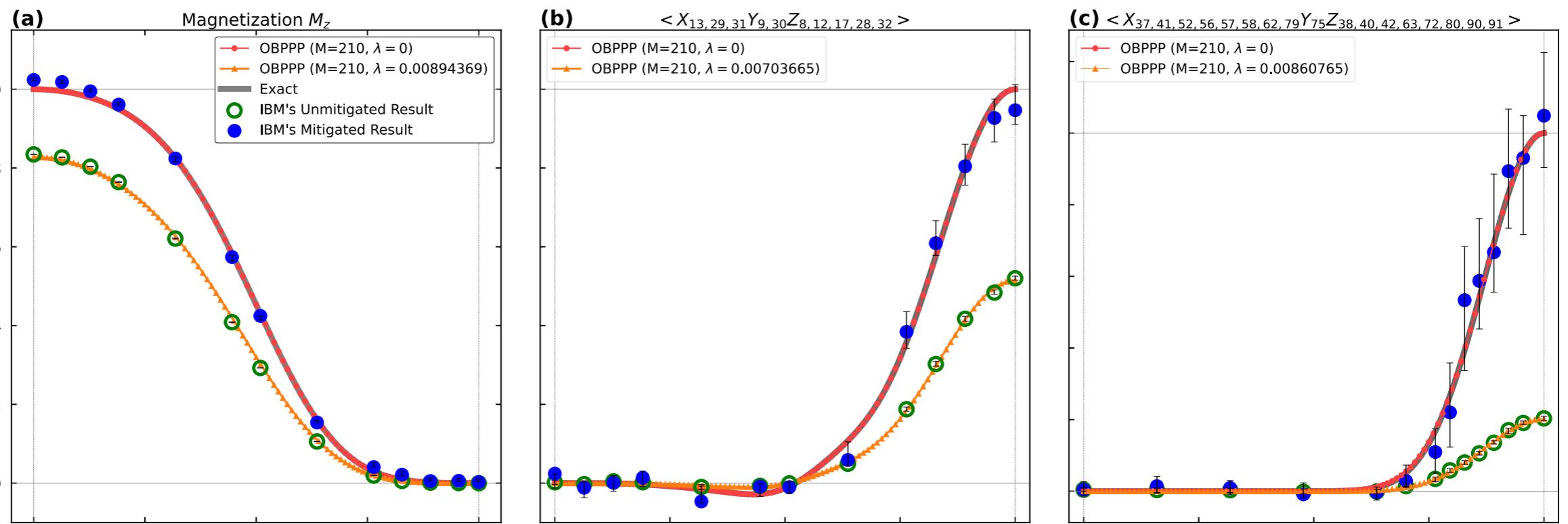
In (a)-(c), $N = 5$ corresponds to a circuit with depth $L = 20$.



- Running time (in 56 core): **40 s, 4 min, 4 min**, respectively.
- Compared with exact results, OBPPP shows higher accuracy in (b) and (c) than other classical simulators.



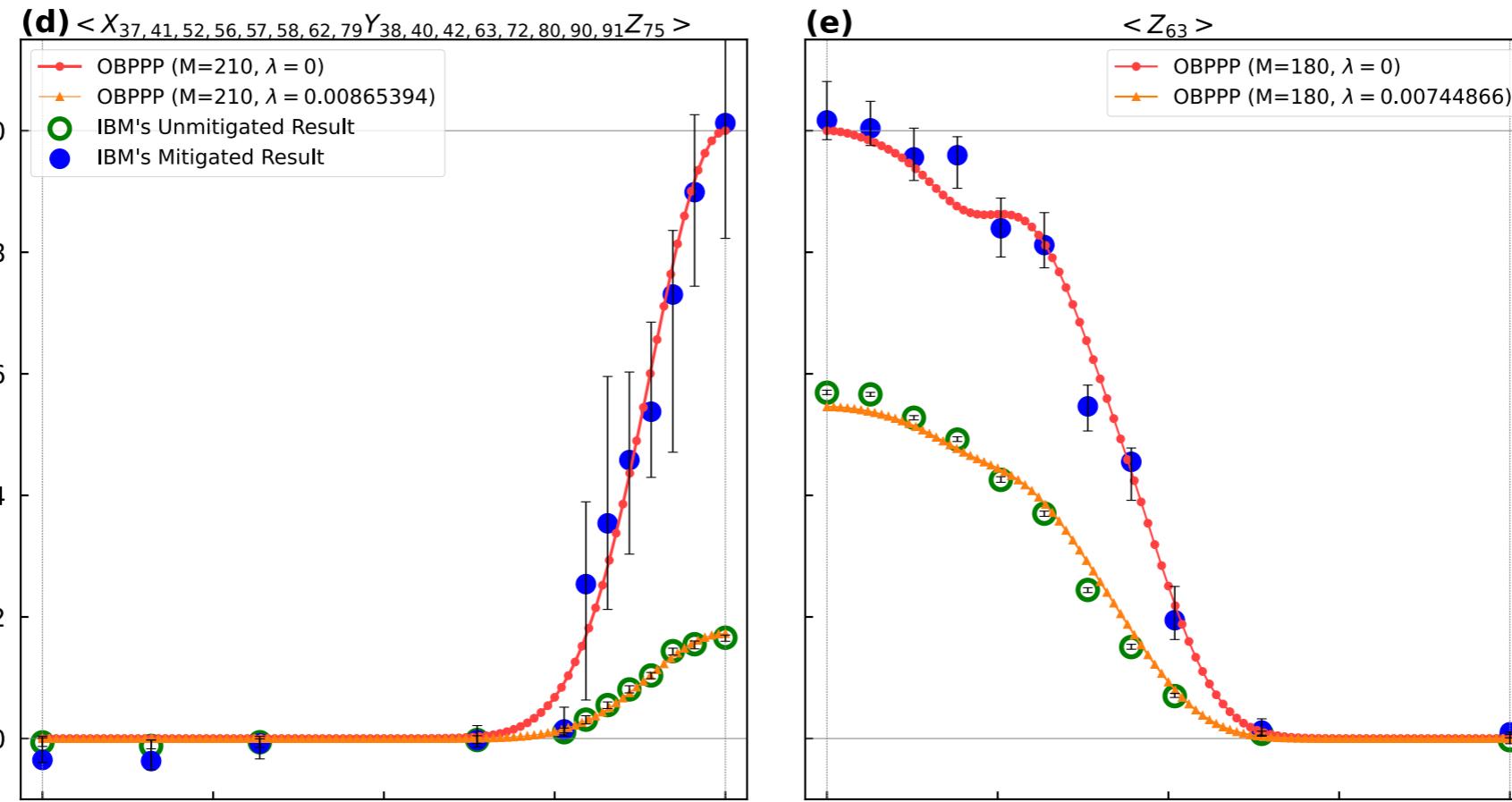
- In (d), $N = 5$ and there is an additional layer of R_X gates applied at the end of the circuit, corresponds to a circuit with depth $L = 21$. Runtime is **10min**.
- In (e), $N = 20$ corresponds to a circuit depth with $L = 80$. Runtime is **33min**.



- In order to compare the unmitigated results, we employed a classical optimizer to minimize the distance between $\widetilde{\langle H \rangle}$ and our approximate observed value $\langle \widetilde{H} \rangle$, formalized as

$$\lambda = \arg \min_{\lambda} \sqrt{\sum_{(\theta_h, y_{\theta_h}) \in \text{dataset}} \left| \langle \widetilde{H} \rangle_{\theta_h} - y_{\theta_h} \right|^2}.$$

- The minimal distance is **below 0.002** for (a)-(d) and **below 0.008** for (e).
- Additionally, the optimal λ ranges from **0.007** to **0.009**, which is also in agreement with the error rates reported by IBM's experiment.



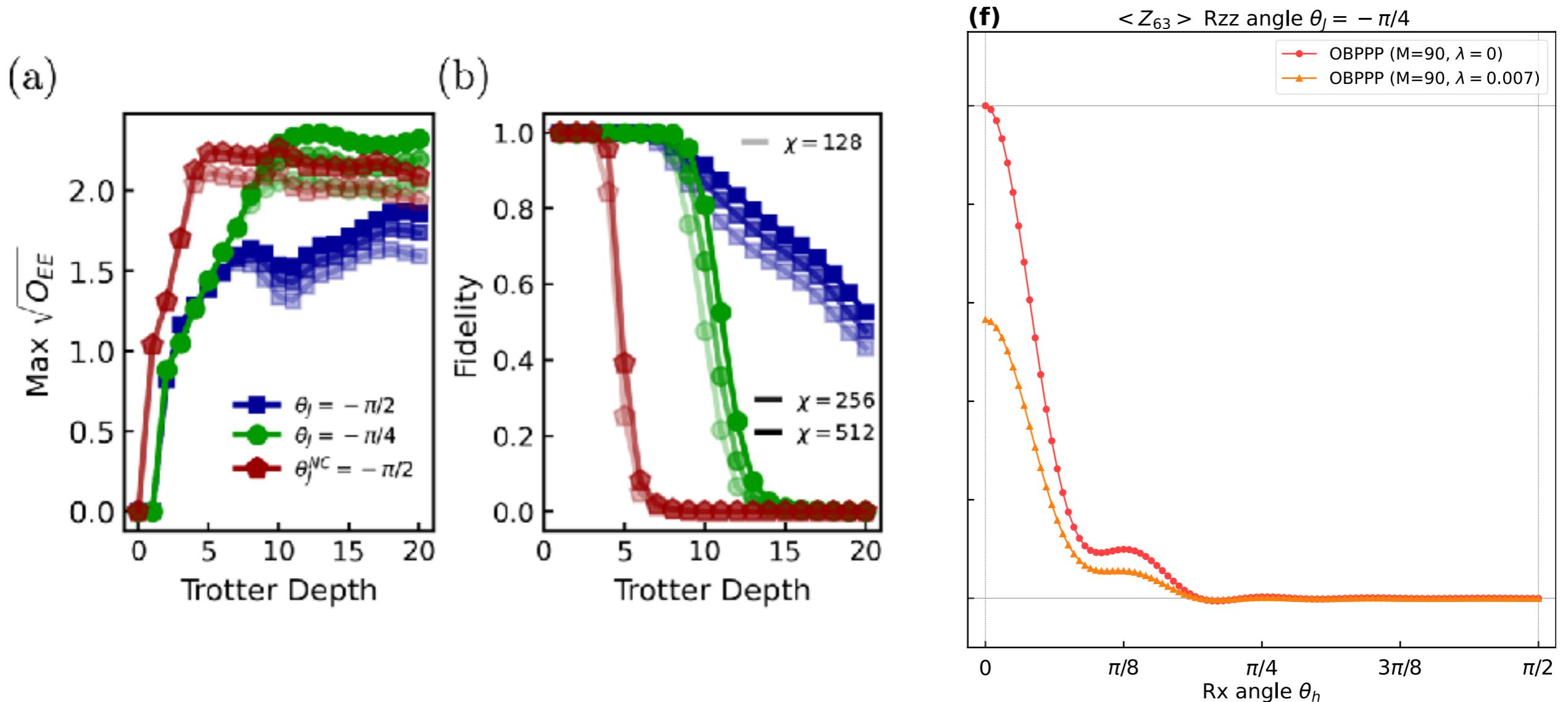
- In order to compare the unmitigated results, we employed a classical optimizer to minimize the distance between $\widetilde{\langle H \rangle}$ and our approximate observed value $\langle \widetilde{H} \rangle$, formalized as

$$\lambda = \arg \min_{\lambda} \sqrt{\sum_{(\theta_h, y_{\theta_h}) \in \text{dataset}} \left| \langle \widetilde{H} \rangle_{\theta_h} - y_{\theta_h} \right|^2}.$$

- The minimal distance is **below 0.002** for (a)-(d) and **below 0.008** for (e).
- Additionally, the optimal λ ranges from **0.007 to 0.009**, which is also in agreement with the error rates reported by IBM's experiment.

Classical benchmarking of zero noise extrapolation beyond the exactly-verifiable regime

Sajant Anand,¹ Kristan Temme,² Abhinav Kandala,² and Michael Zaletel^{1,3}



- In (f), $N = 20$ and the rotation angle of R_{ZZ} is set as $\theta_J = -\frac{\pi}{4}$. While the experiment results are not available. Runtime is **30min**.

Brief Summary in Numerical results

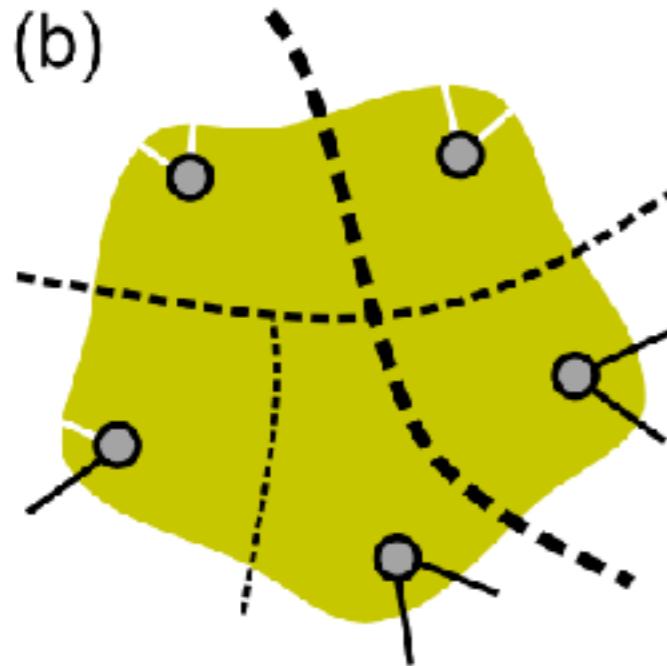
- Successfully performing classical simulations on IBM's Eagle processor in a **shorter runtime** than quantum hardware, while achieving **more accurate** expectation.
- Successfully deriving noise results from noiseless results, resulting in a **good fit to the unmitigated data** of raw experimental observations.
- OBPPP exhibits **faster execution times than BP-TNS** methods. From (a)-(c), OBPPP demonstrates higher accuracy in (b) and (c) than TNS. For (a), the error is within $1e^{-8}$.
- Compared with SPD method, the speed of OBPPP is comparable to that of SPD method, while with higher accuracy.

Final Remark

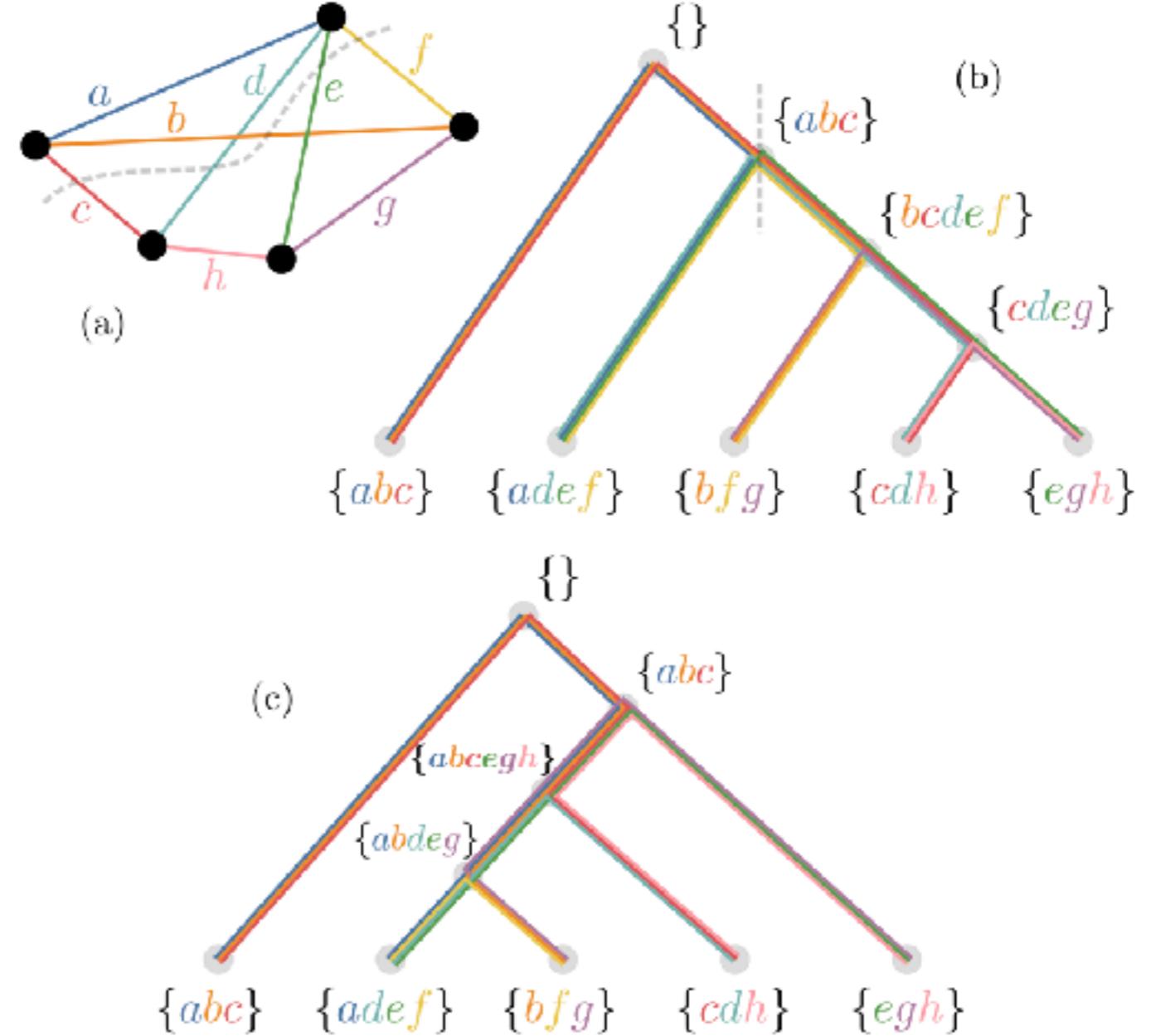
- The dimension of Hilbert space grow exponentially with qubit's number, classical simulating “general states” in polynomial cost is impossible.
- However, producing real “general states” of Hilbert space in experiment is non-trivial (decoherence, planar connection, etc.)
- Experimental physicists do their best to make quantum states look general enough to claim quantum superiority. Classical algorithms attack those superiorities by looking for “specialty” of system and using them to speed up computation.
- In the end, this confrontation game will end in a fiasco of the classic algorithm. But the longer classical algorithms resist, the more better classical algorithms will be designed, and the deeper quantum scientists' understanding of "general states" will be.

Thank You!

Hyper-graph partitioning



- Map TN to a hyper-graph
- Hyper-partitioning to get a contraction tree (e.g. min-cut)
- Optimal/Greedy search contraction order at the bottom



The order of tensor contraction affect both the time and the space complexity significantly

3000 years / amplitude on one Quadro P2000 GPU

Hyper-graph partitioning

↓
Dynamic slicing

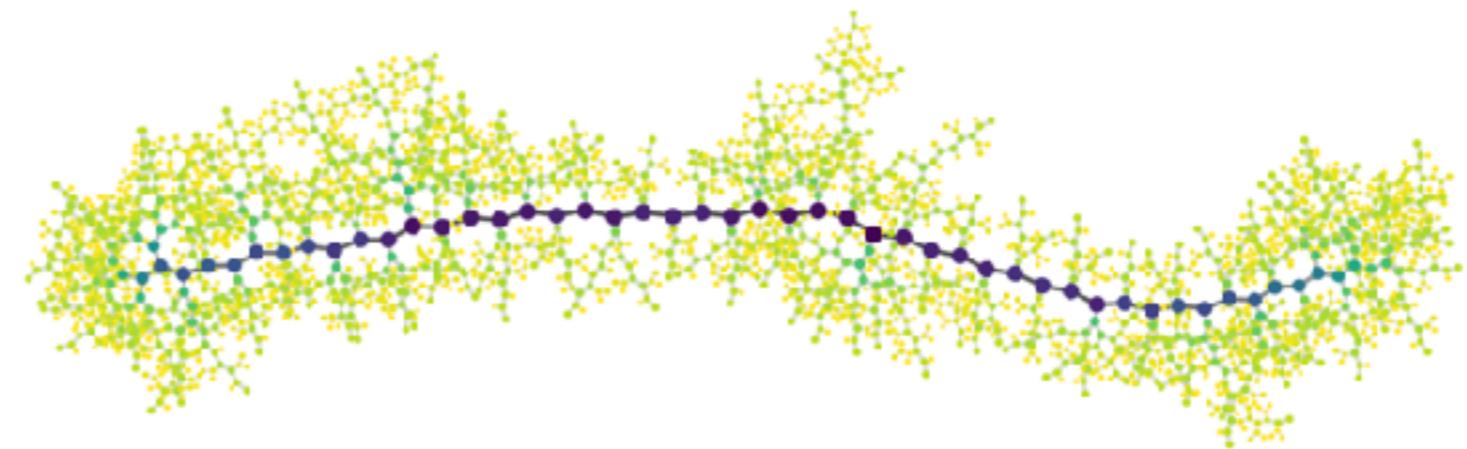
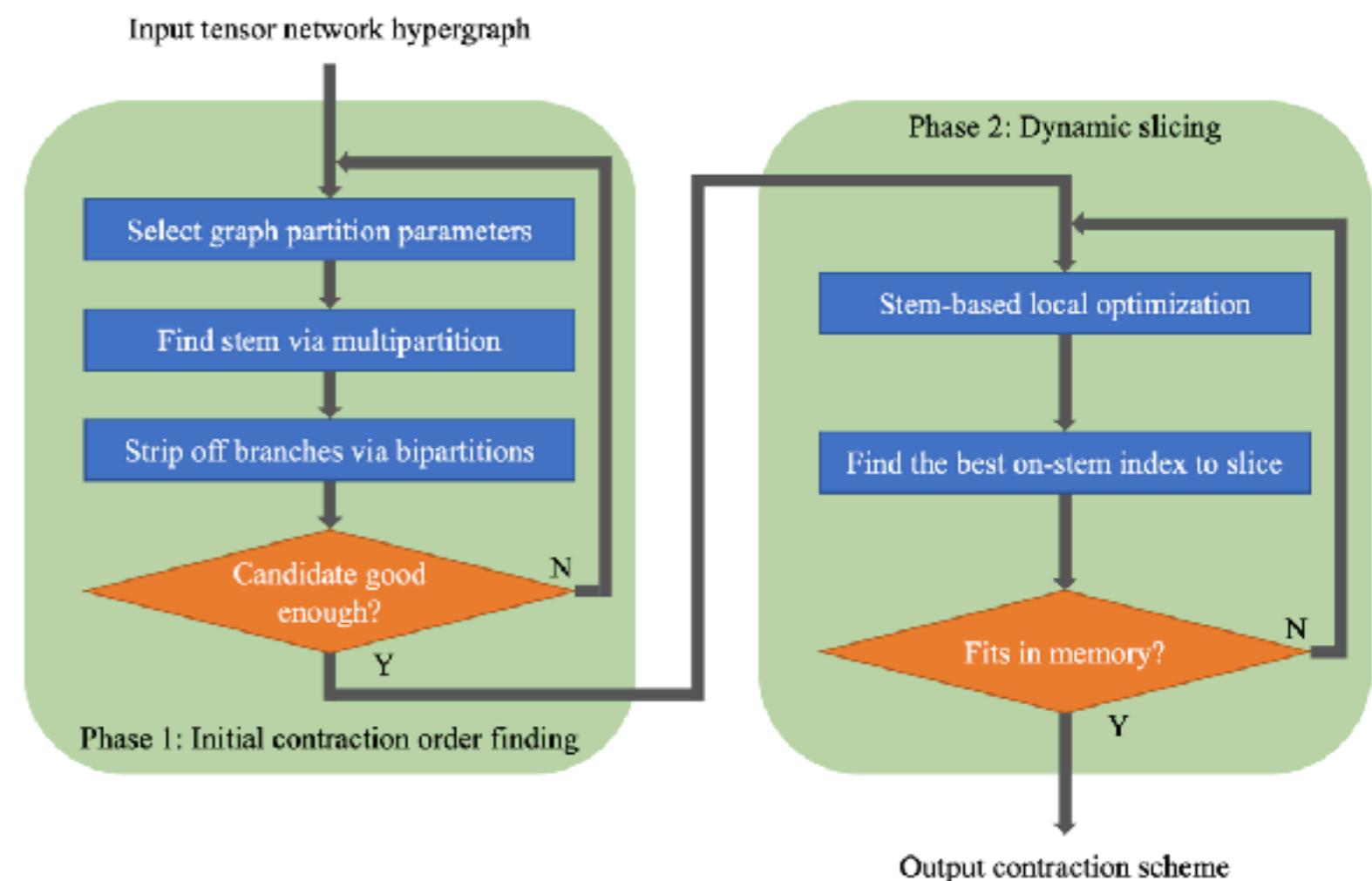


Figure 1: Tensor network contraction of a quantum circuit from the random circuit family [7], visualized as a binary contraction tree. Each node in the tree represents a step in the contraction. Larger, darker nodes indicate more expensive steps. The central stem dominates the overall contraction cost.

TN simplification
+ Hypergraph partitioning
+ greedy optimization
+ Dynamic slicing
+ Rejection sampling

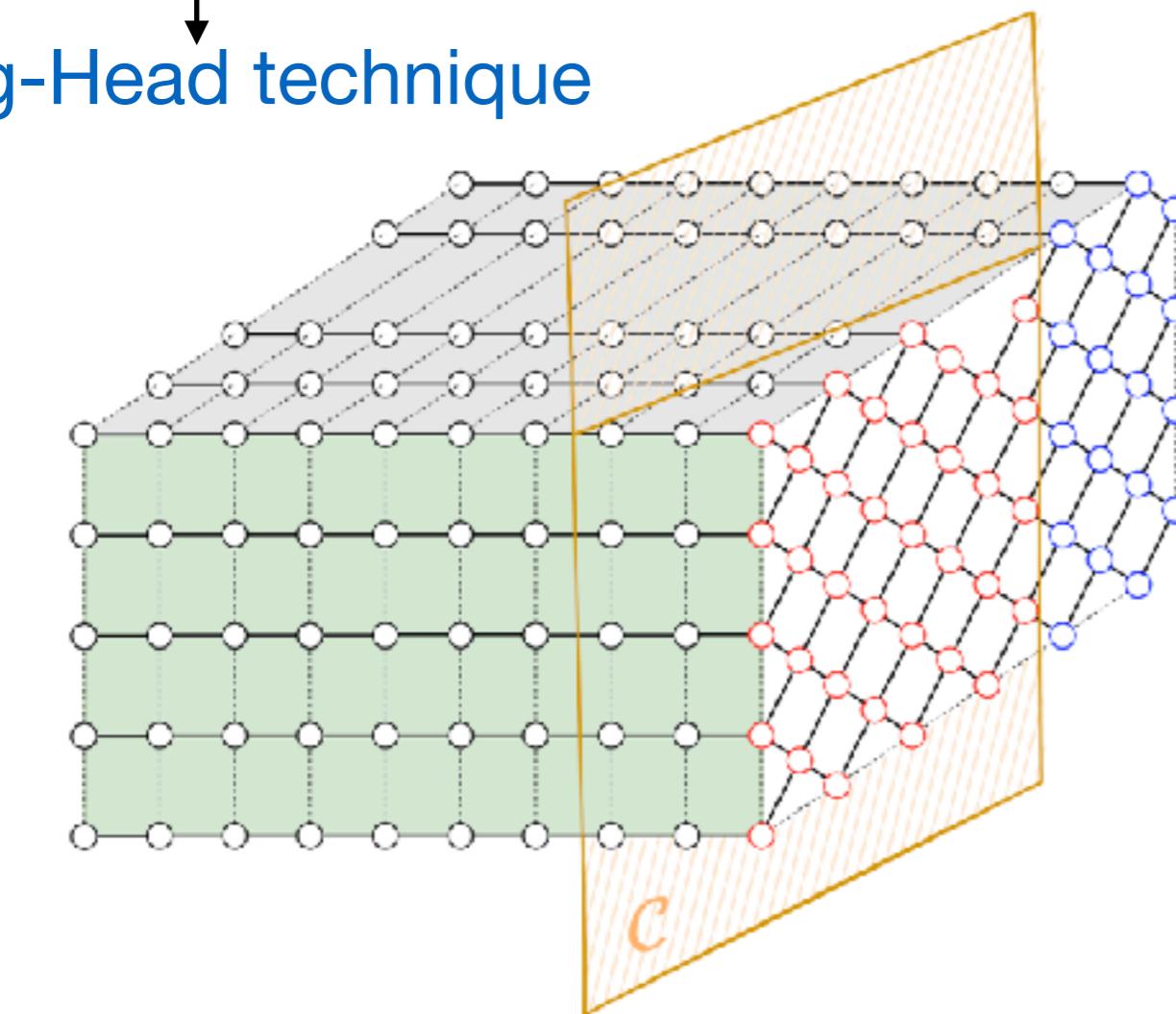


267 days / 64 amplitudes in one V100 GPU

Hyper-graph partitioning

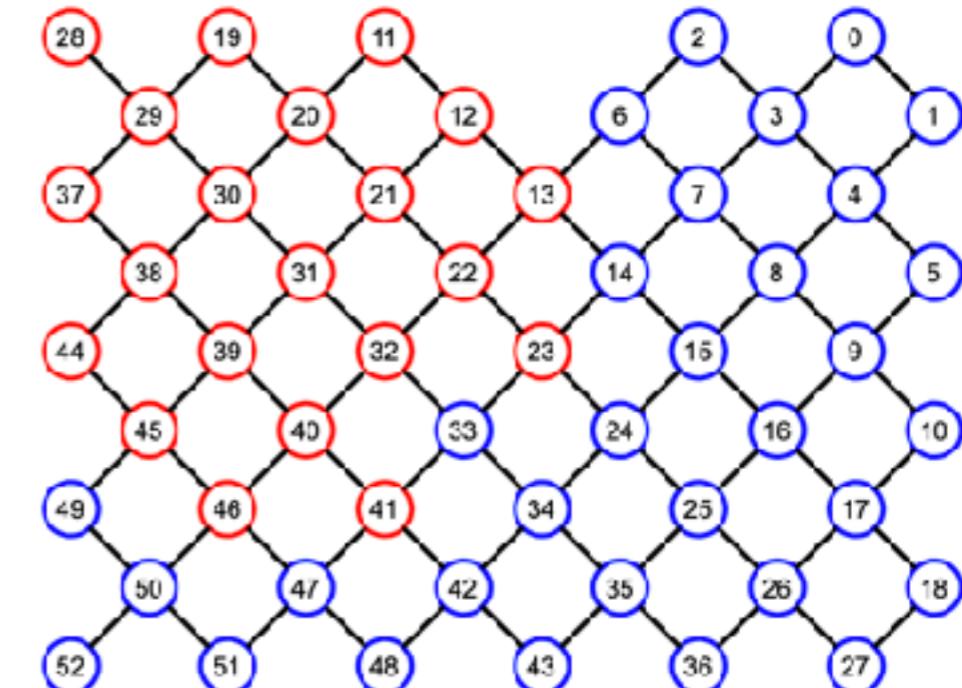
↓
Dynamic slicing

↓
Big-Head technique



bitstring

00000000000000000000100000110100000010000100000000
00100001000000
0000000000000111000001111100001111100110000000
0000000000000111000001111100001111000101000000
00000000000000000000000000000000100000000011100010000000



#subtasks	S_{total}	T_{sub}	T_{head}	T_{tail}	T_{total}
2^{23}	2^{30}	5.37×10^{11}	4.51×10^{18}	2.87×10^{15}	4.51×10^{18}

5 days / 2^{21} amplitudes in 64 GPU

Hyper-graph partitioning

Dynamic slicing

Big-Head technique

Low-rank structure in fSim gate

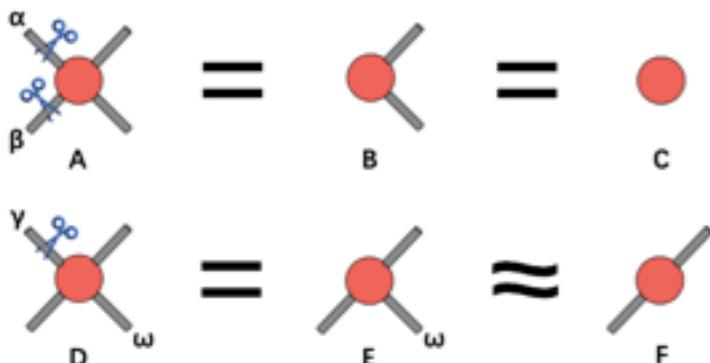


FIG. 2. Two situations that we can detect the low-rank structures in our method. (Top:) When two indices α, β are pinned to 0 for the fSim gate, the result is a rank-one matrix B , effectively equals to a scalar $c = 1$ in tensor network contractions. (Bottom:) When one index γ of a fSim gate is pinned, the resulting tensor E has decomposition rank 2 but very imbalanced singular value spectrum $(\sqrt{\sin^2(\theta) + 1}, \cos(\theta))$ with $\theta \approx \pi/2$ in Sycamore circuits. Thus pinning the index ω gives a slightly decrease to the fidelity.

Solving the sampling problem of the Sycamore quantum supremacy circuits

Feng Pan,^{1,2} Keyang Chen,^{1,3} and Pan Zhang^{1,*}

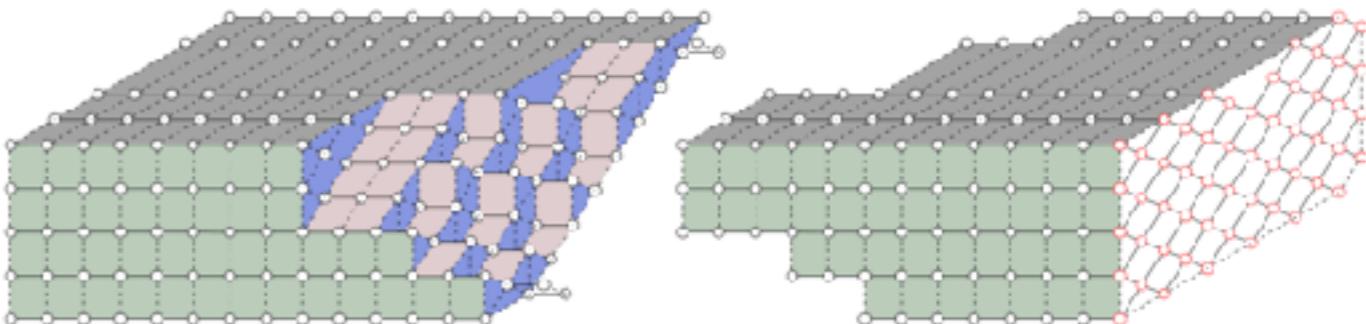
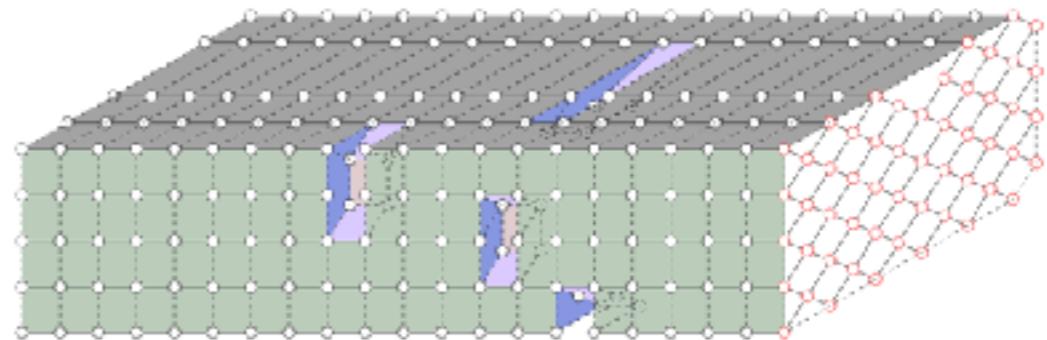


FIG. 4. The 3-dimensional tensor network \widehat{G} (corresponding to the Sycamore circuit of $n = 53$ qubits, $m = 20$ cycles) is split into two parts, $\widehat{G}_{\text{head}}$ (left) and $\widehat{G}_{\text{tail}}$ (right).

$$\text{fSim}(\theta, \phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -i \sin \theta & 0 \\ 0 & -i \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & e^{-i\phi} \end{bmatrix}. \quad (1)$$

$$F_{\text{estimate}} \approx 2^{-K} \prod_{i=1}^k (\sin^2(\theta_i) + 1)/2.$$

$$\prod_{i=1}^{21} (\sin^2(\theta_i) + 1)/2 \approx 0.9565$$

$$F_{\text{estimate}} = 2^{-8} \times 0.9565 \approx 0.0037.$$

Specifically, the parameters in Google's experiments [1] are tuned to $\theta \approx \pi/2$ in order to keep the decomposition rank equal to 4 with a near-flat spectrum. This setting increases

computations. Using our algorithm the simulation for the Sycamore circuits with $n = 53$ qubits and $m = 20$ cycles is completed in about 15 hours using 512 V100 GPUs. There

15 hours / 2^{21} amplitudes in 512 V100 GPU

Hyper-graph partitioning

↓
Dynamic slicing

↓
Big-Head technique

↓
Low-rank structure in
fSim gate

↓
Three-level parallelization
on 42 million cores

Specific Types of Contributions
ACM Gordon Bell Prize
Innovations in applying high-performance computing to science, engineering, and large-scale data analytics

Award Winners Nominations Committee Members

Closing the “Quantum Supremacy” Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer

Yong (Alexander) Liu^{1,3}, Xin (Lucy) Liu^{1,3}, Fang (Nancy) Li^{1,3}, Haohuan Fu^{2,3}, Yuling Yang^{1,3}
Jiawei Song^{1,3}, Pengpeng Zhao^{1,3}, Zhen Wang^{1,3}, Dajia Peng^{2,3}, Huarong Chen^{1,3}
Chu Guo⁴, Heliang Huang⁴, Wenzhao Wu³, Dexun Chen^{2,3}

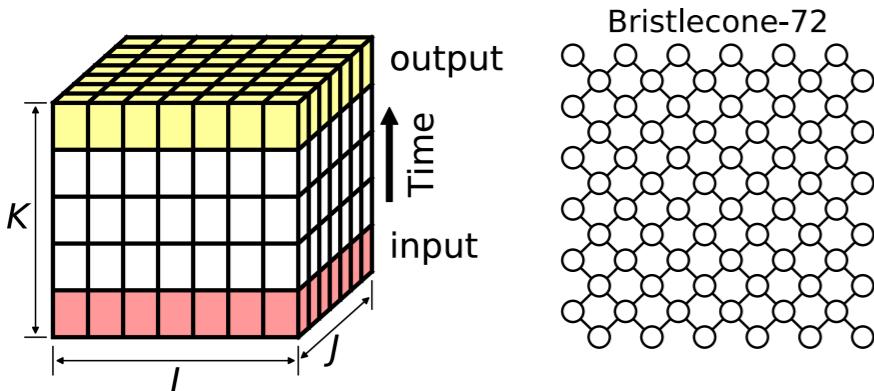
1. Zhejiang Lab, Hangzhou, China
2. Tsinghua University, Beijing, China
3. National Supercomputing Center in Wuxi, Wuxi, China
4. Shanghai Research Center for Quantum Sciences, Shanghai China

Time needed to sample Sycamore	
our simulation	304 seconds
physical Sycamore [1]	200 seconds
Summit[1]	10,000 years
Summit[23]	2.55 days (estimated)
Ali_Cloud[14]	19.3 days (estimated)
60_GPUs(Pan)[21]	5 days

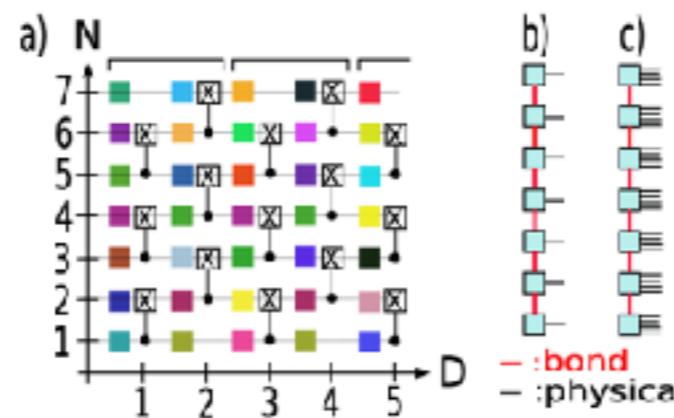
Real-time classical simulation of
Google’s circuit in **304 seconds**
by 42 million cores



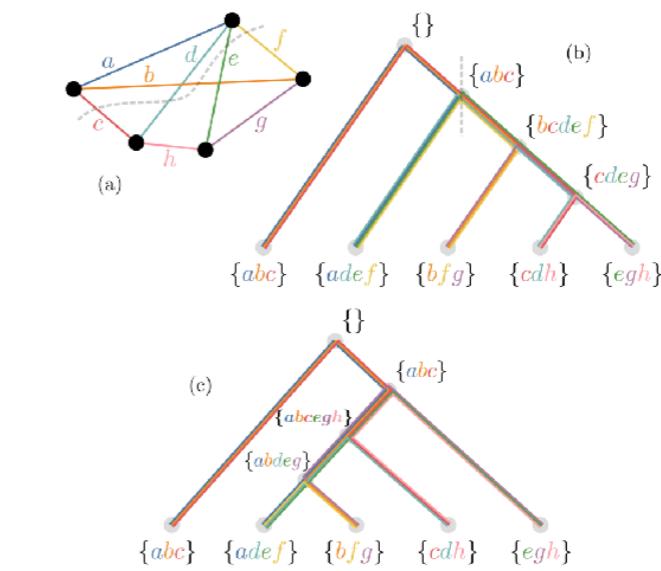
Classical Simulator Based on Tensor Networks



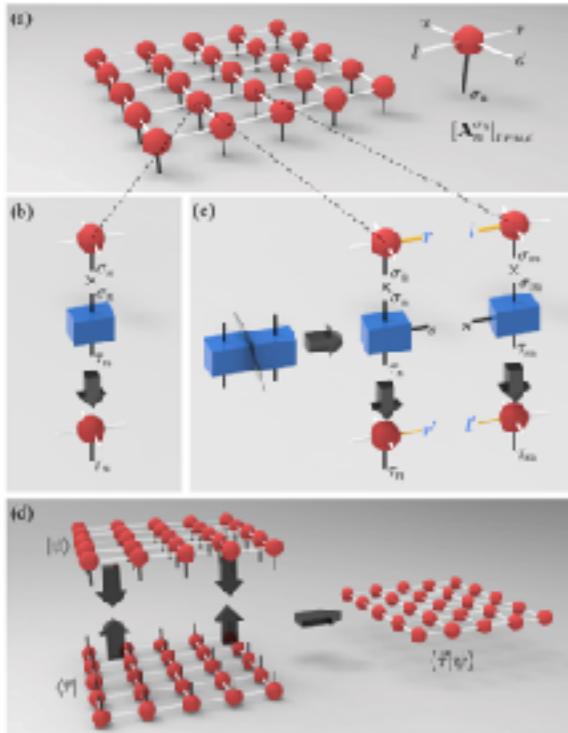
Villalonga et al, 1811.09599



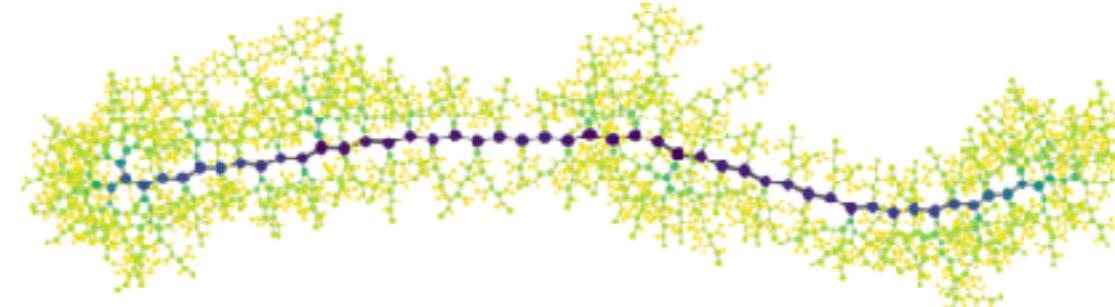
Yiqing Zhou et al. arXiv:2002.07730



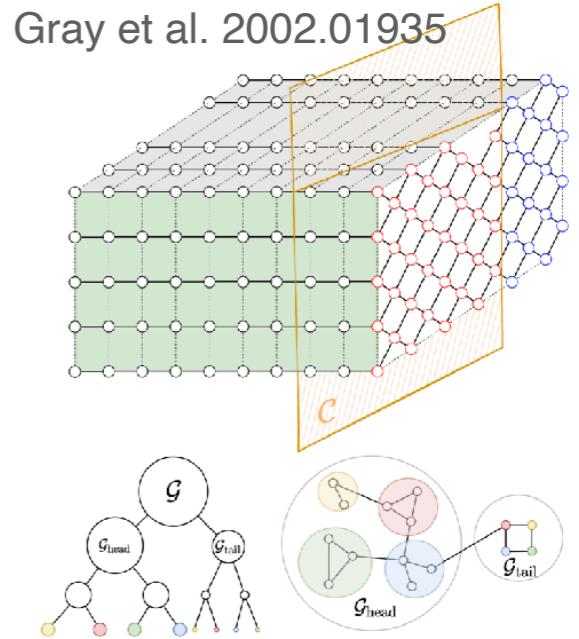
Johnnie Gray et al. 2002.01935



Chu Guo et al, Phys. Rev. Lett. 123, 190501



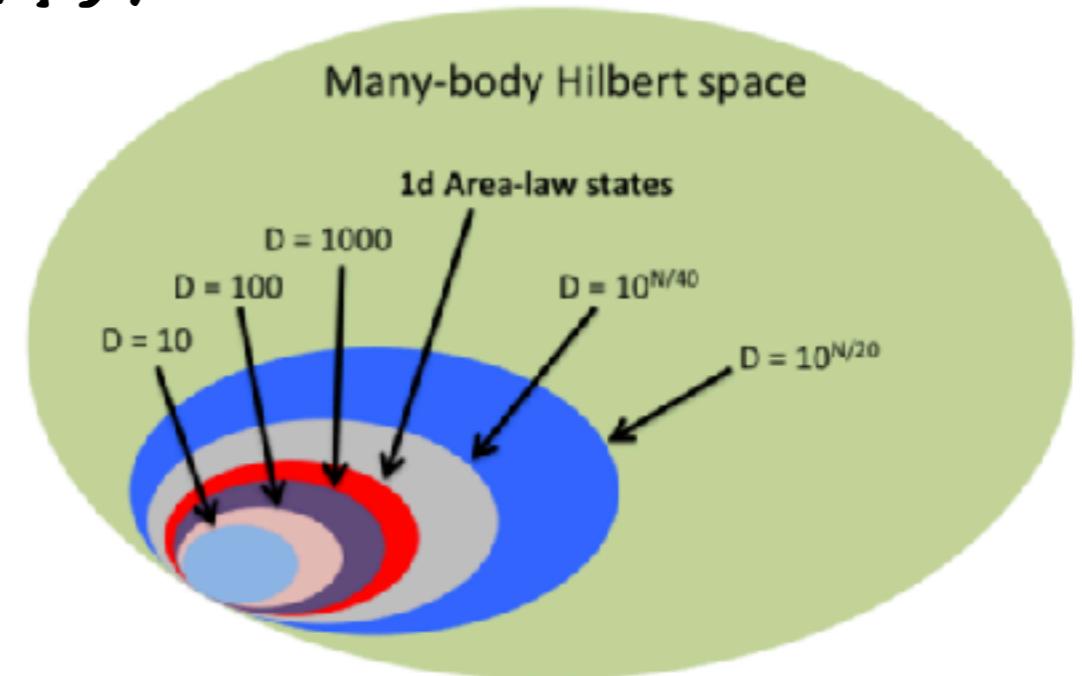
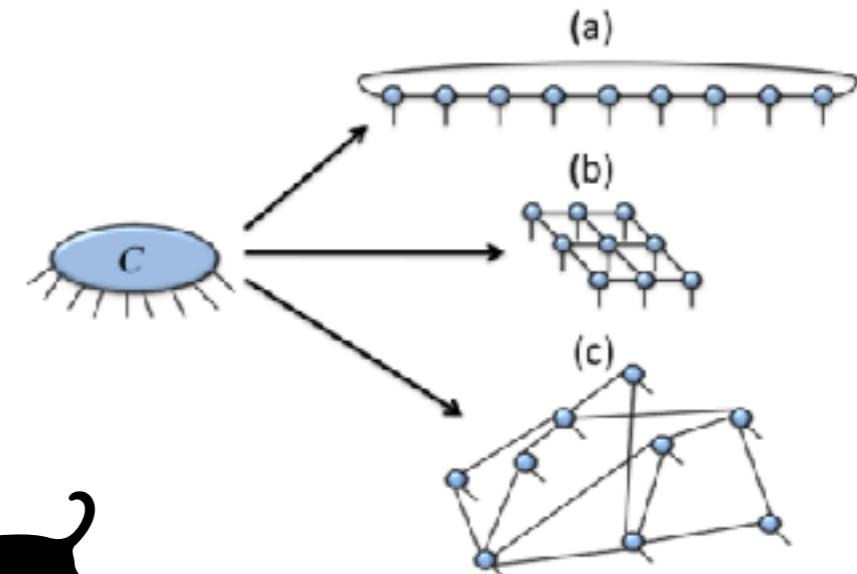
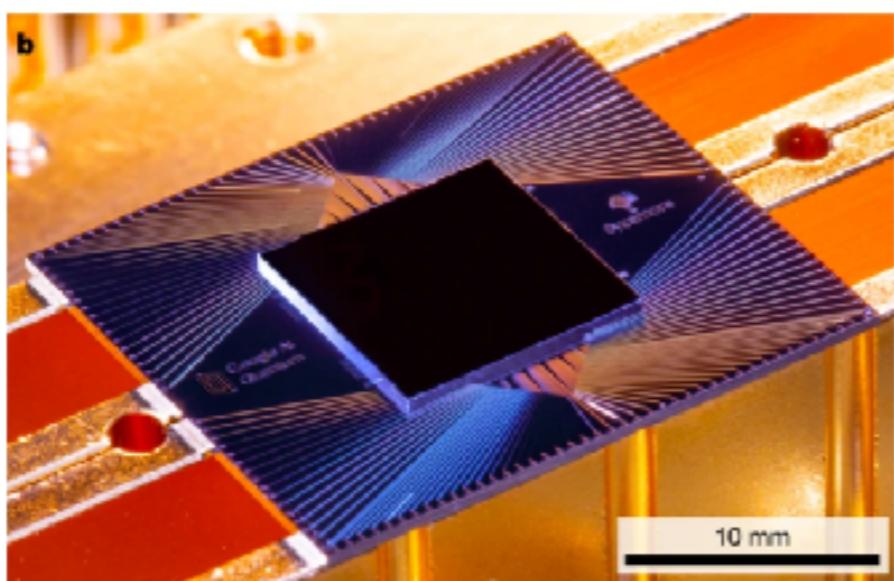
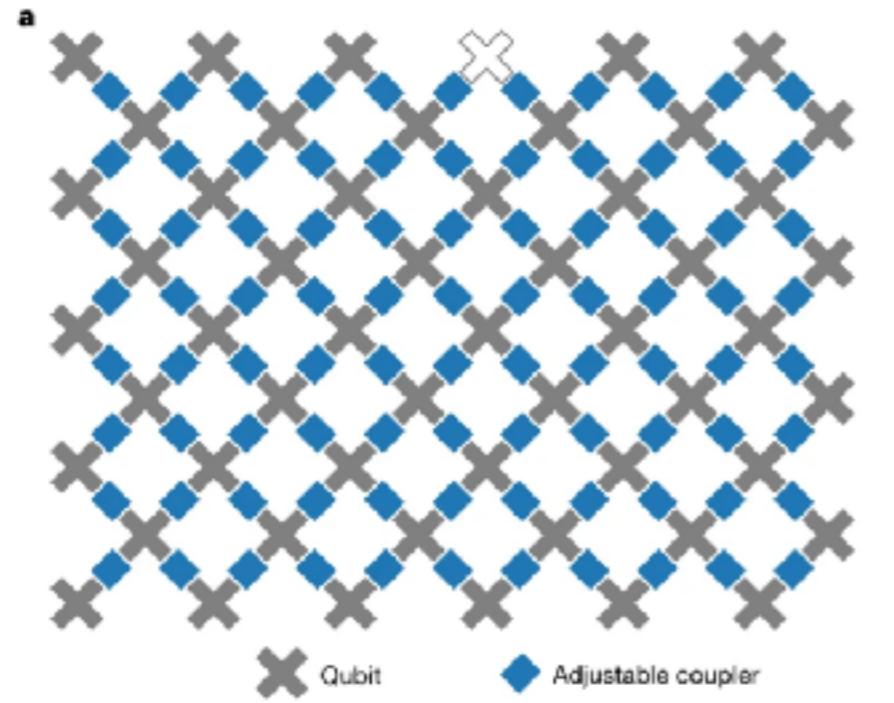
Cupjin Huang et al, 2005.06787



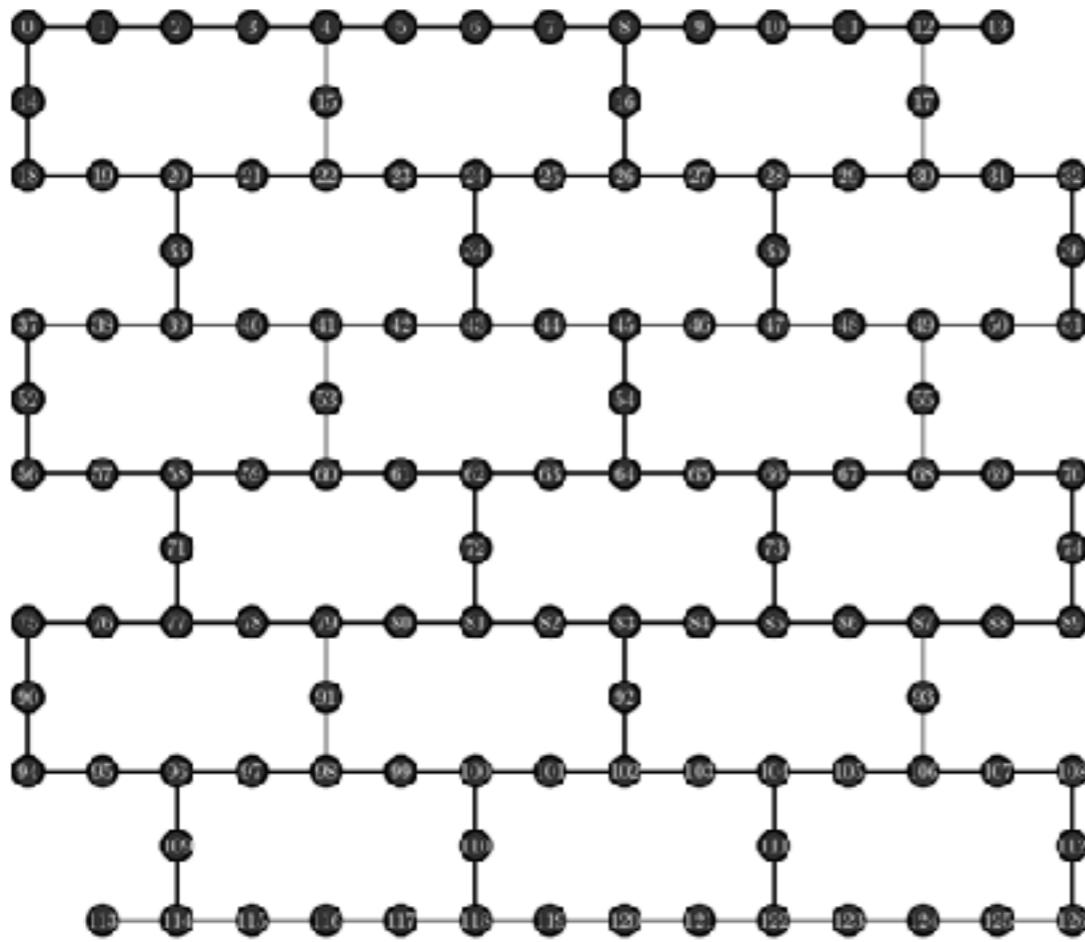
Feng Pan et al. 2103.03074

The above algorithms are all based on tensor networks

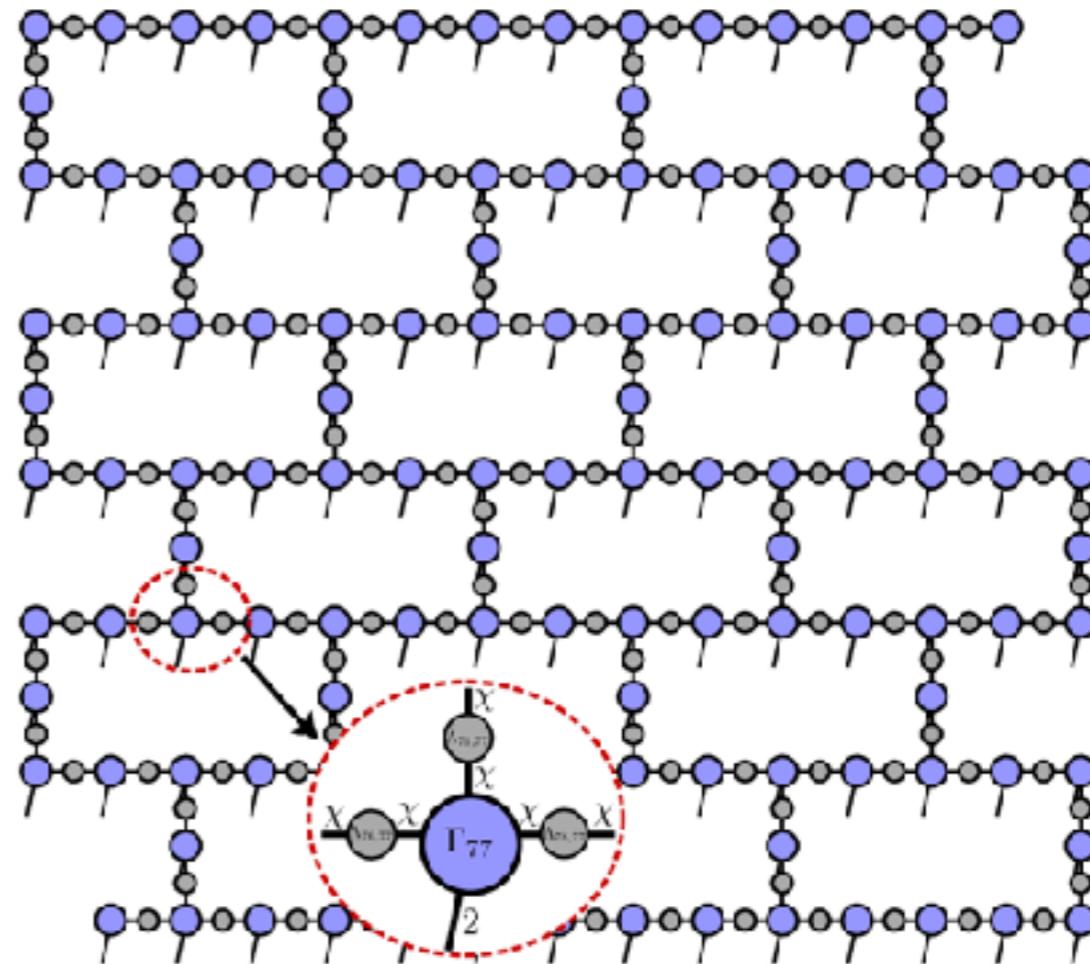
- NISQ circuits introduced a wave state ansatz, the output state is specialized.
- Locality, lower entanglement entropy.
- The ansatz led to the efficiency of the classical simulating algorithms of tensor networks



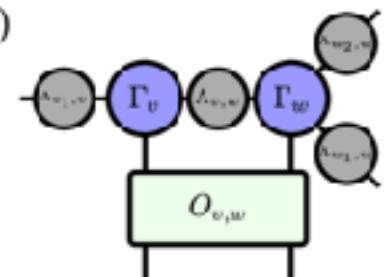
Eagle Processor Layout :



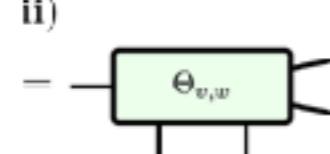
Classical Tensor Network Ansatz :



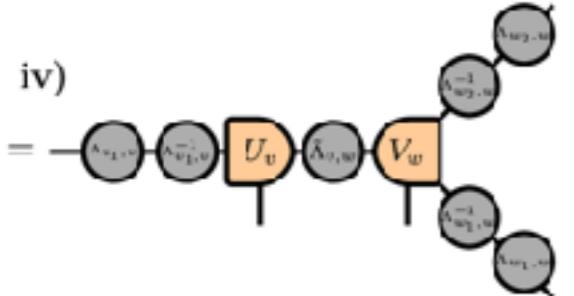
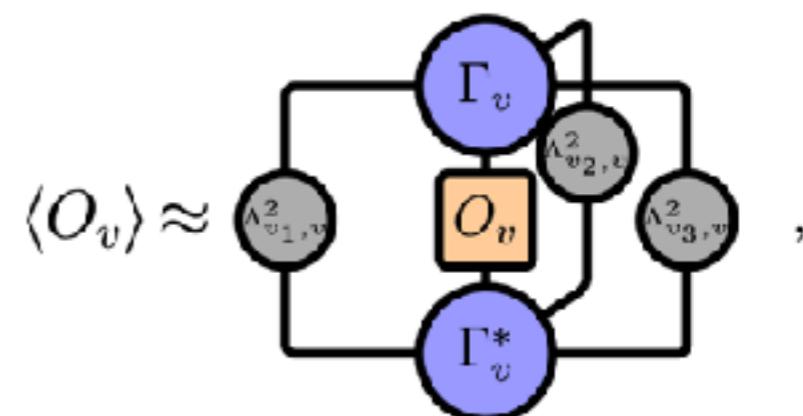
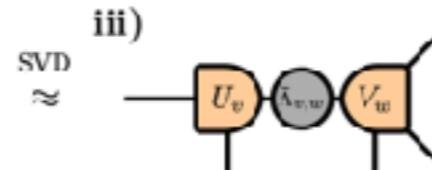
i)



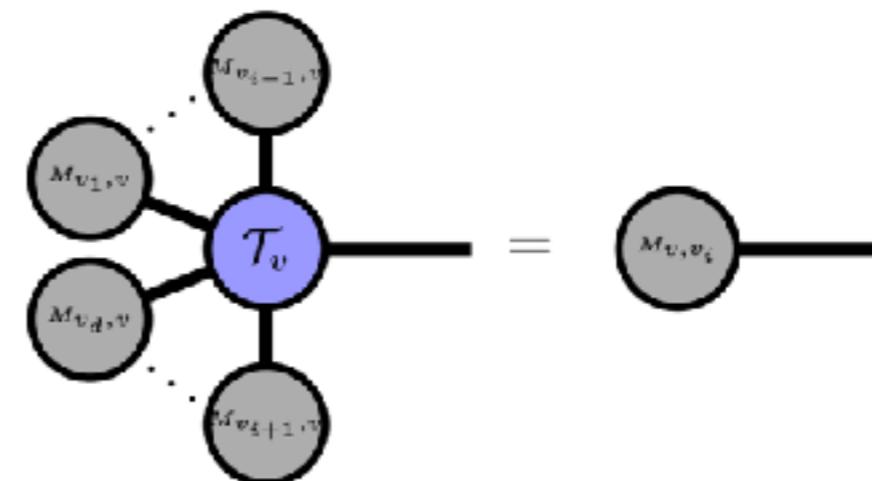
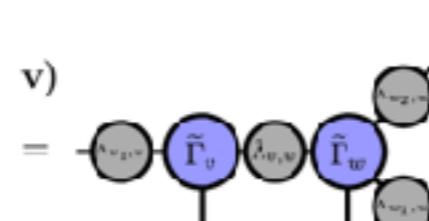
ii)



iii)



v)



“Specialty” of Noisy VQAs

- If system exhibits **sparsity in the Pauli basis**, OBPPP could leverage it to significantly accelerate computations.
- The contributions of **high-weight Pauli paths could be heavily suppressed in the presence of Pauli type noise**(such as depolarizing noise), thereby limiting truncation errors.

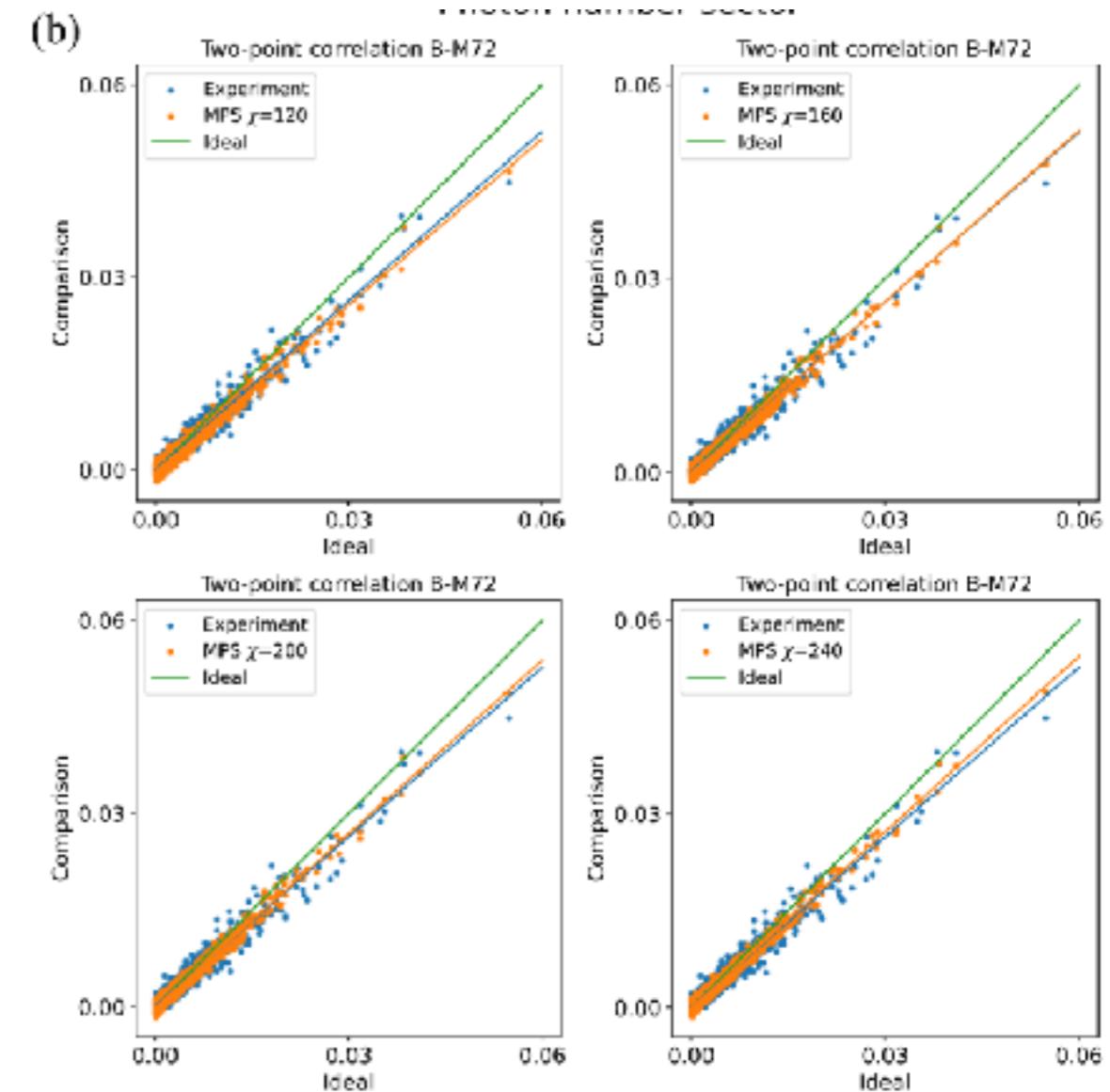
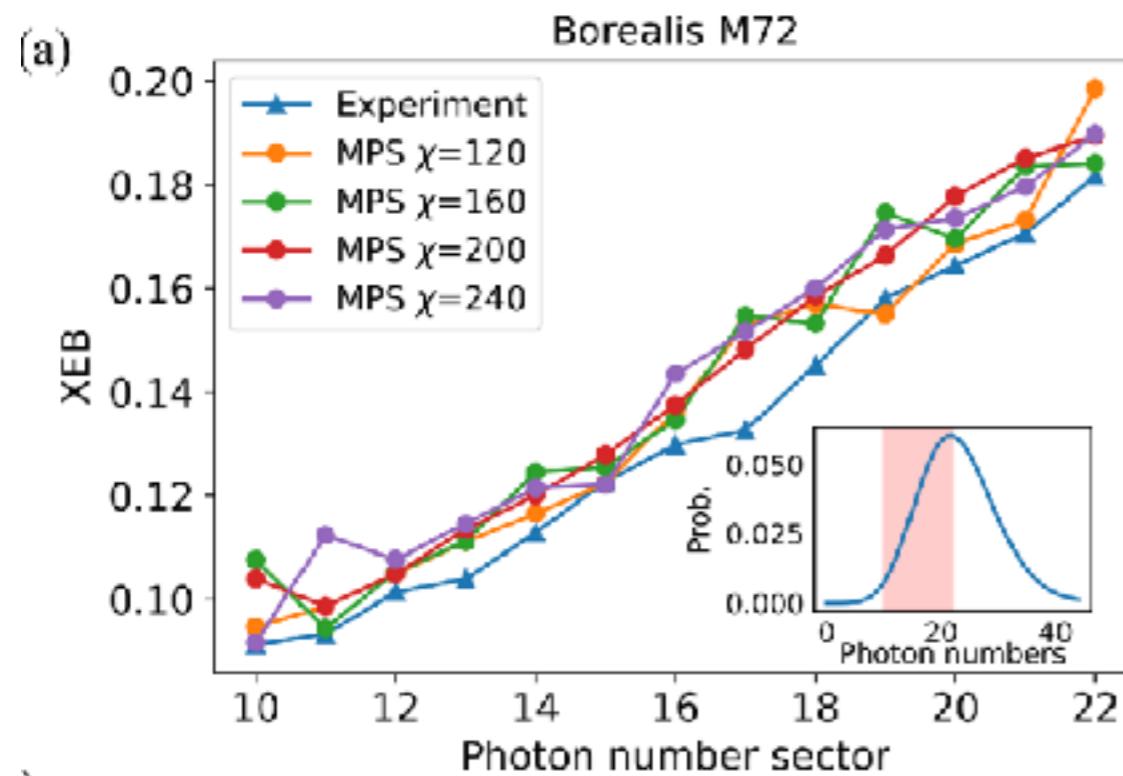
Gaussian Boson Sampling

- ▶ In December 2020 , the team of Pan Jianwei of the University of Science and Technology of China realized quantum superiority based on Bose sampling
 - 200 seconds vs 600 million years



- ▶ It should be noted that the above two computational tasks do not seem to have obvious application value
 - Existing machines cannot run Shor's algorithm or Grover's algorithm on a large scale

Simulating Boson Sampling by TN



we demonstrate for large-scale experiments that our sampler matches the ground-truth two-point and higher-order correlation functions better than the experiment does, exhibiting evidence that our sampler can simulate the ground-truth distribution better than the experiment can.