Quantum Error Detection

Theory and Experiment

Quantum Error Detection

- Referred to by different names:
  - subspace expansion
  - symmetry verification
  - logical shadow tomography

- Little work toward experimental implementation of the technique

- Practical considerations:
  - Logical gate overheads
  - Error propagation within encoding / logical operations
  - Hardware noise / connectivity

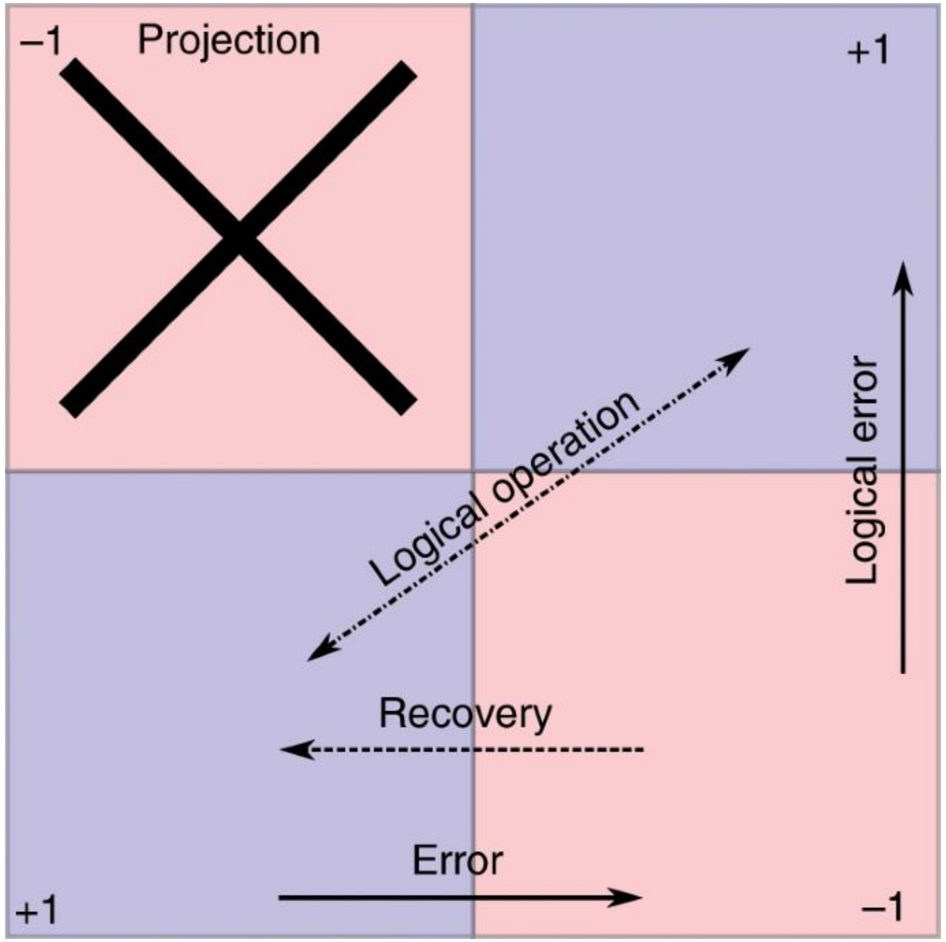


Image: *Decoding quantum errors with subspace expansions (2020)*

Theory

Given observable O, quantum state ρ, and stabilizer code defined by

$$\mathcal{S}=\langle S_1, S_2, \ldots, S_r \rangle$$

we define the error-mitigated expectation value as

$$\langle O \rangle(n) := \frac{\mathrm{Tr}\left[\Pi \bar{\rho} \Pi^\dagger \bar{O}\right]}{\mathrm{Tr}\left[\Pi \bar{\rho} \Pi^\dagger\right]} \approx \mathrm{Tr}[\rho O]$$

where $\bar{\rho}$, $\bar{O}$ are encoded, and

$$\Pi := \prod_{i=1}^{r} \frac{I + S_i}{2} = \frac{1}{2^r} \sum_{S \in \mathcal{S}} S$$

Steps:

1. Given stabilizer code $\mathcal{S}$, compute codewords $|\bar{0}\rangle$ and $|\bar{1}\rangle$
2. Map circuit preparing ρ to logical circuit preparing $\bar{\rho}$
3. Measure circuit, only keeping results which are codewords

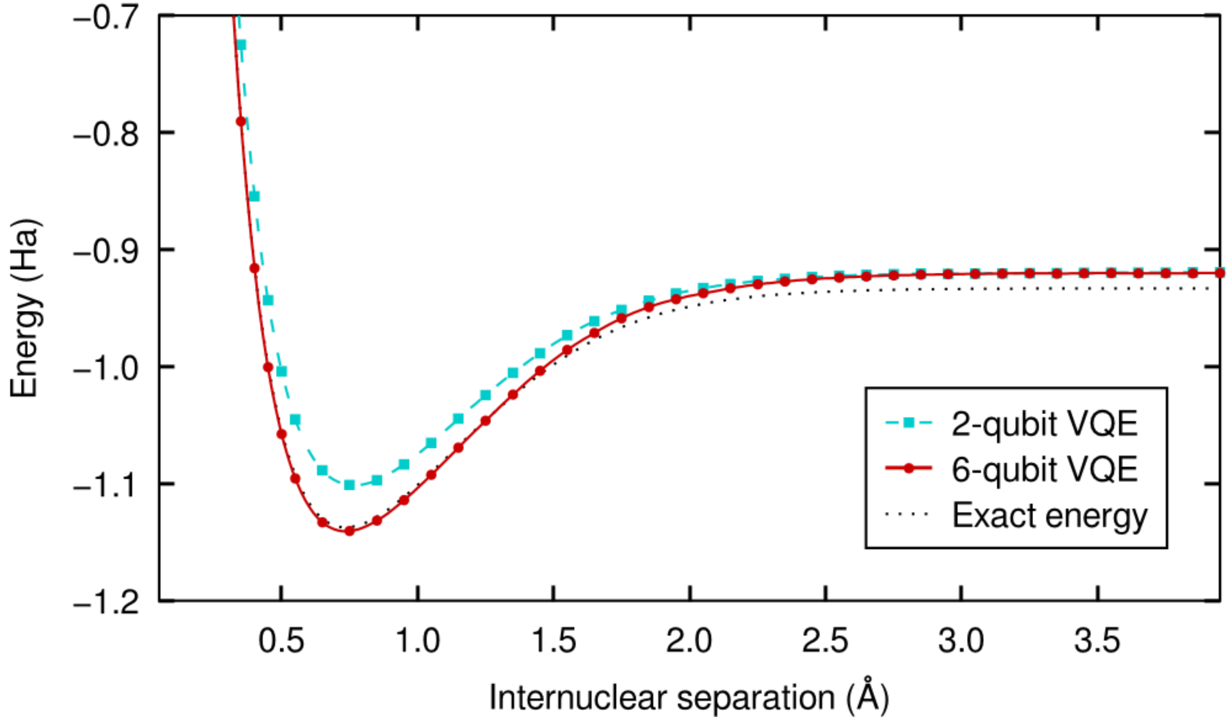
Note:

This scheme requires exponential sampling overhead – important consideration but not the focus of this work

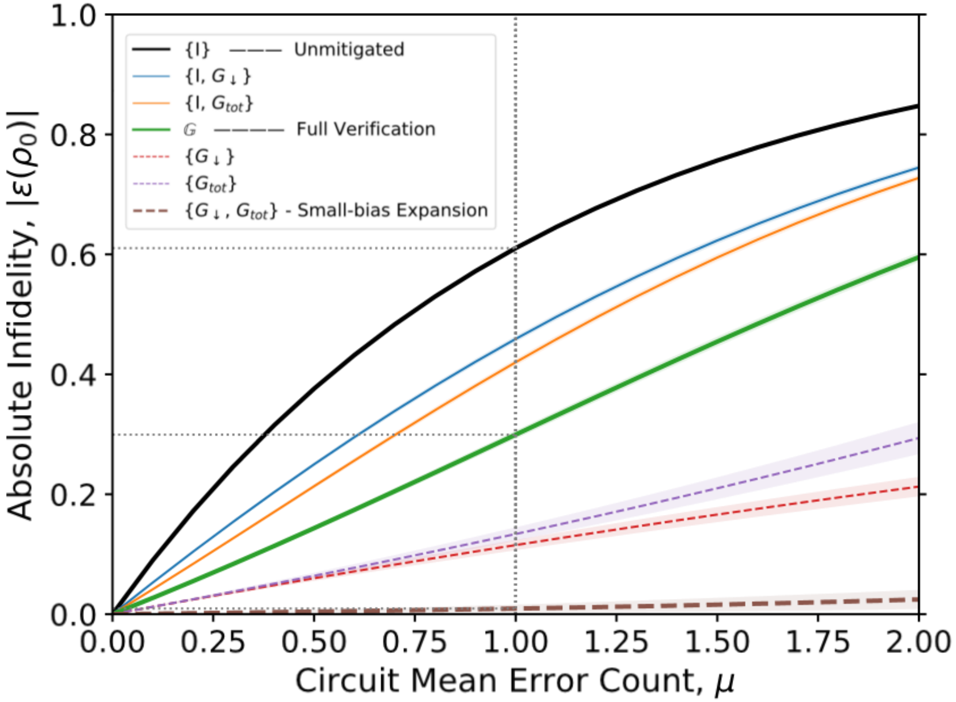
Prior Experiments

Error detection has been performed in small scale experiments and numerical simulations

VQE with H2 using IBMQ Tokyo in
Error detection on quantum computers improving the accuracy of chemical calculations (2020)



12-qubit Hubbard model under depolarizing noise in
Simulated effects in Quantum Error Mitigation using Symmetry Expansion (2021)



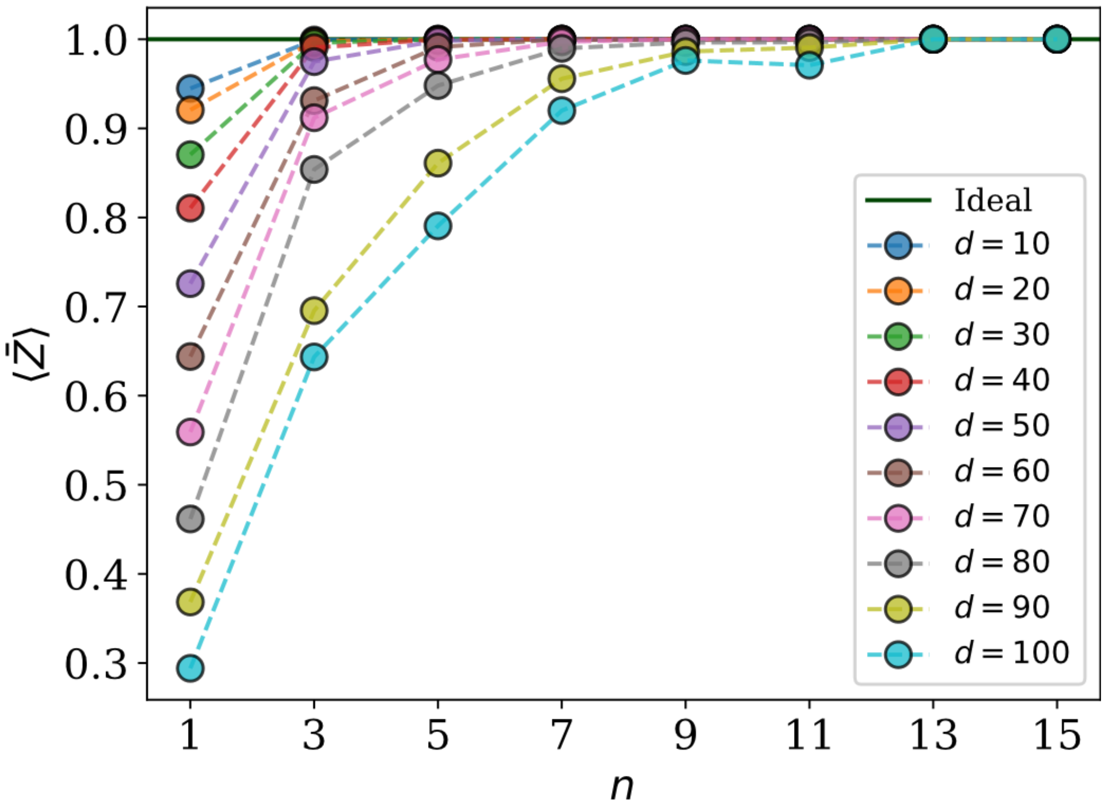
Current Work

Want to investigate a potentially useful but under-studied technique at scale on hardware

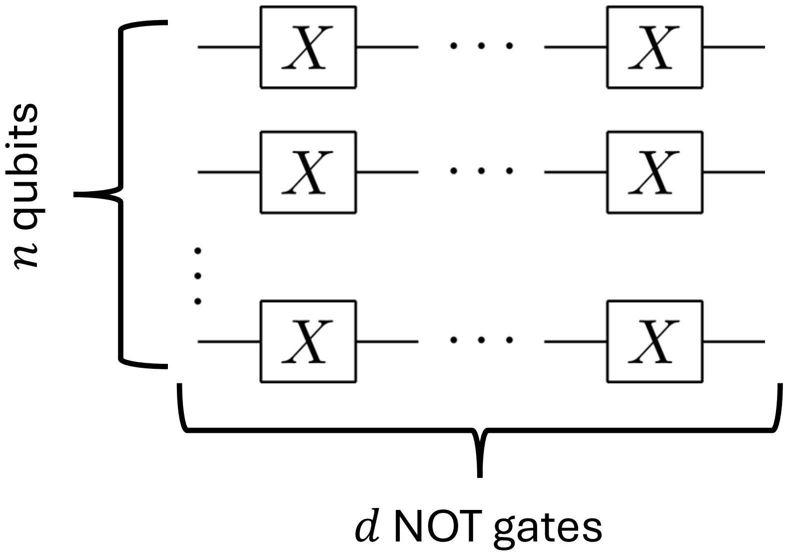
- With classical codes, method scales and performs well
- With the overhead of encoding / logical gates, error detection can perform worse than unmitigated circuits

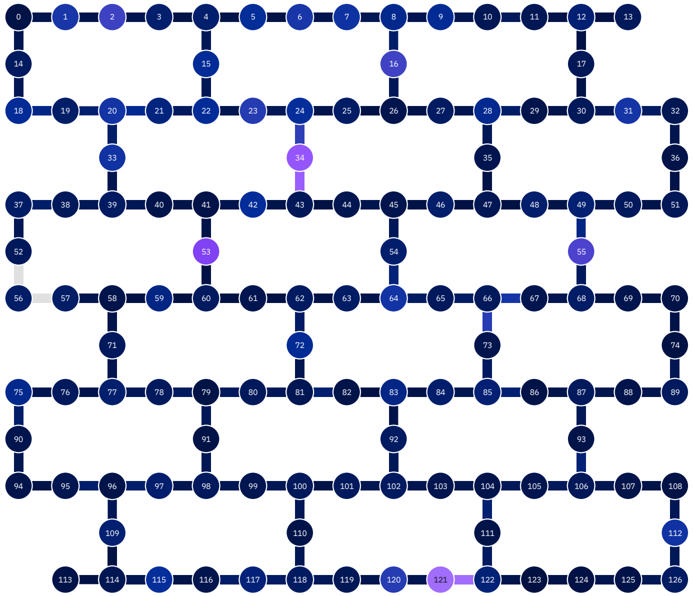
Experimental Results

Repetition Code

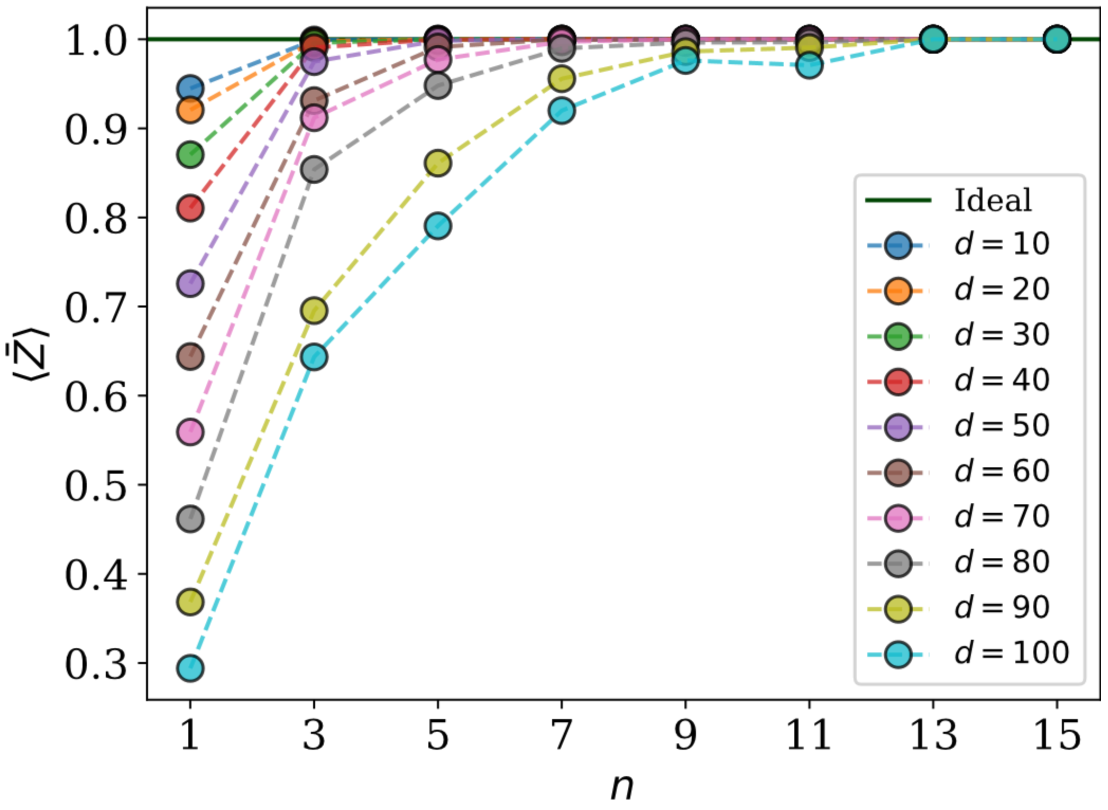


Memory experiment using repetition code on IBM Kyiv

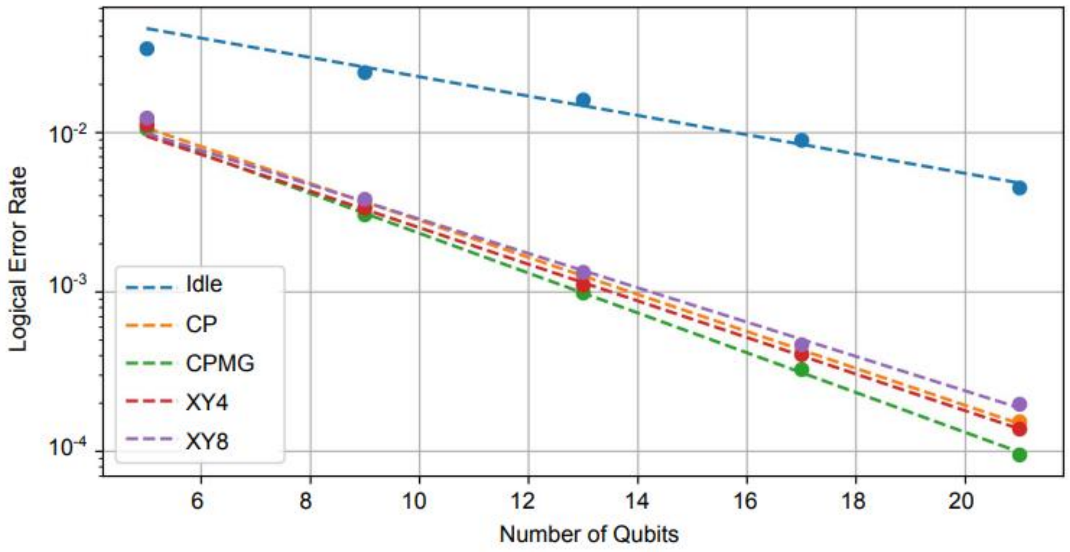




Repetition Code



Memory experiment using repetition code on IBM Kyiv

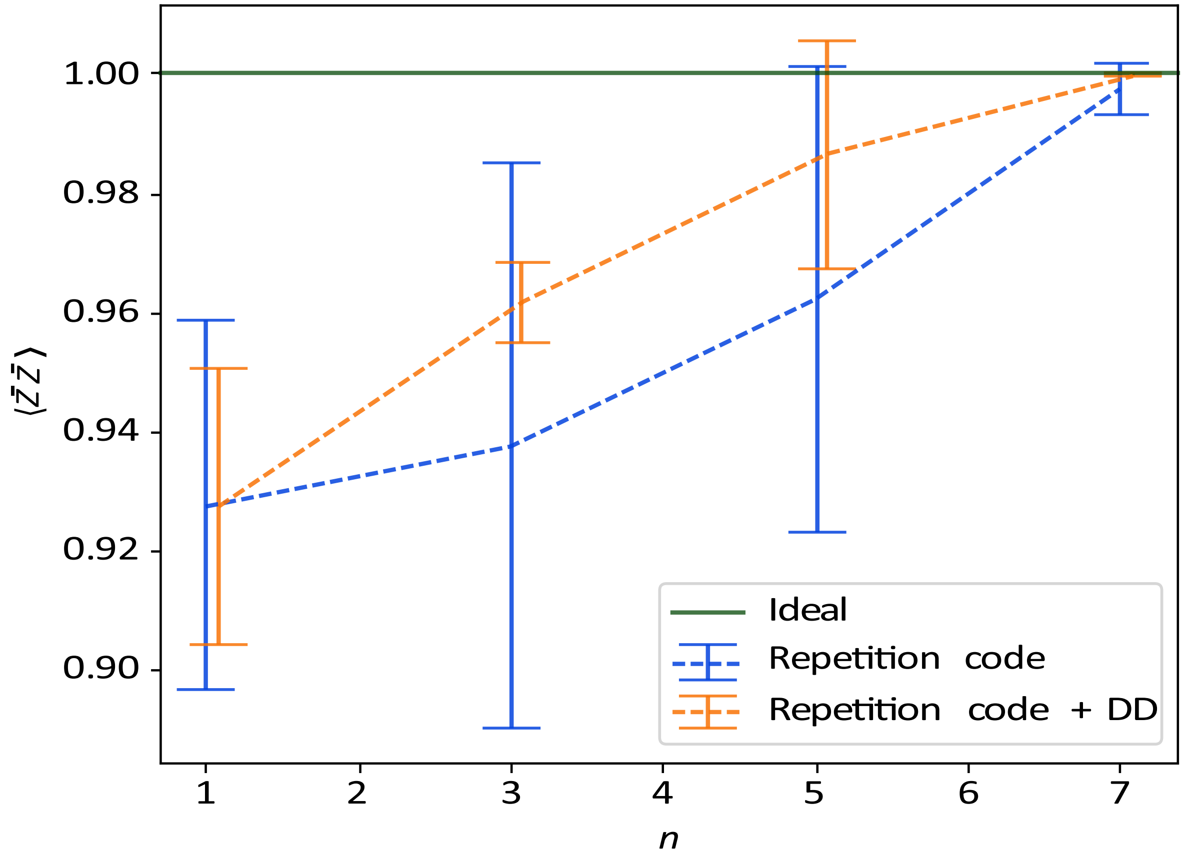


Exponential suppression of bit or phase errors with cyclic error correction (2021)

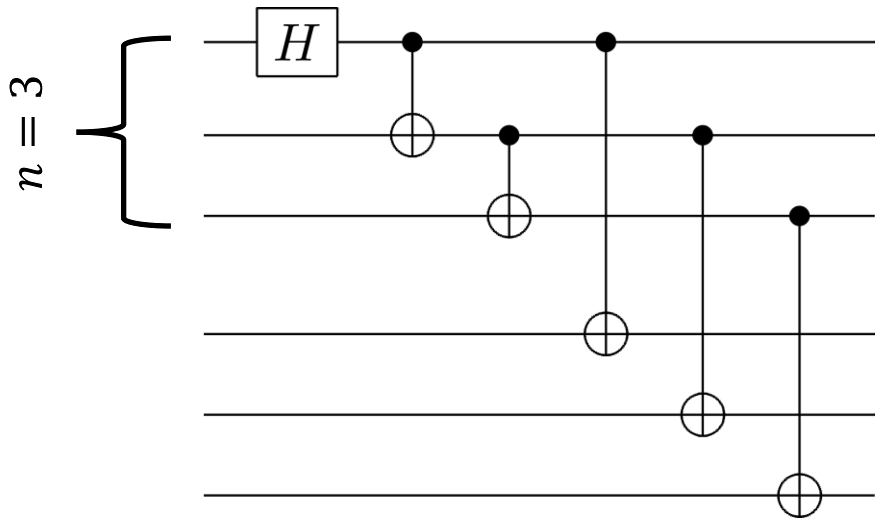
Repetition Code

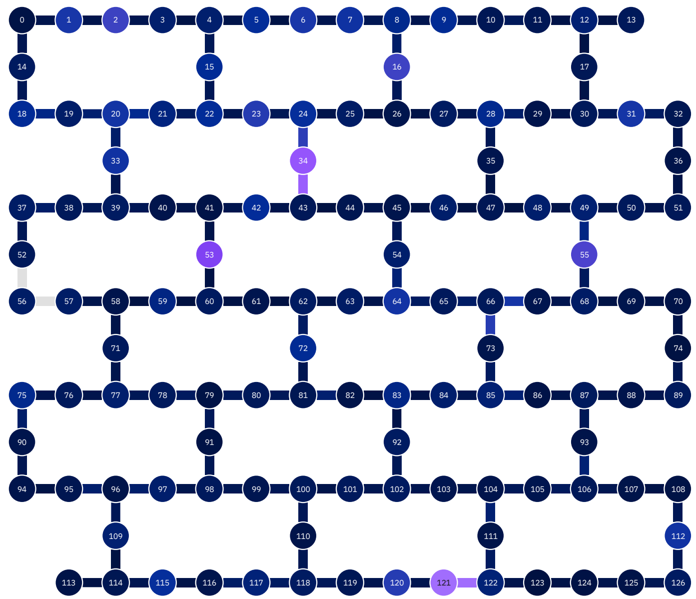
Memory experiments are helpful, but we want to perform quantum computation (logical gates) with this technique

Repetition Code

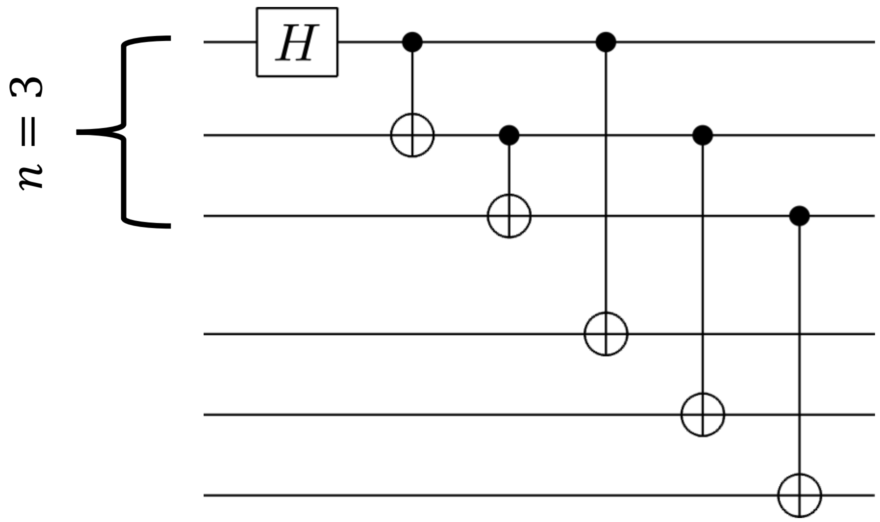


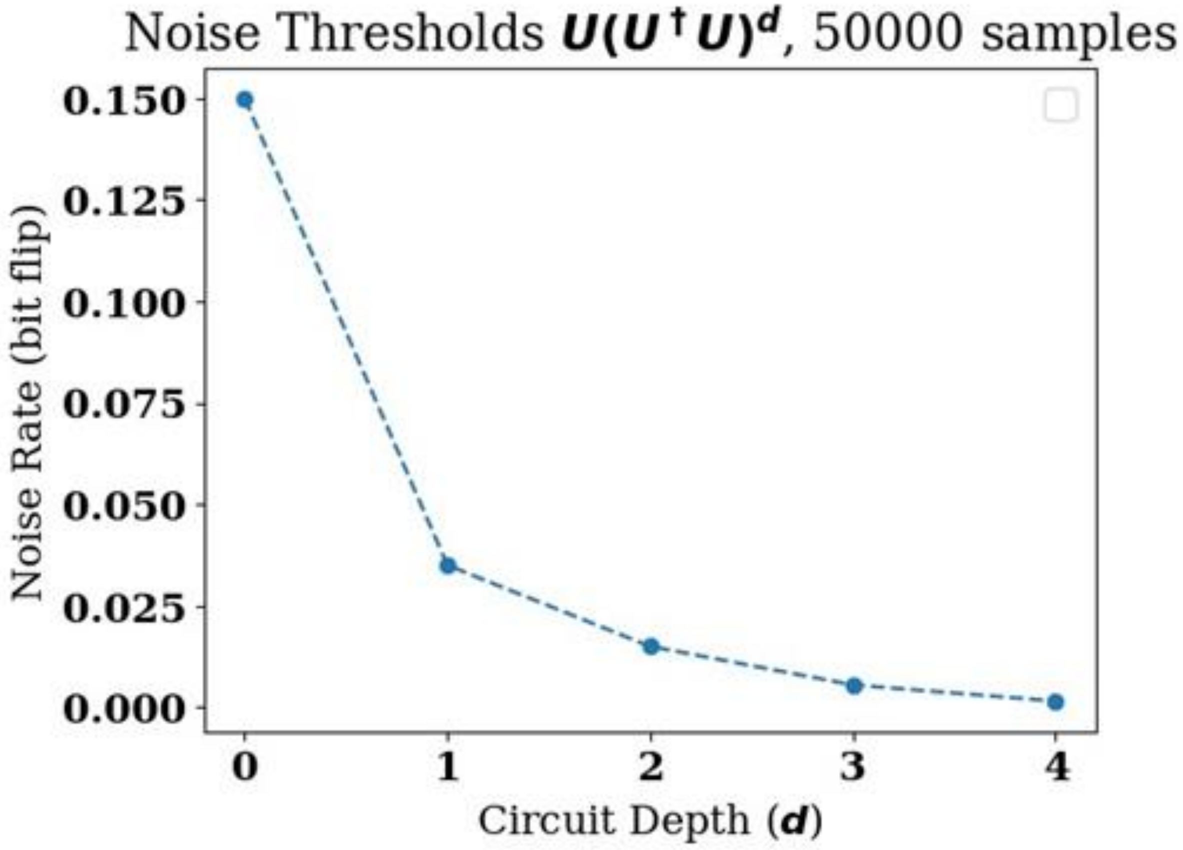
Bell state preparation using repetition code on IBM Kyiv
(Results from 20 trials)





Repetition Code





(Here "threshold" refers to the noise rate at which error detection no longer performs better than physical)

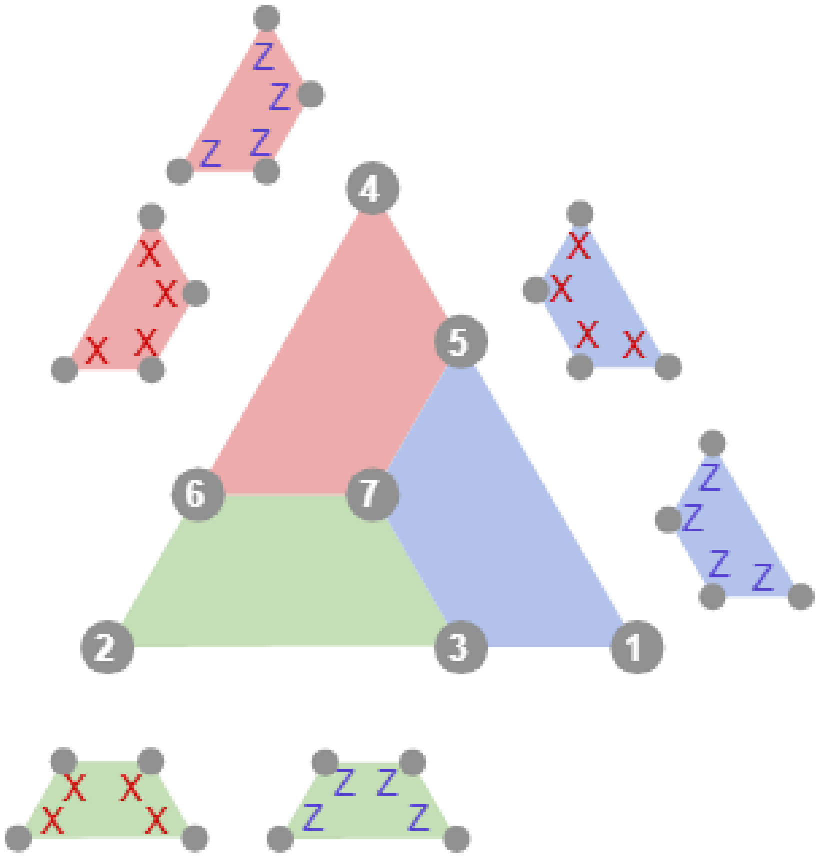
Simulated "thresholds" for Bell state preparation using repetition code

Repetition Code

- This is a classical code (no overhead for encoding / measurement)

- How does technique perform on a quantum code?

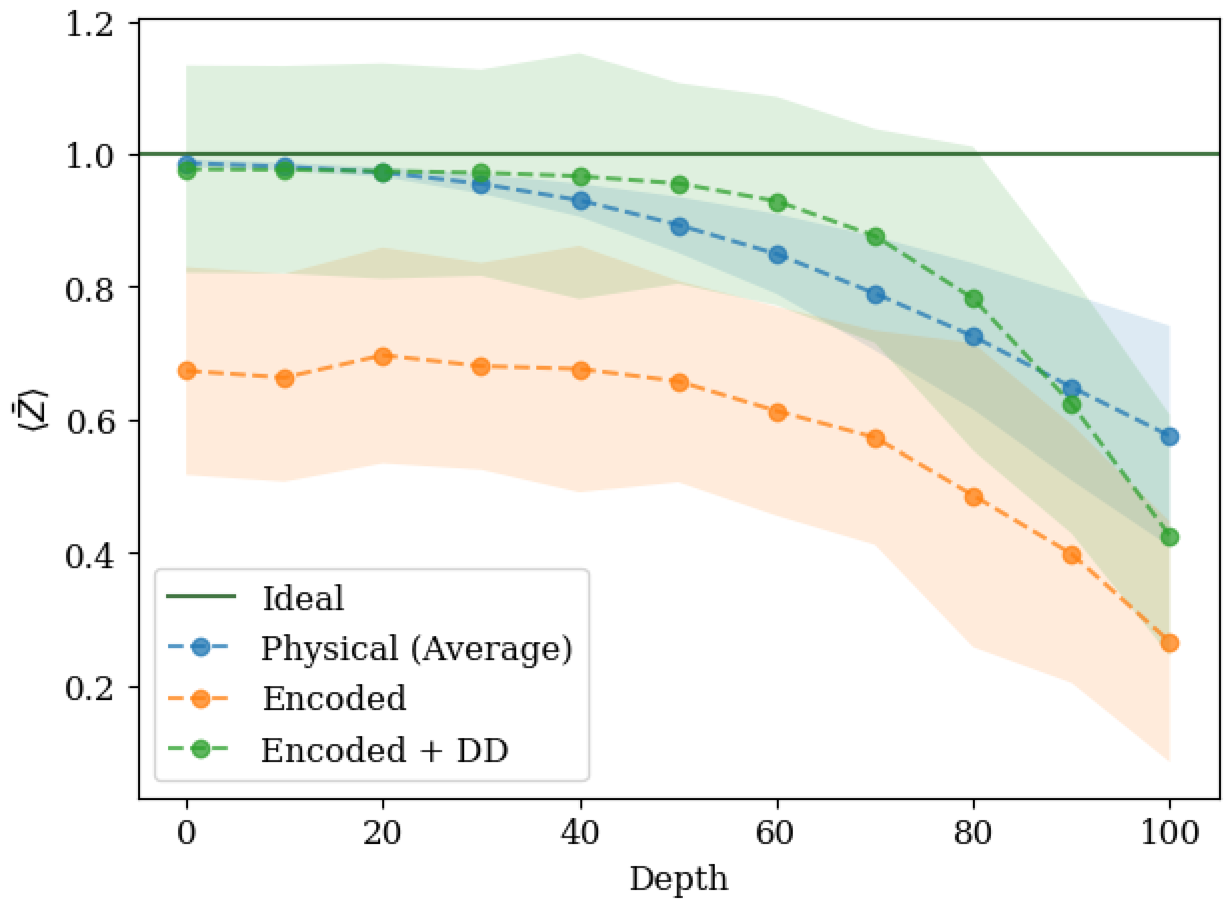
Triangular Color Code



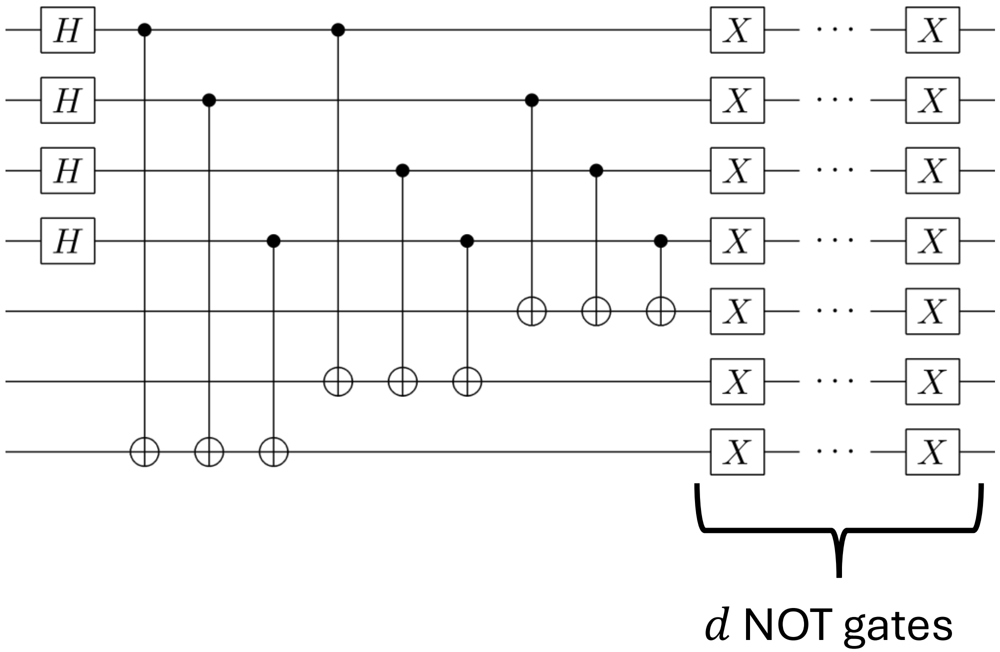
- Family of stabilizer codes where distance can be scaled
- Convenient for our purposes:
  - Transversal H and CNOT gates
  - Symmetric X- and Z-type stabilizers

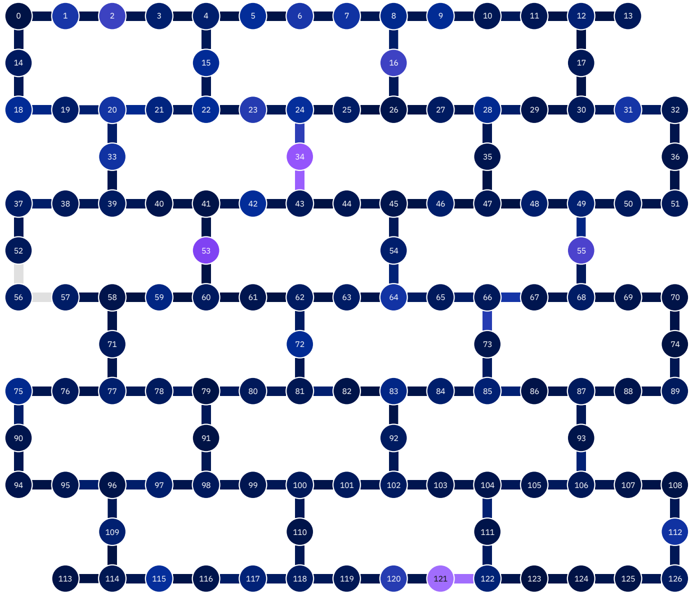
Image: Error Correction Zoo

Triangular Color Code

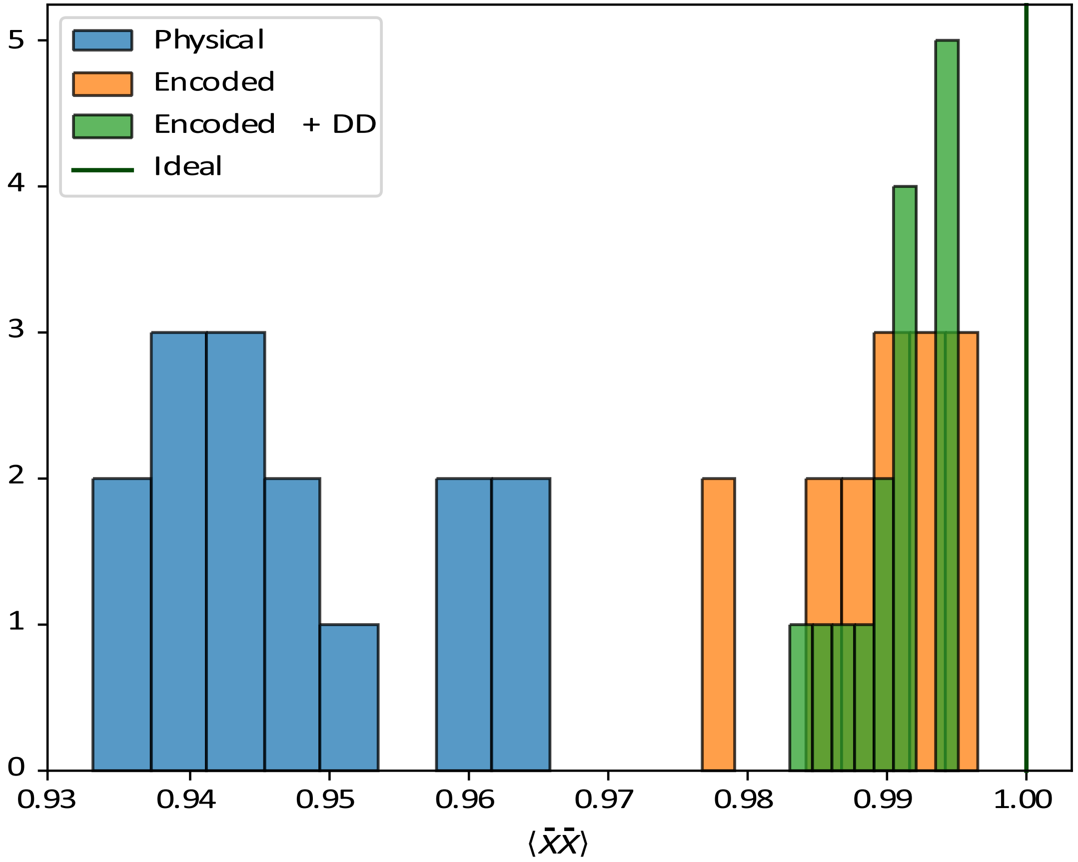


Memory experiment using Steane code on IBM Kyiv
(Results from 10 trials)

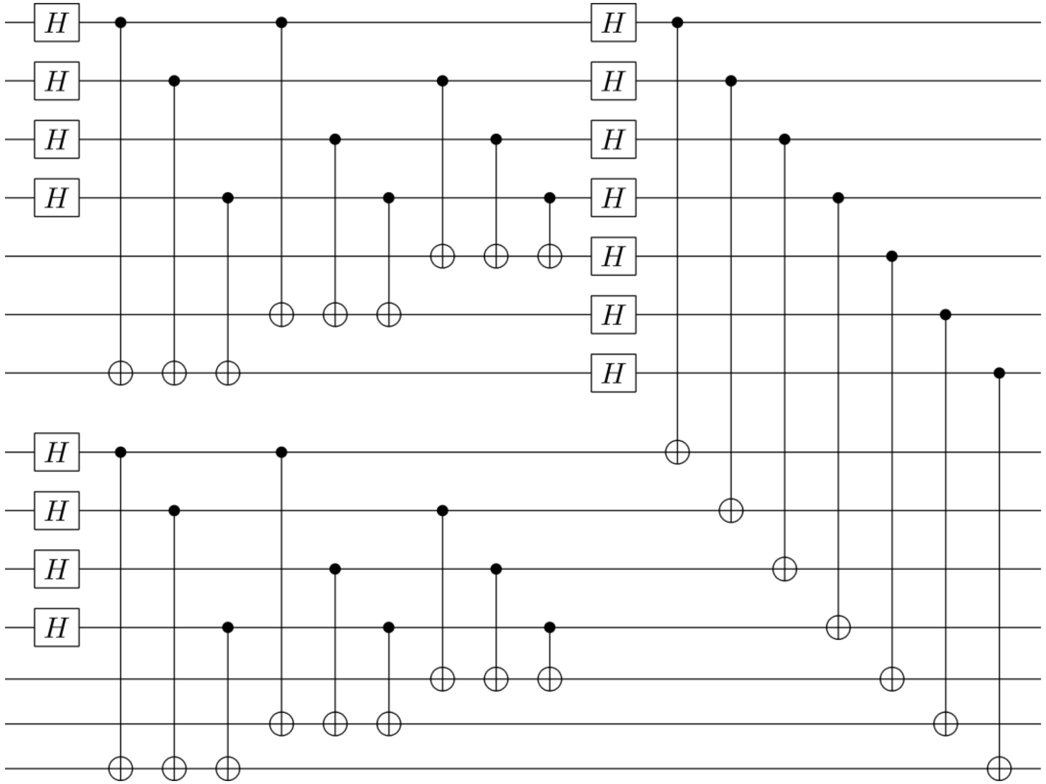


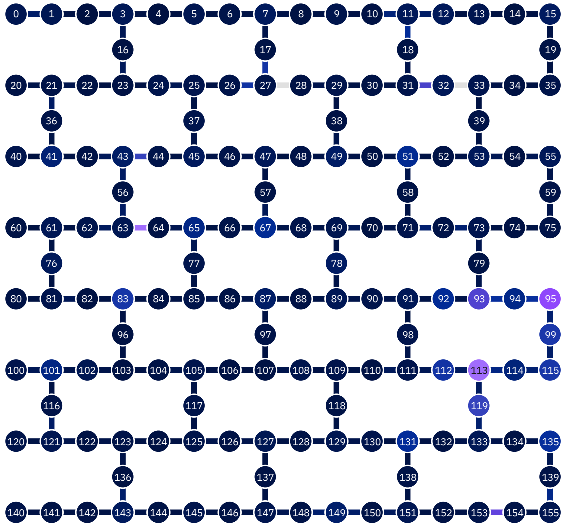


Triangular Color Code



Memory experiment using Steane code on IBM Fez





Future Directions

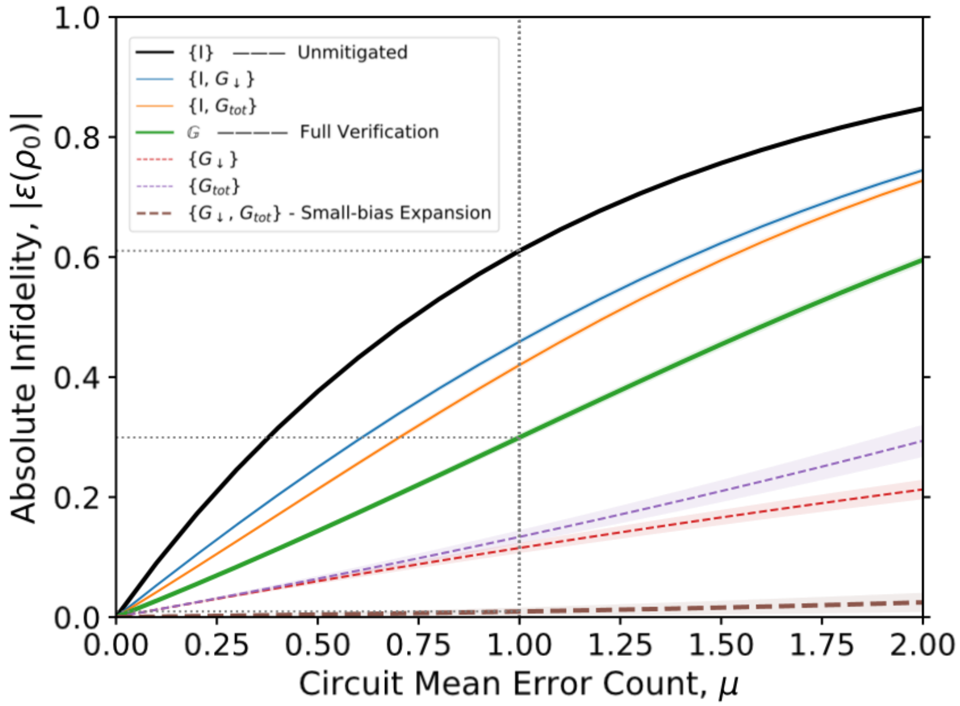
Problem-Tailored Code Construction

Challenge: Effective error detection likely needs steps to limit gate overhead

Solution: Build stabilizer code around problem parameters

Consider $|\psi\rangle$ with symmetry group generated by

$$\mathcal{S}=\langle S_1, S_2, \ldots, S_r\rangle$$



Simulated effects in Quantum Error Mitigation using Symmetry Expansion (2021)

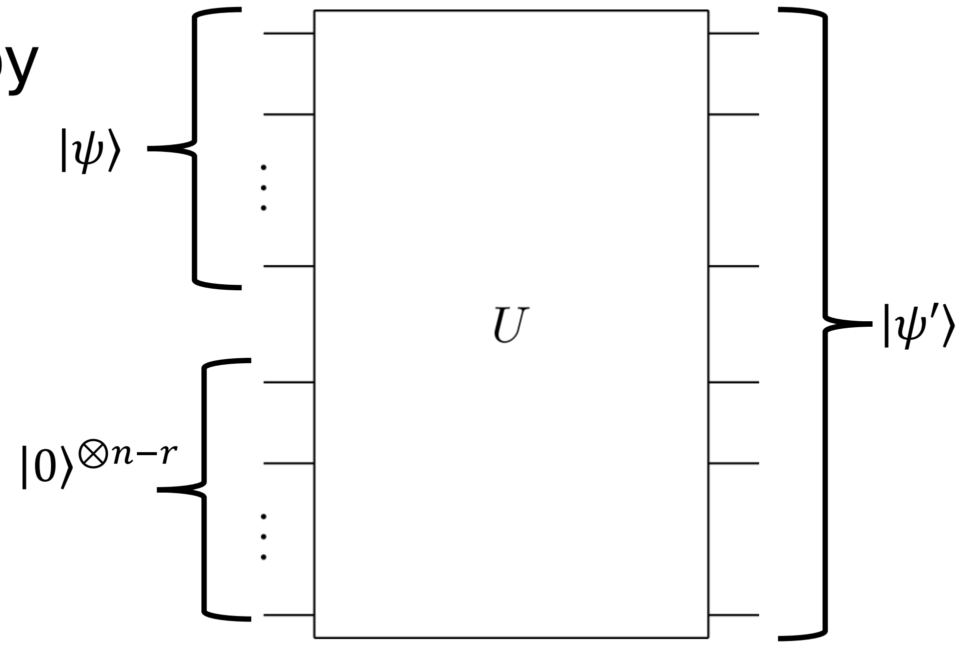
Problem-Tailored Code Construction

Given $|\psi\rangle$ with symmetry group generated by

$$\mathcal{S}=\langle S_1, S_2, \dots, S_r\rangle$$

find additional stabilizers $S_{r+1}, \dots, S_n$ such that resulting code can correct some set of errors

$$\mathcal{E} = \{E_1, E_2, \dots, E_m\}$$



Steps:

For each pair of errors $E = E_i^\dagger E_j$ in $\mathcal{E}$:

1. Check if E is in $\mathcal{S}$, or if E anti-commutes with an element of $\mathcal{S}$; if true skip 2-3
2. Set up system of equations requiring a new stabilizer to commute with all $\mathcal{S}$ and anti-commute with E
3. Solve system and add new stabilizer to $\mathcal{S}$

Combining Error Detection and Correction

“Interpolate” between quantum error detection and active error correction using syndrome measurements of only select stabilizers

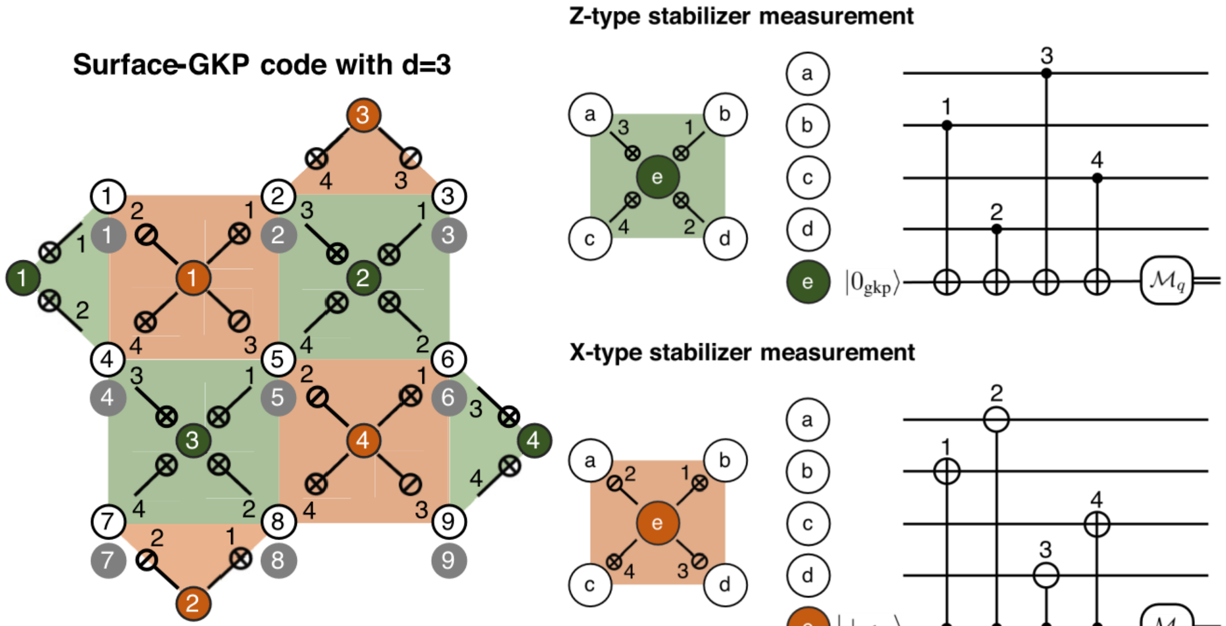


Image: *Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code (2020)*

Combining Error Detection and Correction

“Interpolate” between quantum error detection and active error correction using syndrome measurements of only select stabilizers

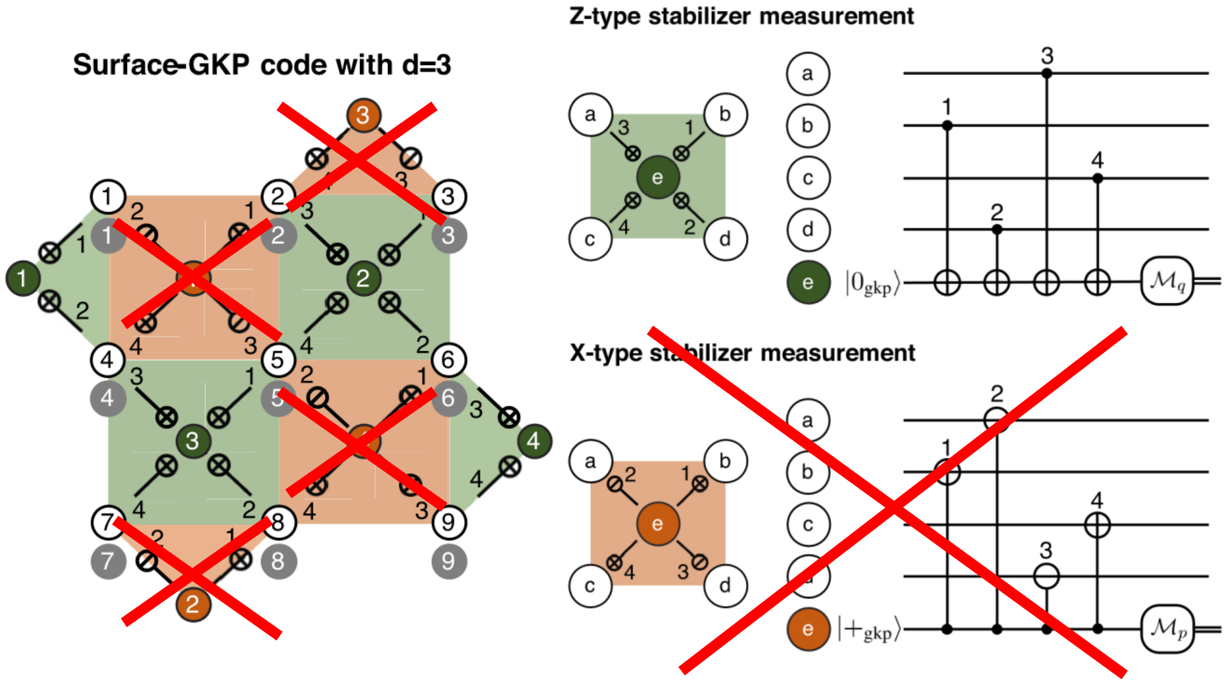
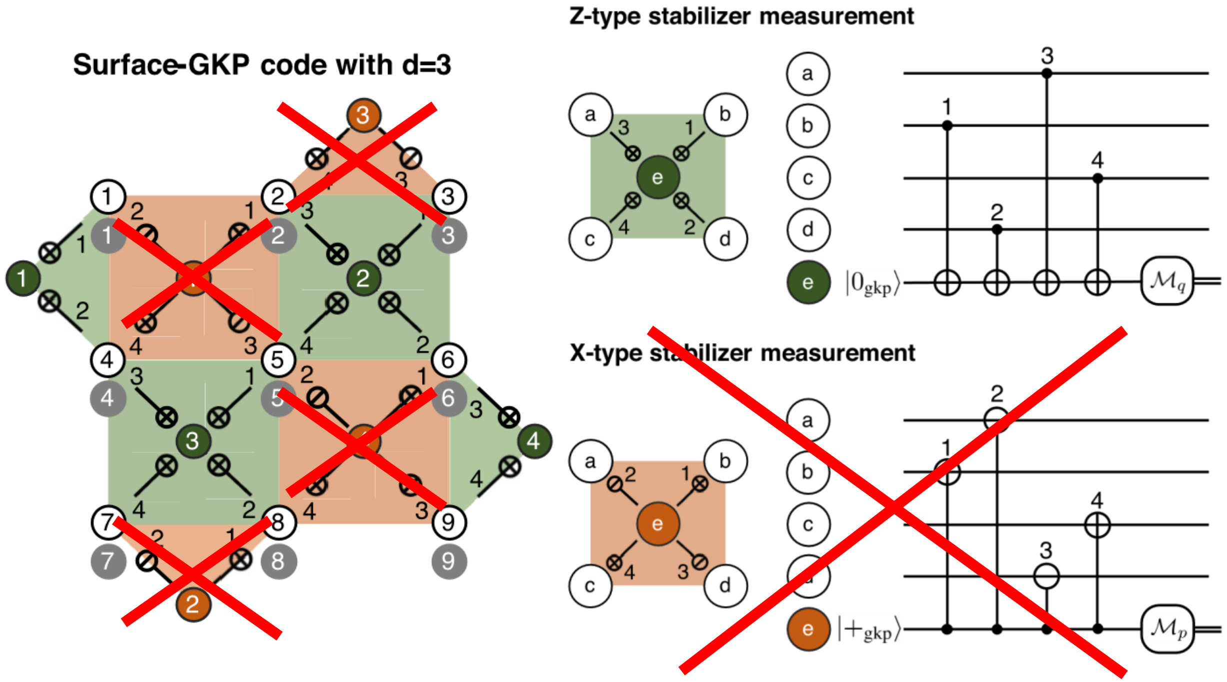


Image: *Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code (2020)*

Combining Error Detection and Correction

“Interpolate” between quantum error detection and active error correction using syndrome measurements of only select stabilizers



- Is this a viable way to combine QEM and QEC?
- How much can gate overhead be reduced by only measuring some stabilizers?
- Is there a notion of a threshold, and how to calculate?

Image: *Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code (2020)*

Collaborators

Yanis Le Fur
Institute of Fundamental Physics IFF-CSIC

Hong-Ye Hu
Harvard University

Ryan LaRose
Michigan State University