Duke Quantum Center

arXiv:2306.12862 (2023)

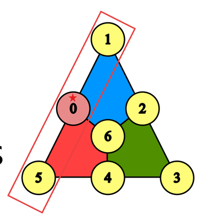
Optimization tools for distance-preserving flag fault-tolerant error correction

Balint Pato, Theerapat Tansuwannont, Shilin Huang, and Kenneth R. Brown

Duke

FLAG FAULT-TOLERANT ERROR CORRECTION

Quantum Error Correcting codes can handle the code capacity noise model – i.e. memory errors between rounds. However, syndrome measurements can be faulty in realistic settings.

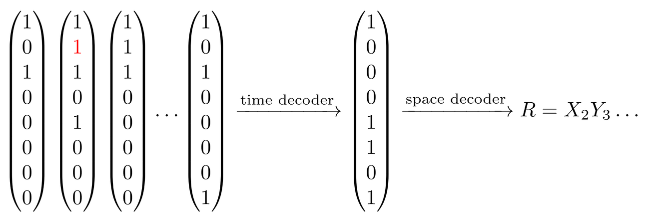


$$H = \begin{pmatrix} 1&0&0&0&1&1&1 \\ 0&0&1&1&1&0&1 \\ 1&1&1&0&0&0&1 \end{pmatrix} \quad \vec{s} = \begin{pmatrix}1\\0\\1\end{pmatrix}$$

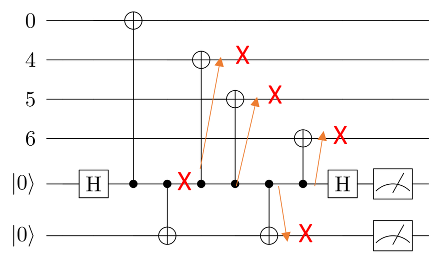
$$E = (1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0)$$

In this work [1], we separate **space decoding** and efficient **time decoding.**

Time decoding can handle untrustworthy syndrome measurements by repetition, e.g. using Shor's method [2] or our special protocols [1].



Faults on ancillas can propagate back to the data qubits as high-weight errors. **Flag qubits** [3] can **distinguish** these concerning errors from low-weight ones even if they have the same syndrome with the right **space decoder.**



To be able to distinguish the accepted syndrome $\vec{s}(E)$ of the combined data error E, we concatenate it with the cumulative flag vector $\vec{f} = \sum \vec{f}^{(i)}$ - the combined object is the **full syndrome**, which is used by the **space decoder** to generate the recovery operator.

SPACE DECODING

We use a lookup table (LUT)-based **distance verification** and **decoder** for an [[n, k, d]] self-dual CSS code under depolarizing noise models.

Naïve LUT	syndrome → recovery operator	2(n-k)+2n=4n-2k bits/entry	100%
Use logical class	syndrome → logical class	2(n-k)+2k=2n bits/entry	~50%
Self-dual CSS code	Same LUT for X and Z stabilizers	n-k+k=n bits/entry	~25%

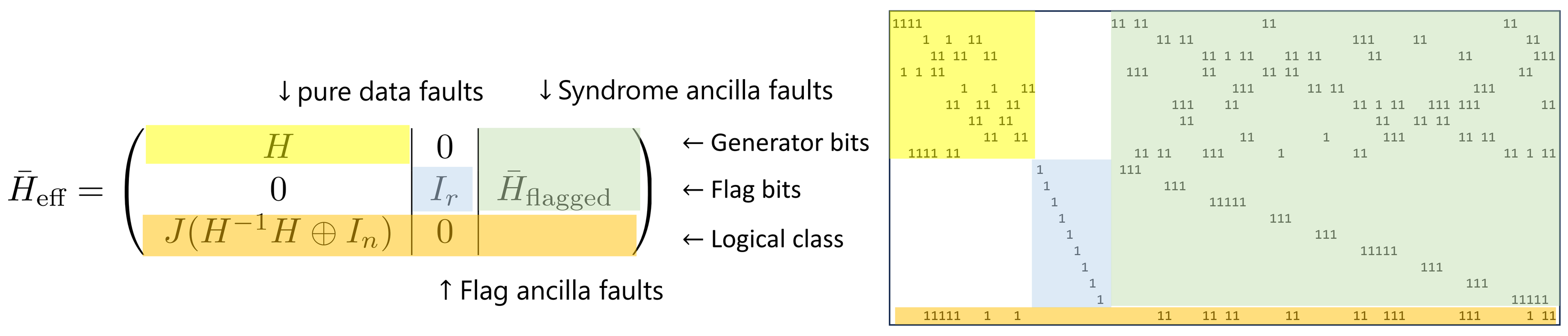
Canonical Recovery Operators (CRO): All Pauli operator $P \in \mathcal{P}_n$ can be decomposed as pure errors (E), a stabilizer (S) and a logical operator $L \in \mathcal{P}_k$, $P = ESL$. We fix a CRO (pure error) for each syndrome bit.

The **logical class of a syndrome** tells what logical correction is required after the application of the CROs based on the syndrome bit. The LUT is built from all possible combinations up to weight-t of single-qubit/location errors. This gives us a code capacity / phenomenological level most-likely-error decoder.

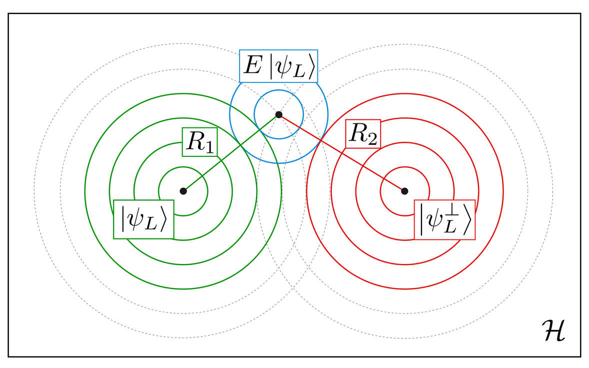
E.g., for the [[7,1,3]] code:

$$H = \begin{pmatrix} 0&0&0&1&1&1&1 \\ 0&1&1&0&0&1&1 \\ 1&0&1&0&1&0&1 \end{pmatrix},\ H^{-1} = \begin{pmatrix} 0&0&1 \\ 0&1&0 \\ 0&0&0 \\ 1&0&0 \\ 0&0&0 \\ 0&0&0 \\ 0&0&0 \end{pmatrix} \qquad L = \underbrace{J}_{\text{J=(11...1) for parity}}(\overbrace{H^{-1}H}^{\text{recoveries}} \oplus \underbrace{I_7}_{\text{1 qubit errors}}) \qquad \begin{pmatrix} H \\ L \\ W \end{pmatrix} = \begin{pmatrix} 0&0&0&1&1&1&1 \\ 0&1&1&0&0&1&1 \\ 1&0&1&0&1&0&1 \\ 0&0&1&0&1&1&0 \\ 1&1&1&1&1&1&1 \end{pmatrix}$$

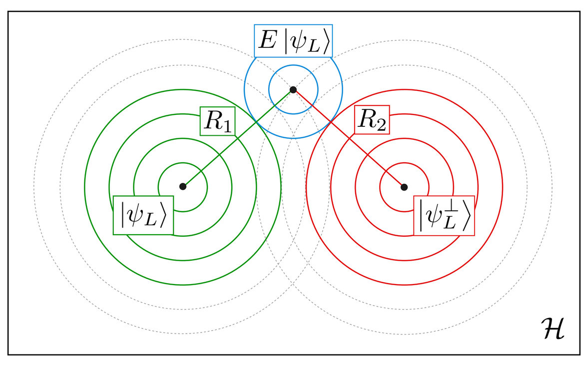
The **full syndrome** (with flags) allows a similar approach for circuit-level decoding: the **fault code** maps a **fault combination** to a full syndrome analogously to how the **error correcting code** maps a combination of single-qubit errors to generator bits.



Meet-in-the-middle optimization to increase the chance of decoding fault combinations with syndromes that are not in the lookup table.



An example case where MIM succeeds in decoding



An example case where MIM decoding leads to a logical error

TIME DECODER OPTIMIZATIONS

We adopt **adaptive syndrome measurements** [6] for flags. For a sequence of full syndromes (syndrome history) of length n, the **difference vector** $\vec{\delta}$ is a binary string of length n-1, with $\vec{\delta}_i = 0$ if $\vec{s}_i = \vec{s}_{i+1}$, and 1 otherwise. Suppose η_i are zero substrings of length γ_i and ν_i is the number of flag bits above 1 in flag syndromes between $\dot{|}$ and $\ddot{|}$, N_{11} is the number of non-overlapping 11 substrings in $\vec{\delta}$ and:

$$\vec{\delta_i} = \underbrace{\eta_1 1 \dots 1\eta_{i-1}}\dot{|}1\eta_i 1\ddot{|}\underbrace{\eta_{i+1}1 \dots 1\eta_c}, \text{ with}$$

min. # of faults	α	β
# of flag bits	μ	ω

Example

rounds	1 2 3 4 5 6 7 8 9 10
# of flag bits	1 0 2 0|0 2 1| 0 0 1
$\vec{\delta_i}$	1 1 0$\dot{\vert}$1 0 0 1$\ddot{\vert}$0 1

$\alpha_j = \beta_j = 1, \gamma_j = 2, \mu_j = 3, \nu_j = \omega_j = 1$

$N_{11} = 1$

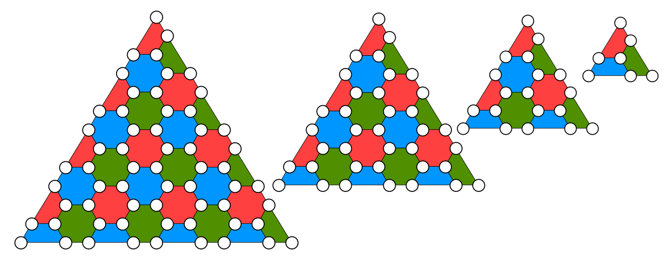
The **Fault Count Assumption (FCA)**: there are at most **t** faults in the syndrome history – this fails with probability p^{t+1}. Using the FCA, stopping conditions ensure with probability p^{t+1} that the accepted zero substring is trustworthy.

- **Shor's:** stop when $\gamma_c = t$, accept η_c
- **One-tailed adaptive:** stop if either $\gamma_c \geq 1$ and $\max(\alpha_c, \mu_c) + \gamma_c + \omega_c \geq t$, accept η_c **OR** $N_{11} \geq t$
- **Two-tailed adaptive:** stop if either $\max(\alpha_j, \mu_j) + \gamma_j + \upsilon' + \max(\beta_j, \omega_j) \geq t$, accept η_i **OR** $N_{11} \geq t$
- **Two-tailed separate XZ/ZX: Two-tailed adaptive** on X-syndromes, estimate t_x, then **Two-tailed adaptive** on Z-syndromes with fault count target $t - t_x$ (for **ZX** swap X and Z)

NOISE MODEL

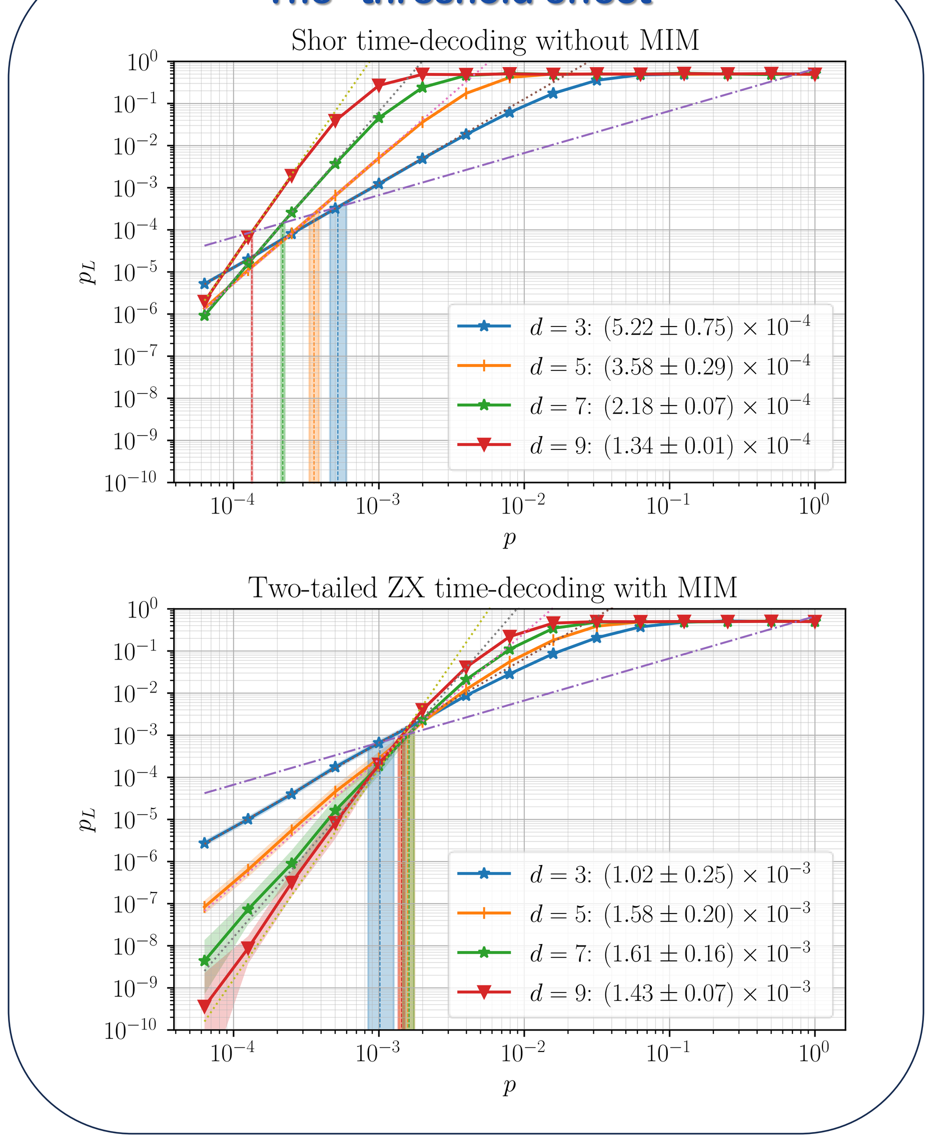
- **Circuit-level depolarizing error model** parametrized by a single physical error parameter p
  - after each 1-qubit (2-qubit) gate 1-qubit (2-qubit) depolarizing channel of strength p, which leaves the state unchanged with probability *1-p*, and with probability p/3 (p/15) applies one of the possible 1-qubit (2-qubit) Pauli operators X, Z, Y($\{I, X, Z, Y\}^{\otimes 2} \setminus \{I \otimes I\}$).
  - With probability p flip the result of a **measurement** operation and *after* a **reset** operation flip the state to the orthogonal state
- Note, that exploring and combating the effect of idling noise is left for future work.

NUMERICAL RESULTS

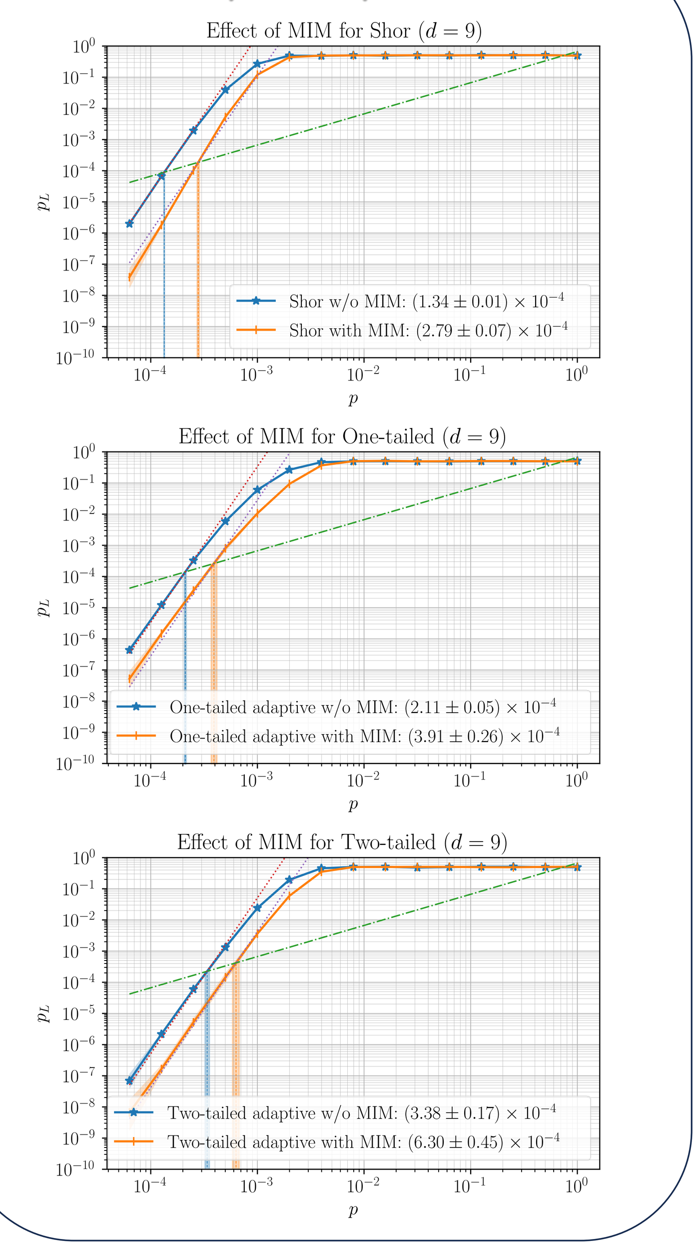


The 6.6.6 color code of distance 3, 5, 7, 9 were tested. These are self-dual CSS codes, thus benefit from all our optimizations.

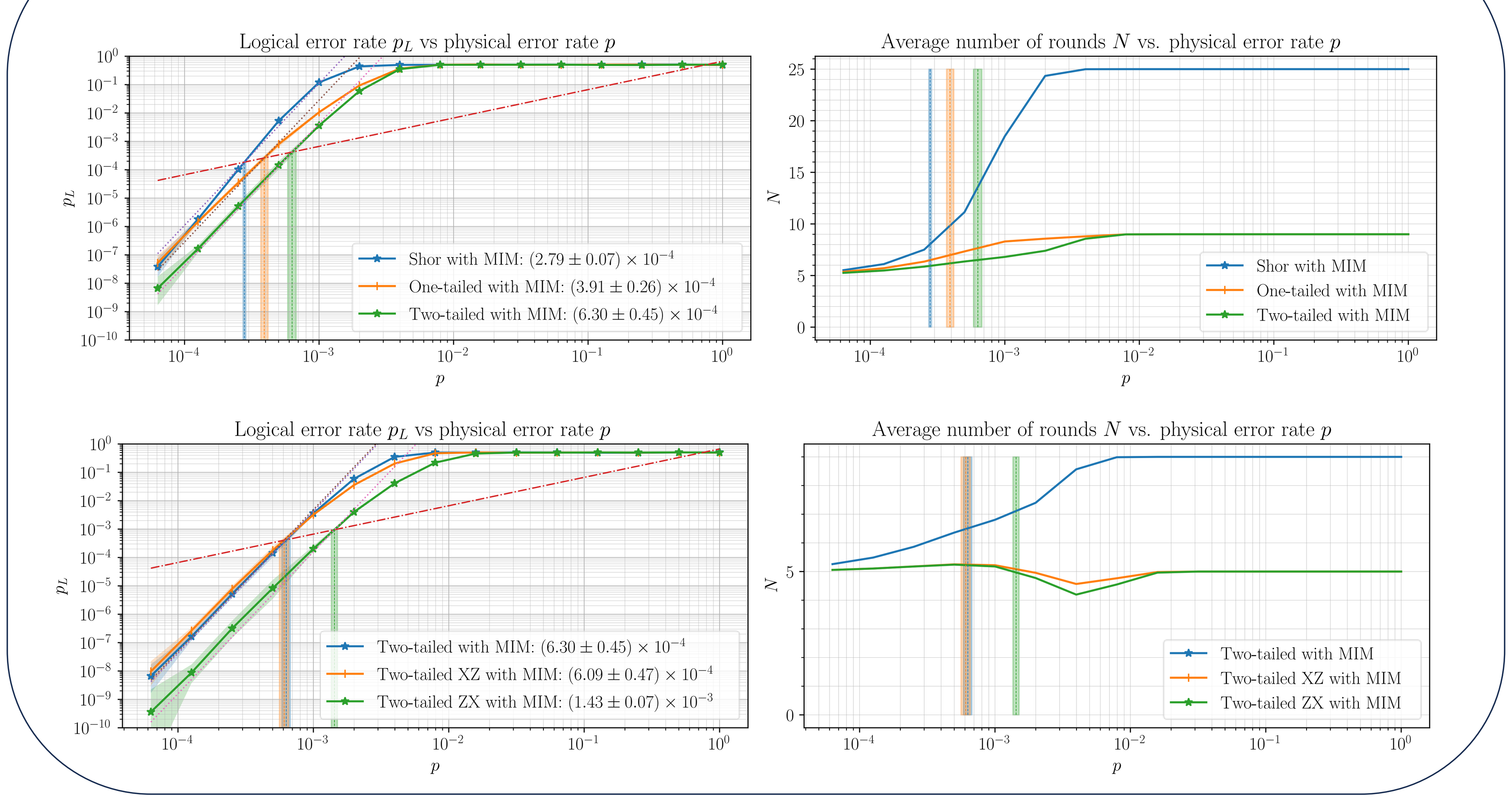
The "threshold effect"



MIM improves performance



Adaptive time decoding improves performance



Methods: Direct sampling using Cirq[7], Stim/Sinter[8], Python/C++, Slurm. Number of samples per data point vary from a minimum of 10^5 to a maximum of 10^9. Code and data will be open-sourced eventually, available for request.

[[61,1,9]] pseudo-threshold summary

Time decoder	No MIM	MIM
Shor's	1.34E-04	2.79E-04
One-tailed adaptive	2.11E-04	3.91E-04
Two-tailed adaptive	3.38E-04	6.30E-04
Two-tailed adaptive XZ		6.09E-04
Two-tailed adaptive ZX		1.43E-03

Conclusions: Both space and time decoding optimizations can have a significant effect on logical error rates. Near-term architectures might find LUT-based methods attractive, but the table is not scalable. Future work should explore scalable decoding of the fault code and the effect of idling noise.

REFERENCES

[1] B. Pato*, T. Tansuwannont*, S. Huang & K. R. Brown arXiv:2306.12862 (2023).
[2] P. W. Shor, *Fault-Tolerant Quantum Computation*, in *Proc. of 37th Conference on Foundations of Computer Science* (1996), pp. 56–65.
[3] R. Chao & B. W. Reichardt, *Quantum Error Correction with Only Two Extra Qubits*, Phys. Rev. Lett. **121**, 050502 (2018).
[4] E. Sabo, A. B. Aloshious & K. R. Brown, *Trellis Decoding For Qudit Stabilizer Codes And Its Application To Qubit Topological Codes*, arXiv:2106.08251 (2022).
[5] T. Tansuwannont & D. Leung, *Fault-Tolerant Quantum Error Correction Using Error Weight Parities*, Phys. Rev. A **104**, 042410 (2021).
[6] T. Tansuwannont, B.Pato & K. R. Brown, *Adaptive Syndrome Measurements for Shor-Style Error Correction*, Quantum **7**, 1075 (2023).
[7] Cirq Developers, *Cirq* (v1.1.0). *Zenodo*. https://doi.org/10.5281/zenodo.7465577 (2022).
[8] C. Gidney, *Stim: A Fast Stabilizer Circuit Simulator*, Quantum **5**, 497 (2021).

Acknowledgments: Funded by IARPA, LPS/ARO, ARO MURI, NSF Institute for Robust Quantum Simulation