Noise-Resilient Near-Term Algorithms with Quantum Error Detection Codes

Dawei Zhong, 11/27/2023
Weekly MURI meeting

Near-Term Algorithms with Exponential

- **Example algorithms**
  - Simulating Time Evolution of Physical System

$$\exp(-iH\Delta t) = \prod_j e^{-iH_j\Delta t}, \text{where } H = \sum_j H_j \text{ and } \forall i, j,\ [H_j, H_k] = 0$$

Nielsen & Chuang (2010)

Near-Term Algorithms with Exponential

- **Example algorithms**
  - Simulating Time Evolution of Physical System

$$\exp(-iH\Delta t) = \prod_j e^{-iH_j\Delta t}, \text{where } H = \sum_j H_j \text{ and } \forall i, j,\ [H_j, H_k] = 0$$

  - UCC Ansatz in VQE algorithms for Quantum Chemistry

$$U(\boldsymbol{\theta}) = \prod_{l,m} U_{lm}(\theta_{lm}) = \prod_{l,m} e^{i\theta_{lm}\sum_j \mu^j_{lm}\sigma^j_n} \simeq \prod_{l,m,j} e^{i\theta_{lm}\mu^j_{lm}\sigma^j_n},\ \sigma^j_n \in \{I, X, Y, Z\}^{\otimes n}$$

arXiv: 2007.14384

Near-Term Algorithms with Exponential

- **Example algorithms**
  - Simulating Time Evolution of Physical System

$$\exp(-iH\Delta t) = \prod_j e^{-iH_j\Delta t}, \text{where } H = \sum_j H_j \text{ and } \forall i,j,\ [H_j, H_k] = 0$$

  - UCC Ansatz in VQE algorithms for Quantum Chemistry'

$$U(\boldsymbol{\theta}) = \prod_{l,m} U_{lm}(\theta_{lm}) = \prod_{l,m} e^{i\theta_{lm}\sum_j \mu^j_{lm}\sigma^j_n} \simeq \prod_{l,m,j} e^{i\theta_{lm}\mu^j_{lm}\sigma^j_n},\ \sigma^j_n \in \{I, X, Y, Z\}^{\otimes n}$$

  - QAOA, with problem and mixer Hamiltonian

$$U(\boldsymbol{\gamma}, \boldsymbol{\beta}) = e^{-i\beta_p H_M} e^{-i\gamma_p H_P} e^{-i\beta_{p-1} H_M} e^{-i\gamma_{p-1} H_P} \ldots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_P}$$

arXiv: 2007.14384

Near-Term Algorithms with Exponential

- **Example algorithms**
  - Simulating Time Evolution of Physical System

$$\exp(-iH\Delta t) = \prod_j e^{-iH_j\Delta t}, \text{where } H = \sum_j H_j \text{ and } \forall i,j,\ [H_j, H_k] = 0$$

  - UCC Ansatz in VQE algorithms for Quantum Chemistry'

$$U(\boldsymbol{\theta}) = \prod_{l,m} U_{lm}(\theta_{lm}) = \prod_{l,m} e^{i\theta_{lm}\sum_j \mu^j_{lm}\sigma^j_n} \simeq \prod_{l,m,j} e^{i\theta_{lm}\mu^j_{lm}\sigma^j_n},\ \sigma^j_n \in \{I, X, Y, Z\}^{\otimes n}$$

  - QAOA, with problem and mixer Hamiltonian

$$U(\boldsymbol{\gamma}, \boldsymbol{\beta}) = e^{-i\beta_p H_M} e^{-i\gamma_p H_P} e^{-i\beta_{p-1} H_M} e^{-i\gamma_{p-1} H_P} \ldots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_P}$$

Near-Term Algorithms with Exponential

- **Example algorithms**
  - Simulating Time Evolution of Physical System

$$\exp(-iH\Delta t) = \prod_j e^{-iH_j\Delta t}, \text{where } H = \sum_j H_j \text{ and } \forall i,j,\ [H_j, H_k] = 0$$

  - UCC Ansatz in VQE algorithms for Quantum Chemistry'

$$U(\boldsymbol{\theta}) = \prod_{l,m} U_{lm}(\theta_{lm}) = \prod_{l,m} e^{i\theta_{lm}\sum_j \mu^j_{lm}\sigma^j_n} \simeq \prod_{l,m,j} e^{i\theta_{lm}\mu^j_{lm}\sigma^j_n},\ \sigma^j_n \in \{I, X, Y, Z\}^{\otimes n}$$

  - QAOA, with problem and mixer Hamiltonian

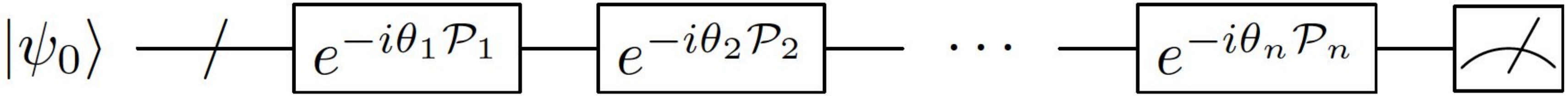
$$U(\boldsymbol{\gamma}, \boldsymbol{\beta}) = e^{-i\beta_p H_M} e^{-i\gamma_p H_P} e^{-i\beta_{p-1} H_M} e^{-i\gamma_{p-1} H_P} \ldots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_P}$$

Hamiltonian -> Pauli Operation!

Jordan-Wigner, Bravyi-Kitaev and parity mapping, arXiv: 2111.05176

Near-Term Algorithms with Exponential

- **General Protocol**



With ***n***-qubits core operation

$$\exp(-i\theta \mathcal{P}_i),\ \mathcal{P}_i = \{I, X, Y, Z\}^{\otimes n}$$

Two types of “Fault-Tolerance”

- **Algorithm Type.** Fault-tolerance quantum simulation.
  - Less or no systematic error, e.g., Trotter error.

arXiv: 2105.12767

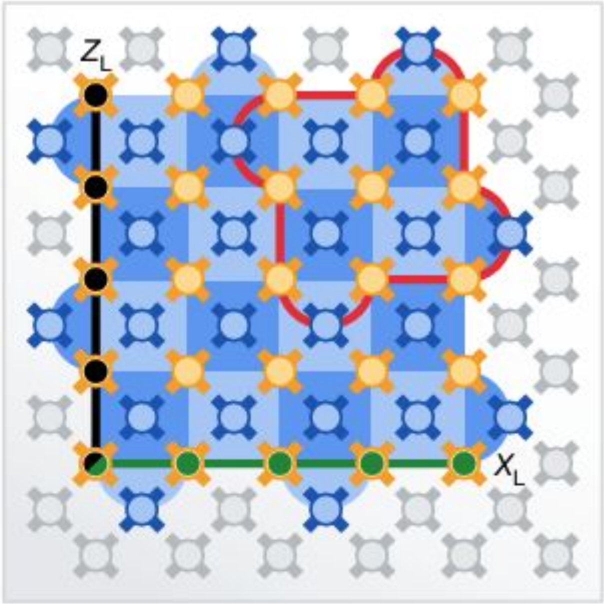
$$e^{-i\sum_{j=1}^{m} H_j t} = \left(\prod_{j=1}^{m} e^{-iH_j t/r}\right)^r + O(m^2t^2/r),$$

- **Noise Type.** Fault-tolerance quantum computation.
  - Less or no error from hardware

Error Reduction Techniques

- **FTQC:** Protect logical operation by error correction code on below-threshold device
- **Challenges:**
  - “above-threshold” device
  - Limited amount of qubits
  - ? all-to-all connectivity?

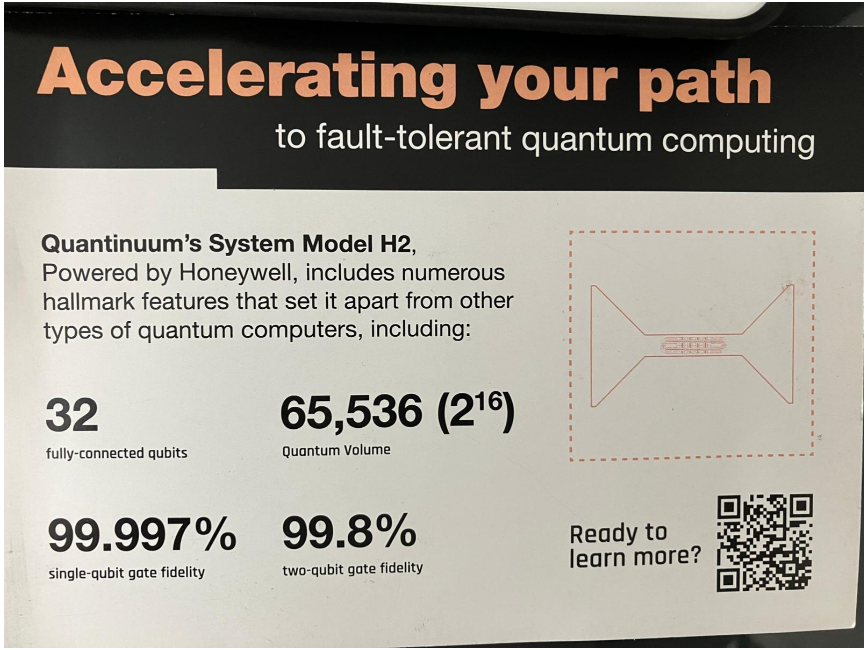
a



Distance 5 code with 25 data qubits and 24 measurement qubits
Google’s paper
arXiv: 2207.06431

Error Reduction Techniques

- **FTQC:** Protect logical operation by error correction code on below-threshold device
- **Challenges for application:**
    - “above-threshold” device
    - Limited amount of qubits
    - ? all-to-all connectivity?



Error Reduction Techniques

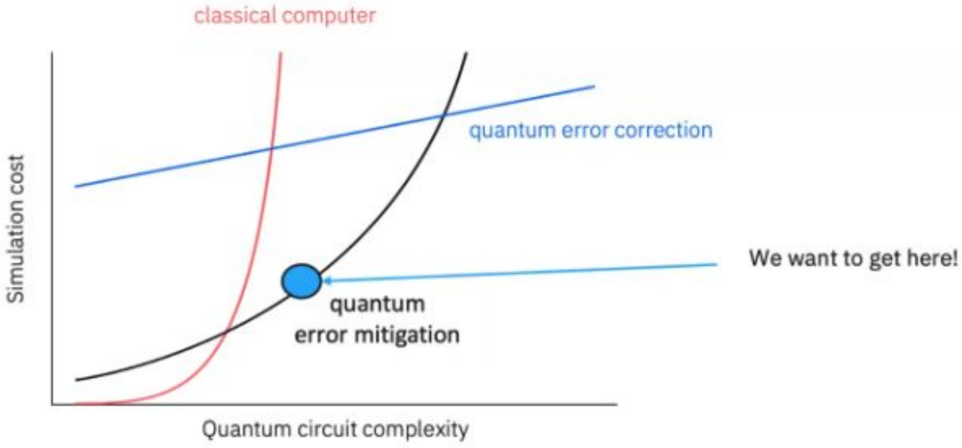
- **FTQC:** Protect logical operation by error correction code on below-threshold device
- **Challenges:**
  - “above-threshold” device
  - Limited amount of qubits
  - ? all-to-all connectivity?
  - non-Clifford gate

arXiv: 1905.06903

Error Reduction Techniques

- **Error Mitigation**: Use quantum information processing to reduce noise

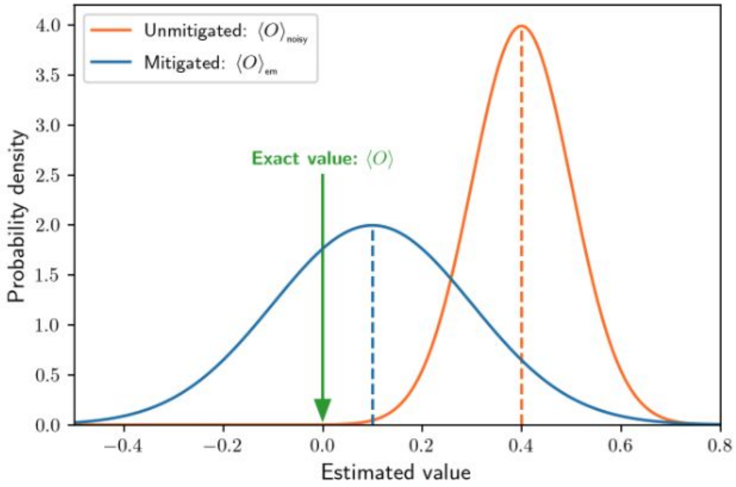
IBM 100 x 100 circuits, 100 qubits and 100 depths
https://qiskit.org/ecosystem/ibm-runtime/tutorials/Error-Suppression-and-Error-Mitigation.html



Error Reduction Techniques

- **Error Mitigation**: Use quantum information processing to reduce noise
- **Challenges:**
  - ZNE and PEC is for expectation value

https://qiskit.org/ecosystem/ibm-runtime/tutorials/Error-Suppression-and-Error-Mitigation.html



Error Reduction Techniques

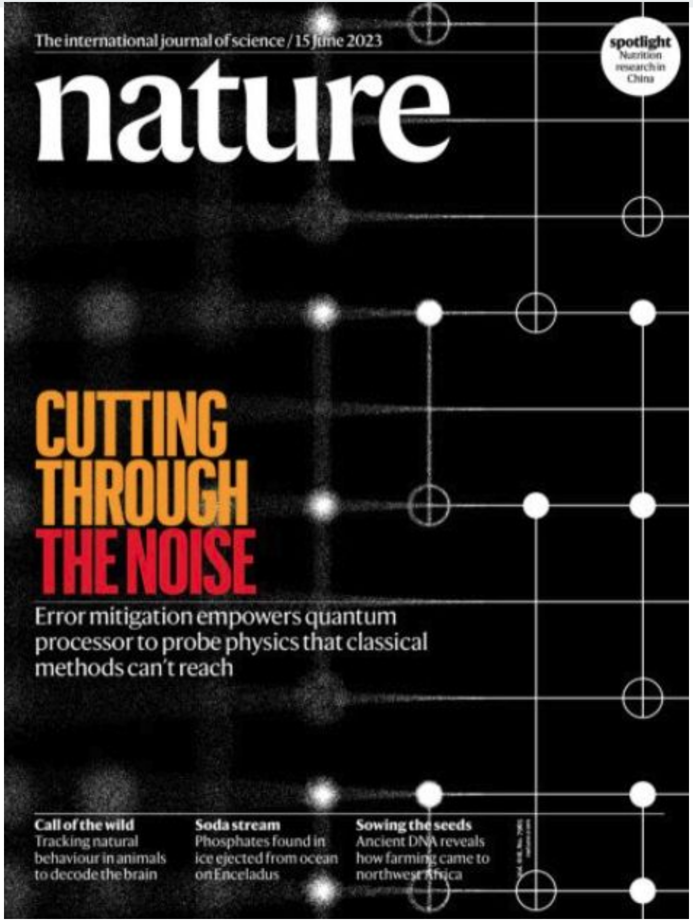
- **Error Mitigation:** Use quantum information processing to reduce noise
- **Challenges:**
  - ZNE and PEC is for expectation value
  - Sampling overhead
  - DD is also bias

Methods	T-REx	ZNE	PEC
Assumptions	None	Ability to scale the noise	Full knowledge of the noise
Qubit overhead	1	1	1
Sampling overhead	2	$N_{\text{noise-factors}}$	$\mathcal{O}(e^{\lambda N_{\text{layers}}})$
Bias	0	$\mathcal{O}(\lambda^{N_{\text{noise-factors}}})$	0

https://qiskit.org/ecosystem/ibm-runtime/tutorials/Error-Suppression-and-Error-Mitigation.html

Error Reduction Techniques

- **Error Mitigation:** Use quantum information processing to reduce noise
- **Challenges:**
  - ZNE and PEC is for expectation value
  - Sampling overhead
  - DD is also bias



https://www.nature.com/articles/s41586-023-06096-3

Error Reduction Techniques for Near-Term?

- **Error Detection Code Family:** $[[n, n-2, 2]]$
  - Qubit overhead as 1
  - No sampling overhead
  - Post-selection but not recovery,
    - Less gates required in “above-threshold” hardware
    - Simple to implement
- **Disadvantages:** all-to-all connectivity, small distance and (perhaps) large post-selection rate

Error Detection Code Basic

- Stabilizers: $X^{\otimes n}, Z^{\otimes n}$
- Logical Operator $\overline{X}_i = X_i X_{n-1}, \overline{Z}_i = Z_i Z_n, \overline{Y}_i = i\overline{X}_i\overline{Z}_i$
- 4 Qubit Example

$$|\overline{00}\rangle = \frac{1}{\sqrt{2}}(|0000\rangle + |1111\rangle)$$

$$|\overline{01}\rangle = \frac{1}{\sqrt{2}}(|0110\rangle + |1001\rangle)$$

$$|\overline{10}\rangle = \frac{1}{\sqrt{2}}(|0101\rangle + |1010\rangle)$$

$$|\overline{11}\rangle = \frac{1}{\sqrt{2}}(|1100\rangle + |0011\rangle)$$

Weakly Fault Tolerance (Weight-2)

- Logical Operator
- Rotation Operation

$$\overline{X}_i = X_i X_{n-1}, \overline{Z}_i = Z_i Z_n, \overline{Y}_i = i\overline{X}_i\overline{Z}_i$$

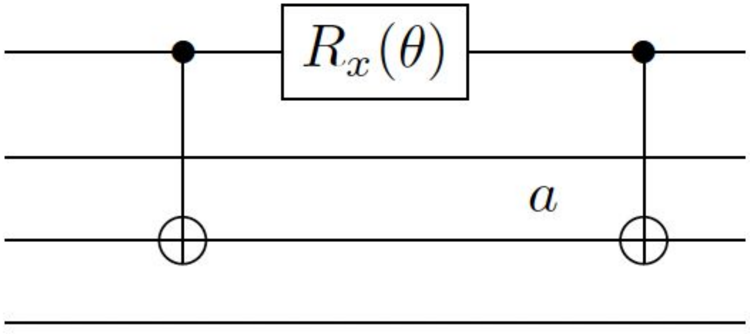
$$R_x(\theta) \equiv e^{-i\theta X/2} \qquad R_z(\theta) \equiv e^{-i\theta Z/2}$$

$$\overline{R_{X_i}}(\theta) = \exp\left(-i\frac{\theta}{2}X_i X_{n-1}\right), \overline{R_{Z_i}}(\theta) = \exp\left(-i\frac{\theta}{2}Z_i Z_{n-1}\right)$$

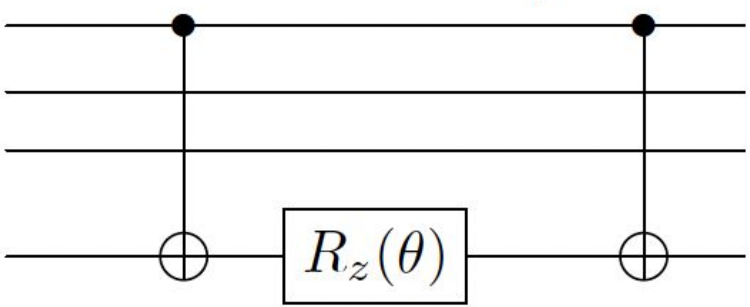
Weakly Fault Tolerance (Weight-2)

- Logical Operator $\overline{X}_i = X_i X_{n-1}, \overline{Z}_i = Z_i Z_n, \overline{Y}_i = i\overline{X}_i\overline{Z}_i$
- Rotation Operation

$$\overline{R_{X_i}}(\theta) = \exp\left(-i\frac{\theta}{2}X_i X_{n-1}\right), \overline{R_{Z_i}}(\theta) = \exp\left(-i\frac{\theta}{2}Z_i Z_{n-1}\right)$$



a

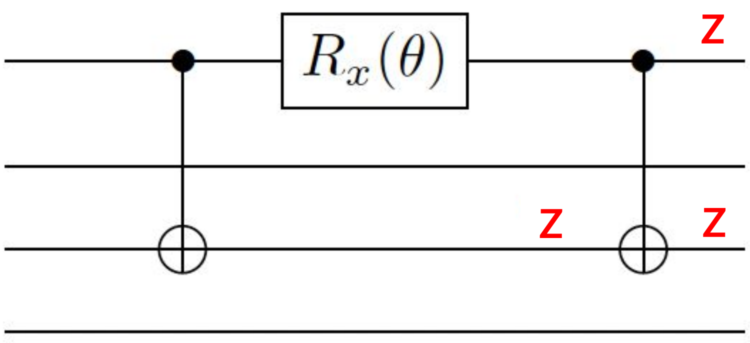
b


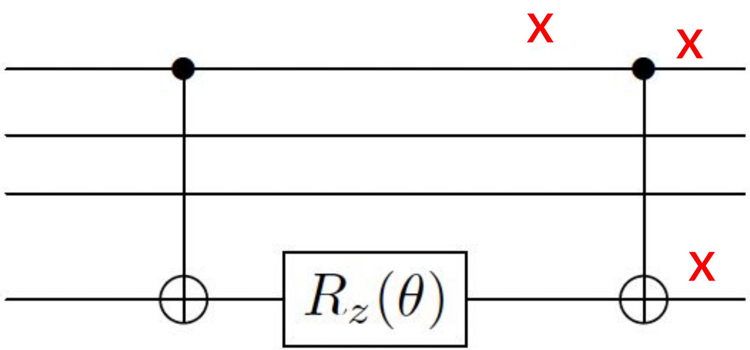
Stabilizers: $X^{\otimes n}, Z^{\otimes n}$

Weakly Fault Tolerance (Weight-2)

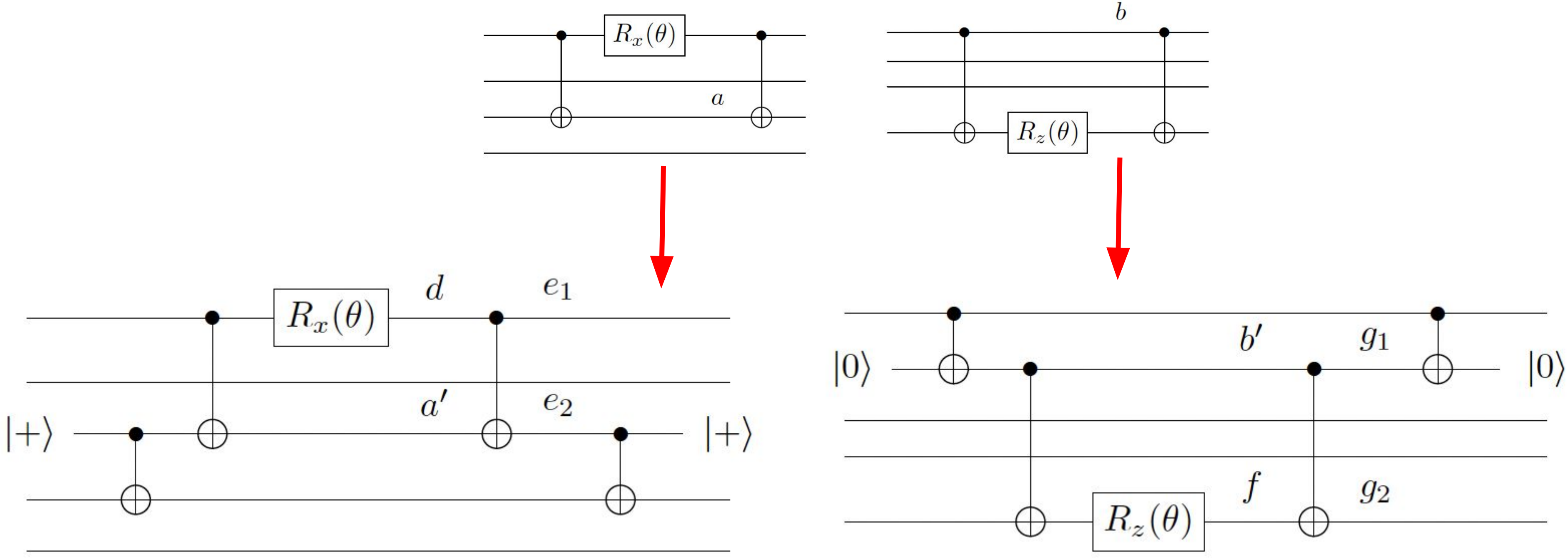
- Logical Operator $\overline{X}_i = X_i X_{n-1}, \overline{Z}_i = Z_i Z_n, \overline{Y}_i = i\overline{X}_i\overline{Z}_i$
- Rotation Operation

$$\overline{R_{X_i}}(\theta) = \exp\left(-i\frac{\theta}{2}X_i X_{n-1}\right), \overline{R_{Z_i}}(\theta) = \exp\left(-i\frac{\theta}{2}Z_i Z_{n-1}\right)$$





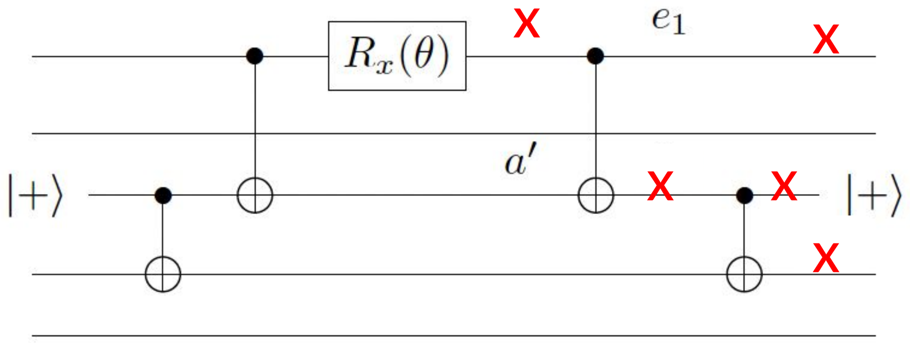
Weakly Fault Tolerance (Weight-2)

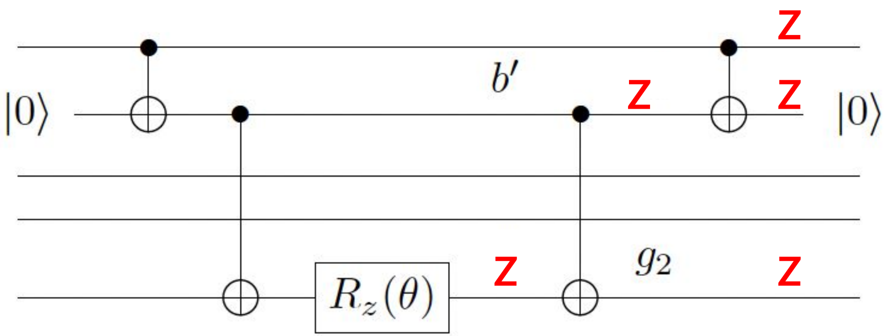


By Prof. Brun and Christopher Gerhard, in preparation

Weakly Fault Tolerance (Weight-2)

- **Weak:** Error from imprecise rotation cannot be detected
- **Fault-tolerance:** All other weight-1 and weight-2 (gate) Pauli error can be detected by either syndrome measurement or ancilla





By Prof. Brun and Christopher Gerhard, in preparation

General Weakly Fault Tolerance (Weight-2)?

- Can we add a few ancilla to achieve weakly fault tolerance for general exponential operator?

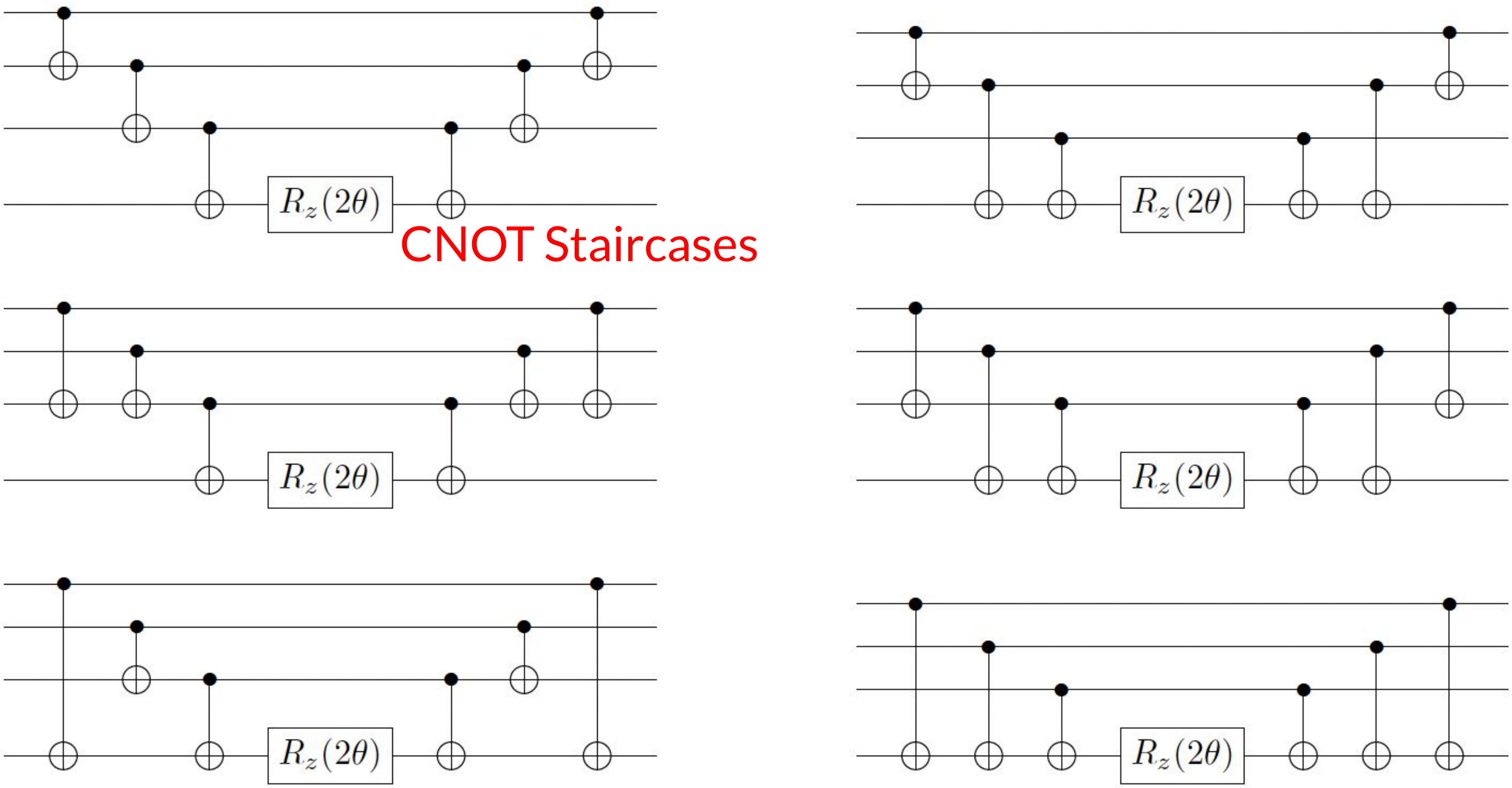
$$\exp(-i\theta \mathcal{P}_i), \ \mathcal{P}_i = \{I, X, Y, Z\}^{\otimes n}$$

Substitute by its physical operator

$$\exp(-i\theta \mathcal{P}_i),\ \mathcal{P}_i = \{I, X, Y, Z\}^{\otimes n}$$

One Observations

- **Observation:**
  - Suppose now we already substitute the Pauli in logical exponential operator by its physical operator
  - There are multiple ways to implement a same exponential operator, and some of them might be better than others on noise reduction

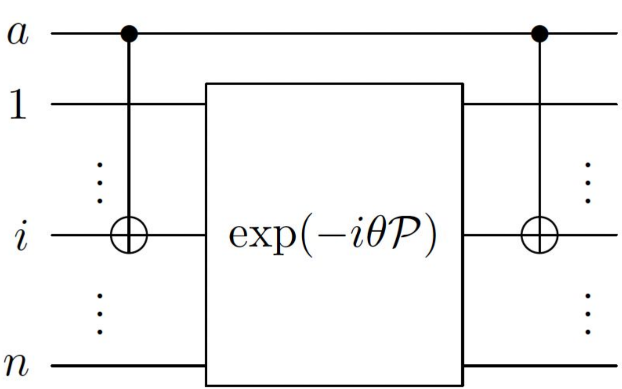


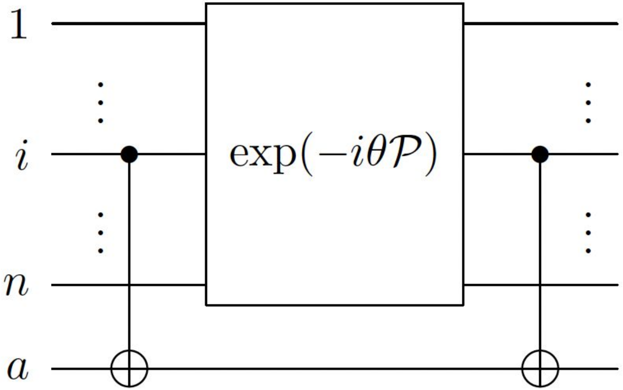
(d) All circuits are $\exp(-i\theta Z_1 Z_2 Z_3 Z_4)$

$$\exp(-i\theta\mathcal{P}_i),\ \mathcal{P}_i = \{I, X, Y, Z\}^{\otimes n}$$

Conclusion from Observations

- **Conclusion.** We might find a way to create these “candidates” and search for the best one
- **Iterative construction**





$$\exp(-i\theta\mathcal{P}_i),\ \mathcal{P}_i = \{I, X, Y, Z\}^{\otimes n}$$

Add Ancilla

- **How to add ancilla:** Due to the limited number of qubit, we would like to add only one ancilla. But, how to add an ancilla to
  - Preserve the codespace
  - Preserve our expected operation
- We can prove that, if using j-th qubit as ancilla

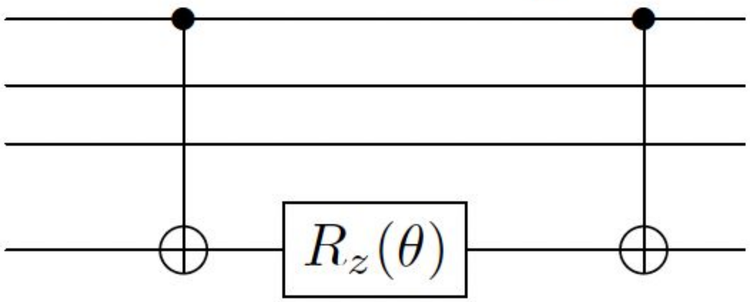
$$\exp(-i\theta\mathcal{P}) \qquad \mathcal{P} = P_1 \otimes P_2 \otimes \cdots \otimes P_{j-1} \otimes P_j \otimes P_{j+1} \otimes \cdots \otimes P_n$$

↓

$$\exp(-i\theta\mathcal{P}') \qquad \mathcal{P}' = P_1 \otimes P_2 \otimes \cdots \otimes \underline{P_{j-1} \otimes P_{j+1}} \otimes \cdots \otimes P_n$$

Weaker than Weakly Fault Tolerance

- **Cannot find weakly fault-tolerant circuit when physical Pauli has weight >= 3**
- **Metric: Remaining error rate** (in the unit of percentage)
  - Definition: number_of_logical_error / number_of_total_error * 100%
  - Example: number_of_total_error = 2 * 15 + 3 = 33

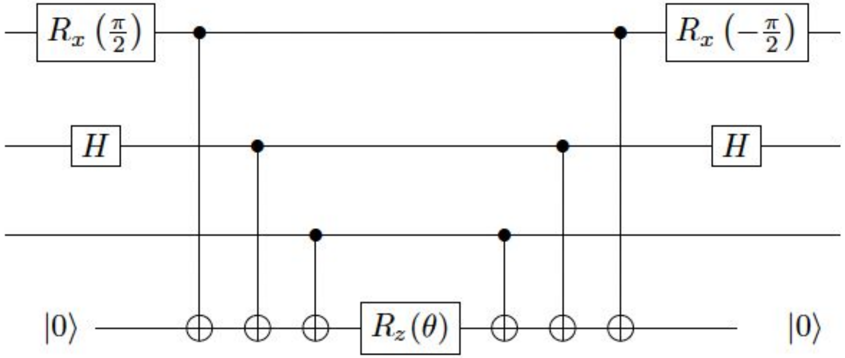


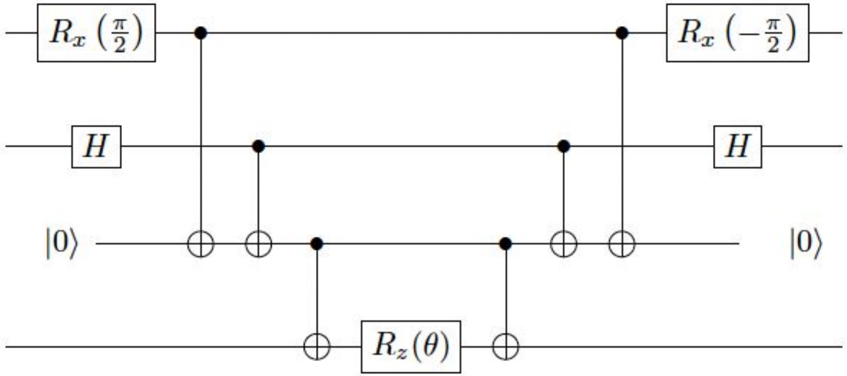
Weaker than Weakly Fault Tolerance

- **Remaining error rate** for weakly fault-tolerant circuit: 4.76%
- **Purpose:** Find circuits with ancilla that with small remaining error rate

Result 1 - Logical Rotation-Y

- **Two equivalent physical circuit.** Remaining error rate, 4.76%

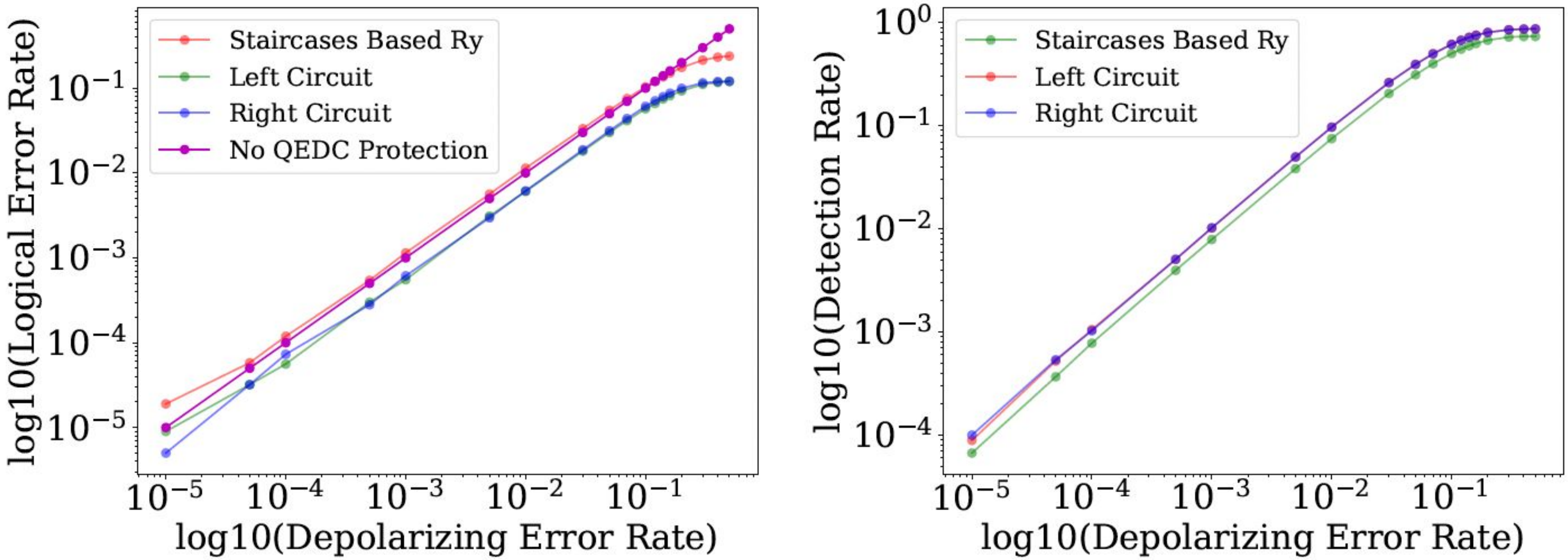




$$\overline{R_{Y_i}}(\theta) = \exp\left(-i\theta Y_i X_{n-1} Z_n/2\right) \text{ in } [[n, n-2, 2]]$$

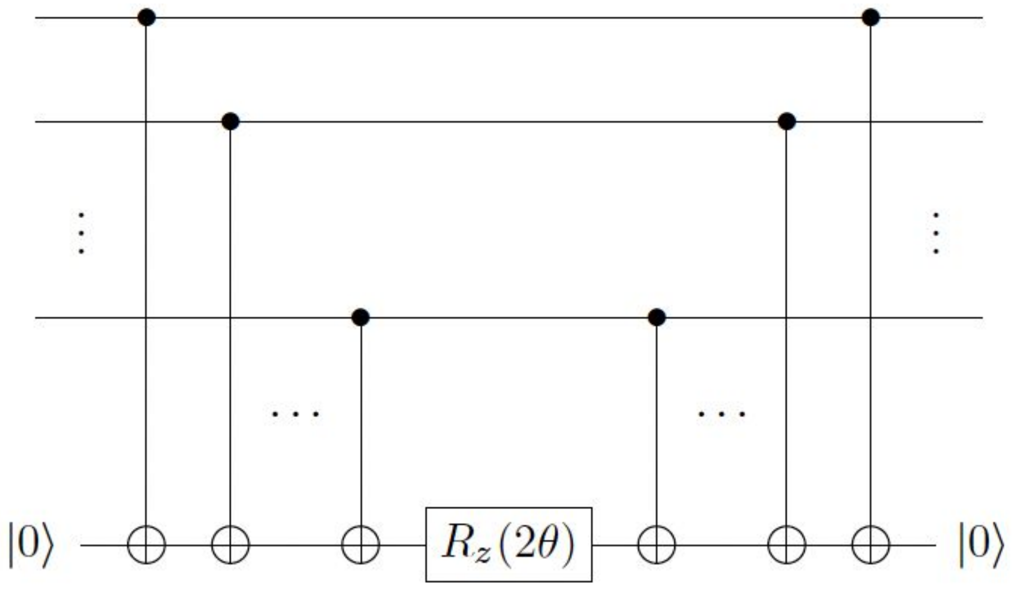
Result 1 - Logical Rotation-Y

- **Two equivalent physical circuit.** Remaining error rate, 4.76%



Result 2 - A More General Exponential

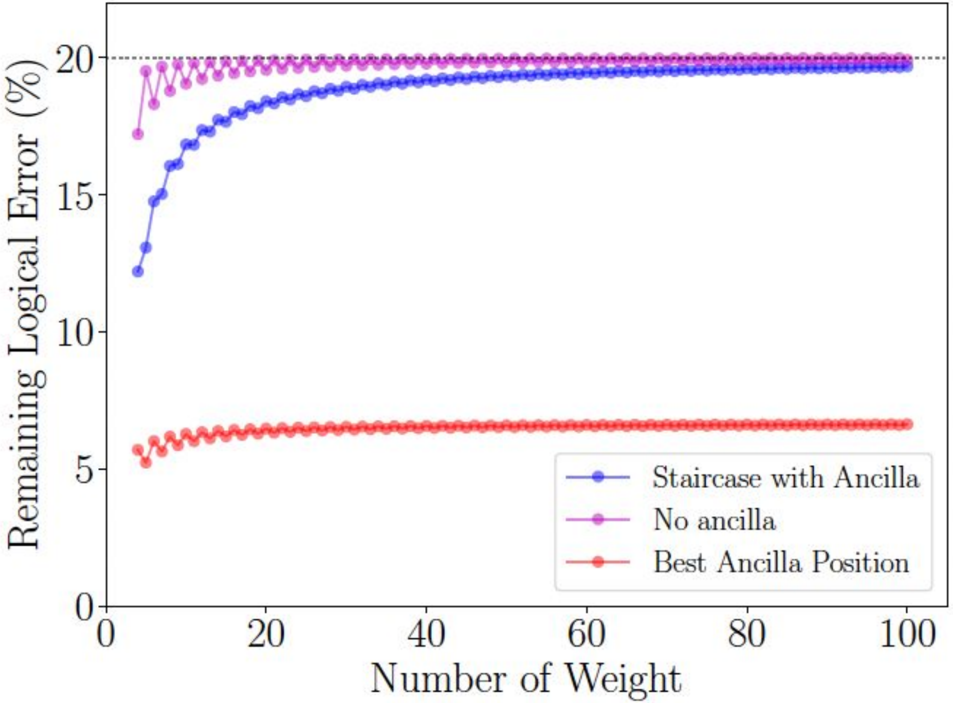
- **One candidate for a weight-n physical Pauli.** (single qubit gates less noisy)



$$\exp(-i\theta Z^{\otimes n})$$

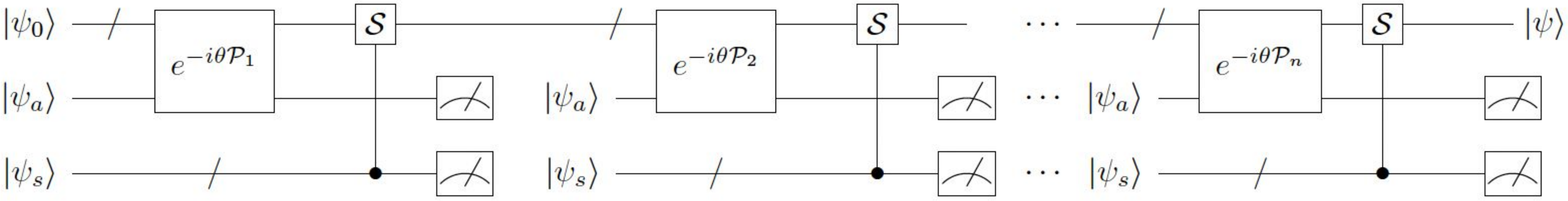
Result 2 - A More General Exponential

- **One candidate for a weight-n physical Pauli.** (single qubit gates less noisy)



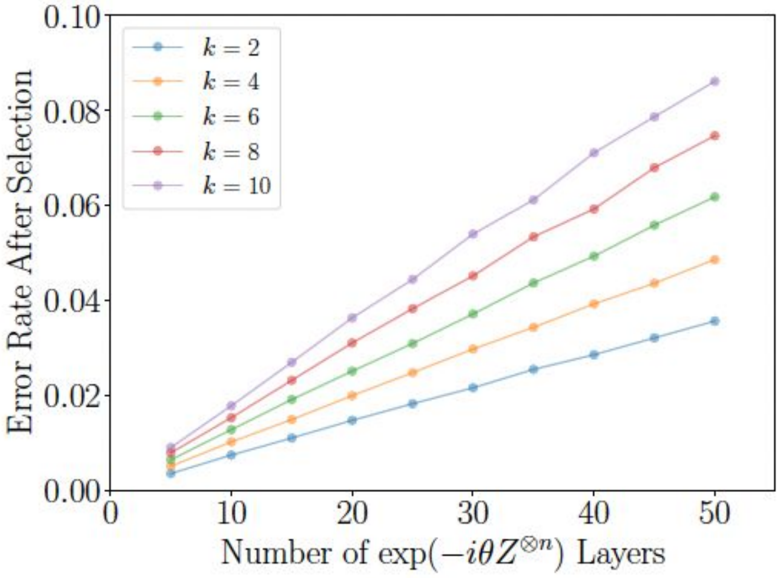
Result 3 - With Mid-Circuit Measurement

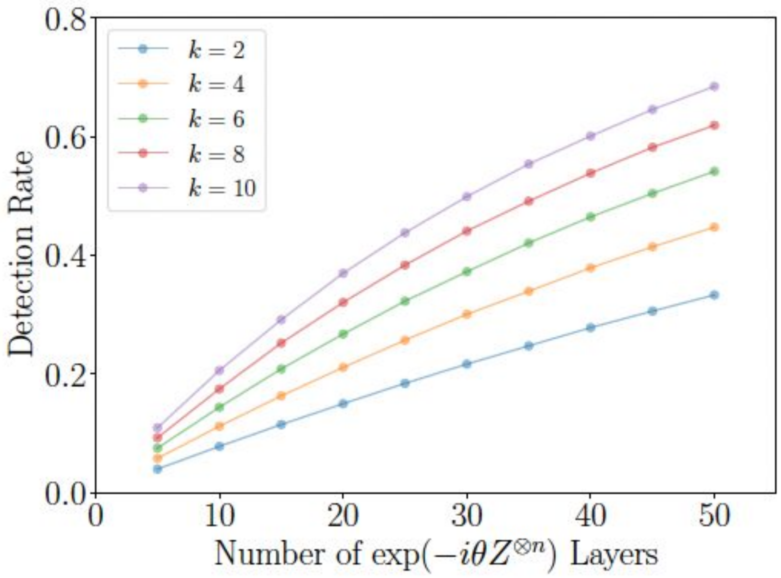
- **Protocol.** Use in deep circuit



Result 3 - With Mid-Circuit Measurement

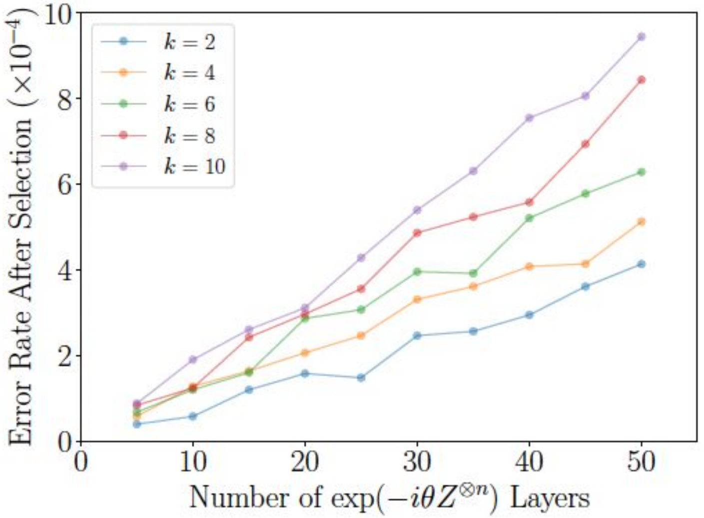
- **With noiseless mid-circuit measurement.**
- Depolarizing error rate = 0.001

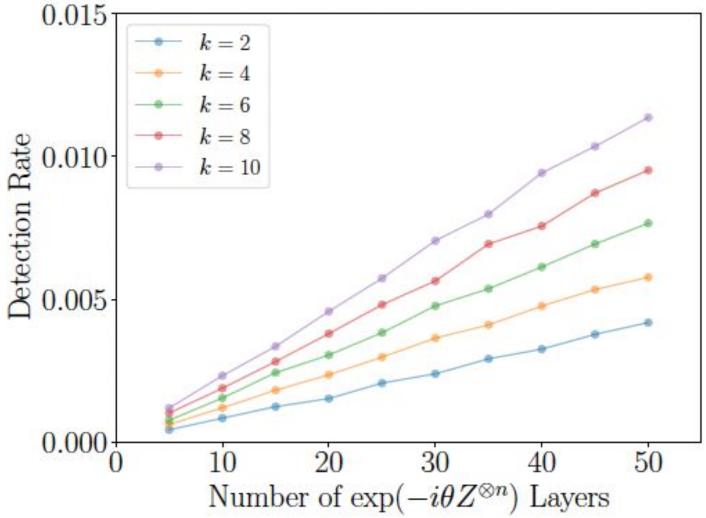




Result 3 - With Mid-Circuit Measurement

- **With noiseless mid-circuit measurement.**
- Depolarizing error rate = 0.00001





Future Work

- Noisy but weakly fault-tolerant syndrome measurement
- Test on some real problems