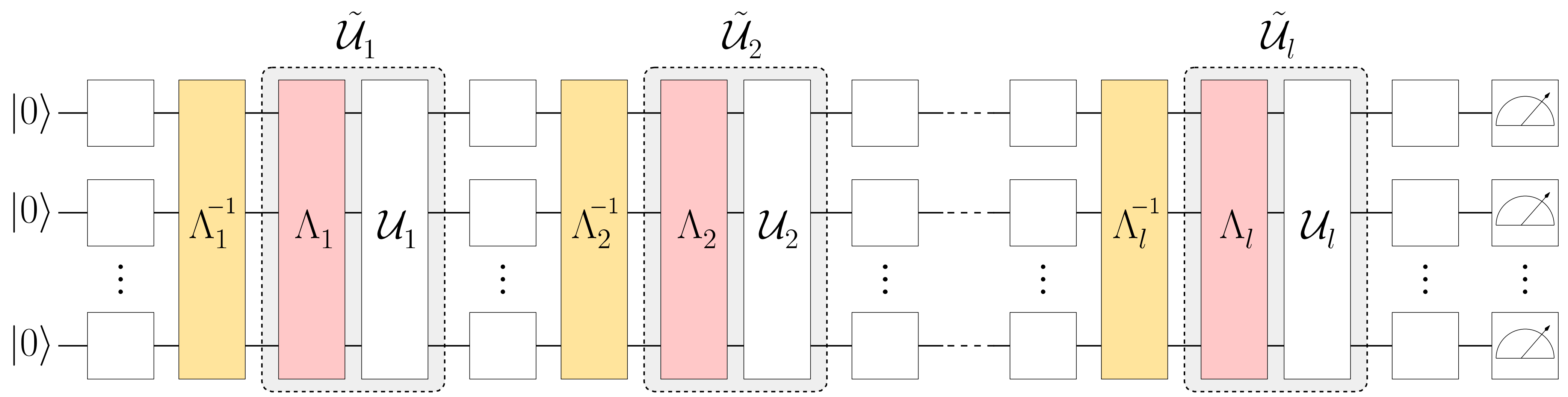
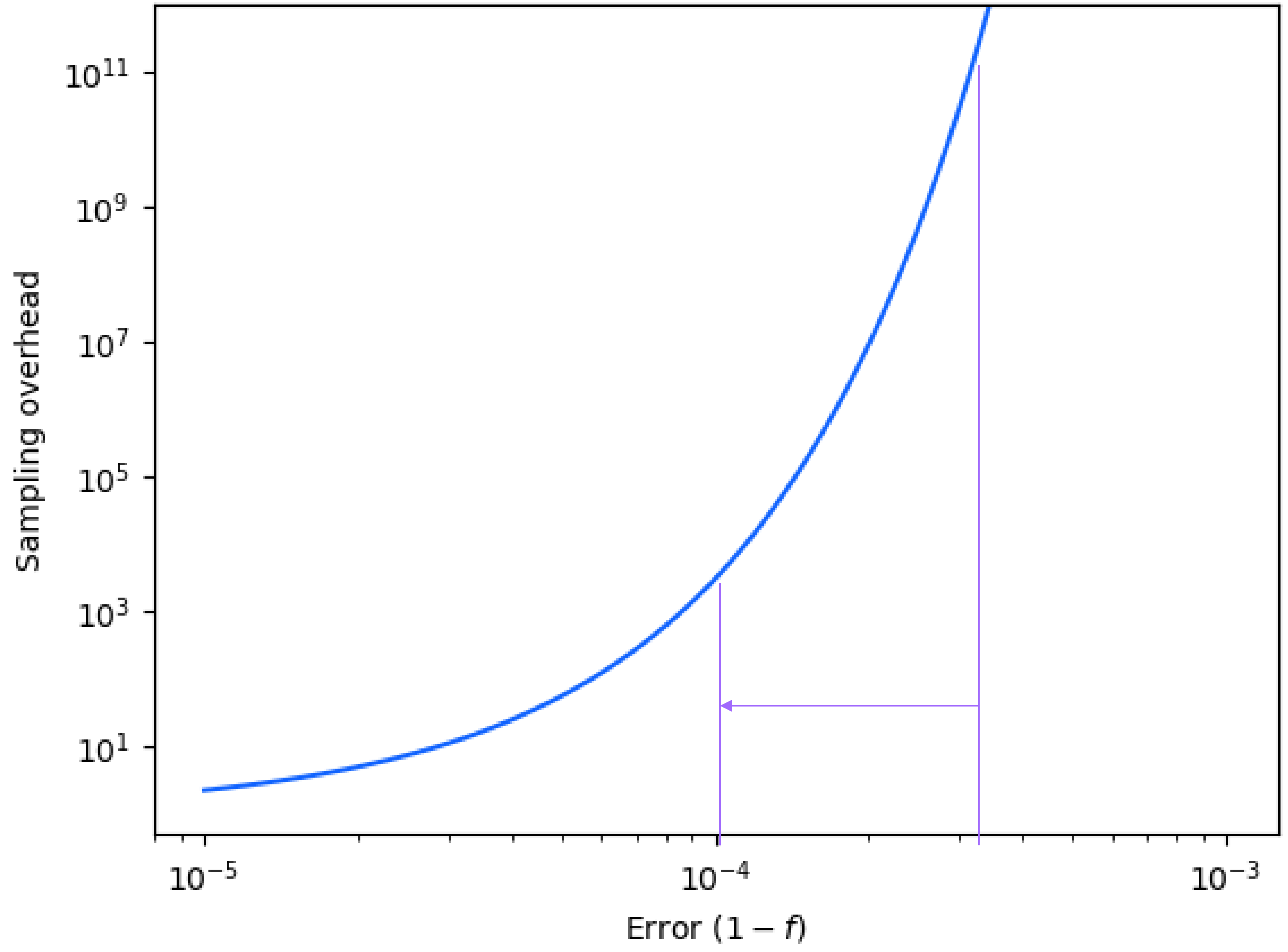
Low-overhead error detection with spacetime codes

Ali Javadi-Abhari
IBM Quantum

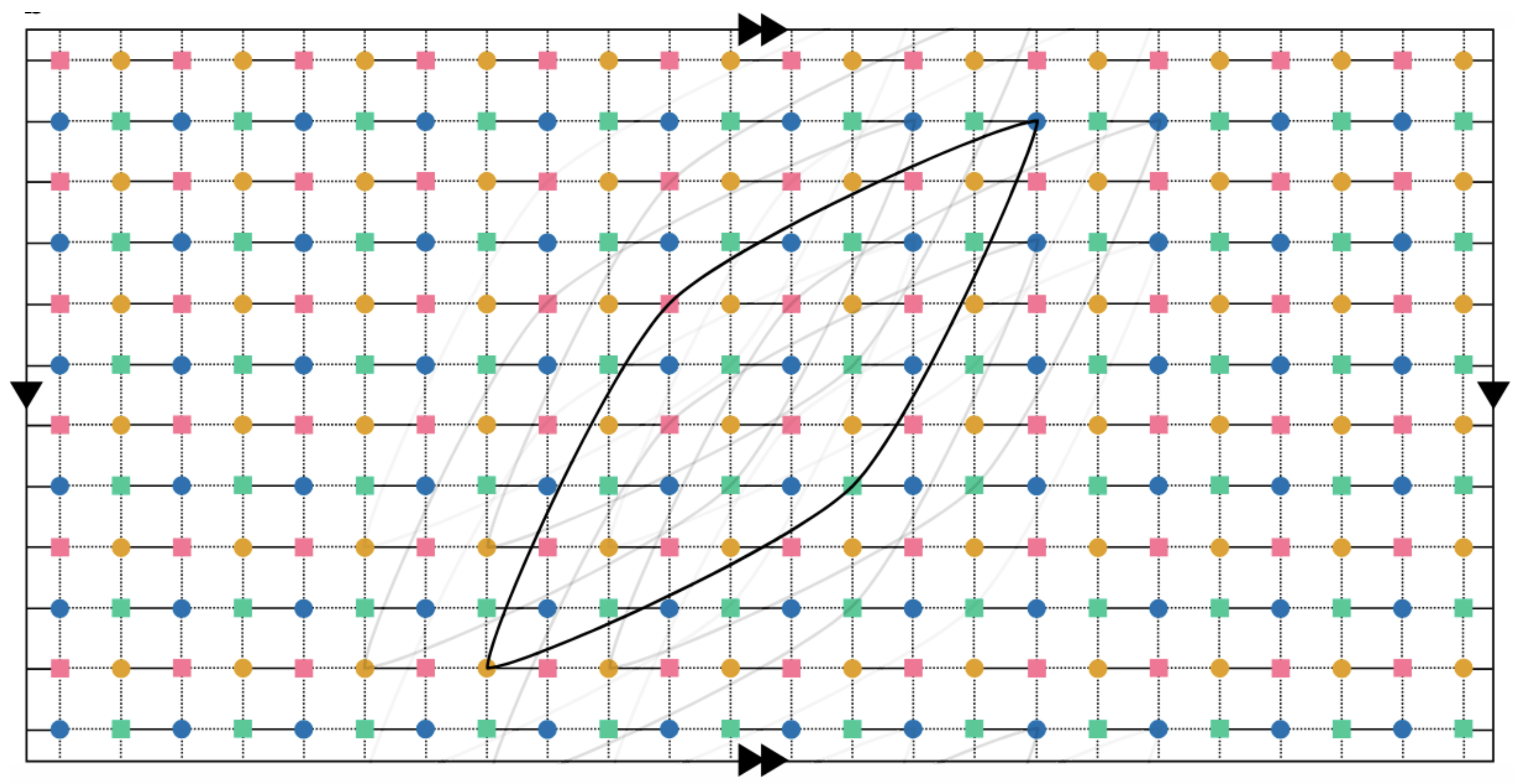
Quantum Error Mitigation

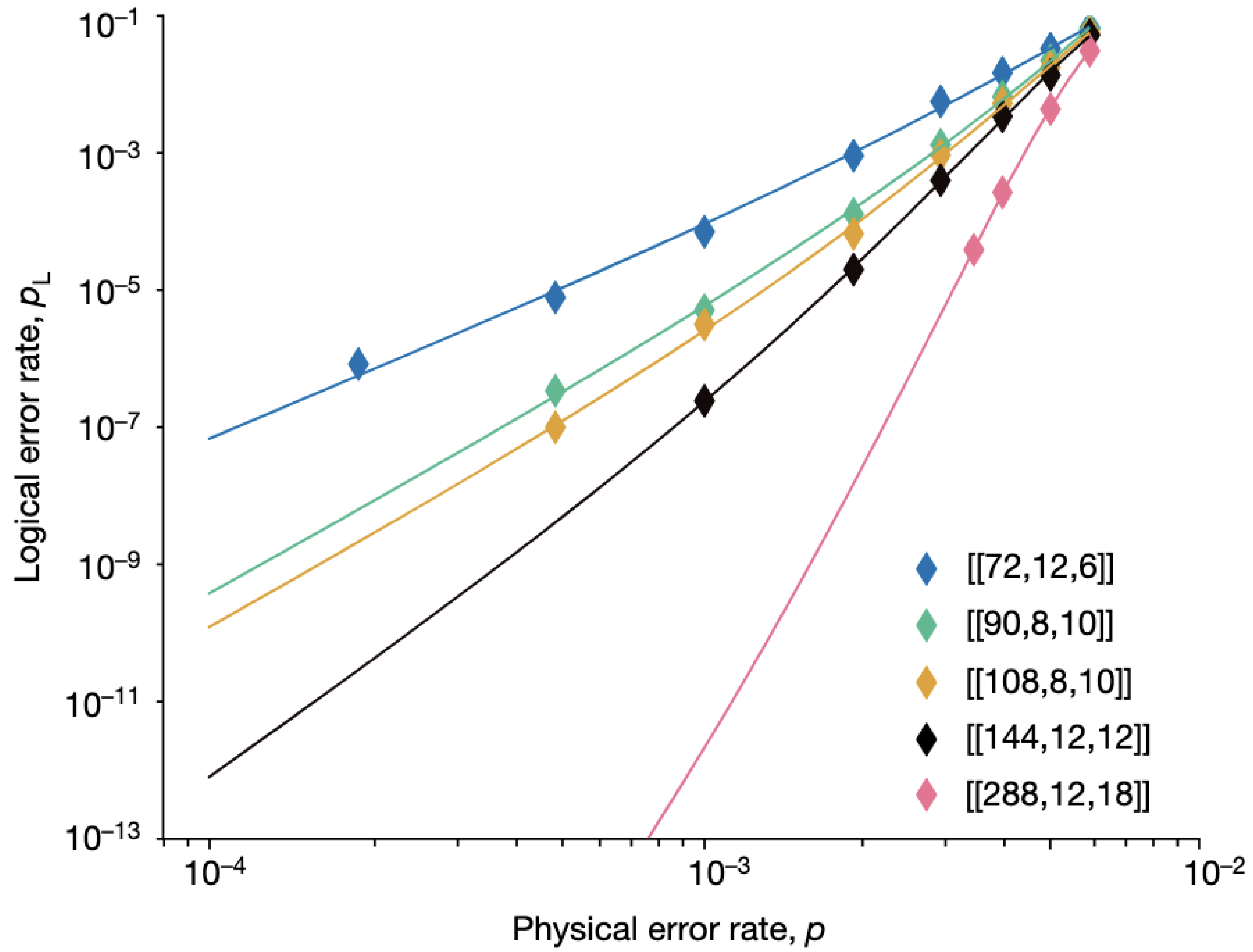




Van den Berg, Minev, Kandala, Temme, Probabilistic error cancellation with sparse Pauli-Lindblad models, *Nature Physics 2023*

Quantum Error Correction





Bravyi, Cross, Gambetta, Maslov, Rall, Yoder, High-threshold and low-overhead fault-tolerant quantum memory, Nature 2024

Trading off samples vs. qubits

Error detection uses extra qubits to reduce sampling cost.
It is also **single shot**.

Trading off samples vs. qubits

Consider an n-qubit quantum circuit with m gates being executed on a quantum computer with gate error p.

We would like to boost the fidelity from $F = (1-p)^m$ to $F' = 1 - \epsilon$.

Method	**Qubit overhead**	**Sampling overhead**	**Single shot**
Mitigation (QEM)	-	$\mathcal{O}\left(\left(\frac{1-\epsilon}{(1-p)^m}\right)^4\right)$	×
Detection (QED)	$\mathcal{O}(1)$	$\mathcal{O}\left(\frac{1-\epsilon}{(1-p)^m}\right)$	✓
Correction (QEC)	$\mathcal{O}\left(n \cdot \log_p^2(\epsilon/m)\right)$	-	✓

But how can we detect errors?

Symmetry verification

Measure conserved quantities of a physical system.

Examples:

Spin parity.
Fermionic particle number.

1. Bonet-Monroig et al. Low-cost error mitigation by symmetry verification, PRA 2018
2. McArdle, Yuan, Benjami, Error-Mitigated Digital Quantum Simulation, PRL 2019

Error detecting codes

Encode qubits in a code. Compute using fault-tolerant gadgets.

Examples:

[[2m, 2m-2, 2]] code.
[[8, 3, 2]] color code.

3. Gottesman, Theory of fault-tolerant quantum computation, PRA 1998
4. Campbell, The smallest interesting colour code, 2016

Coherent Pauli checks

Check that a Clifford circuit maps Paulis to correct Paulis.

Subject of this talk.

5. Debroy, Brown, Extended flag gadgets for low-overhead circuit verification, PRA 2020
6. Van den Berg et al. Single-shot error mitigation by coherent Pauli checks, PRR 2023

What are Coherent Pauli Checks these useful for?

Error detection in Clifford-dominated circuits

Works well when the circuit has large Clifford or near-Clifford chunks .

Universal

Any circuit can be written as a Clifford applied to magic state inputs[1].

Many quantum advantage proposals

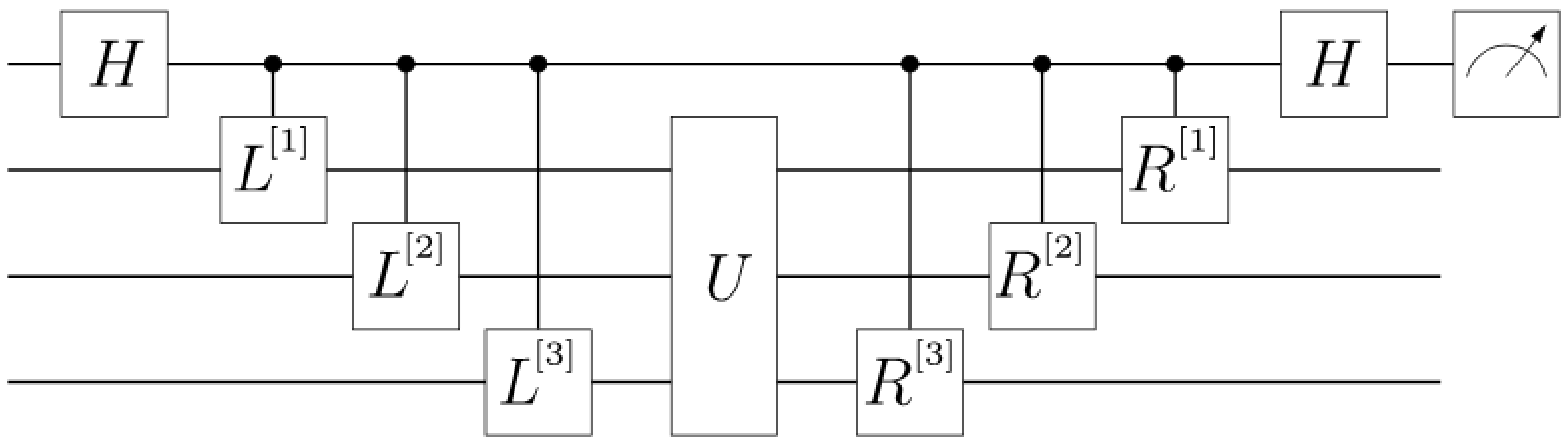
Conjugated Clifford Circuits (CCC) sampling[2].

Extended Clifford circuit sampling[3].

Graph state:
- Hardness of sampling[4].
- Resource for measurement-based quantum computing[5].

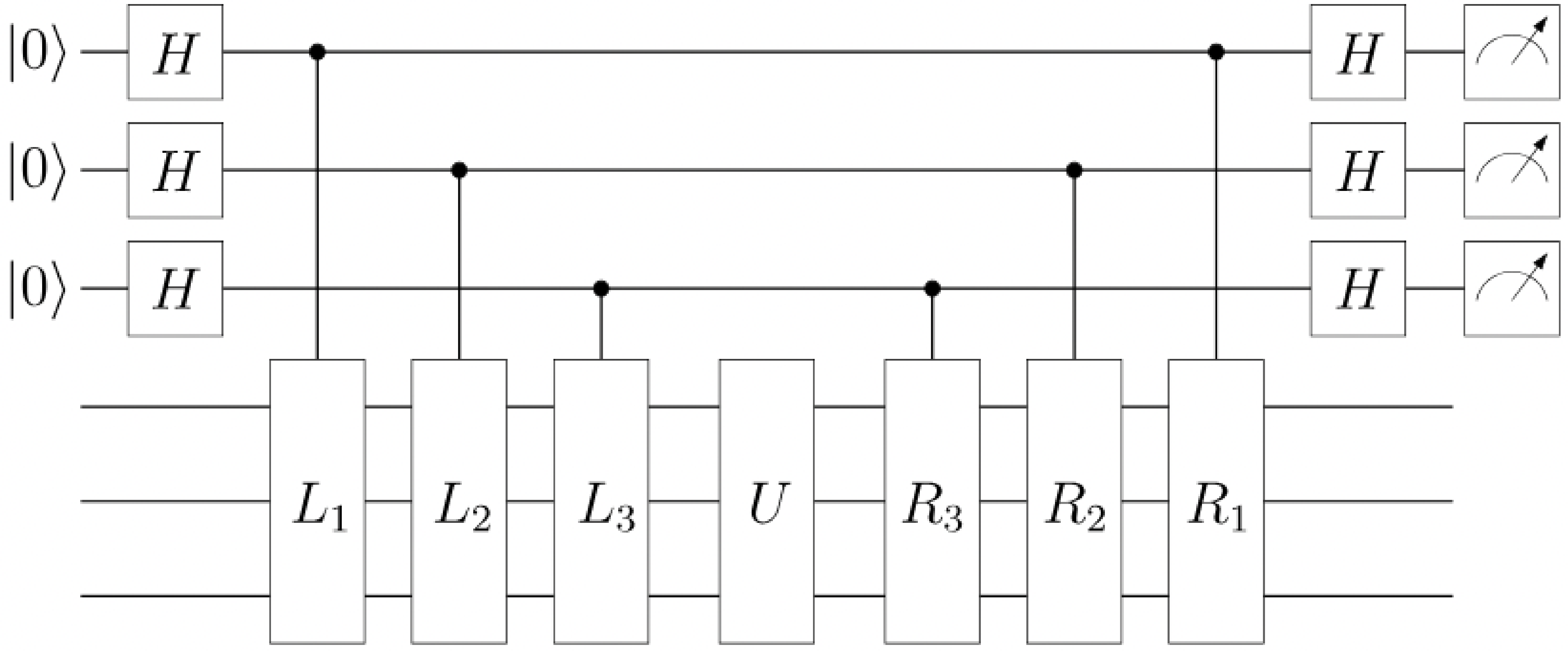
1. Gottesman and Chuang, Quantum teleportation is a universal computational primitive, Nature 1999
2. Bouland, Fitzsimons, Koh, Complexity classification of conjugated Clifford circuits, arXiv:1709.01805
3. Jozsa, Van den Nest, Classical simulation complexity of extended Clifford circuits, arXiv:1305.6190
4. Ghosh et al. Complexity phase transitions generated by entanglement, PRL 2023
5. Van den Nest et al. Universal Resources for Measurement-Based Quantum Computation, PRL 2006

Coherent Pauli checks



Entangle extra "**check**" qubits with a Clifford payload.

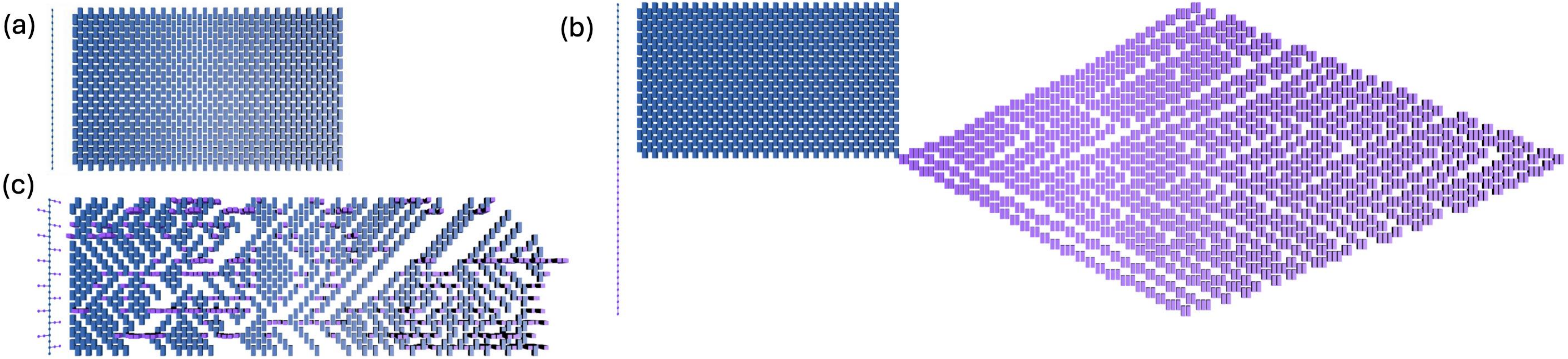
Measuring the check qubit reveals whether the payload correctly maps the **L**eft Pauli to the **R**ight Pauli.

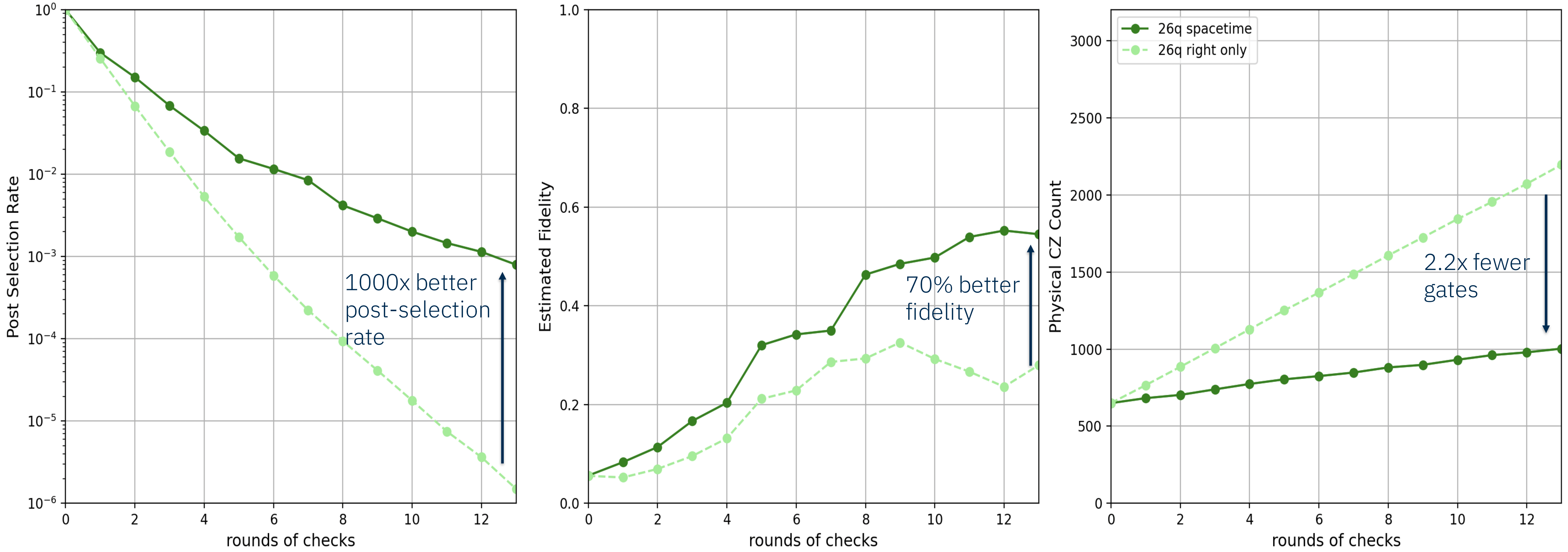


More checks detect more errors. But check overhead can introduce new errors.

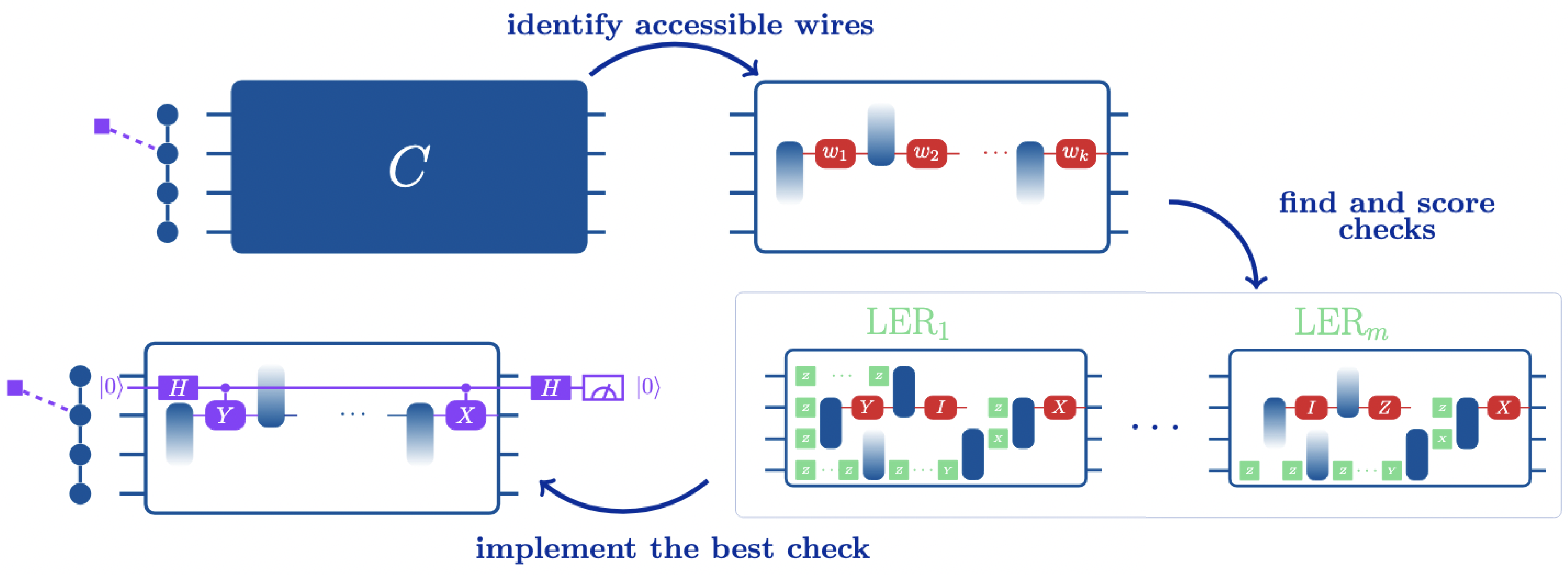
**It is critical to probe for errors with low overhead.
Otherwise, the check introduces more errors than it detects.**

Low-overhead error detection by spacetime codes





How spacetime checks work



Step 1

Start with a payload and consider an adjacent **check qubit** with some **accessible wires**.

Step 2

Find a **valid low-weight** Pauli supported on those wires.

Estimate its logical error rate by computing its “back-cumulant”.

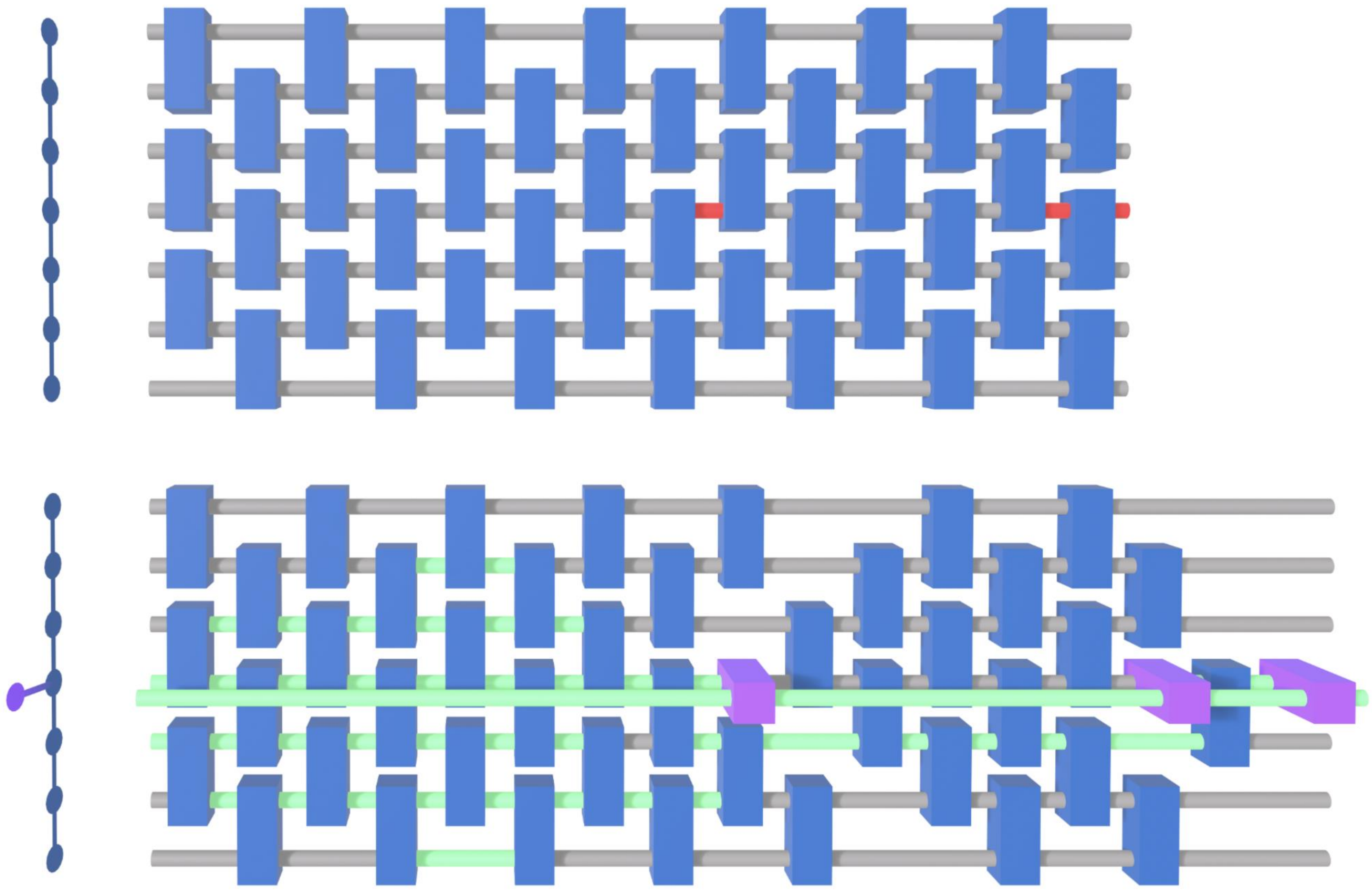
Step 3

Implement the best check using controlled-Paulis.

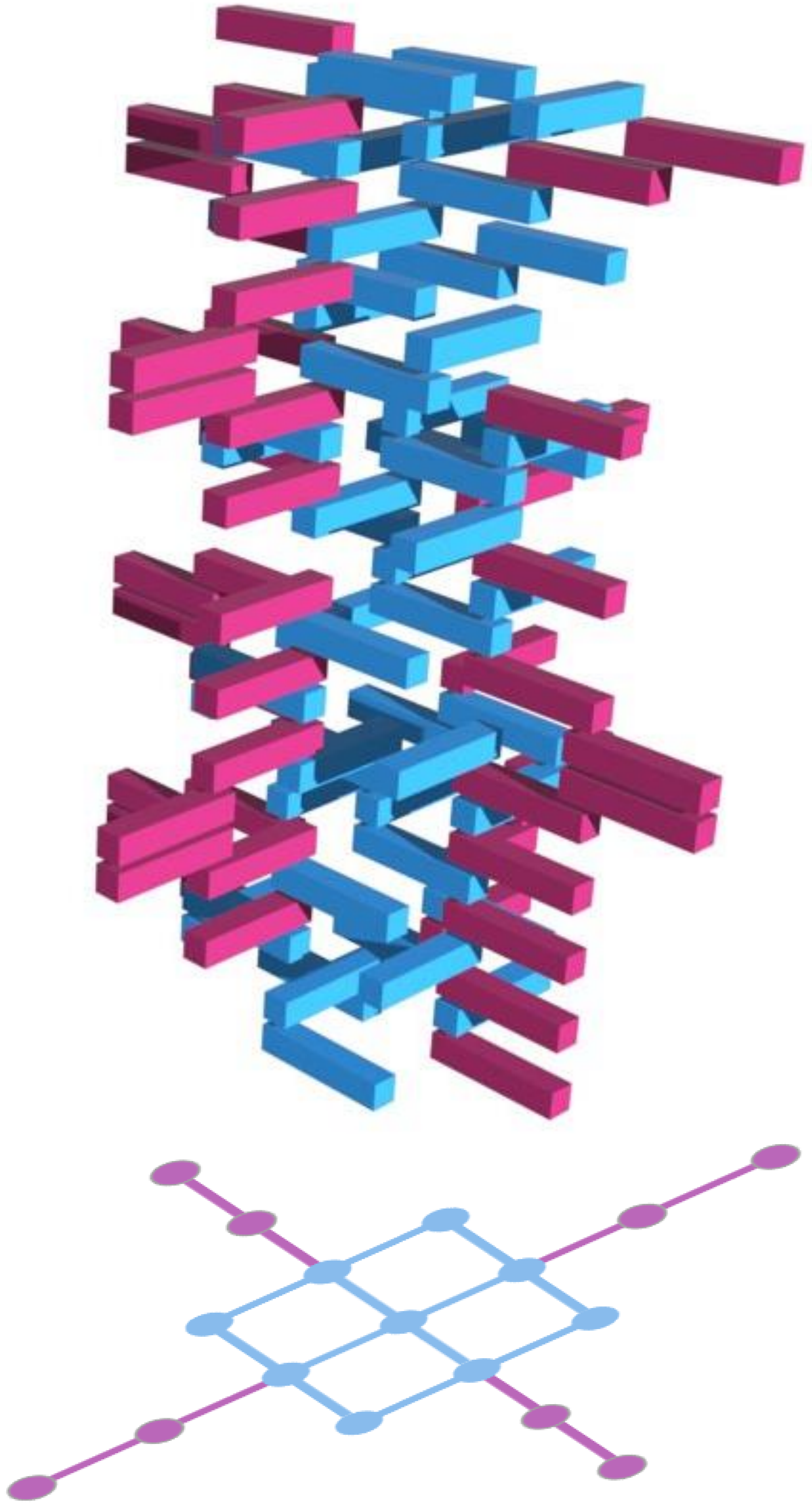
Step 4

Repeat for more checks.

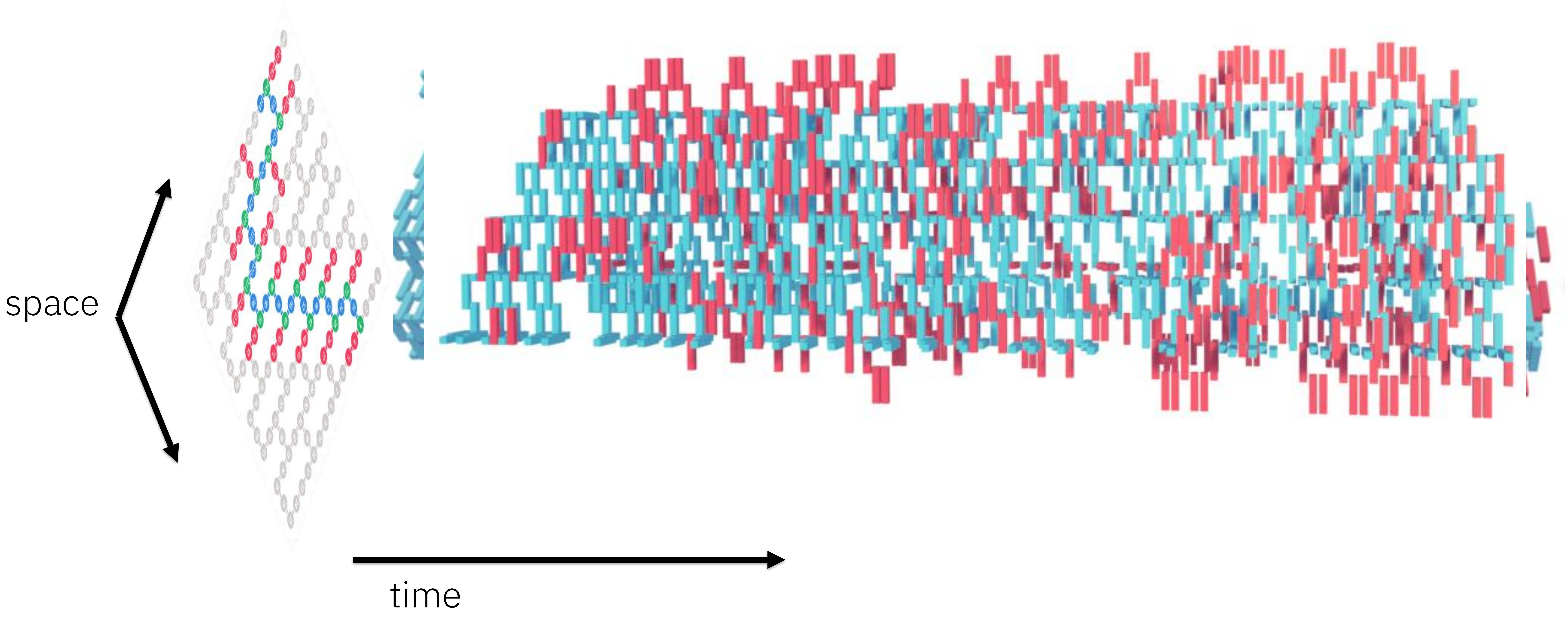
A low-overhead check can yield a large detecting region



Works for any connectivity



Checks localized in space, but distributed in time

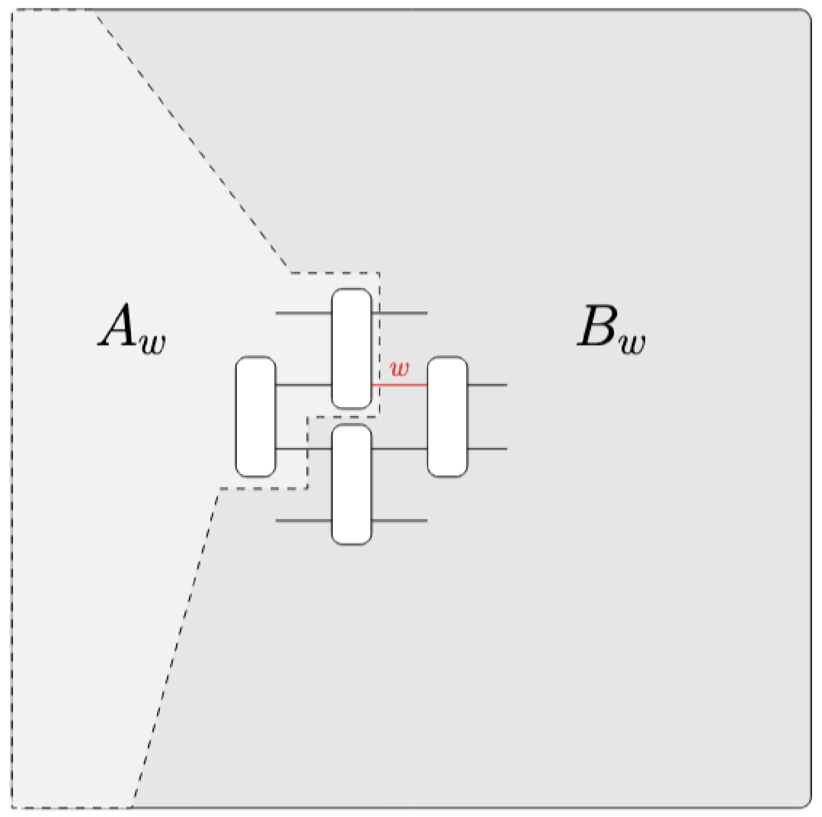


Finding low-weight checks

Back-propagator

Define $B(P,w)$: Pauli operator P on wire w and pulled to the beginning.

$$B(P,w) = A_w^\dagger P_w A_w$$



A check $\{(P_1, w_1), \ldots, (P_k, w_k)\}$ is valid if and only if $\prod B(P_i, w_i) = I$.

Syndrome decoding

We search for checks using a Boolean encoding of Pauli operators.

$$\prod_{w \in L} B(P_w, w) = I \Leftrightarrow xB = 0$$

We can restrict our search on some fixed set of wires L and use a decoding algorithm to find a low-weight x (i.e. low overhead).

$$\mathrm{B} = \begin{bmatrix} 00000100000000 \\ 00000100000010 \\ 00000000000010 \\ 00011100000100 \\ 00011000000110 \\ 00000100000010 \\ 01100100010100 \\ 01100000010110 \\ 00000100000010 \\ 00011000000110 \\ 01111000010000 \\ 01100000010110 \\ 00011000001110 \\ 01111000011000 \\ 01100000010110 \\ 00011100001100 \\ 01100100010100 \\ 01111000011000 \\ 00011101100100 \\ 00000001101000 \\ 00011100001100 \\ 00011001101100 \\ 00011000000100 \\ 00000001101000 \end{bmatrix}$$

Scoring checks

Back-cumulant

Define $\overleftarrow{B}(P, w)$: Pauli operator P on wire w and pulled to the beginning, leaving a Pauli trace on all wires.

For a check C we have

$$\overleftarrow{B}(C) = \prod_i \overleftarrow{B}(P_i, w_i)$$

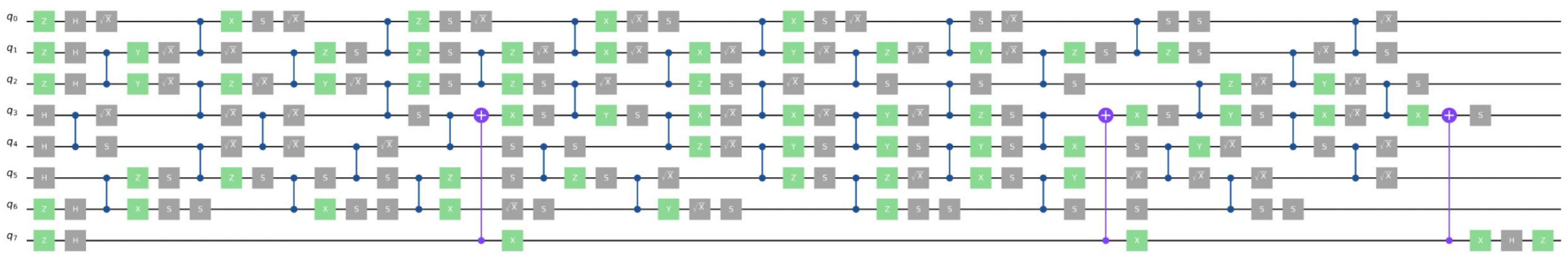
An error P on wire w is detected if and only if $[\overleftarrow{B}(C), P_w] \neq 0$.

Monte-Carlo estimation of logical error

Draw physical errors on the wires according to a noise model $P(E)$ for various error weights k and estimate the logical error of the checked payload.

$$LER(C_1, \ldots, C_l) = \sum_{k=1}^{\infty} \sum_{\substack{E=(P_1,w_1)\ldots(P_k,w_k) \\ \text{not detected} \\ \text{logical error}}} \mathbb{P}(E)$$

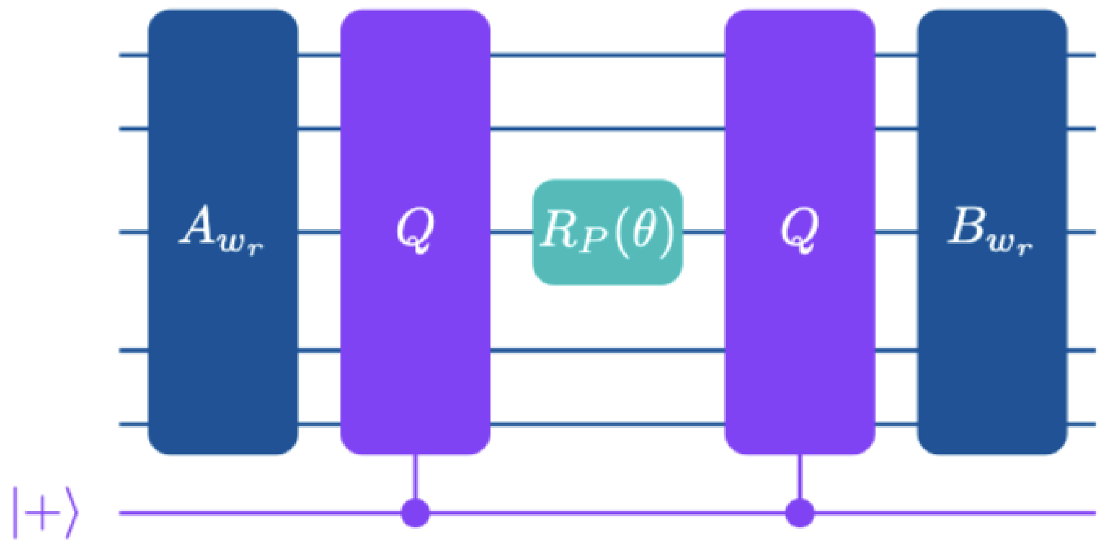
$$= \sum_{k=1}^{\infty} \sum_{\substack{E=(P_1,w_1)\ldots(P_k,w_k) \\ \text{not detected} \\ \text{logical error}}} \prod_{i=1}^{k} \mathbb{P}((P_i, w_i))$$



Beyond Clifford circuits

Valid checks in the presence of rotations

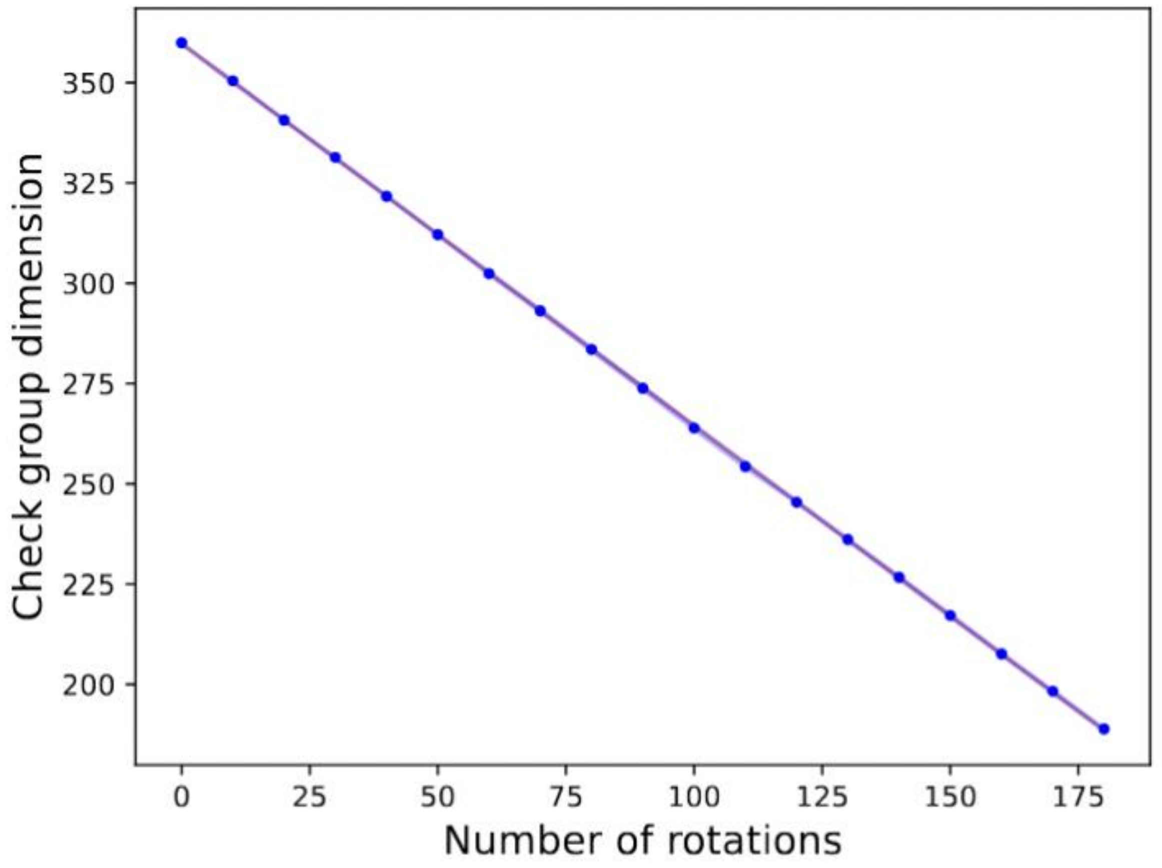
Any non-Clifford rotation R_P partitions the circuit. A check is still valid if and only if, pulled to that partition point, commutes with P.



Space of valid checks

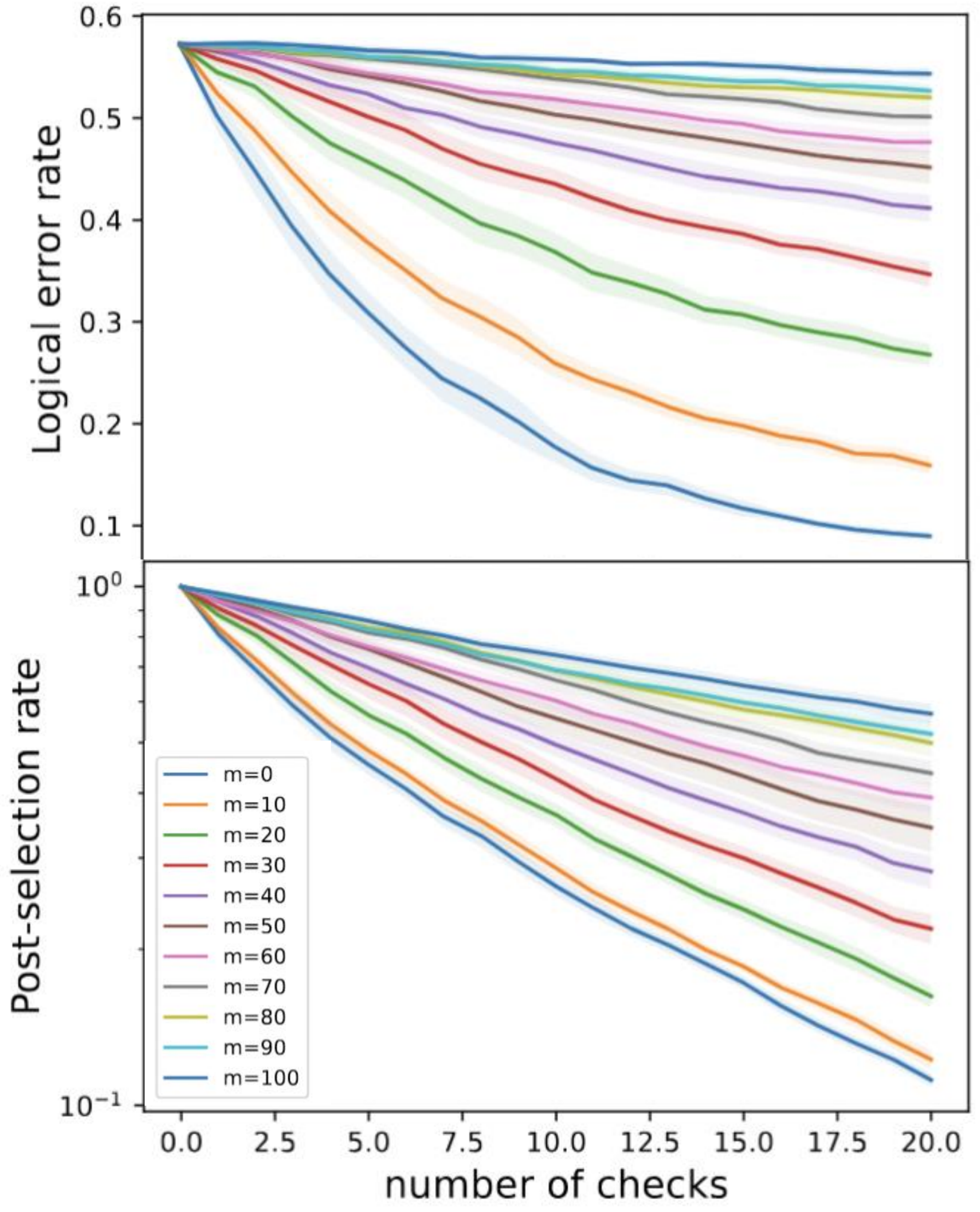
We can still find checks **efficiently** by enforcing the new constraints in our framework.

However, the space of valid checks diminishes exponentially with increasing non-Clifford rotations.



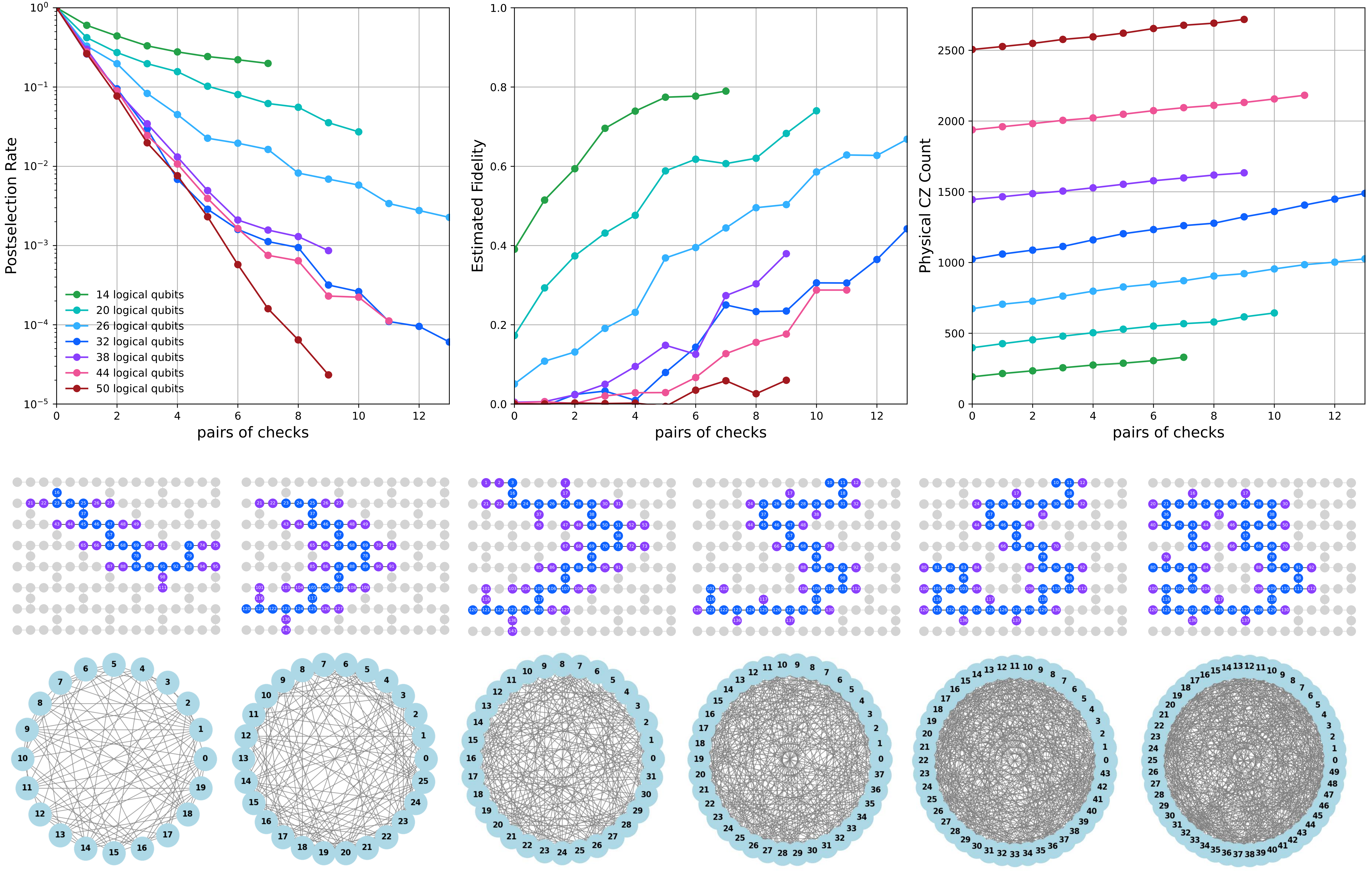
Check performance

The effectiveness of checks starts to diminish with increasing non-Cliffordness.



Experiments

IBM QPU: Heron r2 (Kingston)



Experimental results

We successfully prepare large stabilizer states using deep, **highly entangling** circuits.

Relatively **small sample overhead** and relatively **mild qubit overhead** is enough to boost the signal substantially.

Bare circuit				This work			Overhead estimates	
Qubits	**Gates**	**Ent. width**	**LF**	**Qubit overhead**	**Sampling overhead**	**Fidelity gain**	**QEC qubits**	**PEC samples**
14	182	[4, 5]	0.963	14	5.0	2×	700	1.7×10^1
20	380	[5, 6]	0.937	20	3.7×10^1	4×	1000	3.4×10^2
26	650	[7, 9]	0.907	26	4.4×10^2	13×	2548	3.0×10^4
32	992	[9, 11]	0.861	26	1.7×10^4	109×	5184	1.4×10^8
38	1406	[10, 13]	0.884	18	1.2×10^3	83×	3724	4.8×10^7
44	1892	[9, 15]	0.875	22	8.9×10^3	104×	4312	1.2×10^8
50	2450	[8, 17]	0.870	18	4.3×10^4	235×	2500	3.1×10^9

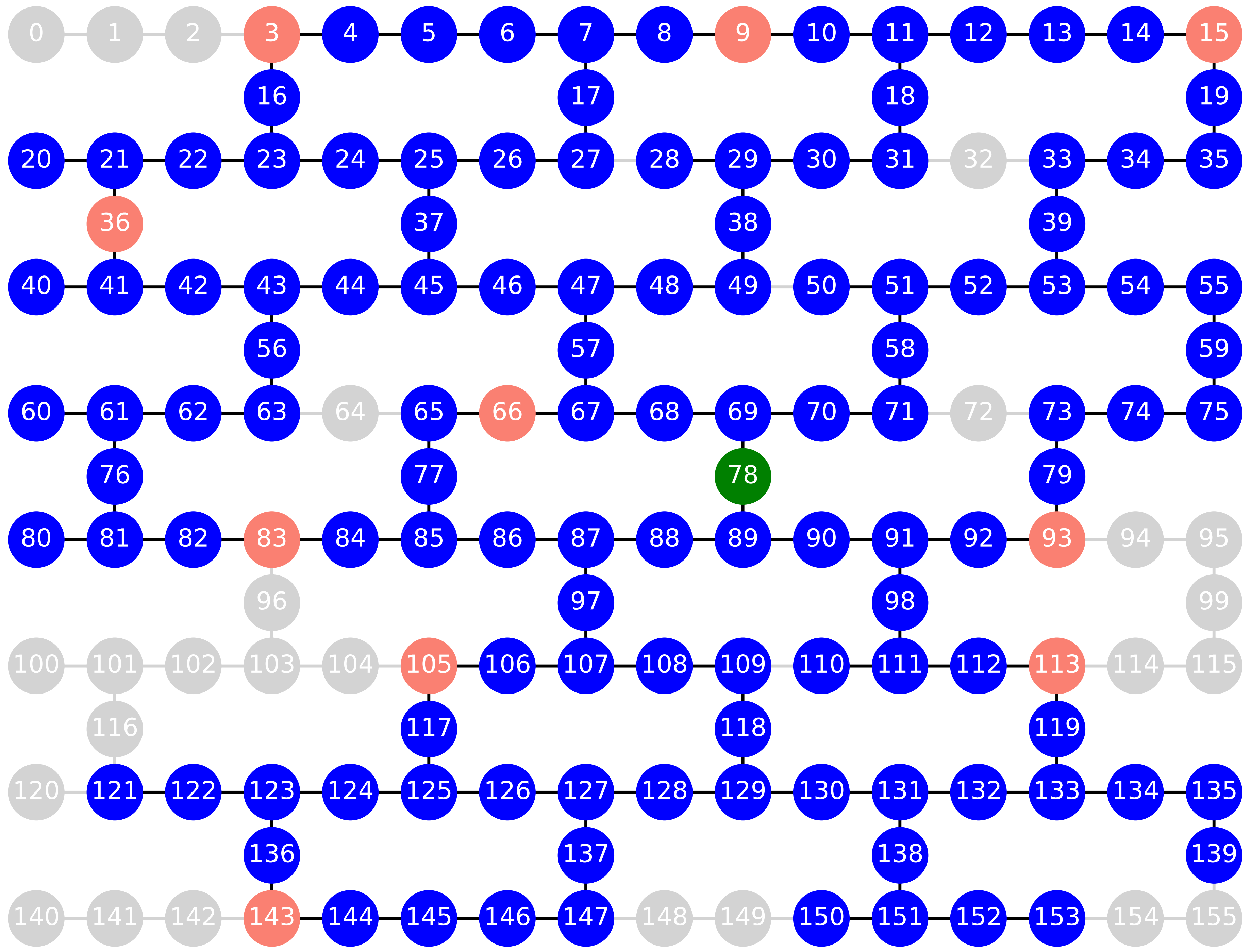
Application: preparation of GHZ states

GHZ states are a common benchmark for the quality of quantum hardware and software.

They can be prepared in short depth by breadth-first search from a root of a tree.

Coherent Pauli checks on GHZ states are particularly simple: the state is permutation-invariant and it has low-weight ZZ stabilizers. Measuring ZZ parity of any qubit pair yields a valid check[1,2].

But not all checks are the same in terms of coverage. We use the tools we developed to optimize the coverage of these checks.



Green: GHZ root, Blue: GHZ qubits, Pink: checks

1. Mooney et al. Journal of physics communications 2021
2. Liao et al, Achieving computational gains with quantum error correction primitives.., arXiv:2411.14638

Recipe for error-detected GHZ states

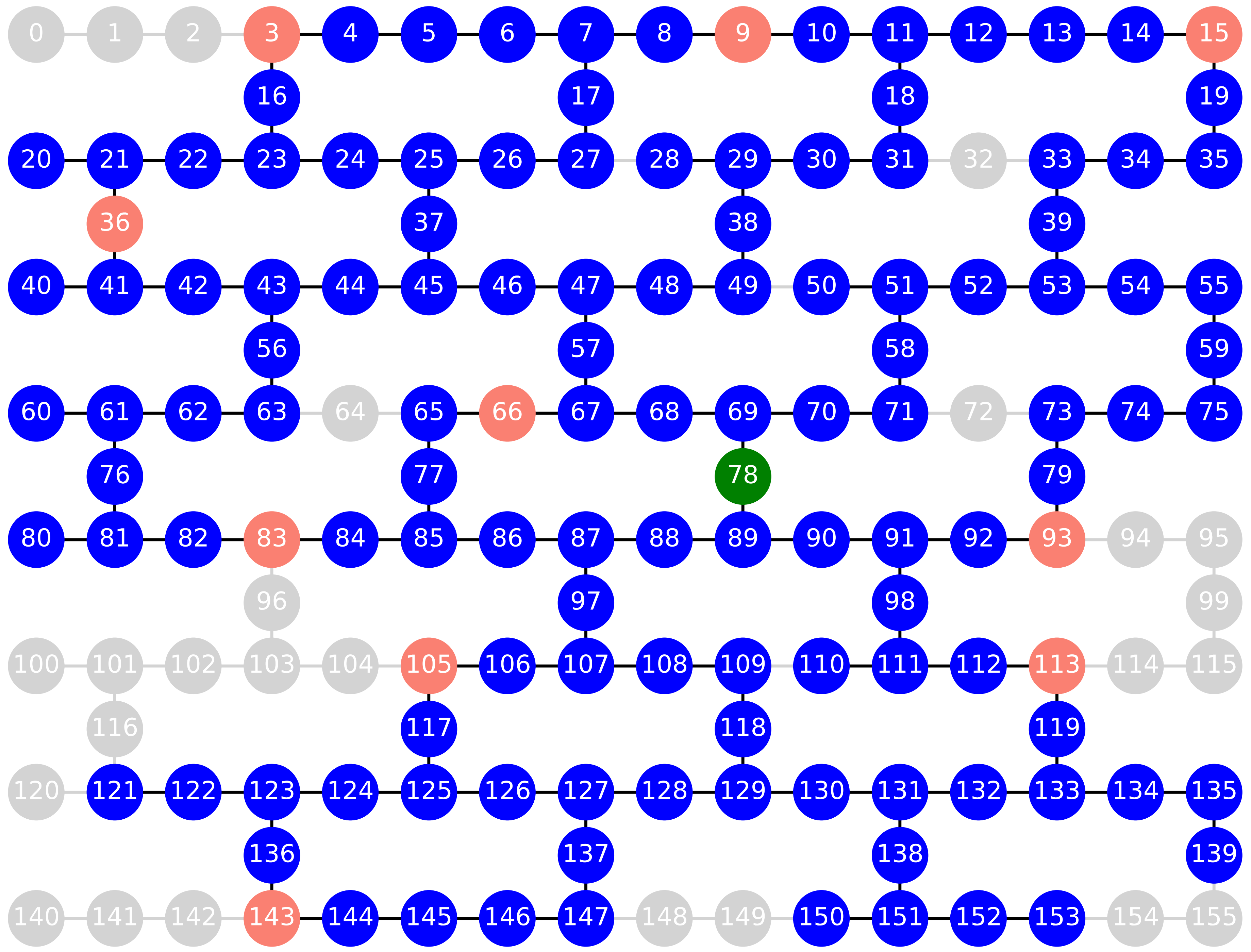
1- Start by removing bad nodes and edges from the graph (errors above a given threshold).

2- For each remaining qubit, build a BFS tree starting at that qubit as root. Choose the root that yields the shortest depth GHZ.

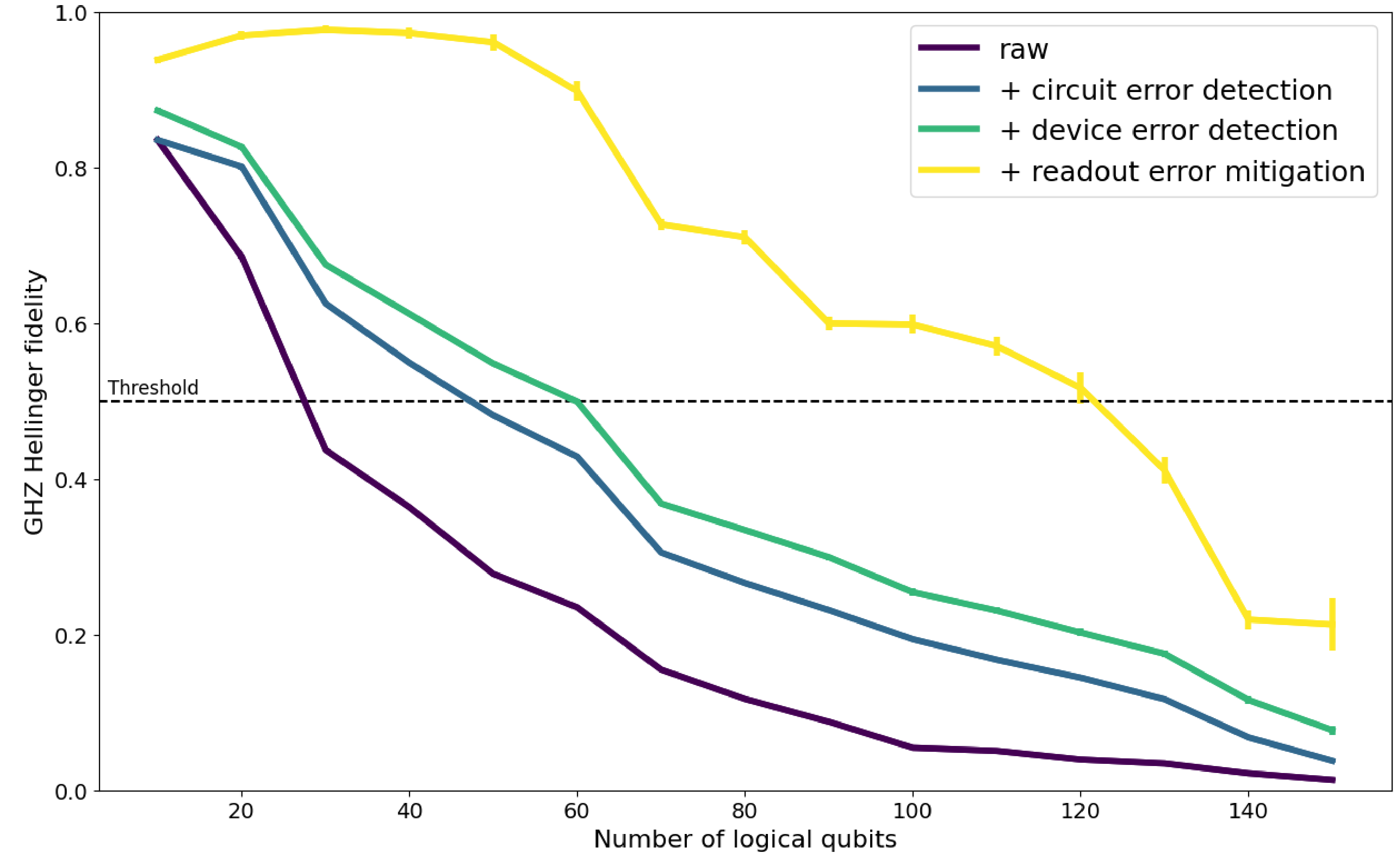
3- Starting at the chosen root, randomly block a small number of remaining nodes and build a BFS tree again.

4- Now find some valid checks and compute their coverage.

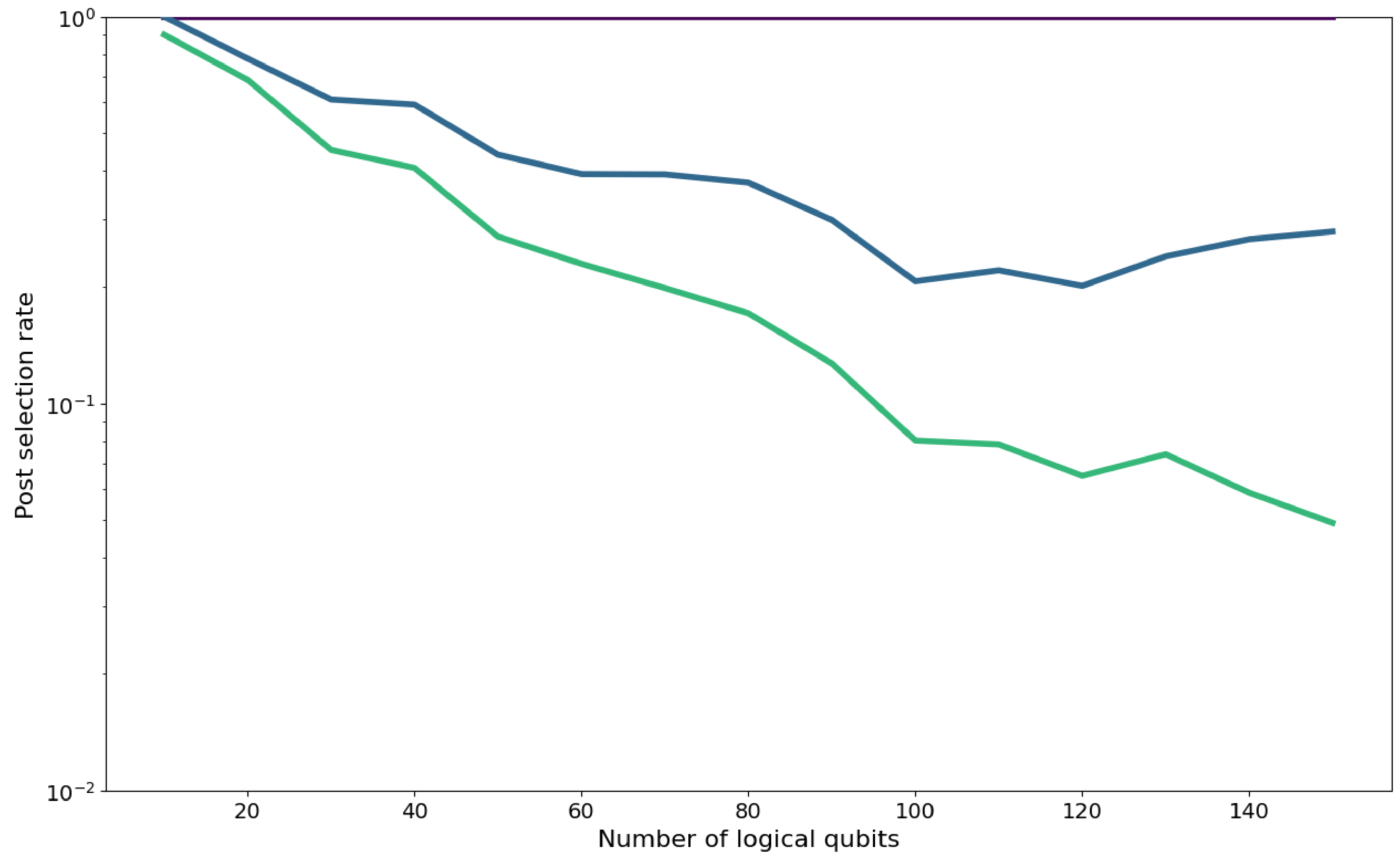
5- Repeat this randomization until finding the best coverage.



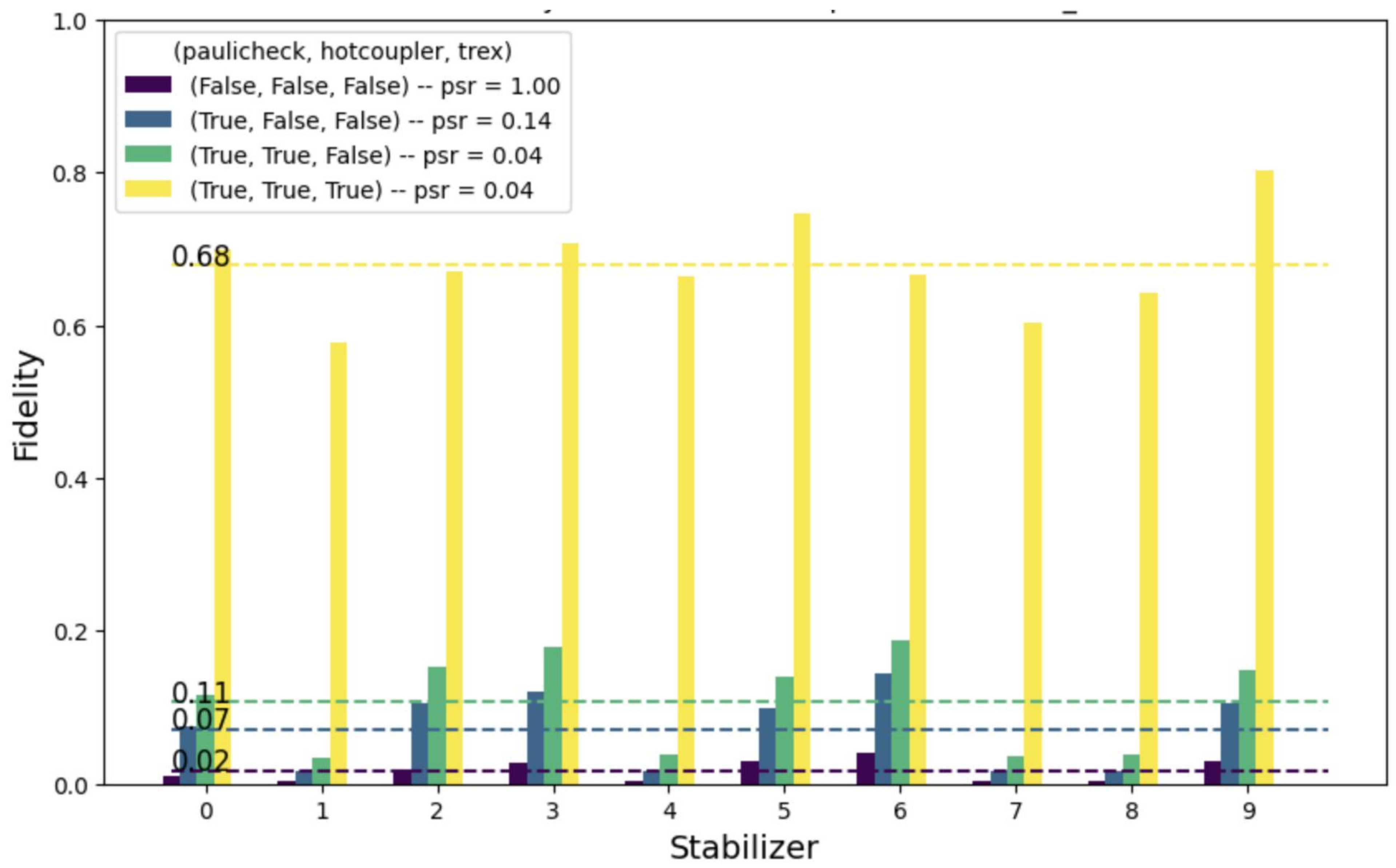
GHZ states beyond 100 qubits



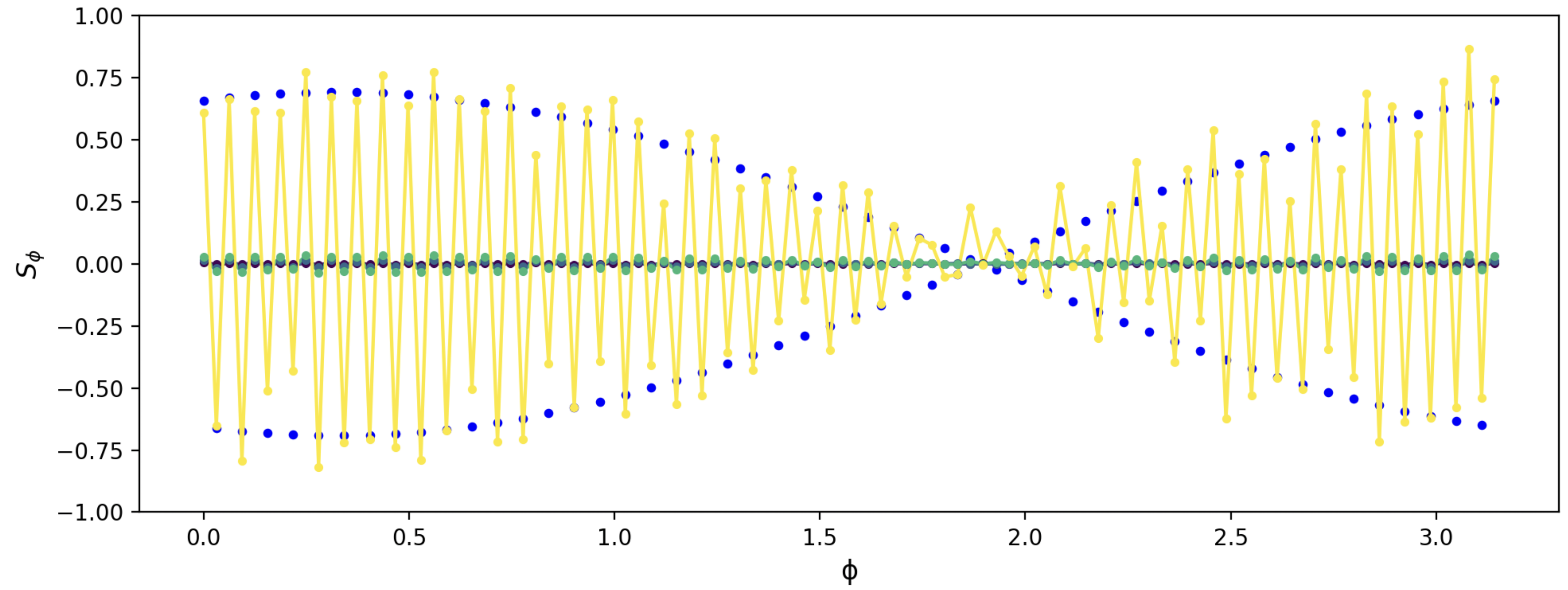
Hellinger fidelity



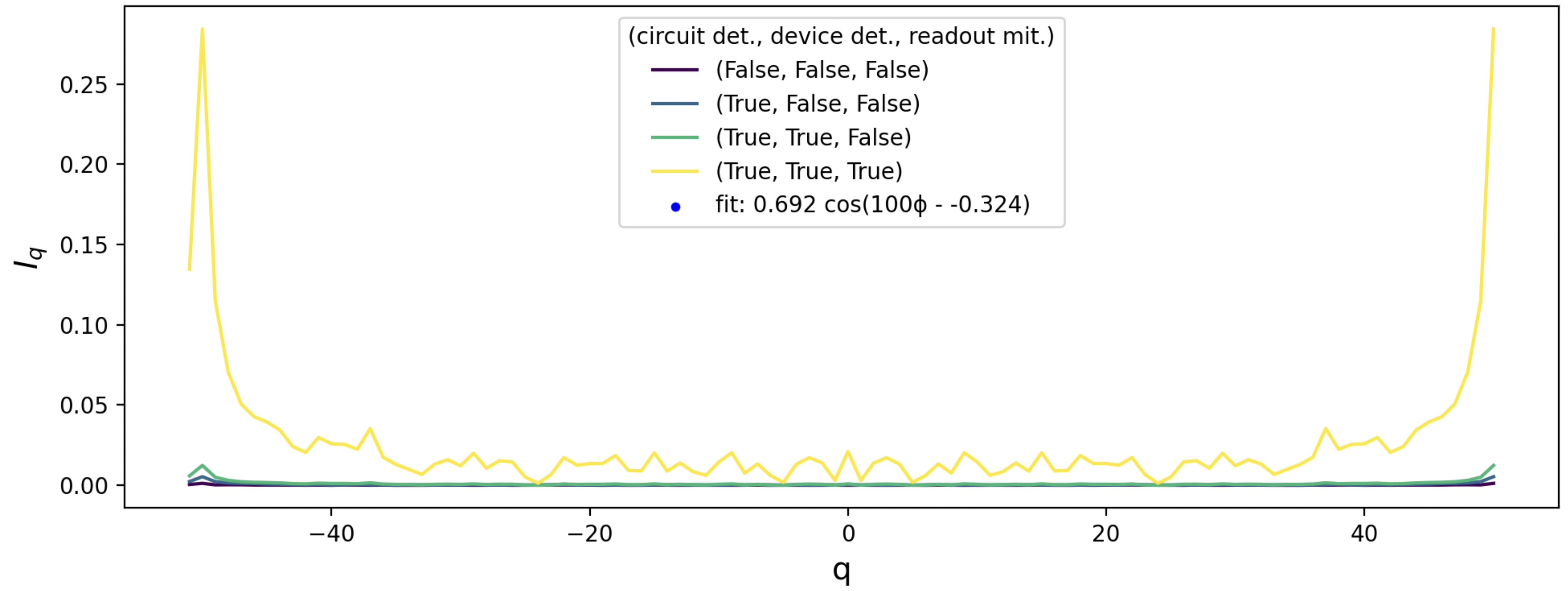
Post-selection rate



Direct fidelity estimation



Parity oscillations



Conclusion

Error detection in Clifford-dominated circuits inspired by spacetime codes.

We develop tools that allow us to find valid checks and score them.

Works for any qubit connectivity graph.

Unlocks large experiments: highly entangled graph states with 2.5k gates; or 100+ qubit GHZ states.

Open questions

Can we pick checks such that we correct errors?

Is there something to exploit in more structured circuits to design better checks?

IBM