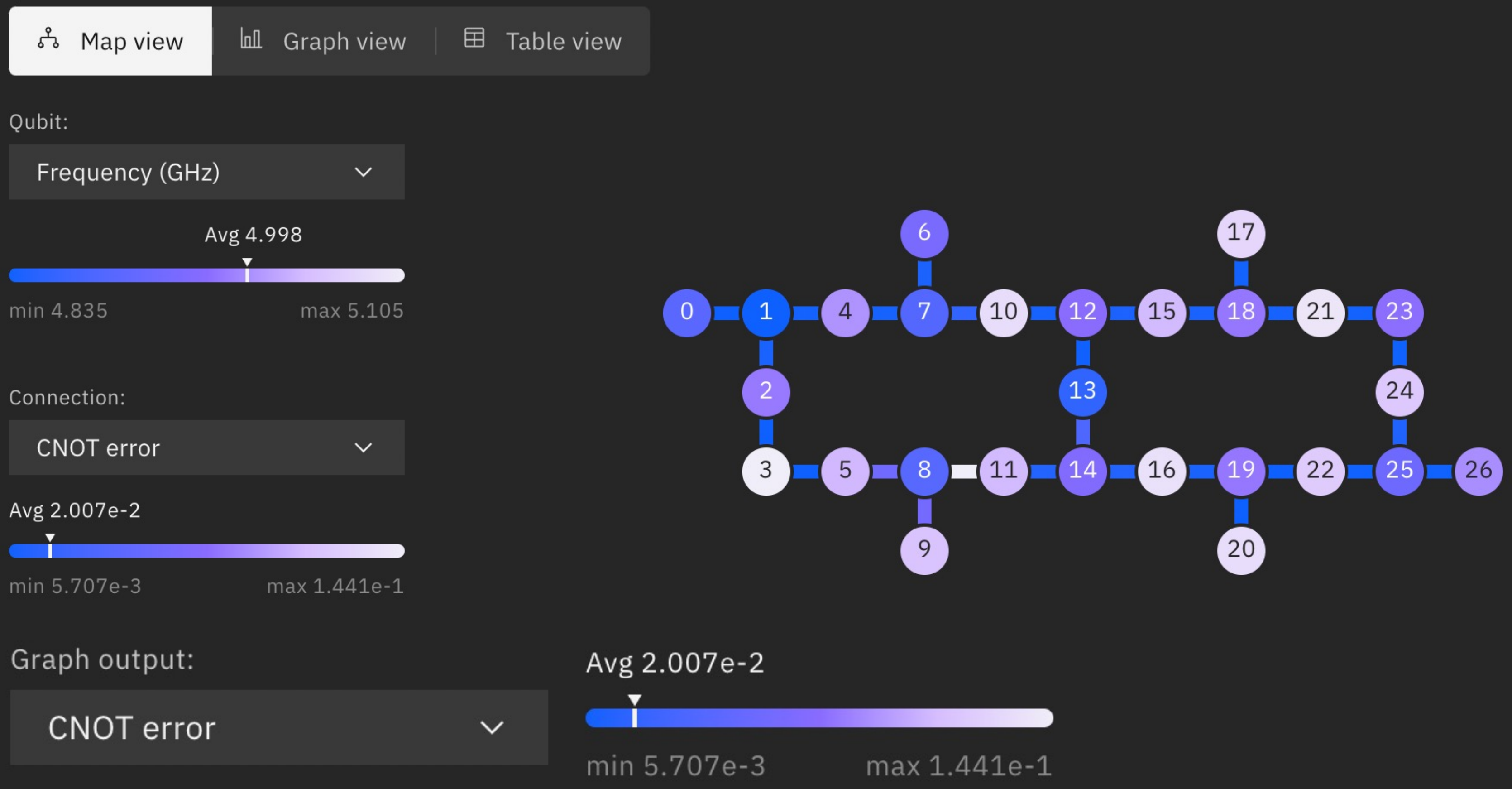
Recent Progress in Quantum Benchmarking

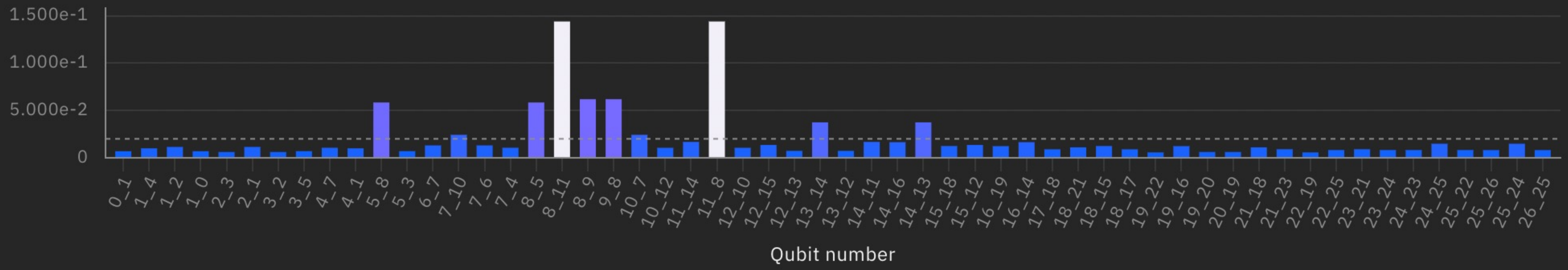
Yunchao Liu (UC Berkeley)

joint works with Senrui Chen, Matthew Otten, Roozbeh Bassirianjahromi,
Alireza Seif, Bill Fefferman, Liang Jiang

arxiv: 2206.06362, 2105.05232



Quantum Benchmarking: design algorithms to learn about noise in quantum devices

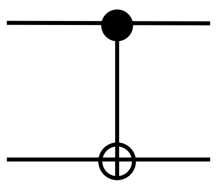


Why benchmarking?

- We need to know what the noise look like in order to
  - Further reduce the noise and build better quantum computers
  - Perform error mitigation in near-term experiments
  - Design suitable error correcting codes for FTQC
- Current status:
  - We have mature methods to estimate total error on a single gate (RB)
  - Single-qubit gates are good ($10^{-3} \sim 10^{-4}$ error)
  - 2-qubit gates are noisy (10^{-2} error)
- Perform benchmarking → obtain knowledge about noise → use the knowledge to reduce noise

What is noise?

What we want

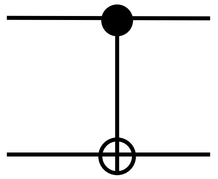


What happened in experiment

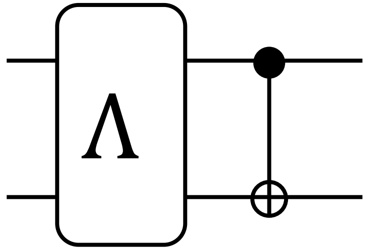
Some other operation

What is noise?

What we want



What happened in experiment



Λ is unknown

$||\Lambda - Id||$ is total error

We have mature methods to estimate total error on a single gate (RB)

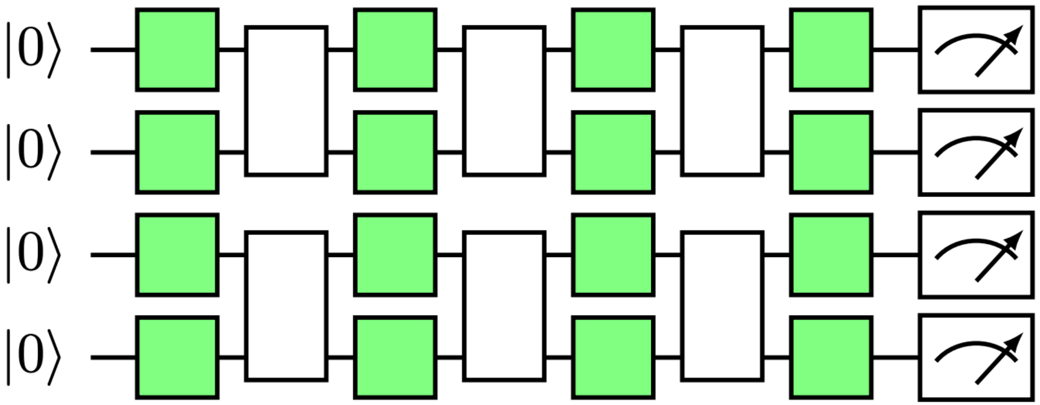
→ Learn more information on 2 qubits (Part I)

→ Learn total error on more qubits (Part II)

Challenges in benchmarking

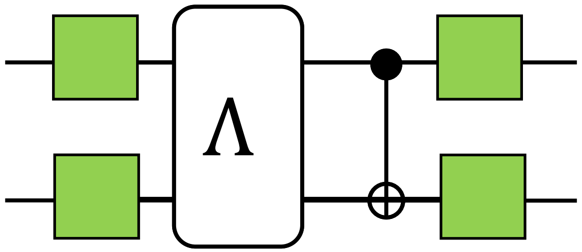
- A general quantum channel is too complicated
- Use Pauli twirling for Clifford circuits

General quantum channel —(Pauli twirling)→ Pauli noise



Can simplify the noise to a Pauli channel $\{p_a\}$, $a \in \{I, X, Y, Z\}^n$ without changing the logic of the circuit

An outstanding issue



- Focus on a single CNOT gate
- We know the total error $1 - p_{II} = p_{IX} + p_{IY} + \cdots + p_{ZZ}$
- Next: only need to learn this 16-dimensional distribution

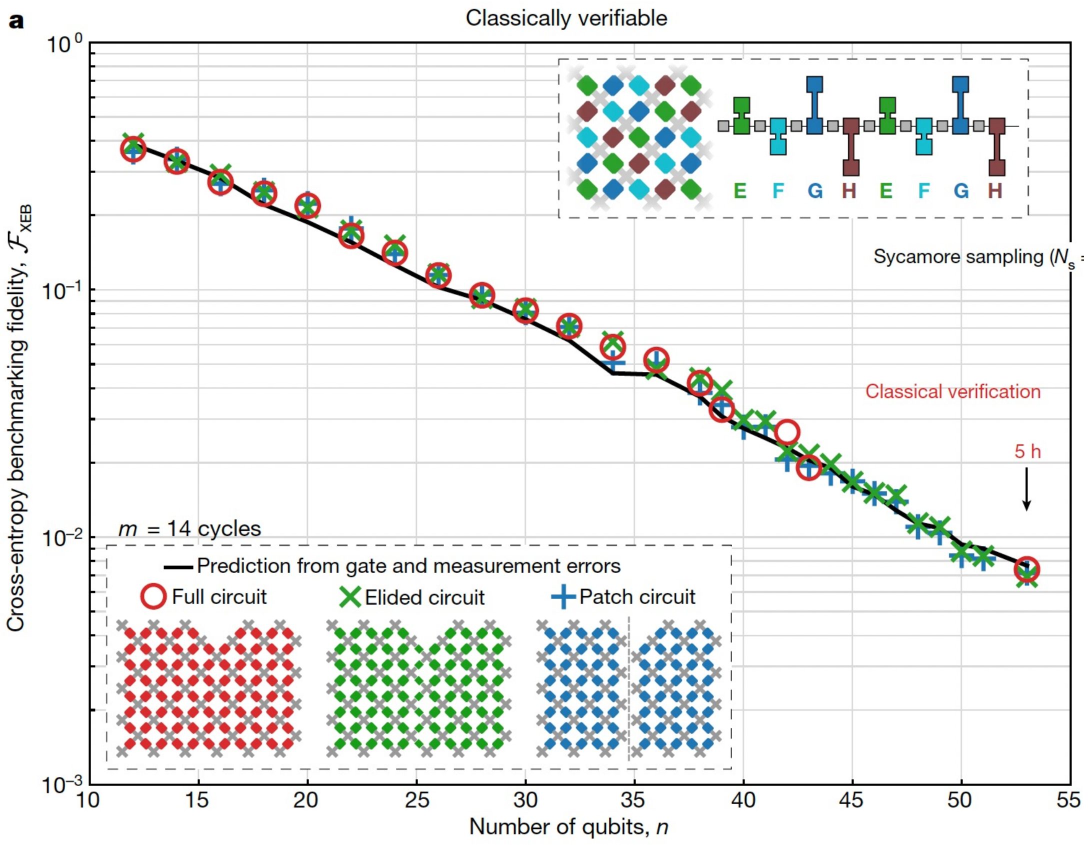
- Even this is not doable!
  - seems to be a fundamental issue
- **Part I of this talk: a precise understanding of what information about noise is learnable for Clifford gates**

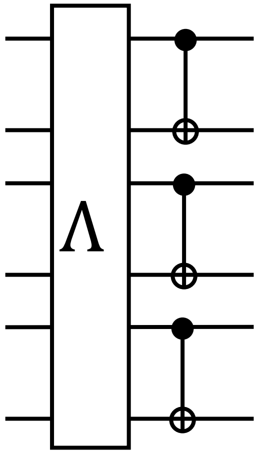
Challenges in benchmarking

- Scalable benchmarking: for large system size (20+ qubits), we want to efficiently estimate the total error on the entire system
  - Previously this is only known for Clifford gates

- **Part II of this talk: scalable benchmarking of non-Clifford gates**
  - Pauli twirling doesn't work in general, but here we still achieve some effective twirling
  - Still think of noise as Pauli channel, want to learn the total error $1 - p_{I^{\otimes n}}$

Why is the total error interesting?





- It is non-trivial: cannot just add up the total error on each gate
  - Because errors can be correlated across gates
- Total error can provide deep insights into the noise model
- Claim: Google's data suggests the noise in their device was uncorrelated
  - We will understand this better by thinking about total error

Outline

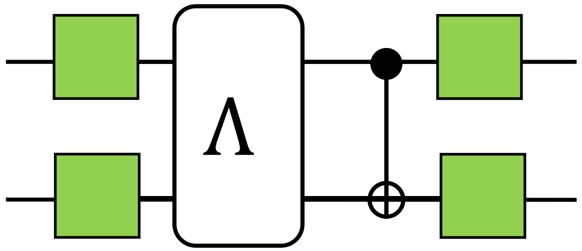
- Always think about noise as a Pauli channel $\{p_a\}$, $a \in \{I, X, Y, Z\}^n$

- **Part I of this talk: a precise understanding of what information about noise is learnable for Clifford gates**
  - Understand CNOT gate
- **Part II of this talk: scalable benchmarking of non-Clifford gates to learn total error**
  - Understand Google's claim

Part I: Clifford benchmarking

Overview

General CPTP noise —Randomized compiling→ Pauli noise



Goal: learn the 16-dimensional probability distribution $\{p_a\}$, $a \in \{I, X, Y, Z\}^2$

What we already have:

Total error $1 - p_{II}$
$= p_{IX} + p_{IY} + \cdots + p_{ZZ}$

What we want:

$p_{IX}, p_{IY}, p_{IZ}, \ldots, p_{ZZ}$

Current status:

Can learn some errors, not all

This talk:

All learnable information:

$p_{IX}, p_{XY}, p_{XZ}, p_{YY}, p_{YZ}, p_{ZI}, p_{ZX},$
$p_{IY} + p_{ZY}, p_{IZ} + p_{ZY}, p_{IY} + p_{ZZ},$
$p_{XI} + p_{XX}, p_{XX} + p_{YI}, p_{XI} + p_{YX}$
(13 equations)

CNOT has 13 learnable degrees of freedom
+ 2 unlearnable degrees of freedom

What's the issue?

- Intrinsic symmetry in a quantum system: gauge freedom
- Example: consider a trivial system with noisy state preparation and measurement
  - We prepare $|0\rangle$, measure, see 1 with probability 5%
  - It could be the case that all 5% noise comes from state preparation (SP)
  - It could be the case that all 5% noise comes from measurement (M)
  - It could be the case that 2% comes from M, 3% comes from SP…

Can't tell the difference

5% SP, 0% M → 0% SP, 5% M

Can move from one point to another along the manifold without changing experiment outcomes, such an operation is called gauge transformation

Our noise model

- Noise model: initial states $\{\rho_i\}$, POVM $\{E_j\}$, gates $\{G_k\}$ are all subject to unknown quantum noise
  - Standard assumption: single-qubit gates are perfect, total error is sufficiently small
- The gauge transformation can be written as
  - $\rho_i \mapsto \mathcal{M}(\rho_i), E_j \mapsto E_j \circ \mathcal{M}^{-1}, G_k \mapsto \mathcal{M} \circ G_k \circ \mathcal{M}^{-1}$
- This does not change measurement outcome statistics; therefore, two different noise models that are related by gauge transformation are indistinguishable by any quantum experiment

Learnable part:
Invariant under any gauge transformation;
Can be learned by an algorithm

Unlearnable part:
Variant under some gauge transformation;
Cannot be learned by any algorithm

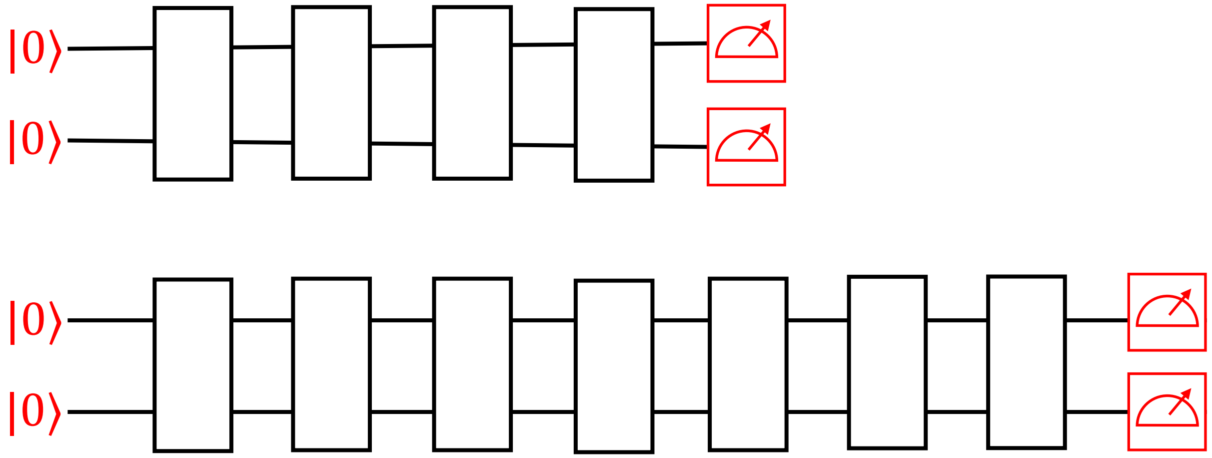
Trivial example

- The noise model has two degrees of freedom {SP, M}
- Learnable part = SP + M, invariant along the manifold

- Our goal: complete this classification for general gate noise

Main idea

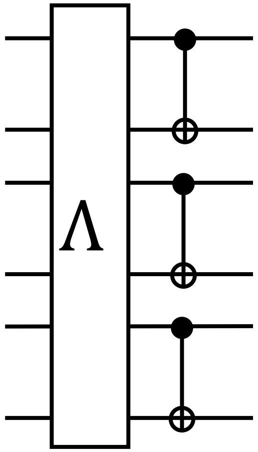
- The main idea of benchmarking: initial state and measurement only appear once in an experiment, but can apply a gate many times
- Exploit this asymmetry to obtain information about gate noise



- Observe different statistics in the two experiments
- The difference is only caused by gates
- Use this to obtain information about gate noise

Formalizing this idea for Pauli noise

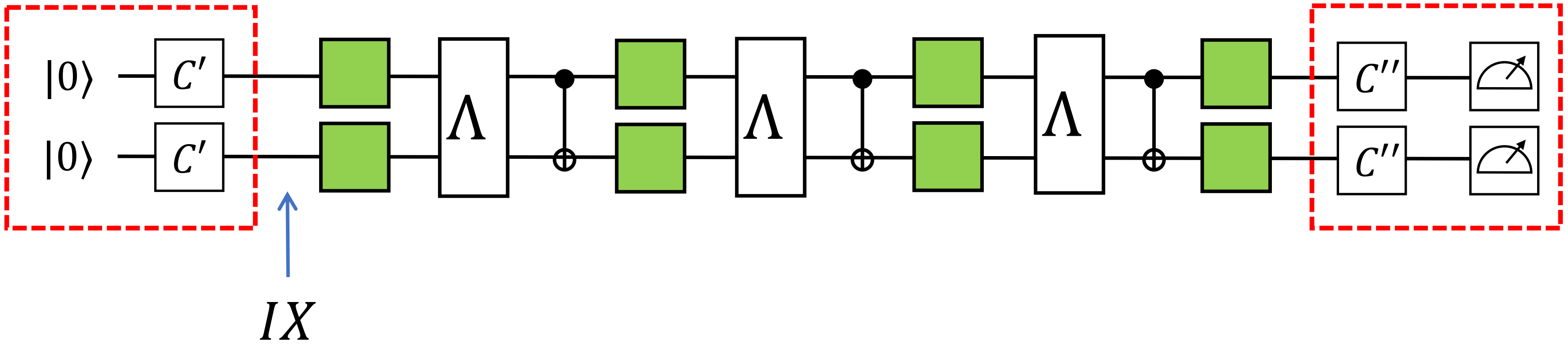
- Consider a n-qubit Pauli noise channel $\{p_a\}$ acting on a n-qubit Clifford
- Λ: $\rho \mapsto \sum_{a \in \{I,X,Y,Z\}^n} p_a P_a \rho P_a$
- Goal: learn the 4^n dimensional distribution $\{p_a\}$



- Idea: we will work in the Fourier domain $\{\lambda_a\}$
  - $\Lambda(P_a) = \lambda_a P_a$, $\lambda_a = \sum_b (-1)^{\langle a,b \rangle} p_b$ called Pauli fidelities
  - Next: learn Pauli fidelities (eigenvalues) $\{\lambda_a\}$ → reconstruct $\{p_a\}$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

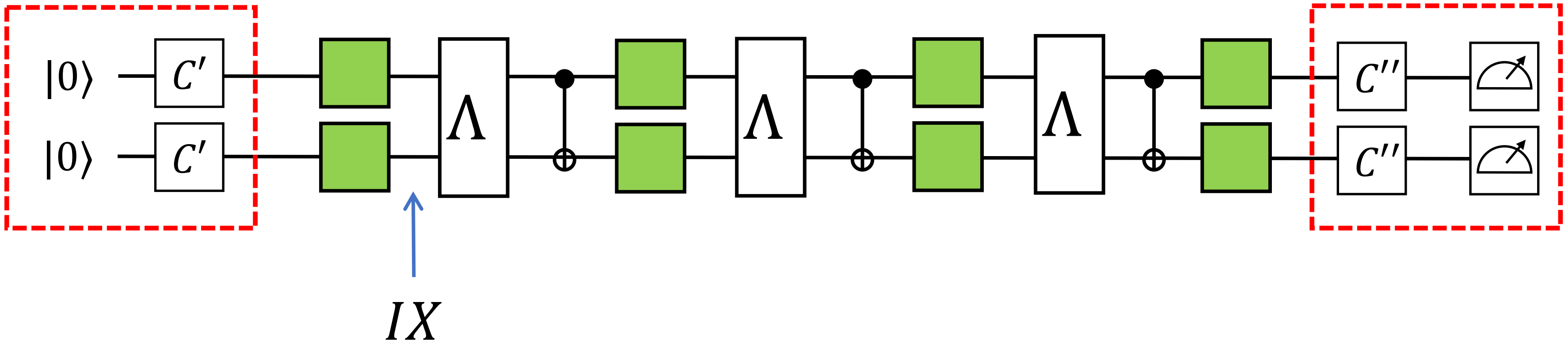
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

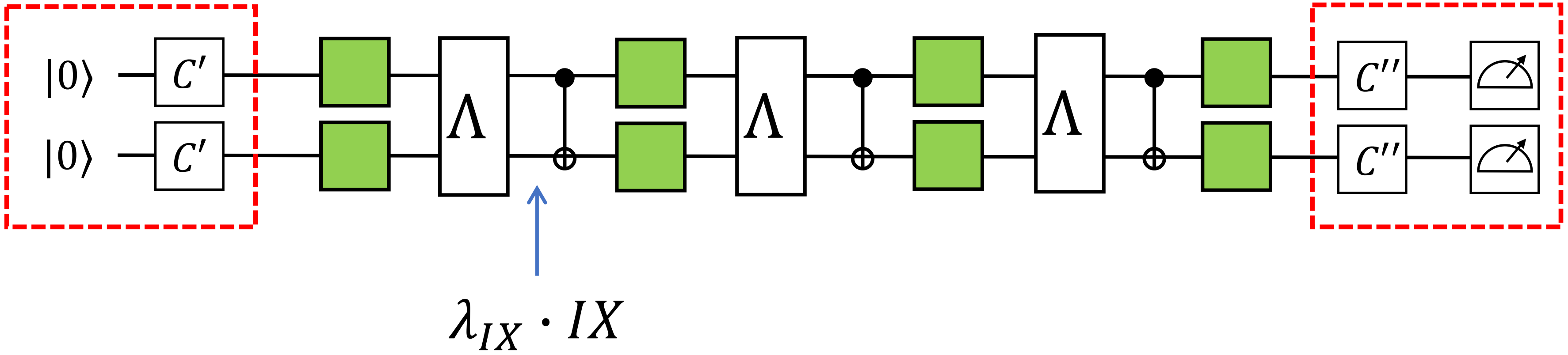
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

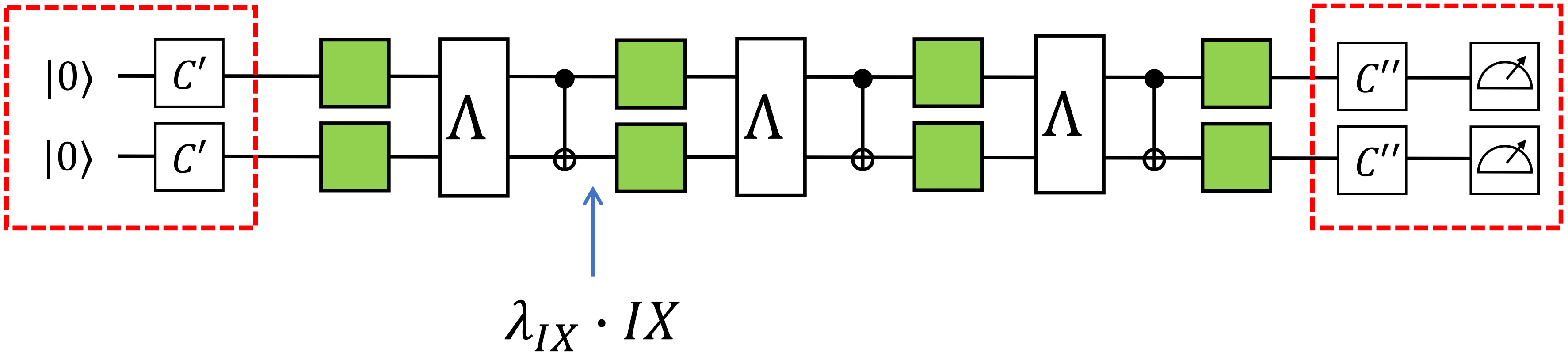
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

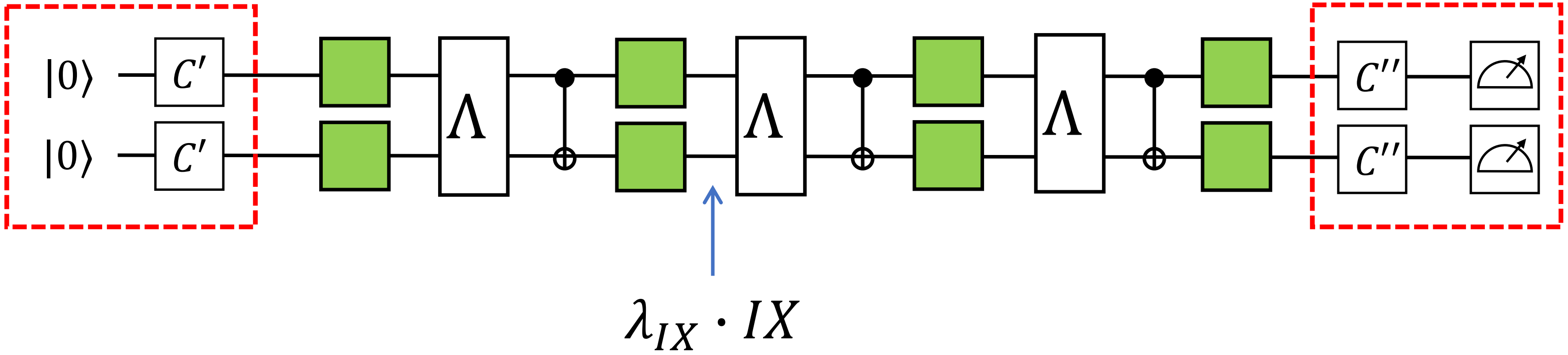
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

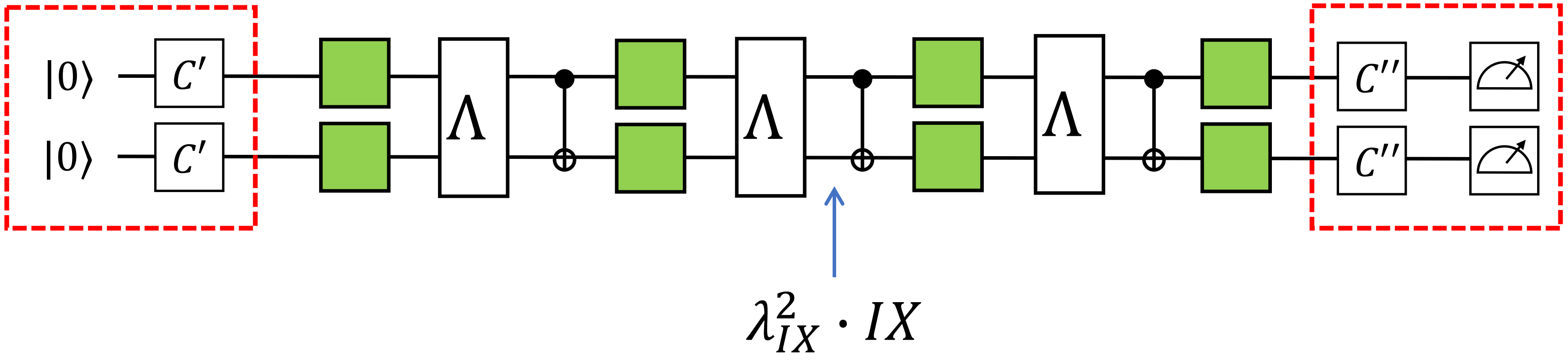
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

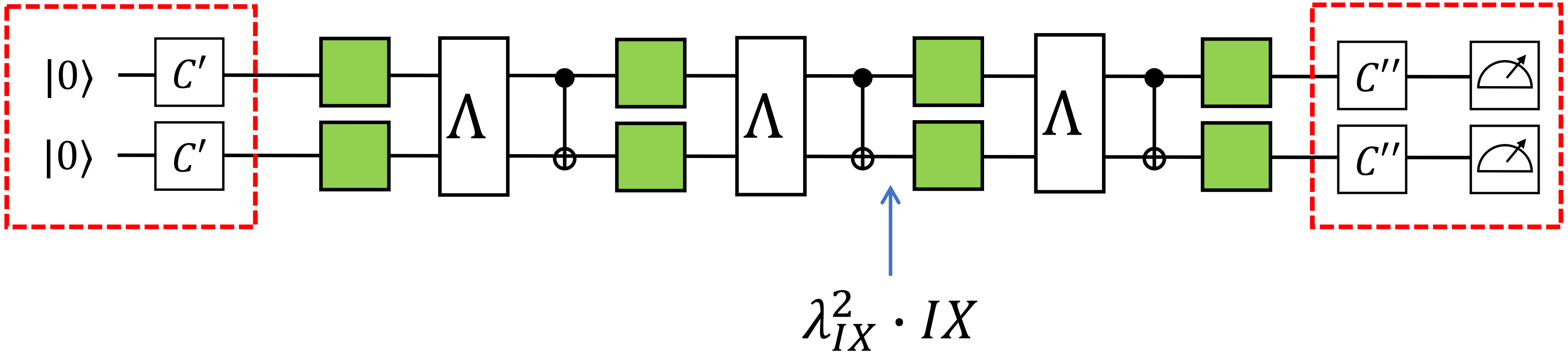
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

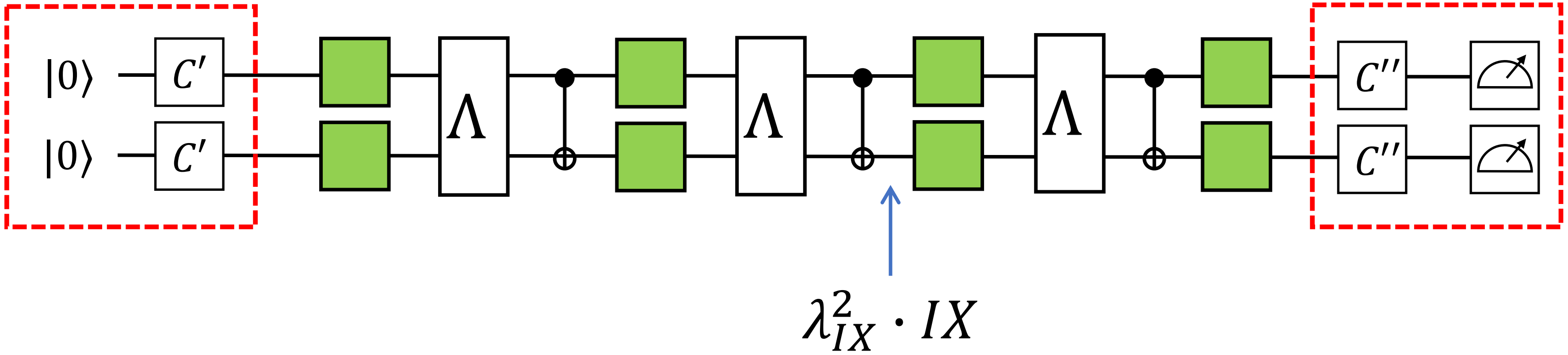
$\Lambda(P_a) = \lambda_a P_a$

In experiments, prepare +1 eigenstate of IX, estimate IX observable at the end, average over random Pauli

$\mathbb{E}\langle IX \rangle = A_{IX} \cdot \lambda_{IX}^d \rightarrow$ perform experiment at different $d \rightarrow$ learn λ_{IX}

Cycle benchmarking

- Cycle benchmarking [Erhard et al'19]: learning Pauli fidelities is the natural way to think about benchmarking



CNOT

$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$

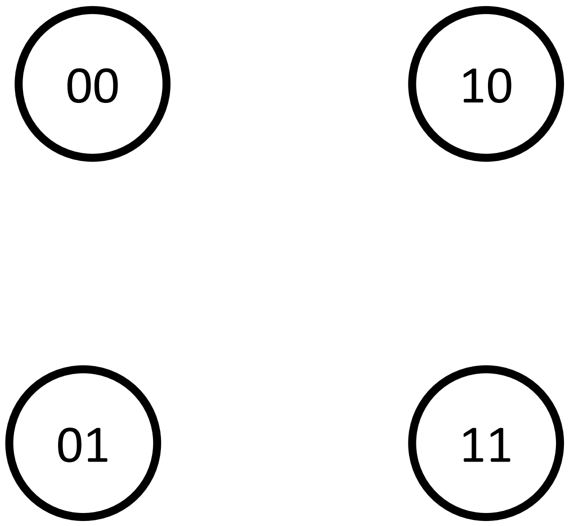
The original cycle benchmarking algorithm learns some specific Pauli fidelities and can be used to learn the total Pauli error

Overview of results

- We augment CB with a trick to learn more information
  - $\lambda_{IX}, \lambda_{ZI}, \lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}, \lambda_{IZ}\,\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}, \lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$
  - Anything beyond this is unlearnable
- This comes from the main result: classification of learnability using a graph representation

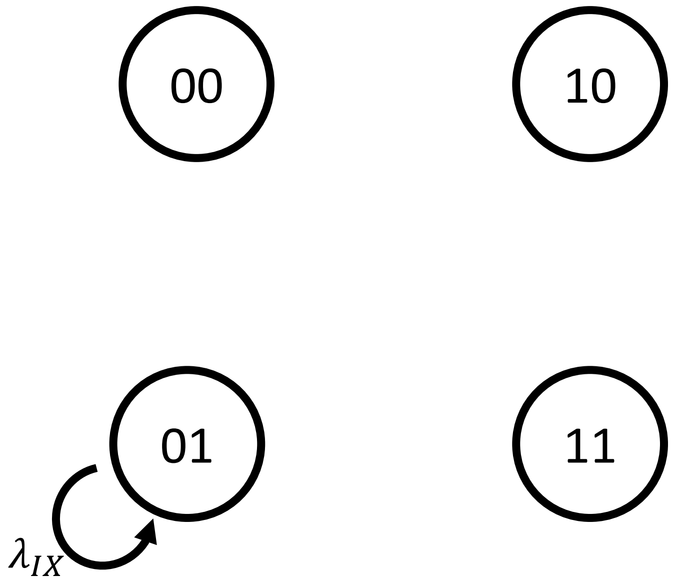
Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford



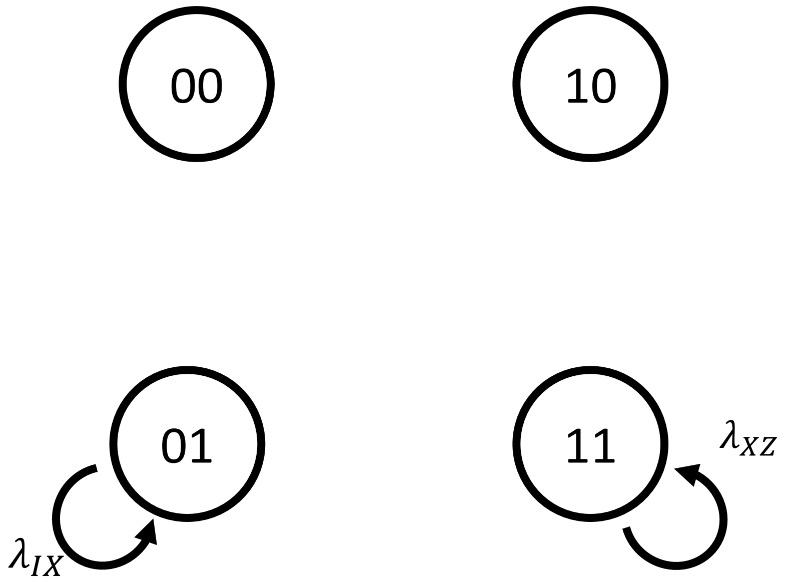
Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford
- CNOT: $IX \rightarrow IX$



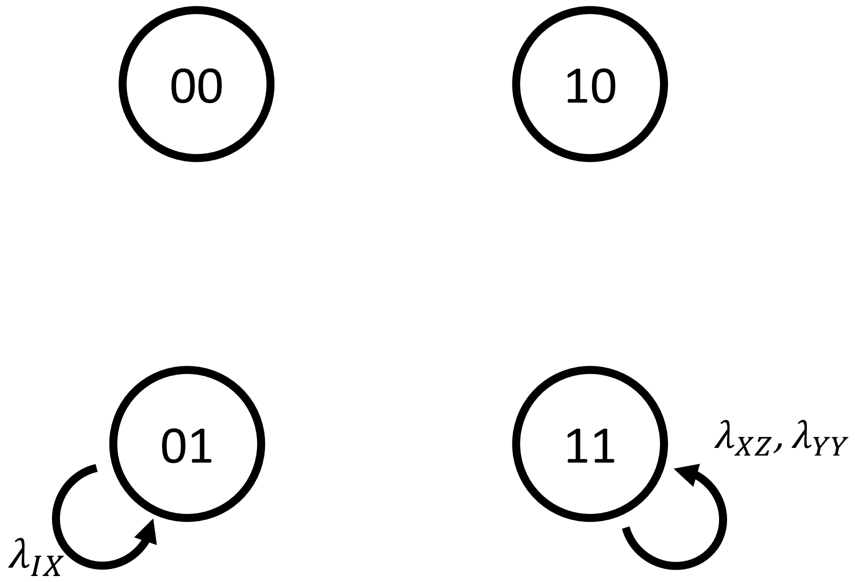
Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford
- CNOT: $IX \rightarrow IX$
- $XZ \rightarrow YY$



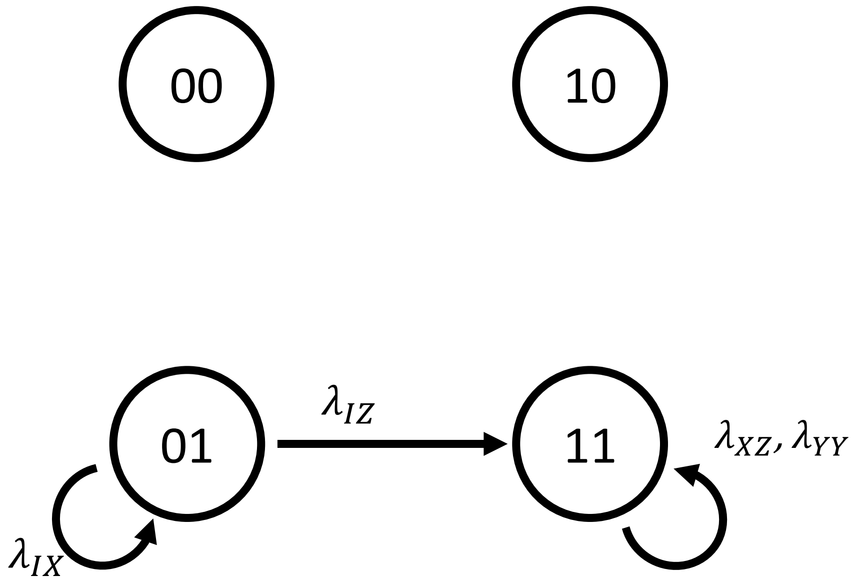
Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford
- CNOT: $IX \rightarrow IX$
- $XZ \rightarrow YY$
- $YY \rightarrow XZ$



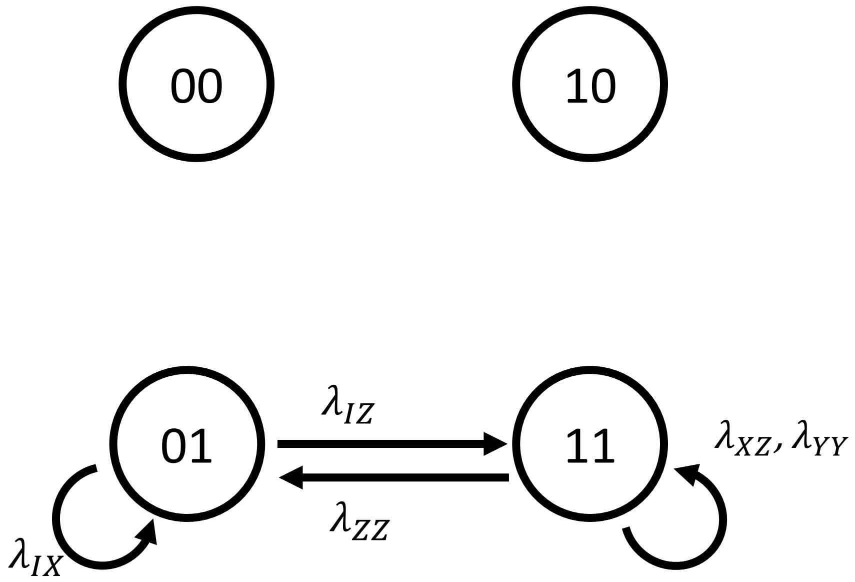
Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford
- CNOT: $IX \rightarrow IX$
- $XZ \rightarrow YY$
- $YY \rightarrow XZ$
- $IZ \rightarrow ZZ$



Building the pattern transfer graph

- Pattern transfer graph: a way to represent the mapping between Pauli operators by the Clifford
- CNOT: $IX \rightarrow IX$
- $XZ \rightarrow YY$
- $YY \rightarrow XZ$
- $IZ \rightarrow ZZ$
- $ZZ \rightarrow IZ$

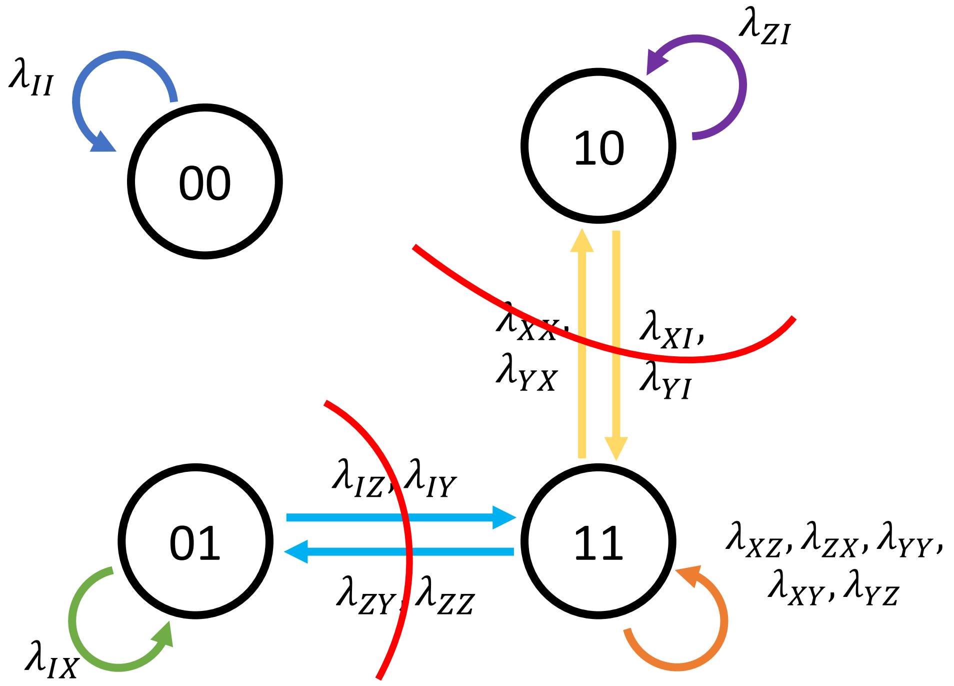


Overview of results

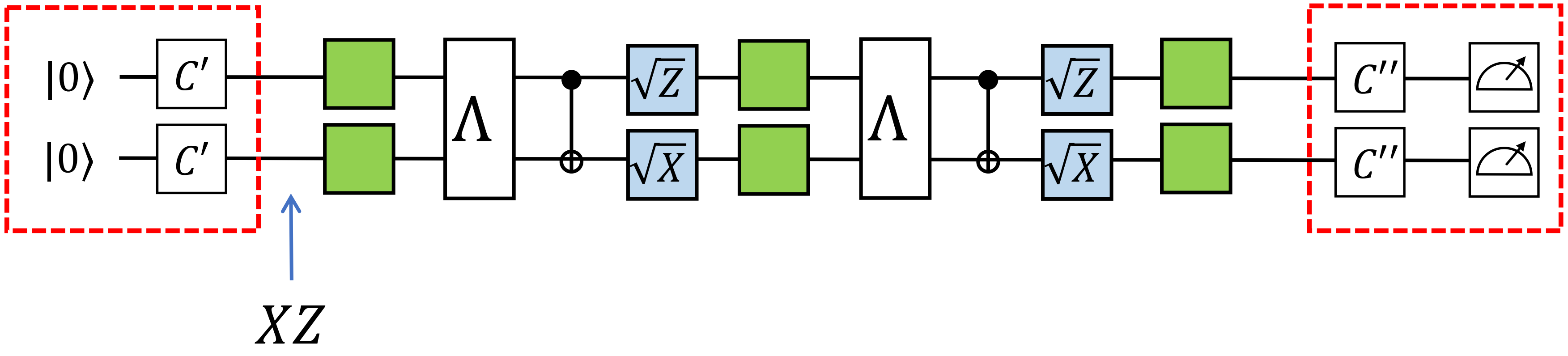
- We augment CB with a trick to learn more information
  - $\lambda_{IX}, \lambda_{ZI}, \lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}, \lambda_{IZ}\,\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}, \lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$
  - Anything beyond this is unlearnable

This comes from the main result: classification of learnability using a graph representation

- The noise model lives on a graph; the cycles in the graph are learnable, cuts are unlearnable
- Corollary: CB + trick is optimal



Cycle benchmarking with trick



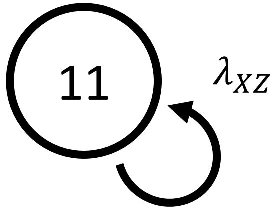
CNOT

$IX \leftrightarrow IX$

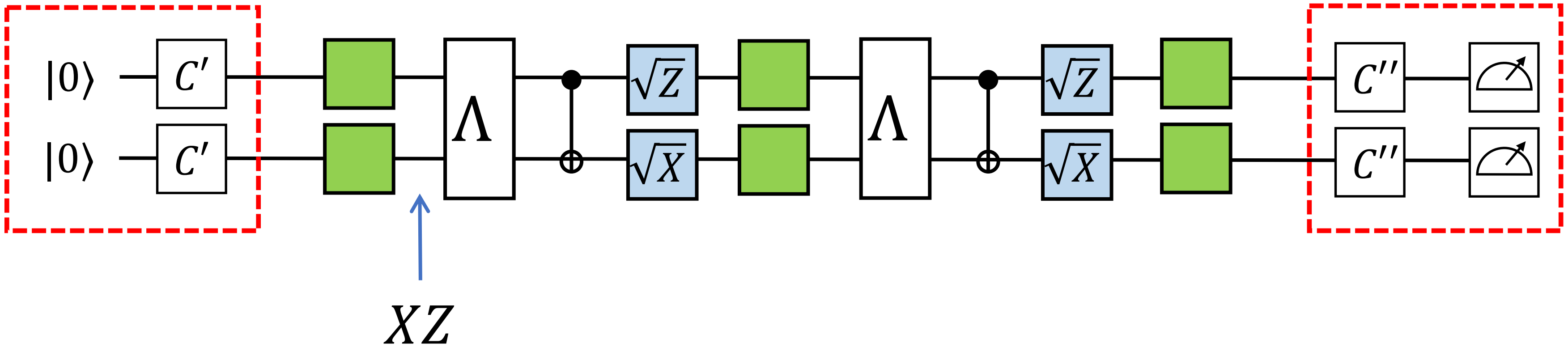
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$



Cycle benchmarking with trick



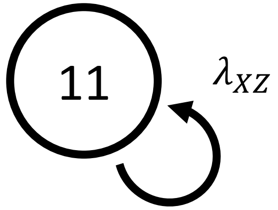
CNOT

$IX \leftrightarrow IX$

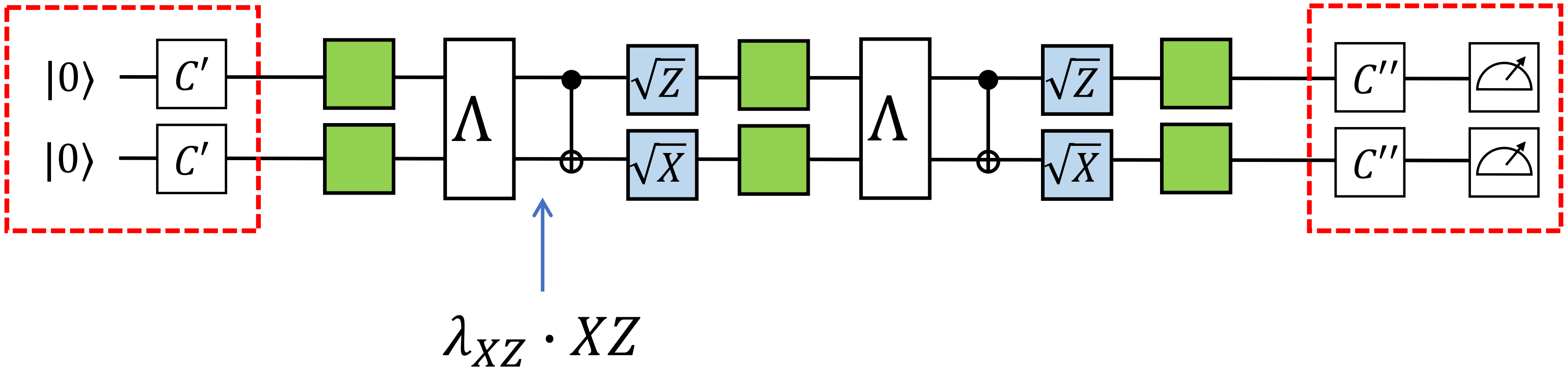
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$



Cycle benchmarking with trick



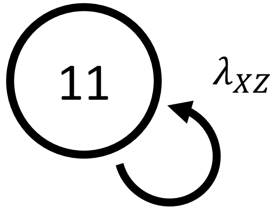
CNOT

$IX \leftrightarrow IX$

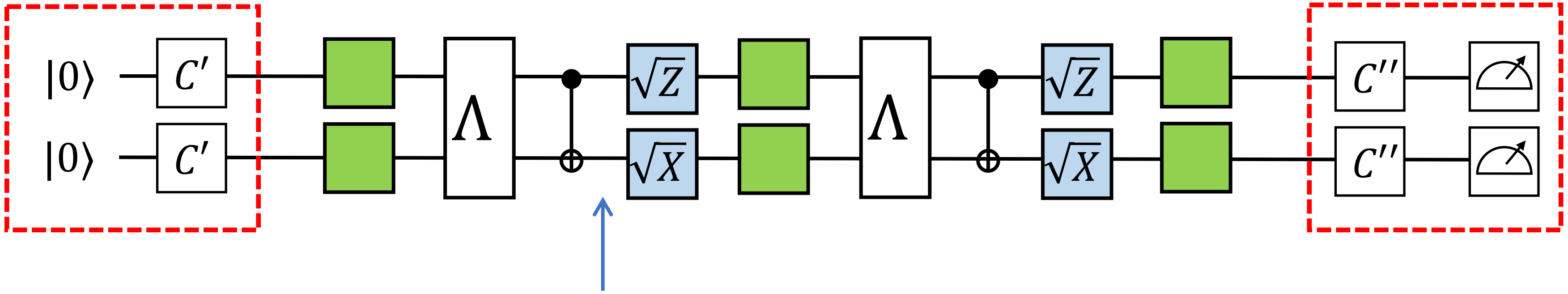
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



$\lambda_{XZ} \cdot YY$

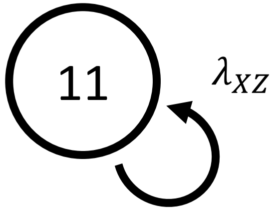
CNOT

$IX \leftrightarrow IX$

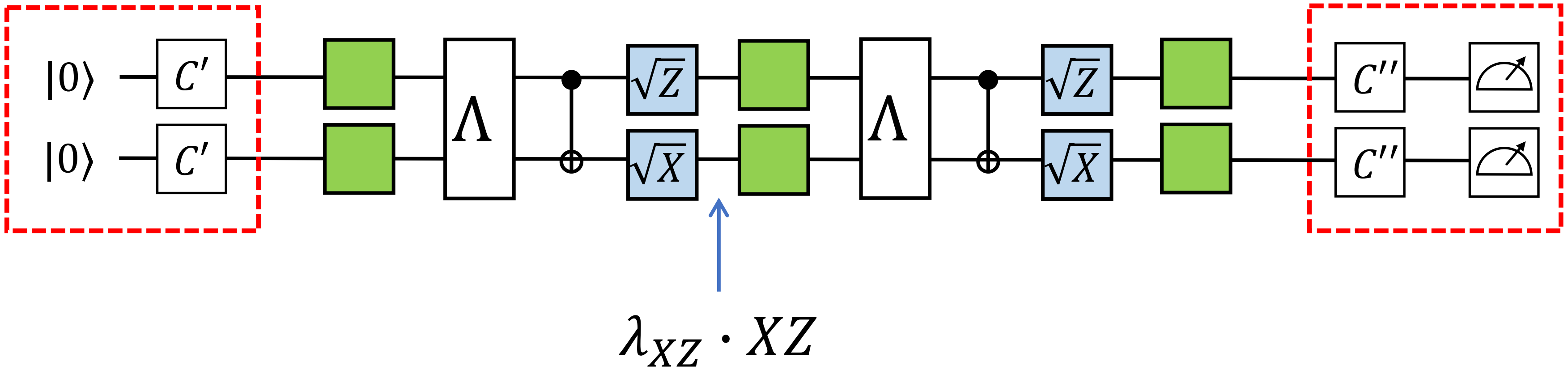
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



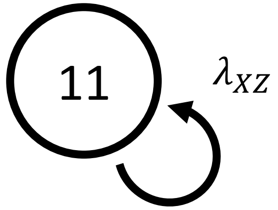
CNOT

$IX \leftrightarrow IX$

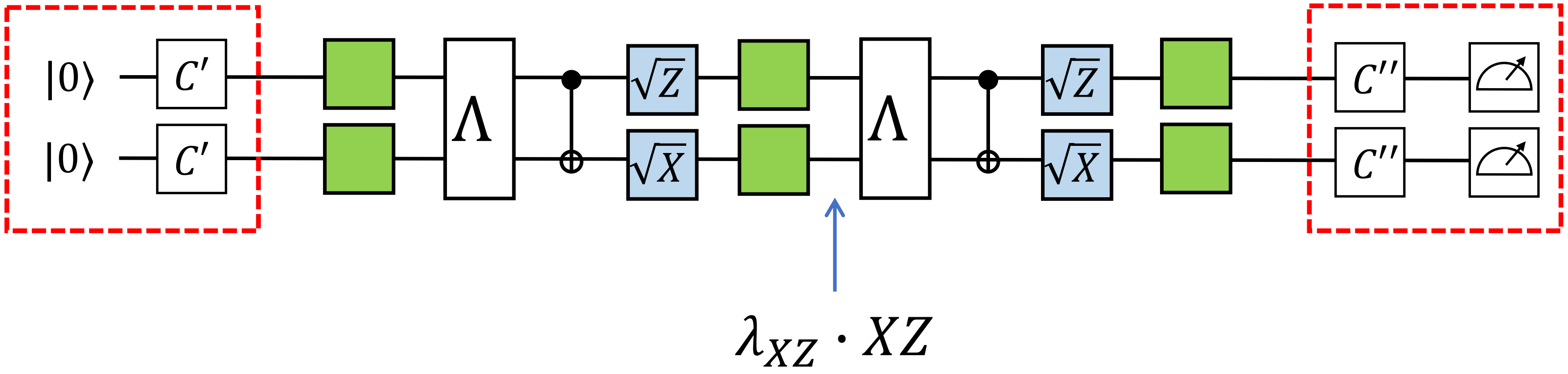
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



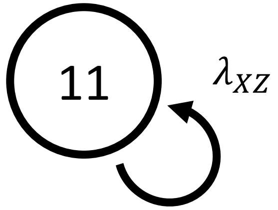
CNOT

$IX \leftrightarrow IX$

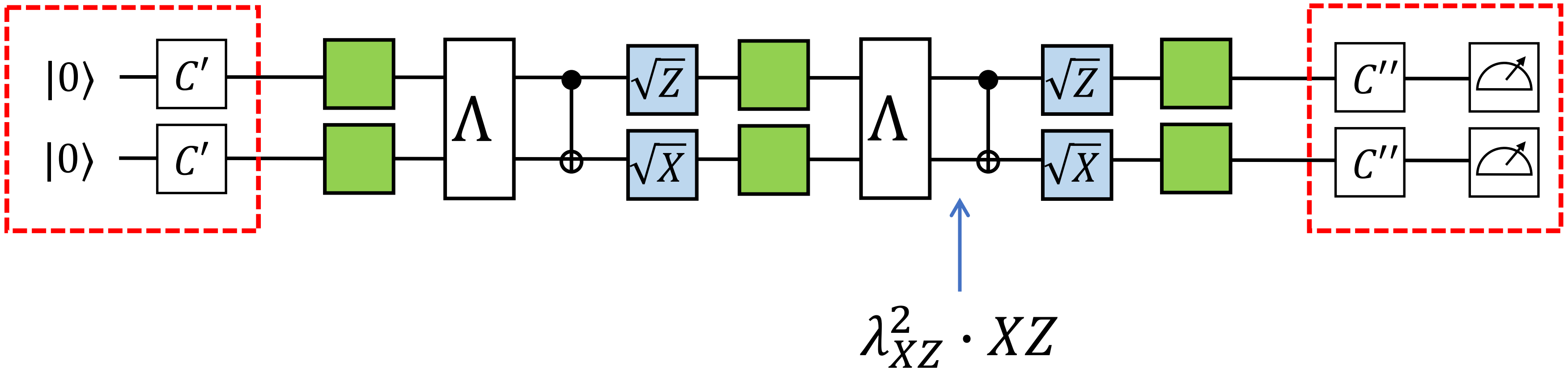
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



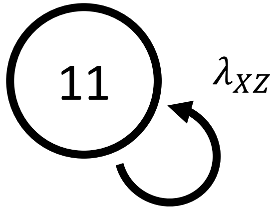
CNOT

$IX \leftrightarrow IX$

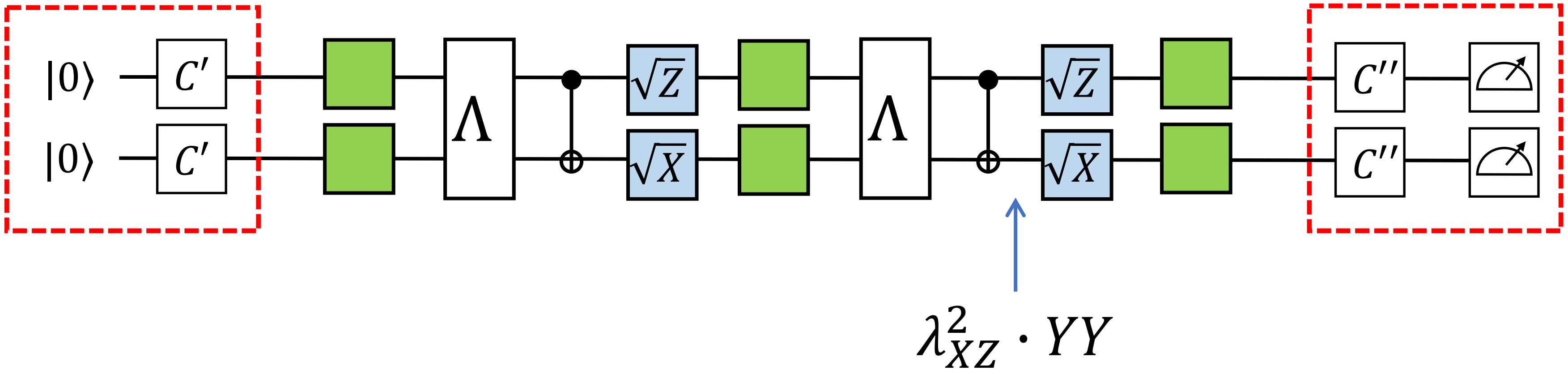
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



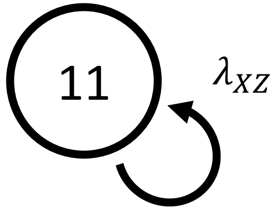
CNOT

$IX \leftrightarrow IX$

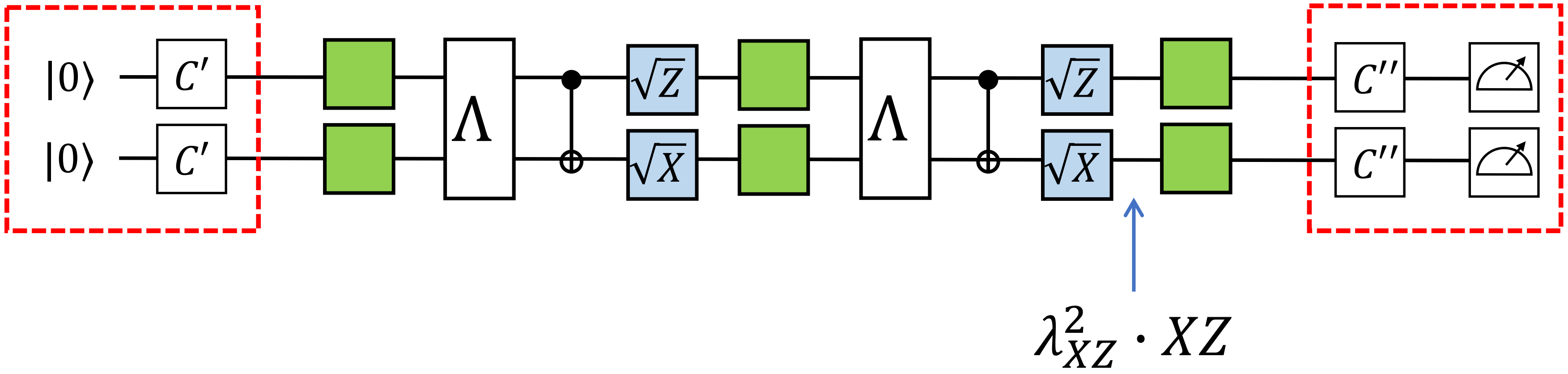
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



CNOT

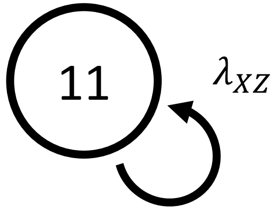
$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

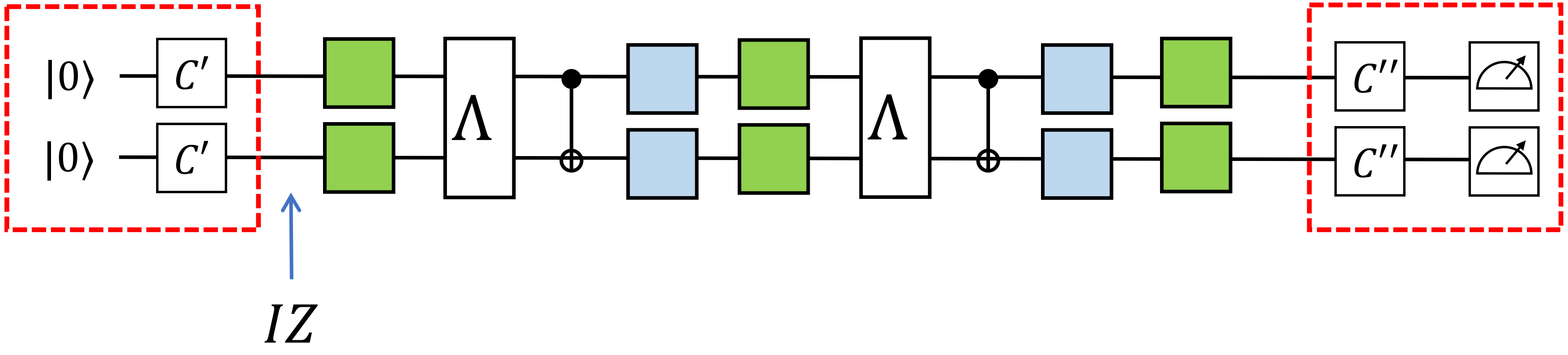
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

Using this single-qubit rotation trick, we can learn λ_{XZ} (as well as λ_{YY})



Cycle benchmarking with trick



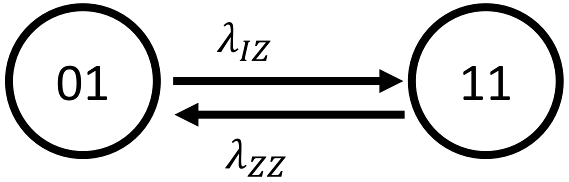
CNOT

$IX \leftrightarrow IX$

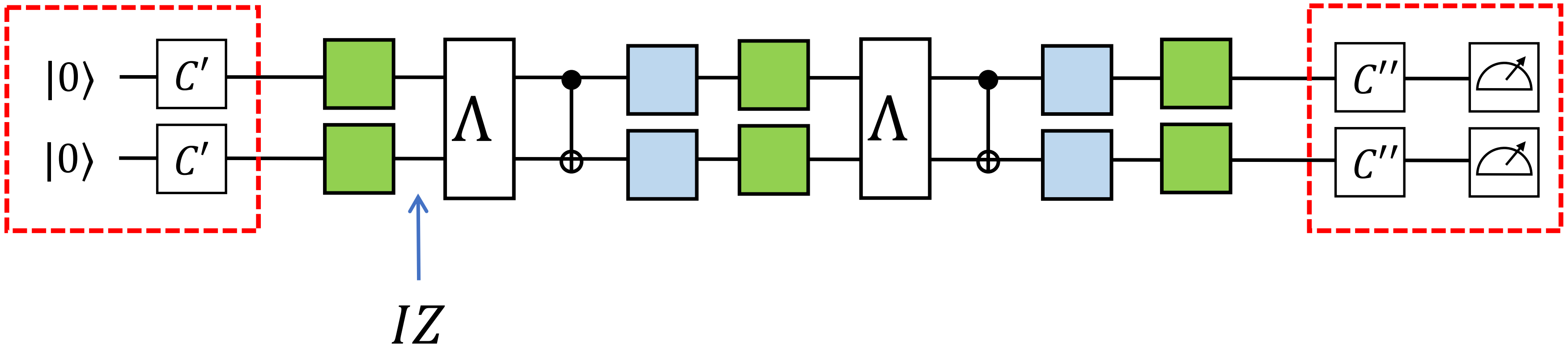
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$



Cycle benchmarking with trick



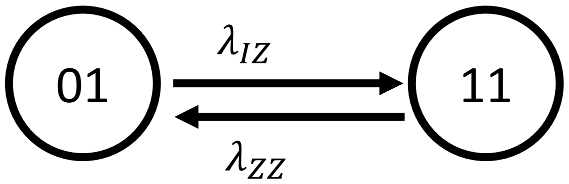
CNOT

$IX \leftrightarrow IX$

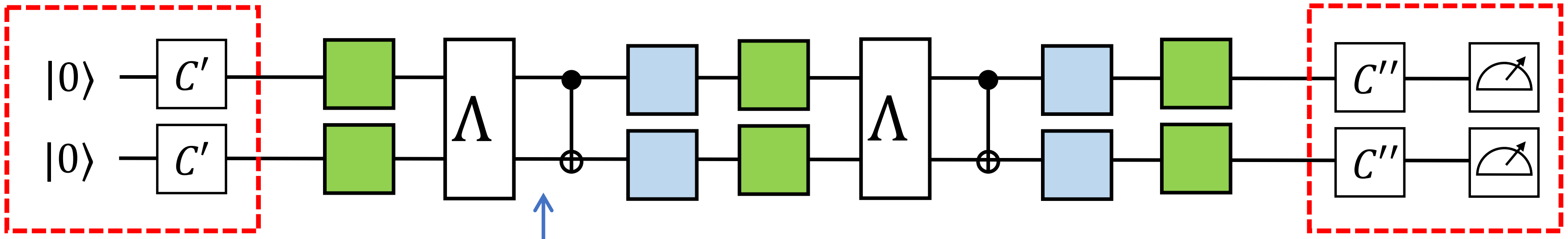
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$$\Lambda(P_a) = \lambda_a P_a$$



Cycle benchmarking with trick



$\lambda_{IZ} \cdot IZ$

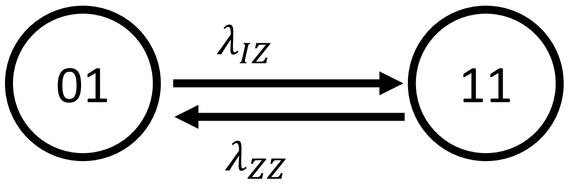
CNOT

$IX \leftrightarrow IX$

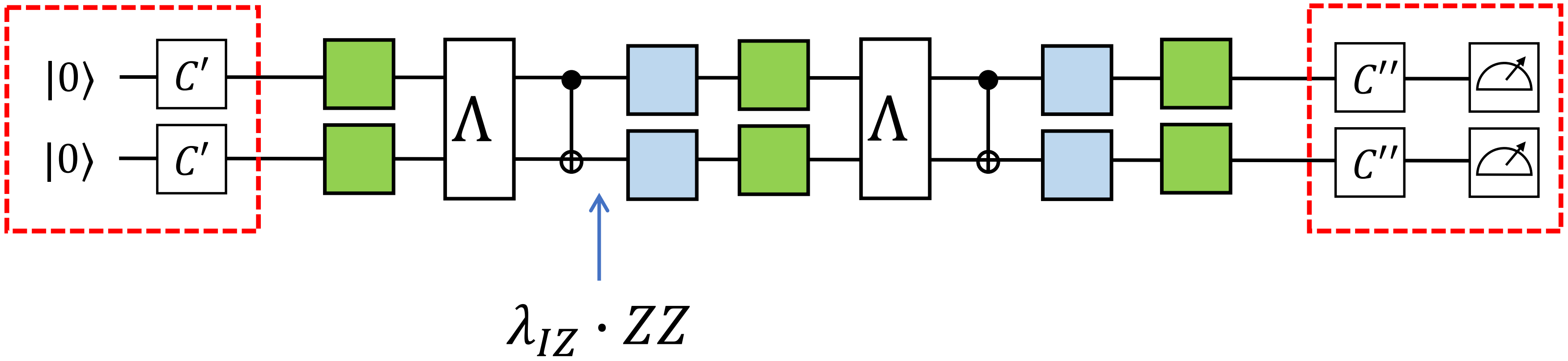
$XZ \leftrightarrow YY$

$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$



Cycle benchmarking with trick



CNOT

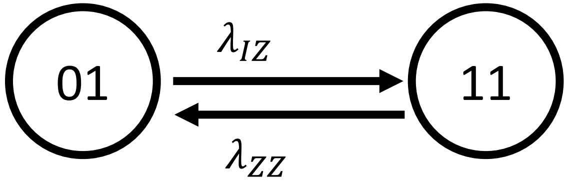
$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

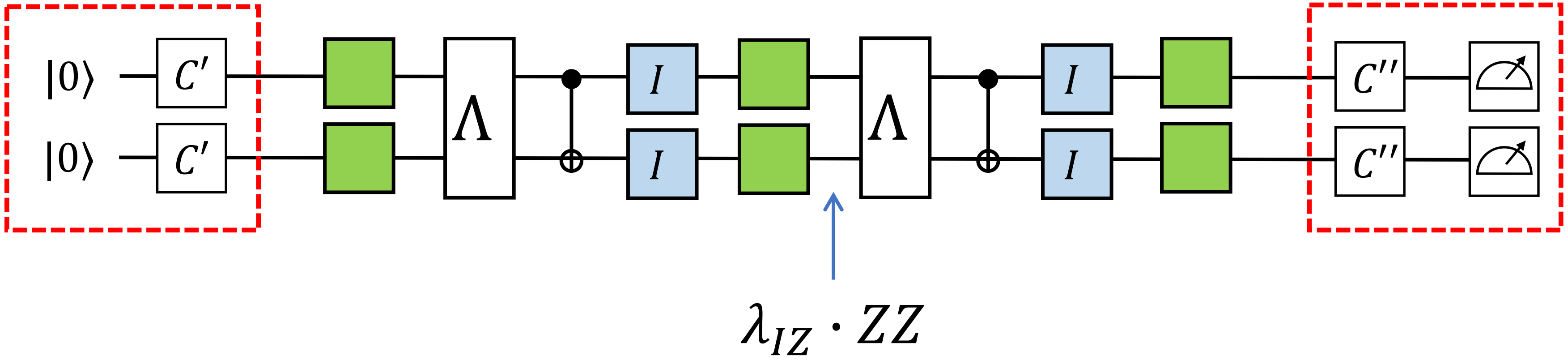
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

Can we use single-qubit gates to rotate ZZ back to IZ? No!



Cycle benchmarking with trick



CNOT

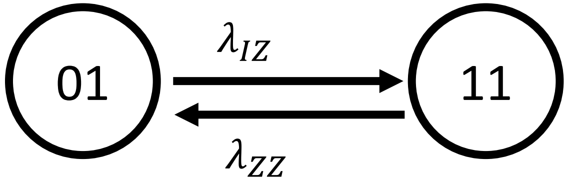
$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

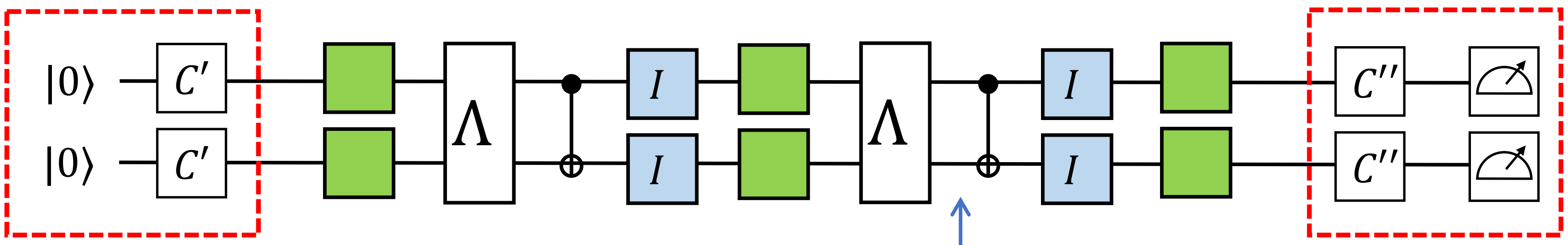
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

Can we use single-qubit gates to rotate ZZ back to IZ? No!



Cycle benchmarking with trick



$\lambda_{ZZ}\lambda_{IZ} \cdot ZZ$

CNOT

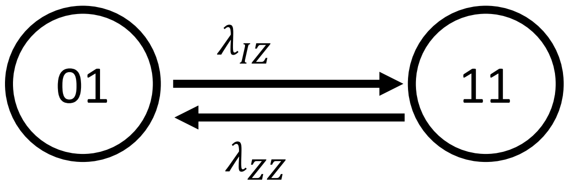
$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

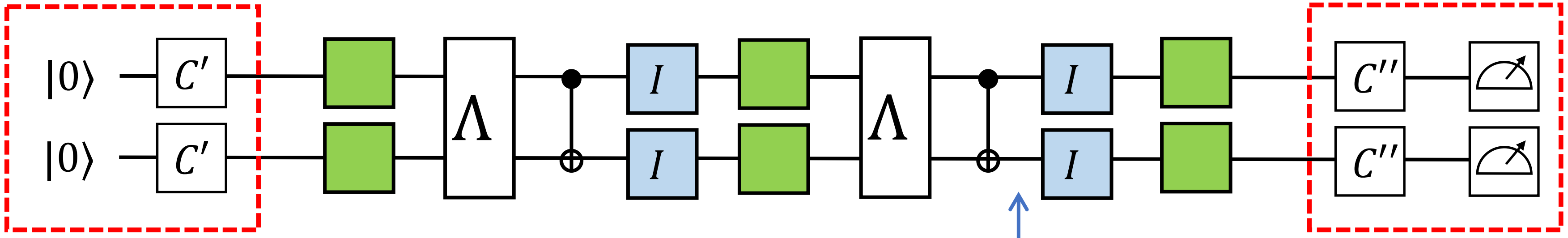
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

Can we use single-qubit gates to rotate ZZ back to IZ? No!



Cycle benchmarking with trick



$\lambda_{ZZ}\lambda_{IZ} \cdot IZ$

CNOT

$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

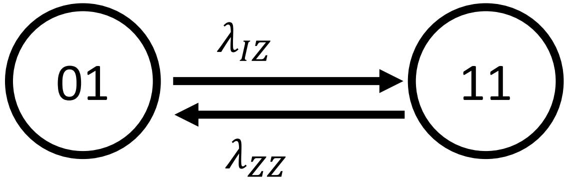
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

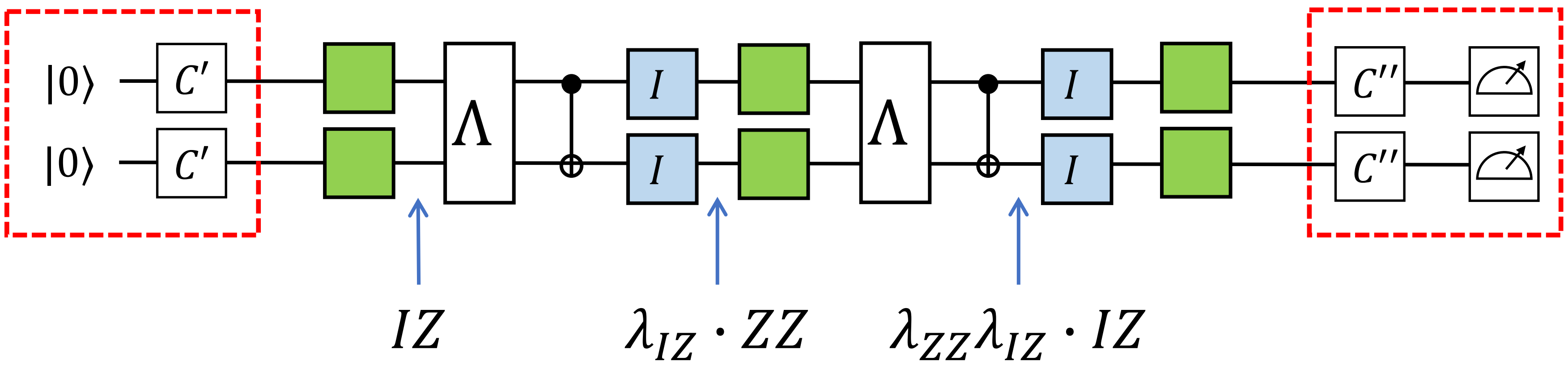
From this experiment we can learn $\lambda_{ZZ}\lambda_{IZ}$, but not individually…

What's the difference in this example?

Pauli weight pattern: $I \leftrightarrow 0, X, Y, Z \leftrightarrow 1$ changes from 01 to 11



Cycle benchmarking with trick



CNOT

$IX \leftrightarrow IX$

$XZ \leftrightarrow YY$

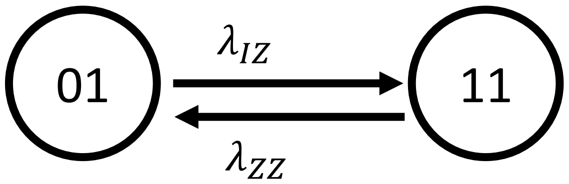
$IZ \leftrightarrow ZZ$

$\Lambda(P_a) = \lambda_a P_a$

The trajectory of the Pauli operator forms a cycle

$$IZ \rightarrow ZZ \rightarrow IZ \rightarrow ZZ \rightarrow \cdots$$

And we can learn the product of Pauli fidelities along the cycle $\lambda_{ZZ}\lambda_{IZ}$

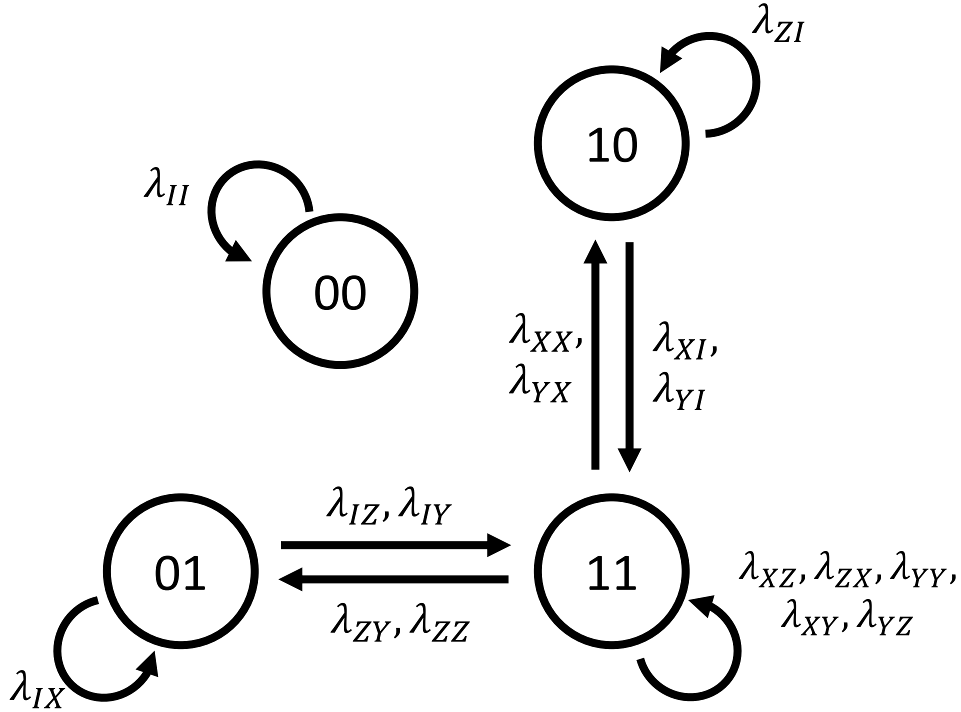


Pattern transfer graph

- For a n-qubit Clifford, the graph has 2^n vertices, 4^n edges
- The vertices correspond to the Pauli weight pattern
  - We don't need to record X/Y/Z in vertices because we can freely rotate among them using the single-qubit rotation trick

Observation: we can learn the product of Pauli fidelities along every cycle in the graph using cycle benchmarking

"So, every cycle is learnable… what's the dual of a cycle in a graph?"



The learnability of Pauli noise

- **Theorem:** in the pattern transfer graph,
- The product of Pauli fidelities along every cycle is learnable
  - Proof: cycle benchmarking
- The product of Pauli fidelities along every cut is unlearnable
  - Proof: construct a gauge transformation for every cut

- This achieves a complete classification of learnability
  - Informally: cycles and cuts span the entire graph space
  - graph space = orthogonal direct sum of cycle space and cut space

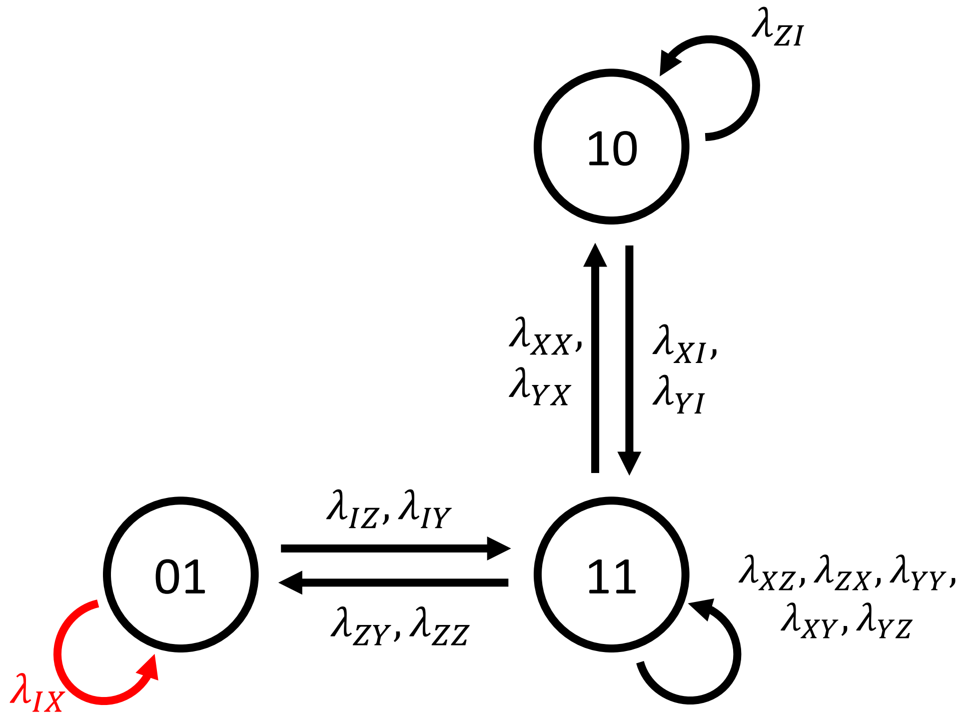
The learnability of Pauli noise

- **Theorem:** in the pattern transfer graph,
- The product of Pauli fidelities along every cycle is learnable
  - Proof: cycle benchmarking
- The product of Pauli fidelities along every cut is unlearnable
  - Proof: construct a gauge transformation for every cut

- This achieves a complete classification of learnability
  - Every function of the noise model can be decomposed as $f = f|_{\text{cycle}} + f|_{\text{cut}}$
  - f is learnable if and only if $f|_{\text{cut}} = 0$

The learnability of Pauli noise: example

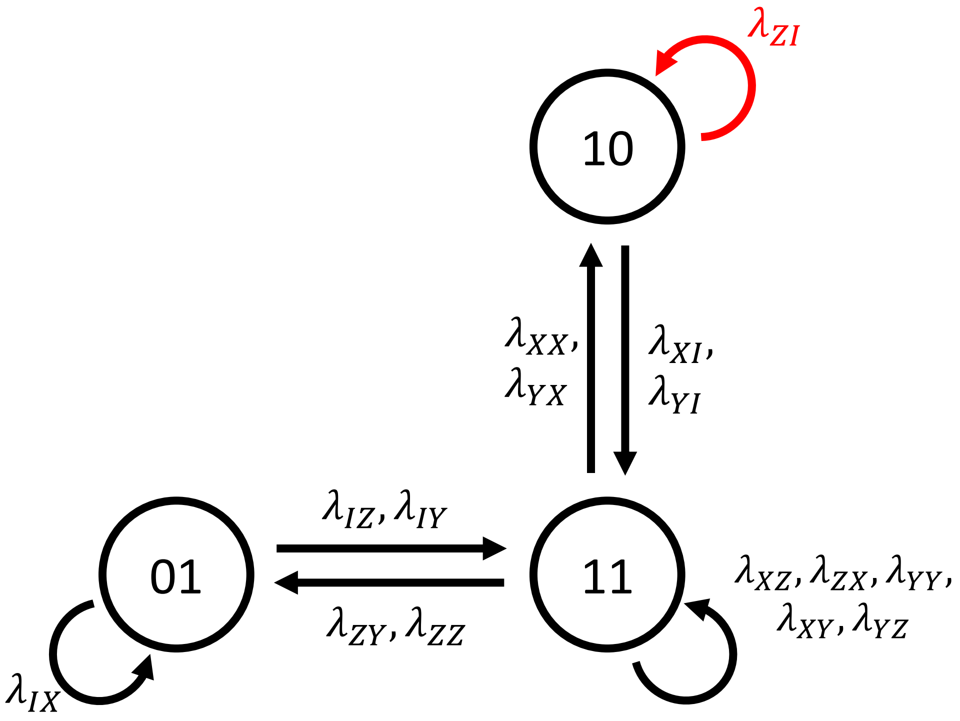
Learnable information
(Cycle space)
λ_{IX}



Unlearnable information
(Cut space)

The learnability of Pauli noise: example

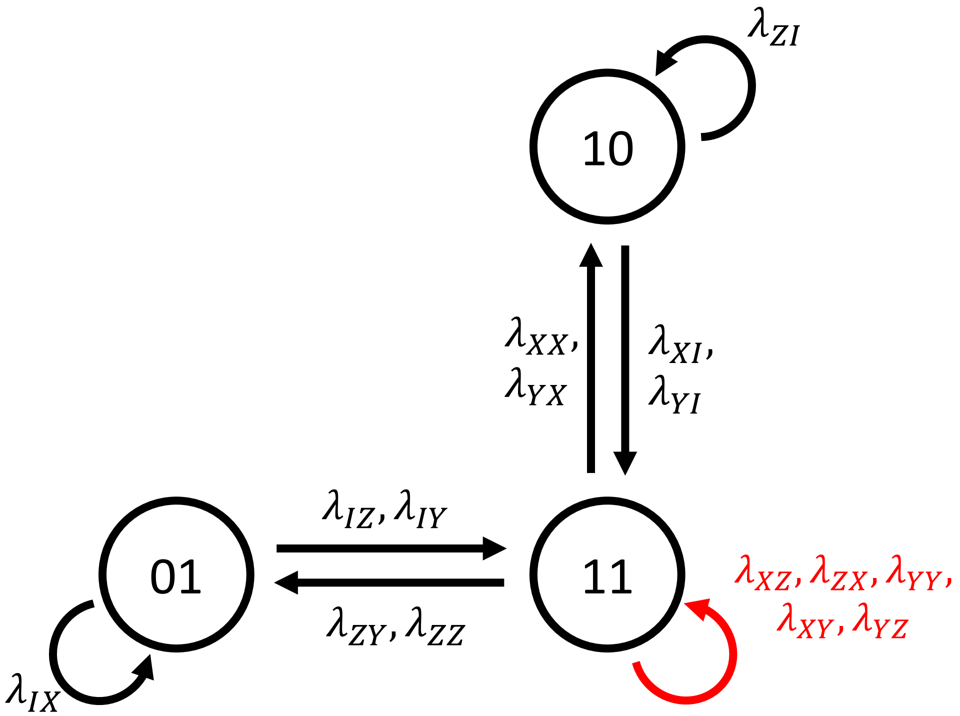
Learnable information
(Cycle space)
λ_{IX}, λ_{ZI}



Unlearnable information
(Cut space)

The learnability of Pauli noise: example

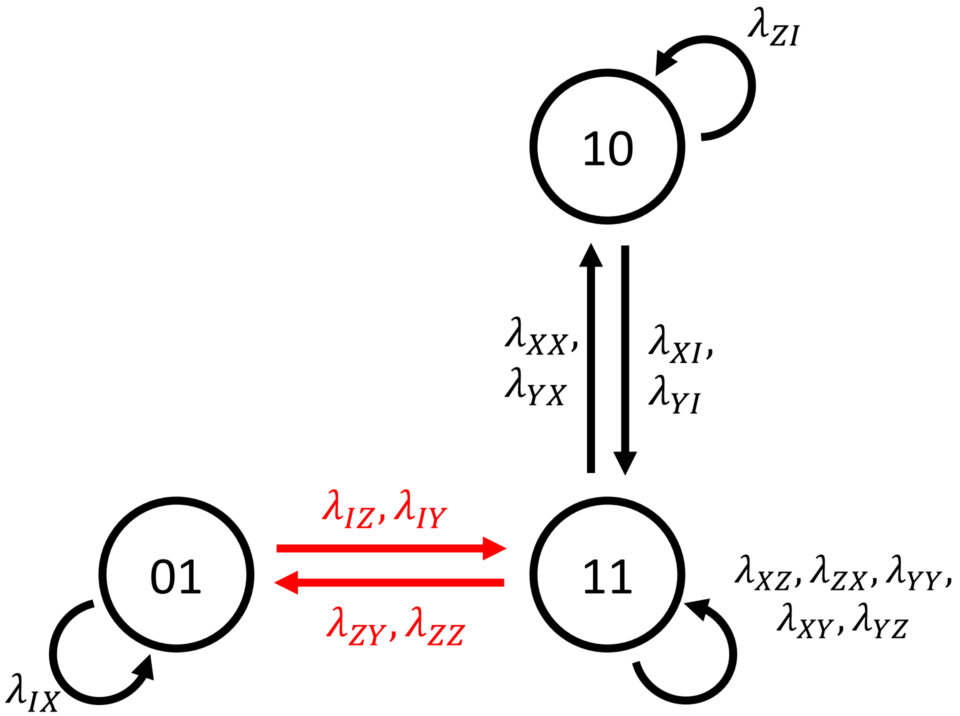
Learnable information
(Cycle space)
$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$



Unlearnable information
(Cut space)

The learnability of Pauli noise: example

Learnable information
(Cycle space)
$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$

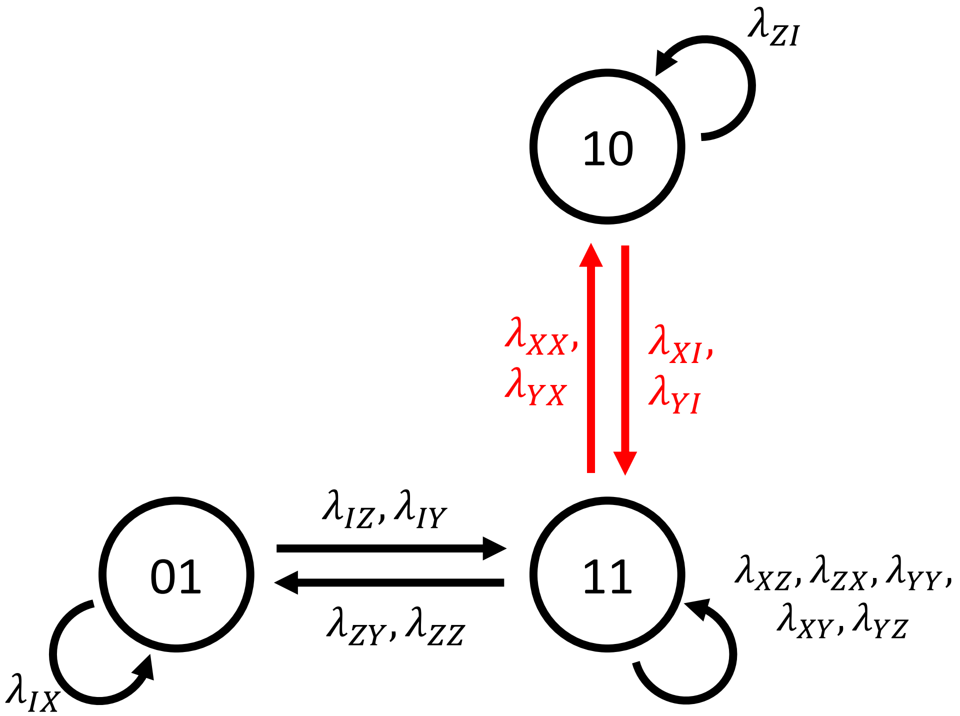


Unlearnable information
(Cut space)

The learnability of Pauli noise: example

Learnable information
(Cycle space)

$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$

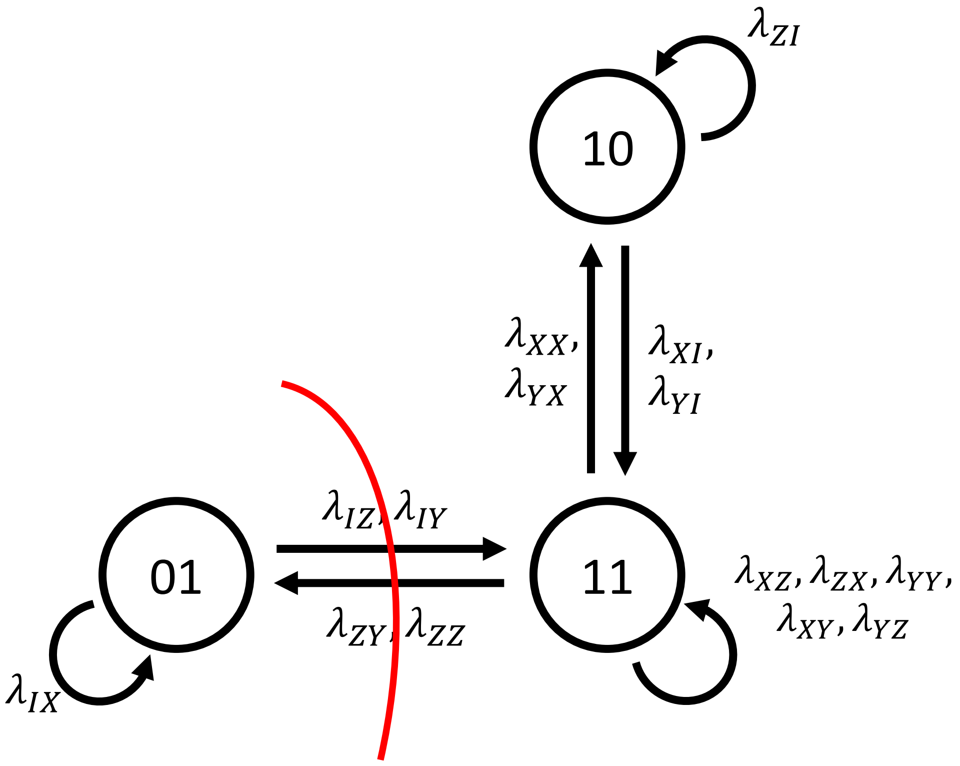


Unlearnable information
(Cut space)

The learnability of Pauli noise: example

Learnable information
(Cycle space)

$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$



Unlearnable information
(Cut space)

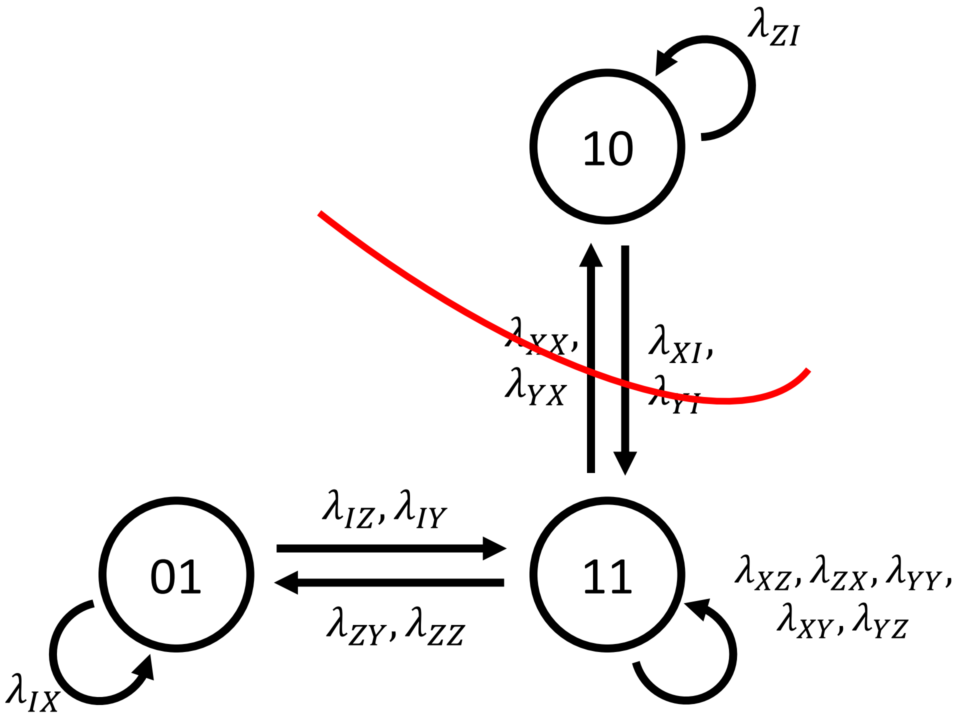
$\lambda_{IZ}\lambda_{IY} / \lambda_{ZY}\lambda_{ZZ}$

Recall: a function is unlearnable means that it is variant under some gauge transformation

The learnability of Pauli noise: example

Learnable information
(Cycle space)

$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$



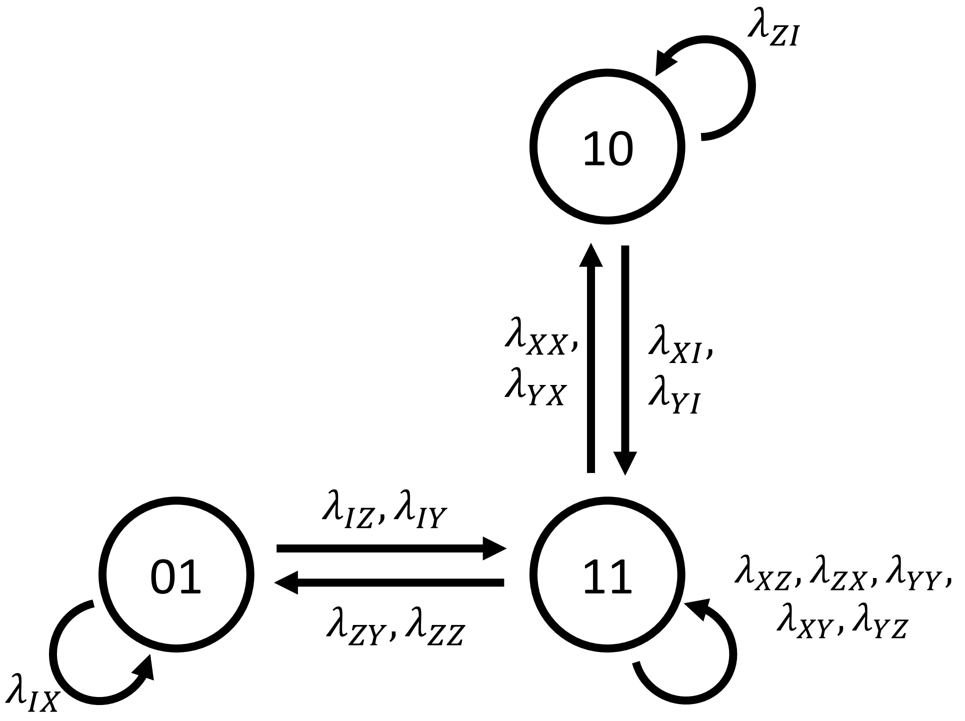
Unlearnable information
(Cut space)

$\lambda_{IZ}\lambda_{IY}/\lambda_{ZY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{YI}/\lambda_{XX}\lambda_{YX}$

The learnability of Pauli noise: example

Learnable information
(Cycle space)

$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$



Unlearnable information
(Cut space)

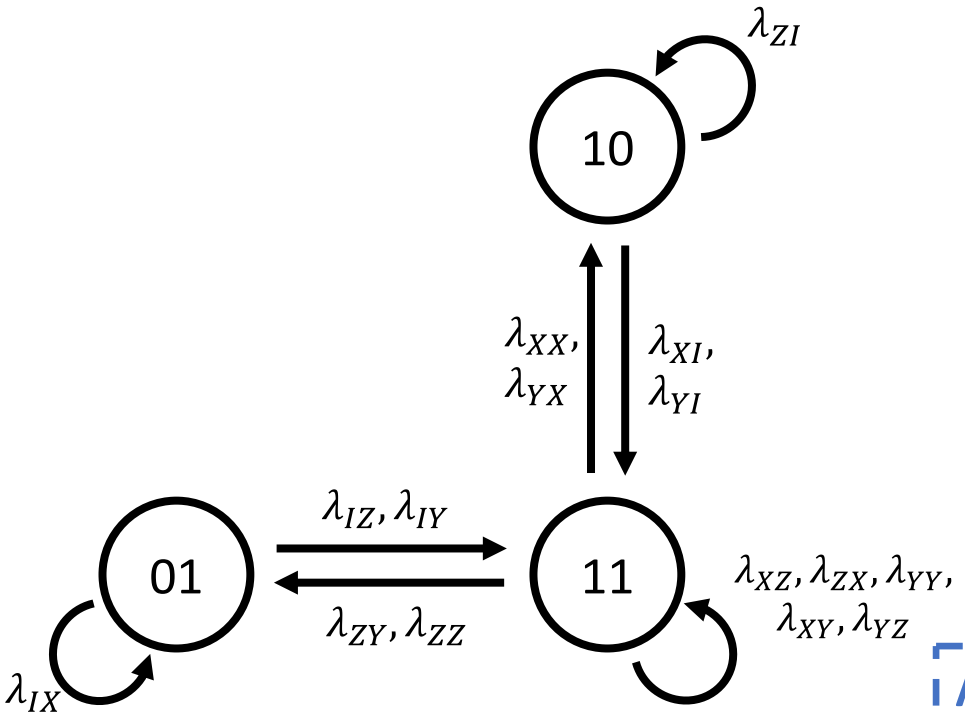
$\lambda_{IZ}\lambda_{IY} / \lambda_{ZY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{YI} / \lambda_{XX}\lambda_{YX}$

CNOT has 15 = 13 learnable degrees of freedom + 2 unlearnable degrees of freedom

The learnability of Pauli noise: example

Learnable information
(Cycle space)

$\lambda_{IX}, \lambda_{ZI},$
$\lambda_{XZ}, \lambda_{ZX}, \lambda_{YY}, \lambda_{XY}, \lambda_{YZ}$
$\lambda_{IZ}\lambda_{ZZ}, \lambda_{IZ}\lambda_{ZY}, \lambda_{IY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{XX}, \lambda_{XI}\lambda_{YX}, \lambda_{YI}\lambda_{XX}$



Unlearnable information
(Cut space)

$\lambda_{IZ}\lambda_{IY}/\lambda_{ZY}\lambda_{ZZ}$
$\lambda_{XI}\lambda_{YI}/\lambda_{XX}\lambda_{YX}$

Finally: the learnability of Pauli errors can be determined from the cycle space via a Fourier transformation

All learnable information:
$p_{IX}, p_{XY}, p_{XZ}, p_{YY}, p_{YZ}, p_{ZI}, p_{ZX},$
$p_{IY} + p_{ZY}, p_{IZ} + p_{ZY}, p_{IY} + p_{ZZ},$
$p_{XI} + p_{XX}, p_{XX} + p_{YI}, p_{XI} + p_{YX}$
(13 equations)

Learnable information = Cycle space

Dimension = $4^n - 2^n + c$

Unlearnable information = Cut space

Dimension = $2^n - c$

The learnability of Pauli noise

- **Theorem:** in the pattern transfer graph,
- The product of Pauli fidelities along every cycle is learnable
  - Proof: cycle benchmarking
- The product of Pauli fidelities along every cut is unlearnable
  - Proof: construct a gauge transformation for every cut

- Corollary: cycle benchmarking learns all learnable information
  - This is because learnable information forms a cycle space
- Main remaining question: how to resolve unlearnability?
  - Must make additional assumptions about noise model (time permits)

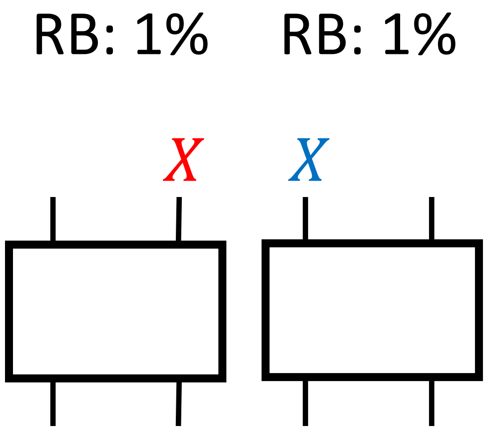
Part II: Non-Clifford benchmarking

Why do we care about non-Clifford benchmarking?

- Non-Clifford two-qubit gates are ubiquitous in current implementations of near-term quantum algorithms
- Use “native” two-qubit gates on hardware to maximize fidelity

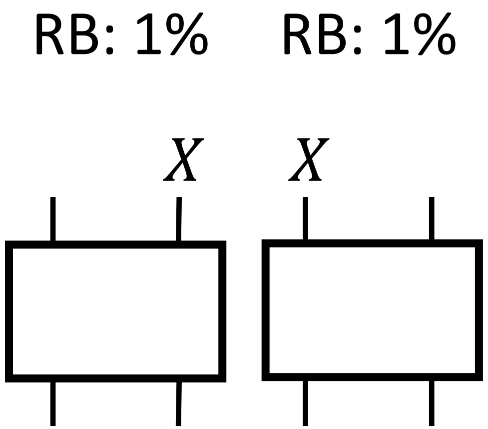
- $\sqrt{\mathrm{i}SWAP}$ used in “Hartree-Fock on a superconducting qubit quantum computer” [Science 369, 1084-1089 (2020)]
- SYC used in “Quantum approximate optimization of non-planar graph problems on a planar superconducting processor” [Nat. Phys. 17, 332-336 (2021)]

Challenge: crosstalk and correlated errors



X with probability 1%
X with probability 1%

Total error = 2%



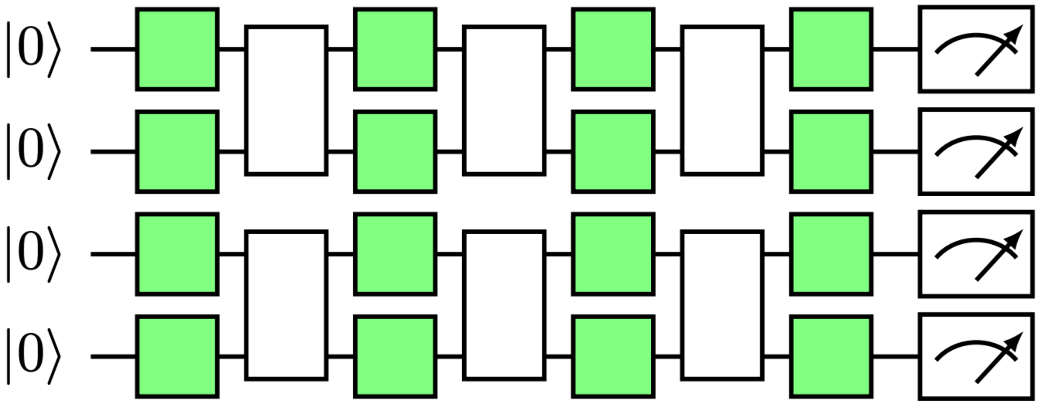
XX with probability 1%

Total error = 1%

This talk: algorithm for estimating the total error in a layer of non-Clifford gates

Scalable noise benchmarking methods

Cycle benchmarking [Erhard et al'19]



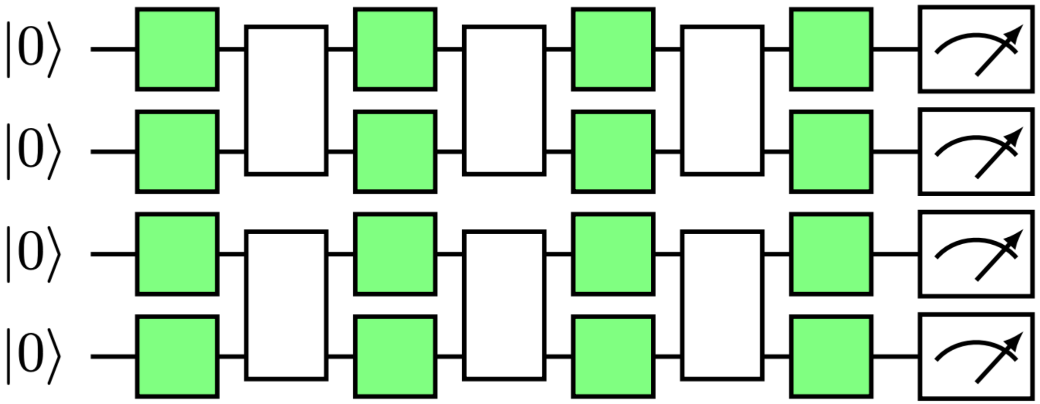
Green: random Pauli gate

Principle: structure of the Clifford and Pauli group
Works for Clifford 2-qubit gates

Challenge: the special structure in the Fourier domain disappears… how to do scalable benchmarking of arbitrary non-Clifford gates?

Scalable noise benchmarking methods

Cycle benchmarking [Erhard et al'19]

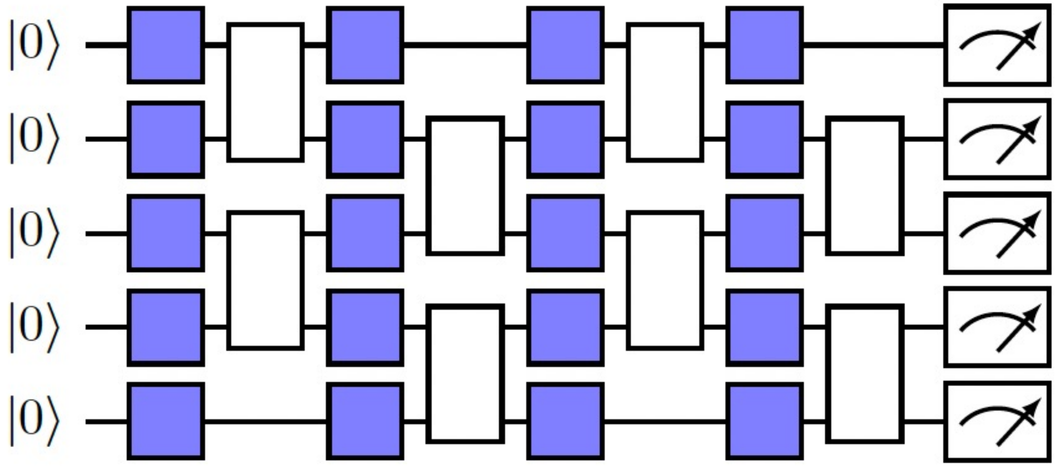


Green: random Pauli gate

Principle: structure of the Clifford and Pauli group

Works for Clifford 2-qubit gates

RCS benchmarking [This talk]

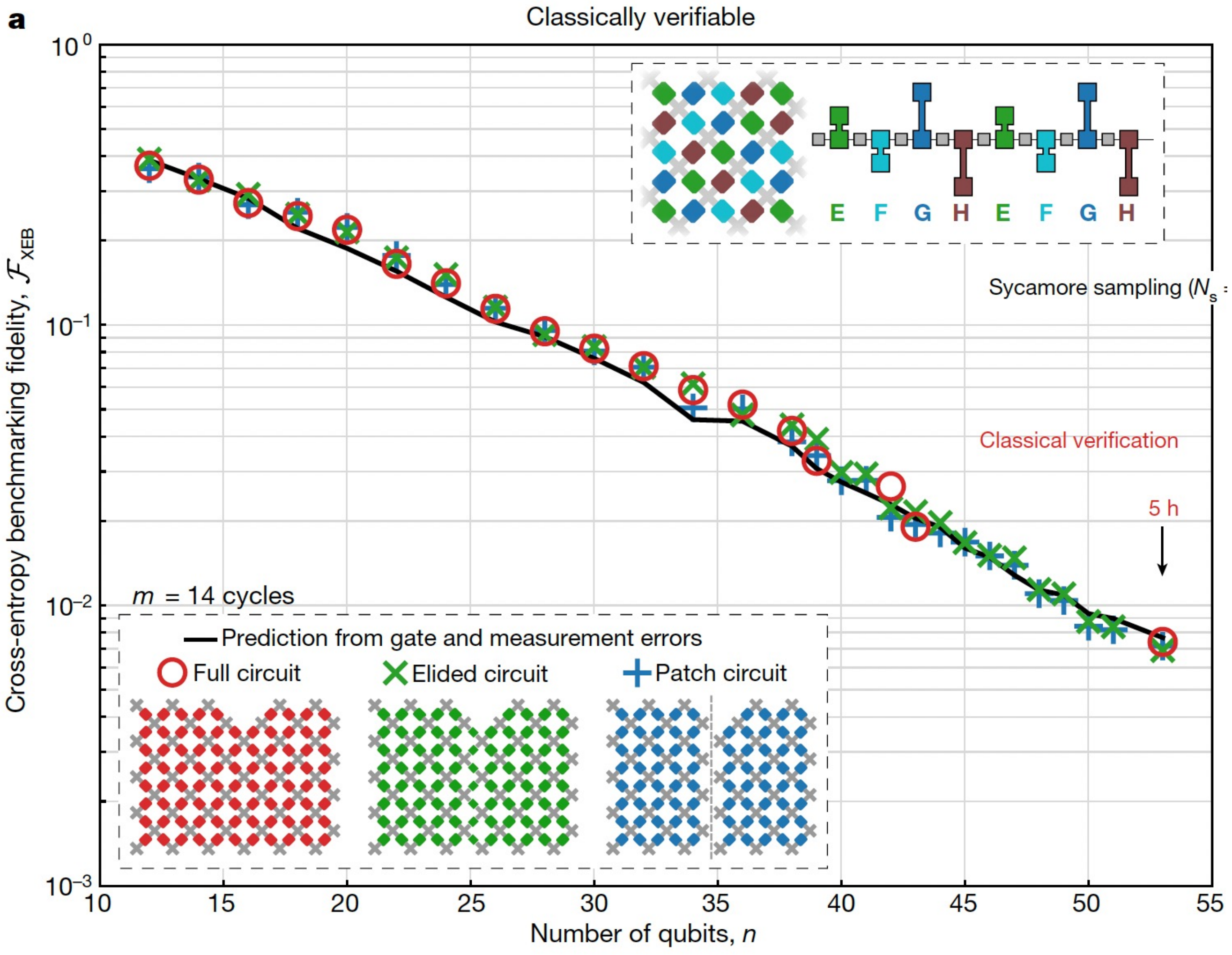


Blue: Haar random single qubit gate

Principle: scrambling effect of random quantum circuits

Works for *any* 2-qubit gates

Motivation: Google's quantum supremacy experiment [Arute et al'19]



Linear cross entropy: m measurement samples,

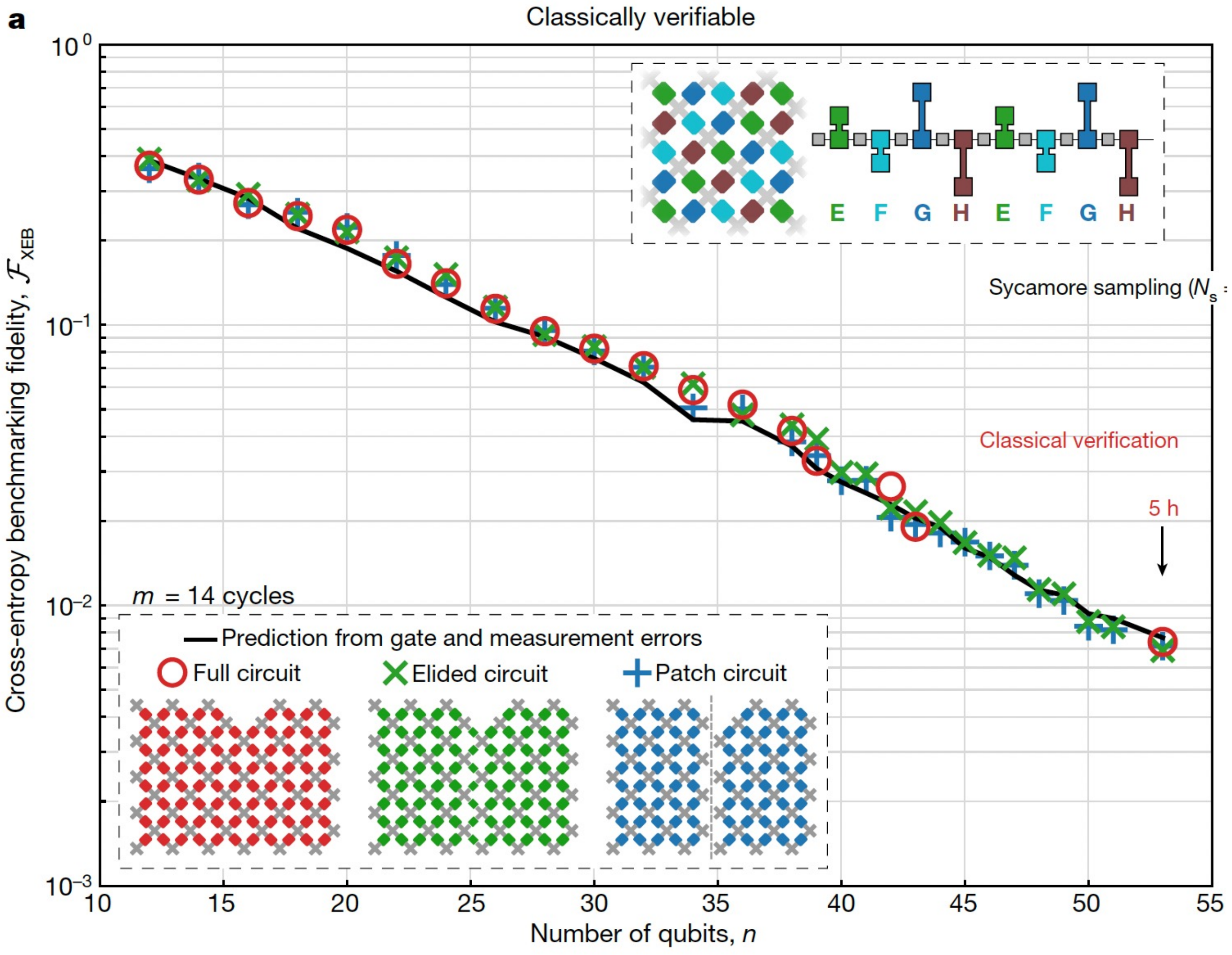
$$XEB = \frac{2^n}{m} \sum_{i=1}^{m} p(x_i) - 1$$

Used as a proxy of the fidelity of their experiment

Claim 1: they have achieved quantum supremacy

Claim 2: the noise in their device was uncorrelated

Motivation: Google's quantum supremacy experiment [Arute et al'19]

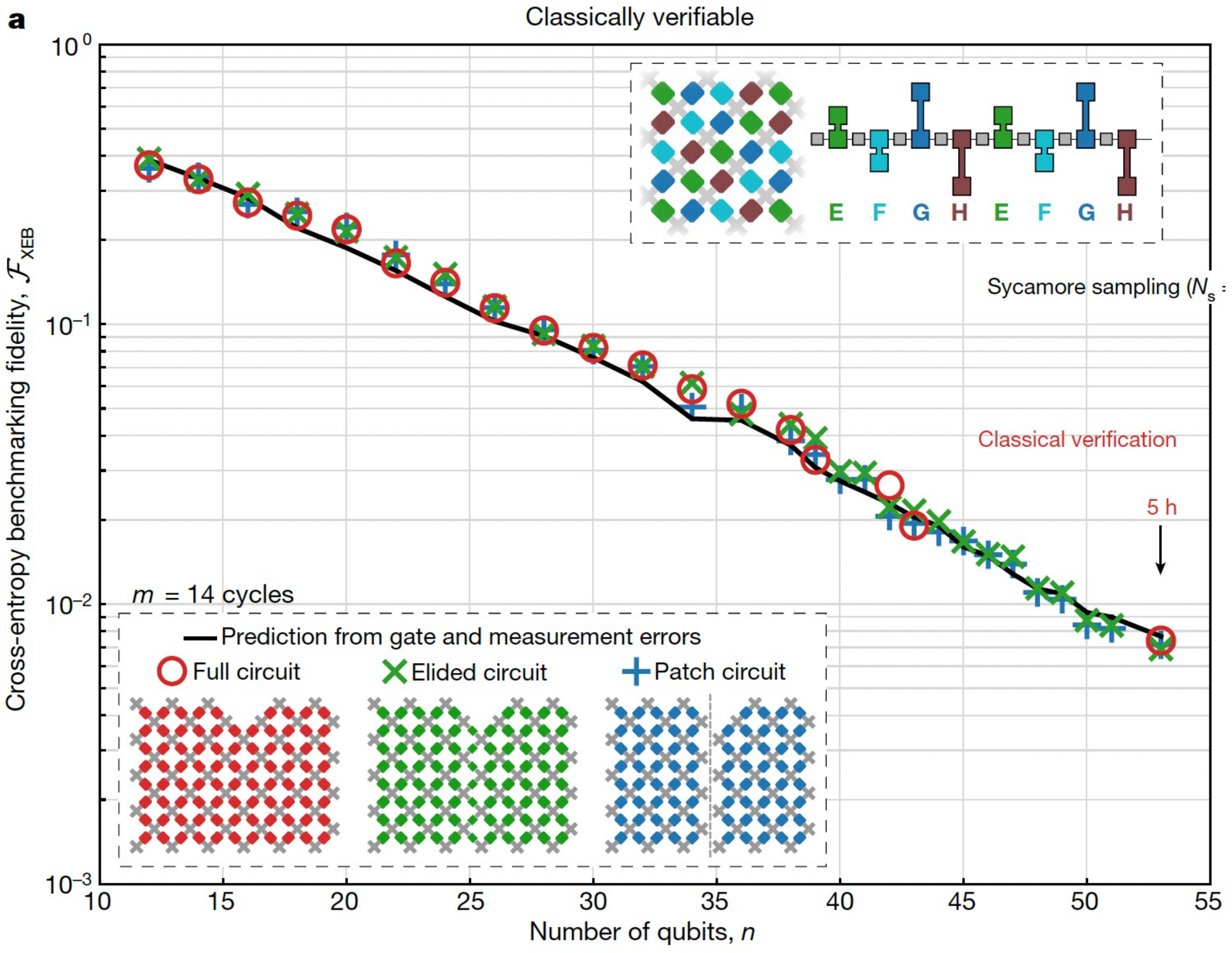


"digital error model" (multiplying individual gate fidelities) $F_{RB} = \prod_{i=1}^{m}(1 - e_i)$

For independent events A, B, P(AB)=P(A)P(B)

"Maybe the errors in our device is uncorrelated? In this case, fidelity=P(no error)= ∏P(no error on gate i). Let's plot both XEB and F_{RB}. If they agree with each other, this suggests that the hypothesis (that noise was uncorrelated) is correct, which would be great news!"

Motivation: Google's quantum supremacy experiment [Arute et al'19]



Observation: the linear cross entropy agrees with the "digital error model" (multiplying individual gate fidelities)

Claim: this coincidence indicated that the noise in Google's device is uncorrelated across each 2-qubit gate

Can we understand this observation and claim from the theoretical perspective?

Could this observation be the hint of a scalable noise benchmarking algorithm for non-Clifford gates?

Overview of RCS benchmarking

- Result: $XEB \approx e^{-td}$, where t is the total amount of noise in an arbitrary noise model acting on each layer of gates
  - Therefore, t can be learned by measuring XEB
- Corollary: with correlated noise, XEB would deviate from the digital error model F_{RB}
  - Evidence that supports Google's claim

Theory of RCS benchmarking

- Consider arbitrary n-qubit Pauli noise channel acting on a layer of 2-qubit gates, the goal is to estimate total error $t = \sum_{a \neq I^n} p_a$
- We show that the average fidelity of random circuits at depth d scales as $\mathbb{E}F \approx e^{-td}$
- In experiments, estimate average fidelity by measuring XEB → get t

Exponential decay of average fidelity

- For a random circuit C, the ideal output state is $|\psi\rangle = C|0^n\rangle$
- Experiment implementation of C creates a mixed state ρ
- The fidelity of C is given by $F = \langle\psi|\rho|\psi\rangle$

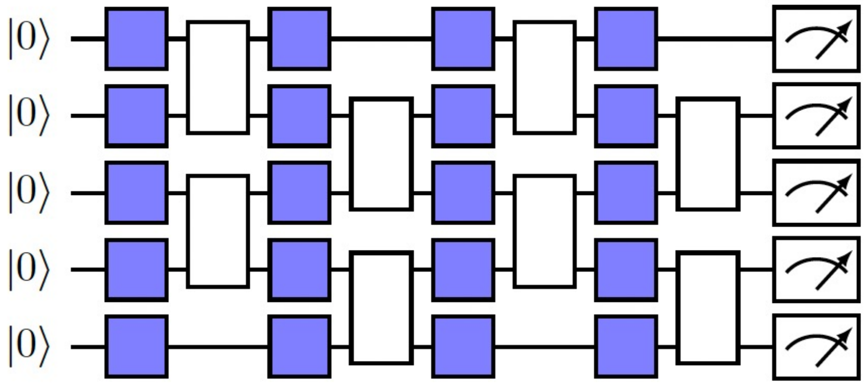
- Theorem: $\mathbb{E}F \approx e^{-td}$ when the total error t is upper bounded by a small constant
- Proof idea: maps $\mathbb{E}F$ into the partition function of a classical spin model, then bound the partition function

RCS benchmarking

Select a few depths, at each depth, sample a few random circuits

Estimate the fidelity of each circuit via XEB, compute the average $\mathbb{E}F$

Fit exponential decay $\mathbb{E}F = Ae^{-td}$, obtain t



Fidelity estimation via cross entropy

- Why not directly measure fidelity?
- Problem: fidelity is hard to estimate
    - Direct fidelity estimation (DFE) has exponential sample complexity $O(2^n/\varepsilon^2)$ in the worst case
- Intuition from Google's experiment: for random circuits, linear cross entropy appears to be a sample-efficient estimator of fidelity
    - $O(1/\varepsilon^2)$ samples suffice
- Recently, theoretical evidence of XEB=fidelity (when total error is small) has been obtained by [Dalzell, Hunter-Jones, Brandão'21] [Gao et al'21]

RCS benchmarking

Select a few depths, at each depth, sample a few random circuits

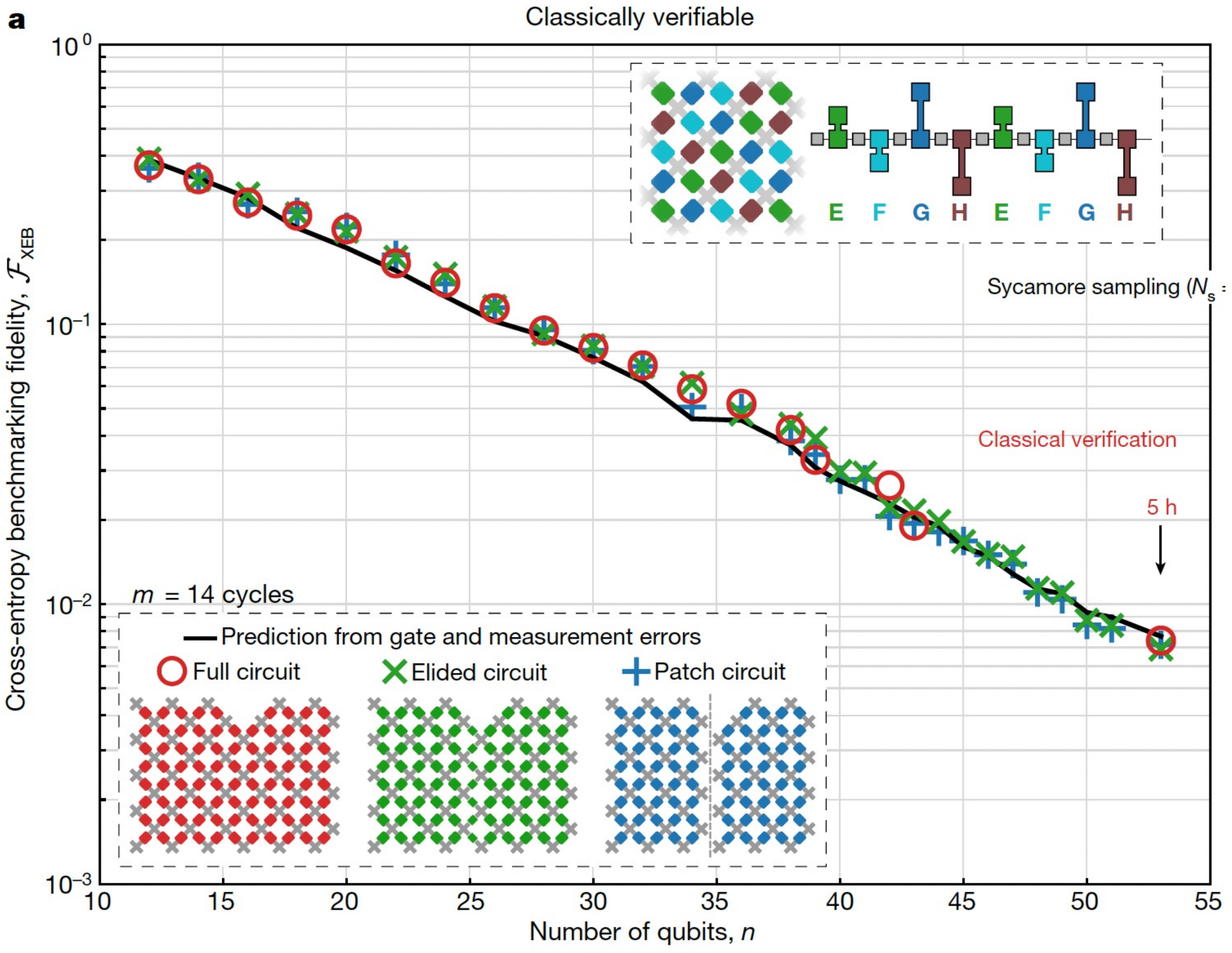
Estimate the fidelity of each circuit via XEB, compute the average $\mathbb{E}F$

←Use linear cross entropy as a proxy for fidelity

Fit exponential decay $\mathbb{E}F = Ae^{-td}$, obtain t

t: the effective noise rate on a layer of arbitrary two-qubit gates

Google's quantum supremacy experiment [Arute et al'19]



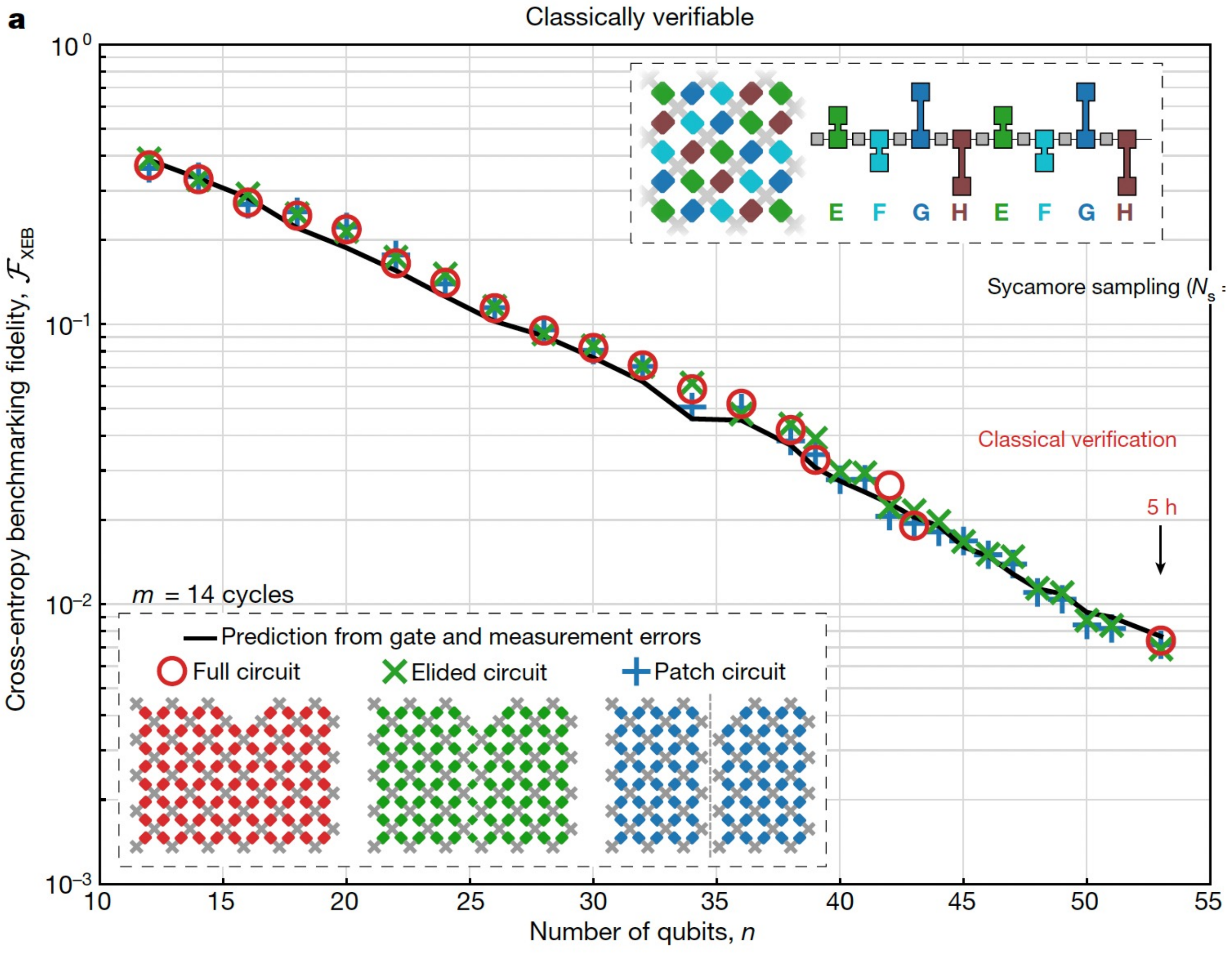
Observation: the linear cross entropy agrees with the "digital error model" (multiplying individual gate fidelities)

Claim: this coincidence indicated that the noise in Google's device is uncorrelated across each 2-qubit gate

Can we understand this observation and claim from the theoretical perspective?

Could this observation be the hint of a scalable noise benchmarking algorithm for non-Clifford gates? ✅

Google's quantum supremacy experiment [Arute et al'19]

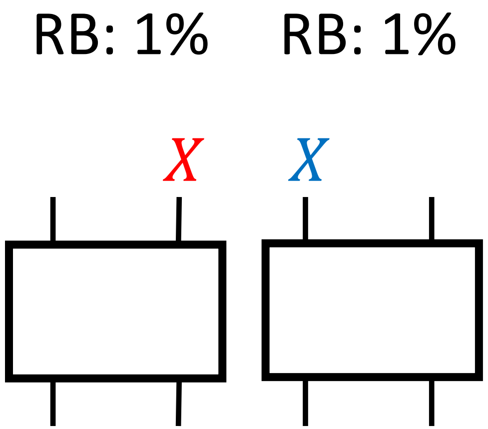


Observation: the linear cross entropy (fidelity) agrees with $F_{RB} = \prod_{i=1}^{m}(1 - e_i)$

Claim: The noise is uncorrelated across each 2-qubit gate

Can we understand this observation and claim from the theoretical perspective?

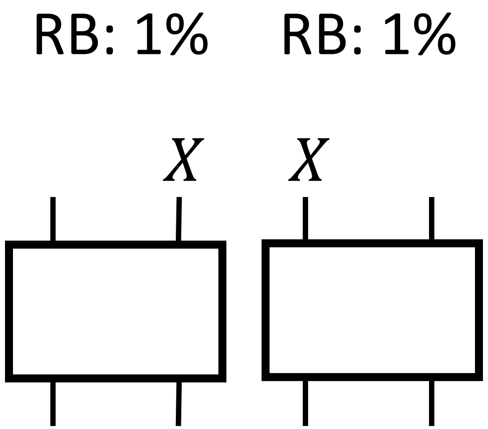
Correlated errors in fidelity estimation



X with probability 1%
X with probability 1%

Total error = 2%

- Contributes 2% to cross entropy and fidelity
- Contributes 2% to F_{RB}



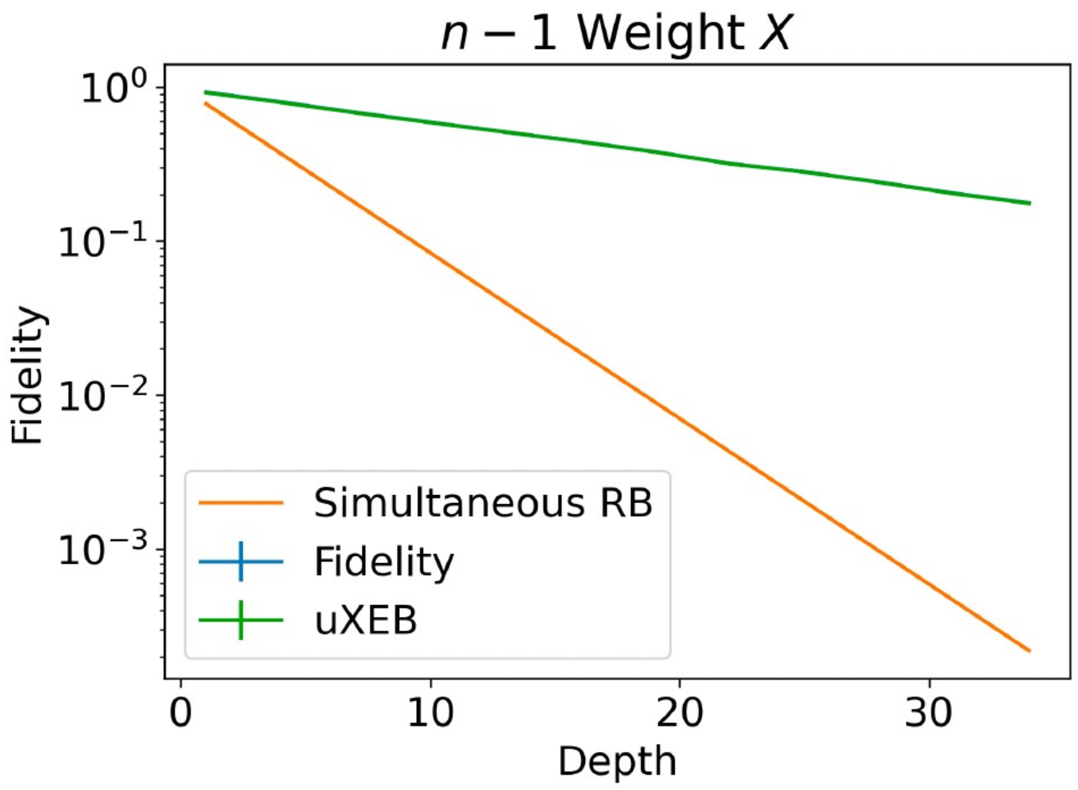
XX with probability 1%

Total error = 1%

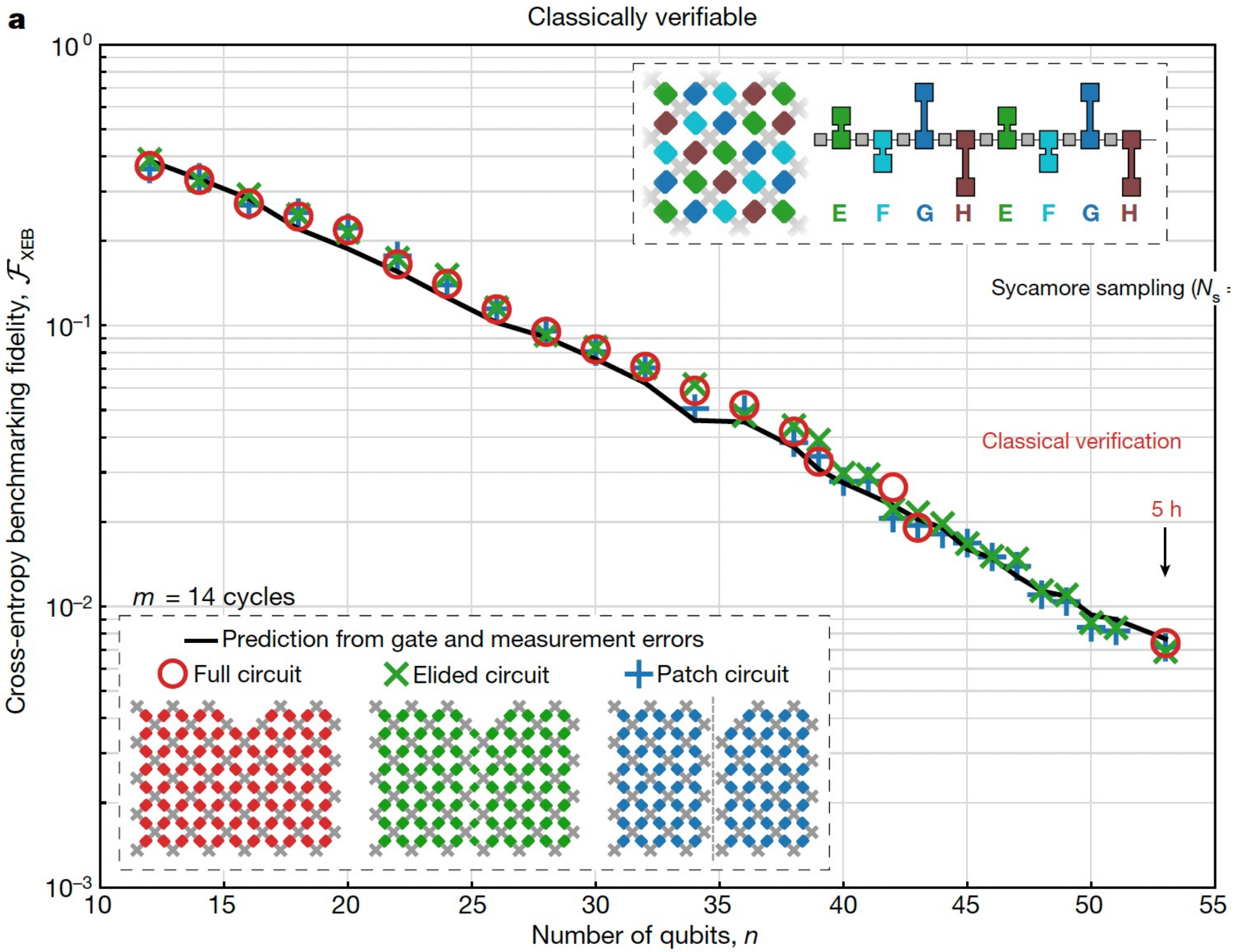
- Contributes 1% to cross entropy and fidelity
- Contributes 2% to F_{RB}

F_{RB} overestimates correlated noise

Correlated errors in fidelity estimation



Google's quantum supremacy experiment [Arute et al'19]



Observation: the linear cross entropy (fidelity) agrees with $F_{RB} = \prod_{i=1}^{m}(1 - e_i)$

Claim: The noise is uncorrelated across each 2-qubit gate

Can we understand this observation and claim from the theoretical perspective? ✅

Conclusion

- We develop a sample-efficient algorithm to estimate the total amount of noise, including all crosstalks, on a layer of non-Clifford two-qubit gates
  - Can’t scale beyond 50 qubits

- As an application, our result provides formal evidence to support Google’s claim that the coincidence between linear cross entropy and the digital error model indicated that the noise in their device was uncorrelated

Summary

- For Clifford gates, the cycle space of the pattern transfer graph determines which part of the noise model is learnable
  - Cycle benchmarking learns all learnable information
- We also discuss ways to resolve unlearnability (time permits)

- For non-Clifford gates, we show how to learn total error by introducing RCS as a powerful new tool
  - A practical application of quantum supremacy experiments
- Can RCS learn more information about noise? [Kim et al’21]

References

- Part I: “The learnability of Pauli noise”
- with Senrui Chen, Matthew Otten, Alireza Seif, Bill Fefferman, Liang Jiang
  - Arxiv: 2206.06362

- Part II: “Benchmarking near-term quantum computers via random circuit sampling”
- with Matthew Otten, Roozbeh Bassirianjahromi, Liang Jiang, Bill Fefferman
  - Arxiv: 2105.05232

How to resolve unlearnability?

- We know that unlearnability comes from gauge freedom
  - $\rho_i \mapsto \mathcal{M}(\rho_i), E_j \mapsto E_j \circ \mathcal{M}^{-1}, G_k \mapsto \mathcal{M} \circ G_k \circ \mathcal{M}^{-1}$

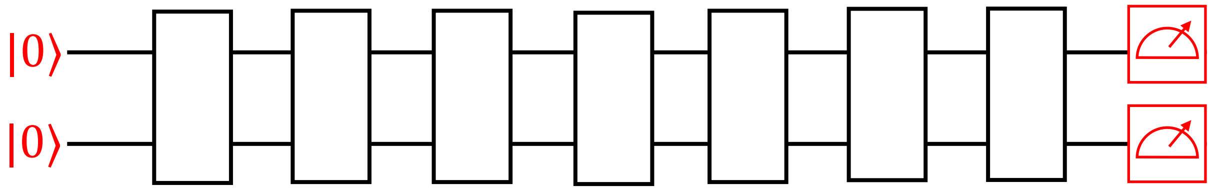
- Idea 1: unlearnability does not apply if the initial state is perfect
  - Experiments (time permits), conclude that SP noise is not small

How to resolve unlearnability?

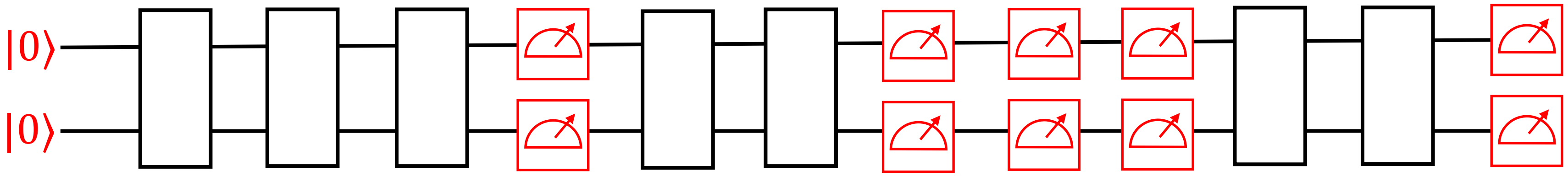
- We know that unlearnability comes from gauge freedom
  - $\rho_i \mapsto \mathcal{M}(\rho_i), E_j \mapsto E_j \circ \mathcal{M}^{-1}, G_k \mapsto \mathcal{M} \circ G_k \circ \mathcal{M}^{-1}$

- Idea 1: unlearnability does not apply if the initial state is perfect
  - Experiments (time permits), conclude that SP noise is not small
- Idea 2: use quantum non-demolition (QND) measurements

Current experiments:



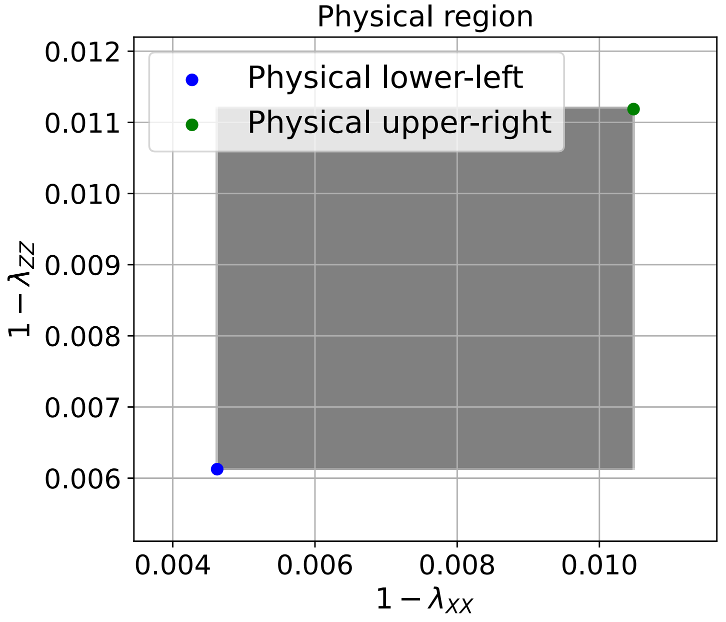
Future experiments: breaking the symmetry between state preparation and measurement

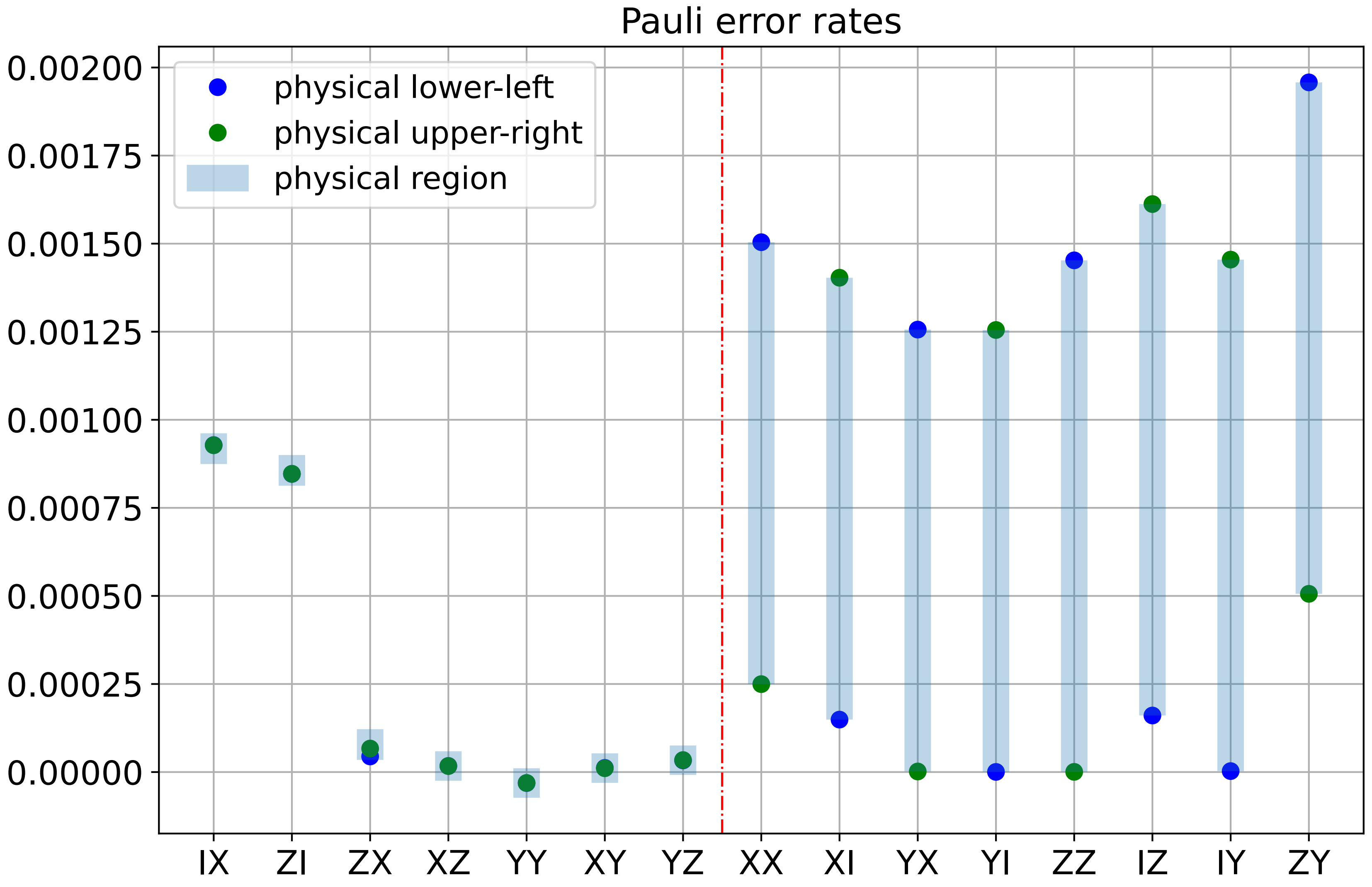


How to resolve unlearnability?

- We know that unlearnability comes from gauge freedom
  - $\rho_i \mapsto \mathcal{M}(\rho_i), E_j \mapsto E_j \circ \mathcal{M}^{-1}, G_k \mapsto \mathcal{M} \circ G_k \circ \mathcal{M}^{-1}$
- Idea 1: unlearnability does not apply if the initial state is perfect
  - Experiments (time permits), conclude that SP noise is not small
- Idea 2: use quantum non-demolition (QND) measurements
- Idea 3: parameterize the noise model using underlying physics
  - E.g. Hamiltonians and Lindbladians
  - Could have much less than 4^n parameters

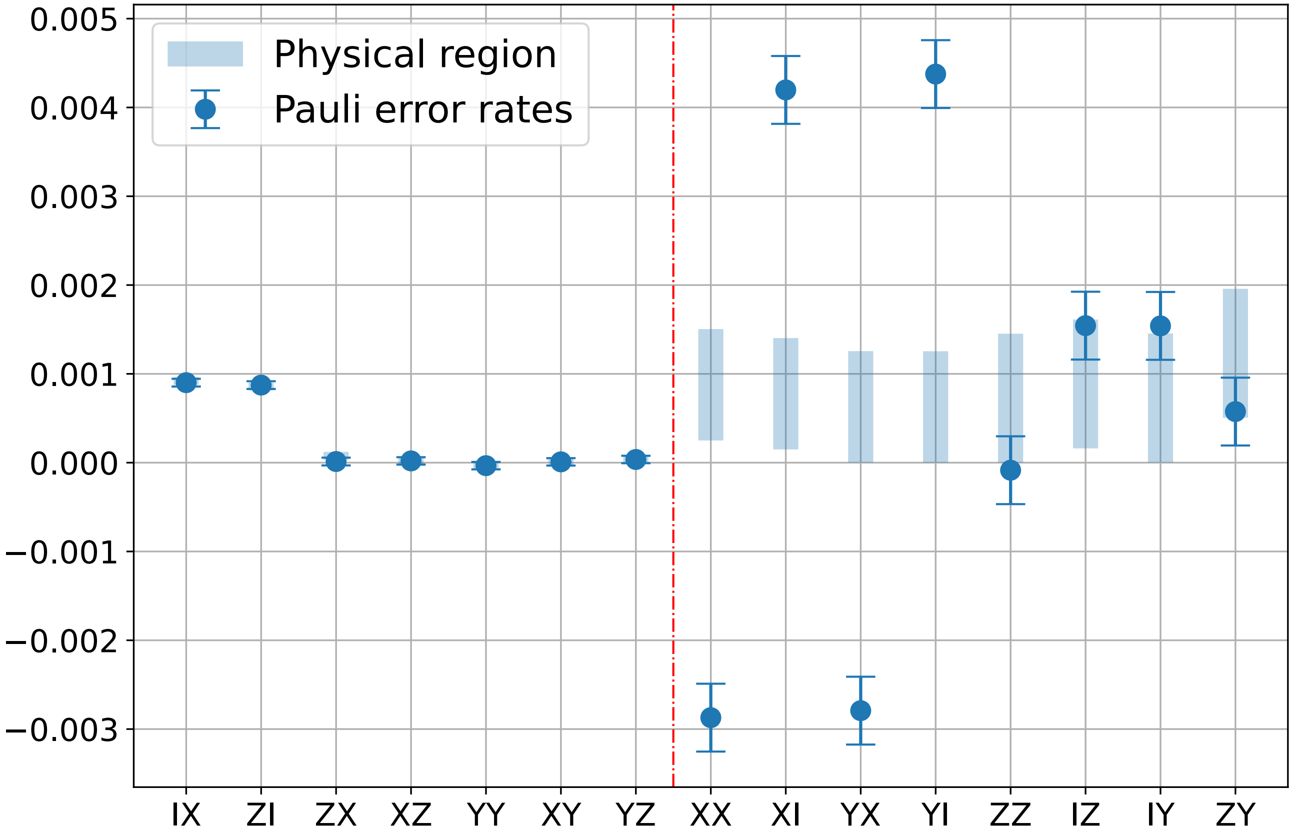
Experiments on IBM Quantum hardware





All learnable information:

$p_{IX}, p_{XY}, p_{XZ}, p_{YY}, p_{YZ}, p_{ZI}, p_{ZX},$
$p_{IY}+p_{ZY}, p_{IZ}+p_{ZY}, p_{IY}+p_{ZZ},$
$p_{XI}+p_{XX}, p_{XX}+p_{YI}, p_{XI}+p_{YX}$
(13 equations)



The result (assuming perfect initial state) is unphysical

Conclusion: state preparation noise is at least 0.6%