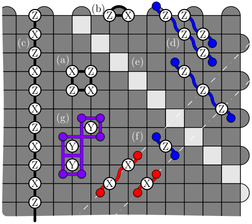
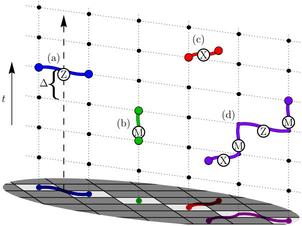
The XZZX surface code

Benjamin J. Brown

arXiv:2009.07851



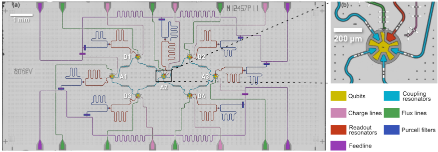


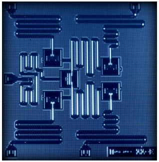
With thanks to my coauthors:
J. Pablo Bonilla Ataides,
David K. Tuckett,
Stephen D. Bartlett
and Steven T. Flammia

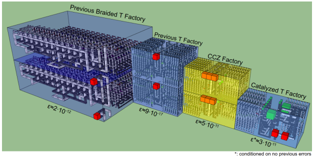
Scaling quantum computers

To scale a quantum computer we must:

Control a large number of qubits below threshold as they perform repeated stabilizer measurements[1,2].







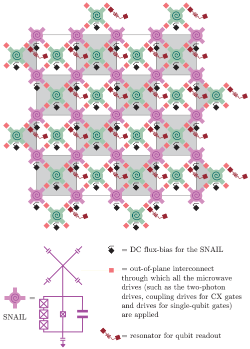
Furthermore, there is a huge overhead cost to perform fault-tolerant quantum logic operations.[3]

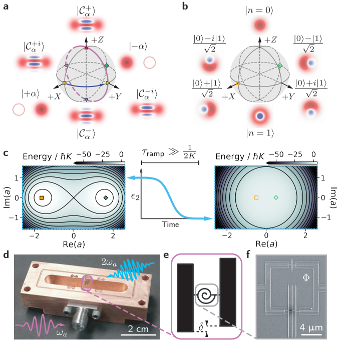
---

[1] M. Takita *et al.*, Phys. Rev. Lett. **119**, 180501 (2017)
[2] C. Kraglund Andersen *et al*, Nat. Phys. **16**, 875 (2020)
[3] C. Gidney and A. Fowler, Quantum **3**, 135 (2019)

Scaling quantum computers

Cat qubits [4, 5] (that experience highly biased noise) have been proposed as high-quality qubits (with bias preserving gates [6, 7]) for surface code implementations[8].





[4] A. Grimm *et al.* Nature **584**, 205 (2020)

[5] R. Lescanne *et al.*, Nat. Phys. **16**, 509 (2020)

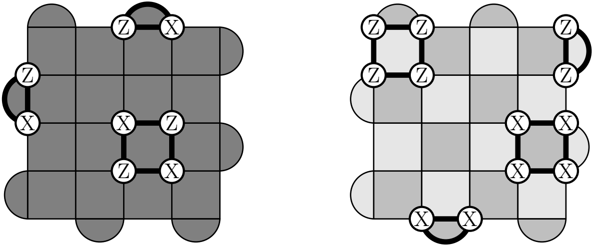
[6] J. Guillaud and M. Mirrahimi, Phys. Rev. X **9**, 041053 (2019)

[7] S. Puri *et al.* Sci. Adv. **6**, eaay5901 (2020)

[8] C. Chamberland *et al.* arXiv:2012.04108 (2020)

The XZZX surface code

The XZZX surface code[9] is locally equivalent to the standard surface code[10]

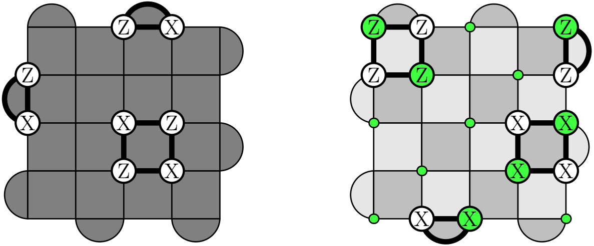


[9] X.-G. Wen, Phys. Rev. Lett. **90** 16803 (2003)

[10] A. Kitaev, Ann. Phys. **303**, 2 (2003)

The XZZX surface code

. . . except every other qubit is rotated by a hadamard gate.
(the green qubits are rotated)

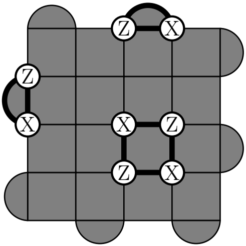


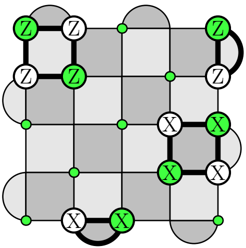
The XZZX surface code

The XZZX code demonstrates:

- Thresholds that match the hashing bound for all uniform single-qubit Pauli noise channels.
- Exceptionally high fault-tolerant thresholds for biased noise models
- Reduced overheads (by a factor $O\left(1/\eta^{d/4}\right)$) at low error rates and high bias η

- We also argue that we can maintain these advantages while performing computation

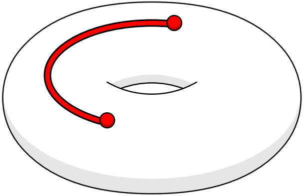


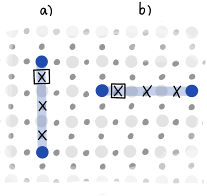


Error correction with the surface code

We use stabilizers to measure (Pauli) errors E.

$$SE|\psi\rangle = (-1)E|\psi\rangle$$





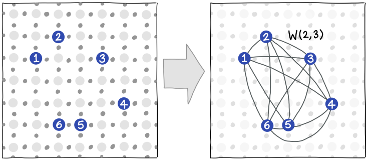
On the surface code, errors appear as strings, and defects appear in pairs at their endpoints[11].

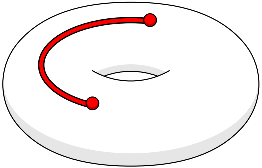
This can be viewed as a defect parity conservation law[12,13].

[11] E. Dennis *et al.*, J. Math. Phys. **43**, 4452 (2002)
[12] A. Kitaev, Ann. Phys. **303**, 2 (2003)
[13] BJB and D. J. Williamson, Phys. Rev. Research **2**, 013303 (2020)

Symmetries and conservation laws

MWPM is possible because the surface code respects a defect conservation symmetry[14,15]





Mathematically, the symmetry is apparent in the stabilizer group

$$\prod_v A_v = \mathbb{1} \quad \Rightarrow \quad \prod_v a_v = 1$$

($a_v = \pm 1$ eigenvalues of stabilizers A_v.)

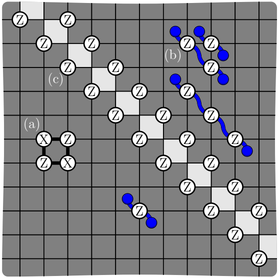
$\Rightarrow \quad \#v$ with $a_v = -1$ is even $\equiv$ defect conservation (mod2)

[14] E. Dennis *et al.*, J. Math. Phys. **43**, 4452 (2002)

[15] BJB and D. J. Williamson, Phys. Rev. Research **2**, 013303 (2020)

A symmetry with respect to Pauli-Z errors[16, 17]

We find a richer space of symmetries if we restrict our error model.



Pauli-Z errors commute with the product of stabilizers along the diagonal and therefore respect a $1D$ symmetry on the XZZX code.

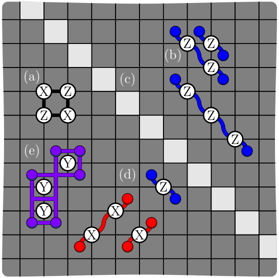
Decoding Z errors is therefore equivalent to decoding the repetition code (50% threshold).

---

[16] BJB and D. J. Williamson, Phys. Rev. Research **2**, 013303 (2020)
[17] D. K. Tuckett, Phys. Rev. Lett. **124**, 130501 (2020)

Symmetries for all Pauli errors[18, 19]

The XZZX code has many symmetries for other error models.



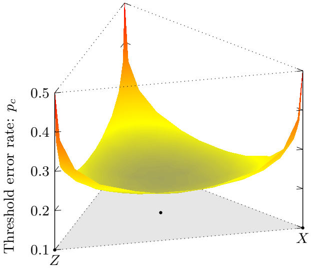
The XZZX code has symmetries with respect to Pauli-X errors along perpendicular diagonals to those for Pauli-Z errors.

Pauli-Y errors have the same symmetry as the CSS surface code (we exploited these symmetries in earlier work)

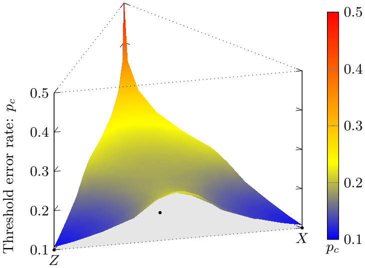
[18] BJB and D. J. Williamson, Phys. Rev. Research **2**, 013303 (2020)
[19] D. K. Tuckett, Phys. Rev. Lett. **124**, 130501 (2020)

High threshold error rates

The XZZX code has a threshold that matches the zero-rate hashing bound for all uniform single-qubit Pauli channels.



XZZX



CSS

$$R \equiv k/n \geq 1 - H(\vec{p}), \quad (1)$$

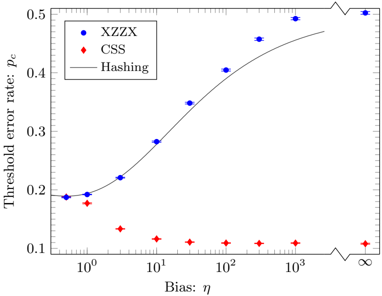
$H(\vec{p})$ - Shannon entropy.

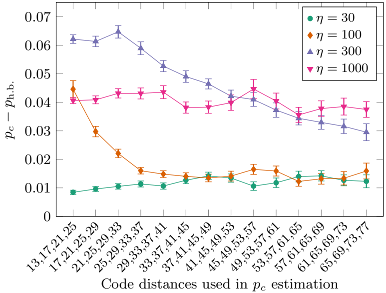
$\vec{p} = (p_X, p_Y, p_Z)$ with p_O the probability that Pauli error O occurs.

R - rate with $R = 0$ the zero-rate hashing bound.

Numerics exceeding hashing

At high bias, our numerics demonstrate a threshold above the zero-rate hashing bound





See also other work on exceeding hashing for instance[20,21,22]

[20] D. DiVincenzo *et al.*, Phys. Rev. A **57**, 830 (1998)
[21] G. Smith and J. Smolin, Phys. Rev. Lett. **98**, 030501 (2007)
[22] J. Bausch and F. Leditzky, arXiv:1910.00471 (2019)

Exceeding hashing

A zero-rate code with a threshold above hashing implies we can send information at non-zero rate with p above hashing.

We need the following:

- A finite rate code $R_{\mathrm{out}} = K_{\mathrm{out}}/N_{\mathrm{out}} > 0$
- An inner code of constant size N_{in} with threshold $p_{\text{th.}} > p_{\text{h.b.}}$

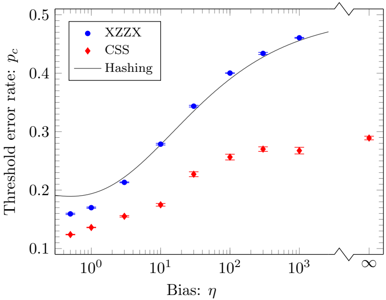
Concatenating gives a family of codes with rate:

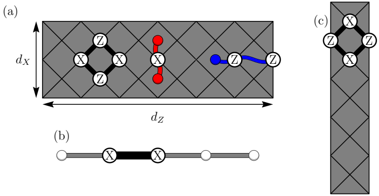
$$R' = R_{\mathrm{out}}/N_{\mathrm{in}} > 0. \tag{2}$$

using qubits with $p_{\text{h.b.}} < p < p_{\text{th.}}$ for some constant N_{in}.

Numerics exceeding hashing

We also exceed hashing with a suboptimal matching decoder



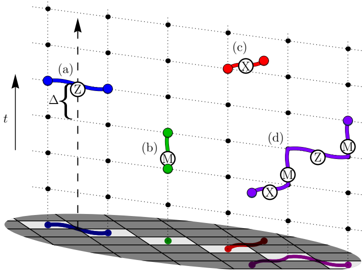


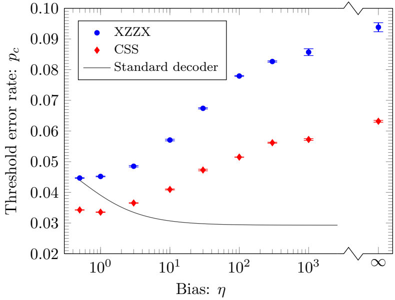
We achieved this using XZZX codes with different boundary conditions.

The high-speed matching decoder allows us to probe very large system sizes

Fault-tolerant thresholds with biased noise

Matching decoders generalise readily to the fault-tolerant setting.[23, 24, 25]





We find exceptional fault-tolerant thresholds for the XZZX code under phenomenological biased noise.

[23] E. Dennis *et al.*, J. Math. Phys. **43**, 4452 (2002)
[24] BJB and D. J. Williamson, Phys. Rev. Research **2**, 013303 (2020)
[25] D. K. Tuckett, Phys. Rev. Lett. **124**, 130501 (2020)

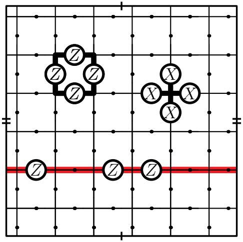
Logical failure rates below threshold

We need a low logical failure rate $\overline{P}$
with a small number of qubits n.

At low error rate p we can generically achieve

$$\overline{P} \sim p^{d/2} \tag{3}$$

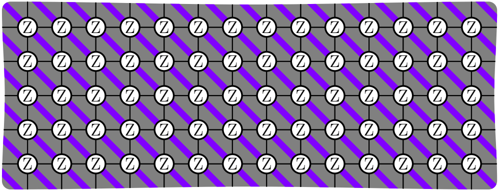
To leading order, this is the probability of $d/2$ errors occurring.



The surface code uses $n = O(d^2)$ qubits.

Logical failure rates below threshold

We can choose an XZZZX code with a weight n Pauli-Z logical.



At infinite bias[26], we can expect a logical failure rate like

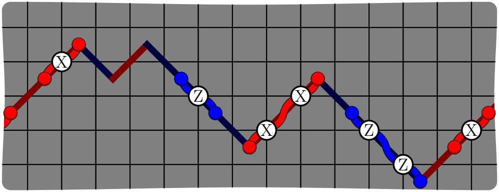
$$\overline{P} \sim p^{n/2}. \quad (4)$$

...but what happens at finite bias?

[26] D. Tuckett *et al.* Phys. Rev. X **9**, 041031 (2019)

Logical failure rates below threshold

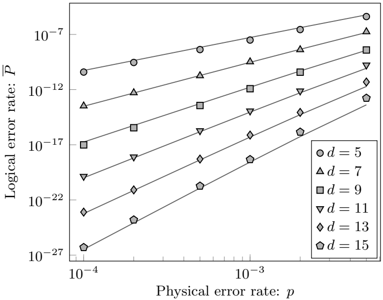
In practice we find errors of
$d/4$ low rate errors
and $d/4$ high rate errors.



$$\overline{P} \sim \underbrace{p^{d/4}}_{\substack{d/4 \text{ high} \\ \text{rate errors}}} \times \underbrace{\left(\frac{p}{\eta}\right)^{d/4}}_{\substack{d/4 \text{ low} \\ \text{rate errors}}} = \underbrace{\left(\frac{1}{\eta}\right)^{d/4}}_{\substack{\text{factor} \\ \text{of} \\ \text{improvement}}} p^{d/2}. \tag{5}$$

Logical failure rates below threshold

We test the ansatz at low p using the splitting method[27,28]



$$\overline{P} \sim \underbrace{p^{d/4}}_{\substack{d/4 \text{ high} \\ \text{rate errors}}} \times \underbrace{\left(\frac{p}{\eta}\right)^{d/4}}_{\substack{d/4 \text{ low} \\ \text{rate errors}}} = \underbrace{\left(\frac{1}{\eta}\right)^{d/4}}_{\substack{\text{factor} \\ \text{of} \\ \text{improvement}}} p^{d/2}. \tag{6}$$

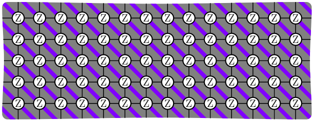
---

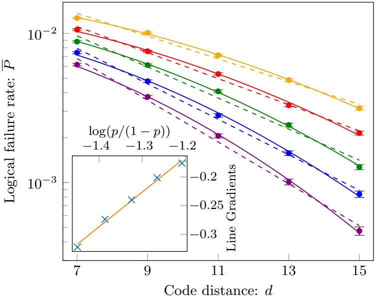
[27] C. Bennett, J. Comput. Phys. **22**, 245 (1976)

[28] Bravyi and Vargo, Phys. Rev. A **88**,062308 (2013)

Logical failure rates below threshold

For small n and high bias we identify signatures of p^{d^2} logical failure rate scaling





We observe this when

$$p^{d^2} \gg \left(\frac{1}{\eta}\right)^{d/4} p^{d/2} \tag{7}$$

(in other words, at moderate p and small-ish n)

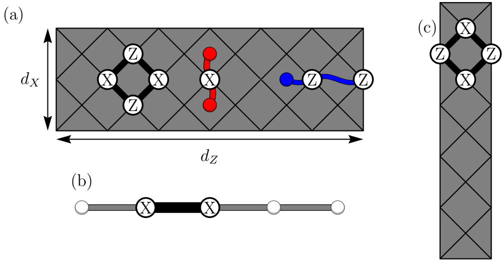
Logical failure rates below threshold

It is favourable to find codes with open boundary conditions.

Changing the lattice geometry means we have logical failure rates like

$$\overline{P}_X \sim \left(\frac{p}{\eta}\right)^{d_X/2} \quad \text{and} \quad \overline{P}_Z \sim p^{d_Z/2}, \tag{8}$$

at low p.

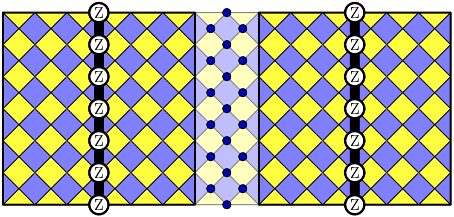


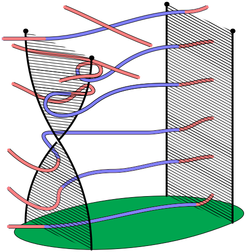
We can therefore save resources by choosing small d_X without compromising $\overline{P}_Z$.

Fault-tolerant quantum computation

We can perform computations using code deformations[29,30], e.g.,

Braiding twists[31,32]



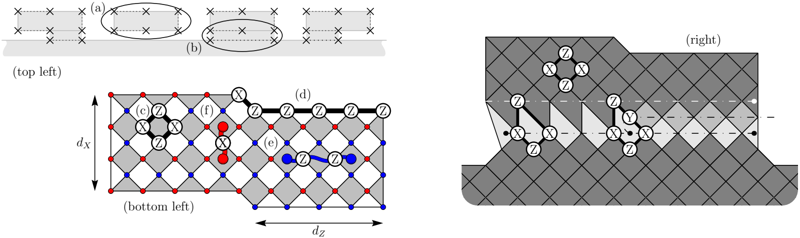


Lattice surgery[33]

[29] R. Raussendorf *et al.* Phys. Rev. Lett. **98**, 190504 (2007)
[30] H. Bombin and M. A. Martin-Delgado, J. Phys. A **42**, 095302 (2009)
[31] H. Bombin, Phys. Rev. Lett. **105**, 030403 (2010)
[32] BJB *et al.*, Phys. Rev. X **7**, 021029 (2017)
[33] C. Horsman *et al.*, New J. Phys. **14**, 123011 (2012)

Fault-tolerant quantum computation

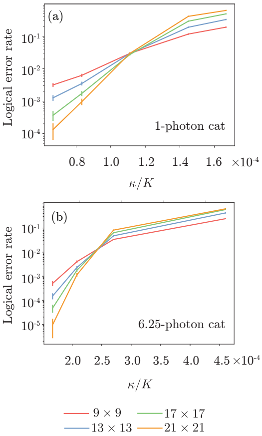
We can maintain these advantages undergoing fault-tolerant quantum computation[34].

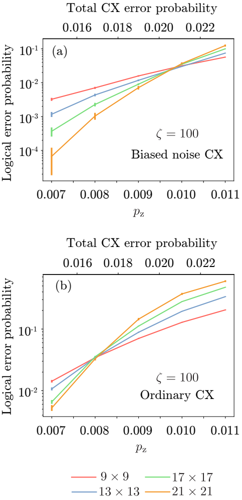


The one-dimensional symmetries we need for high thresholds are maintained under initialisation and XZZX codes with twists.

[34] D. Litinski, Quantum **3**, 128 (2019)

Forthcoming work:
XZZX thresholds with cat qubit circuits



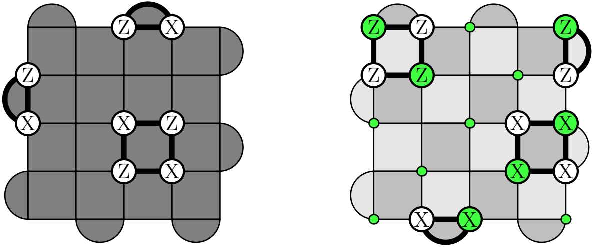


work with A. Darmawan, A. Grimsmo, S. Puri and D. Tuckett.

Outlook

We have shown the XZZX code:

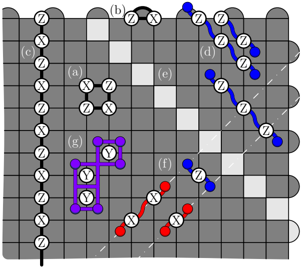
- has threshold error rates that match hashing
- has exceptionally high fault-tolerant thresholds
- significantly reduced resource costs below threshold
- can maintain its performance during computational operations

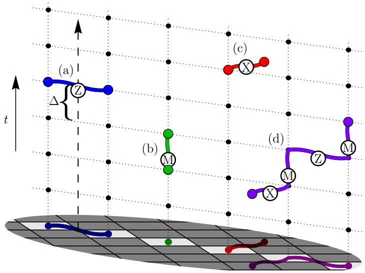


Outlook

Our work raises questions in several areas, such as:

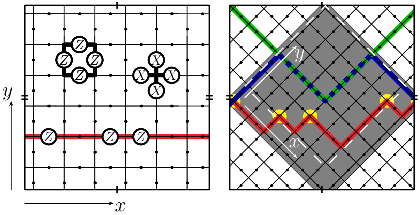
- Have we really exceeded the hashing bound?
- How do we specialise codes for a given noise model in general?
- What is the potential for non-CSS codes?

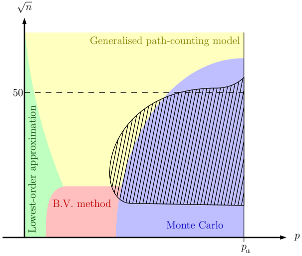


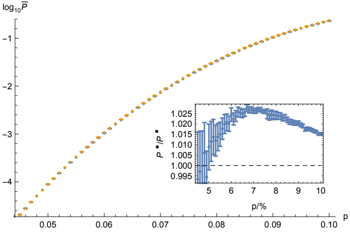


Backup slide: Different geometries

Previous work[35] has shown negligible difference in logical error rates against bit-flip noise for modest p as a function of number of qubits n.







[35] M. Beverland *et al.* J. Stat. Mech.:Theo. Exp. **2019**, 073404 (2019)