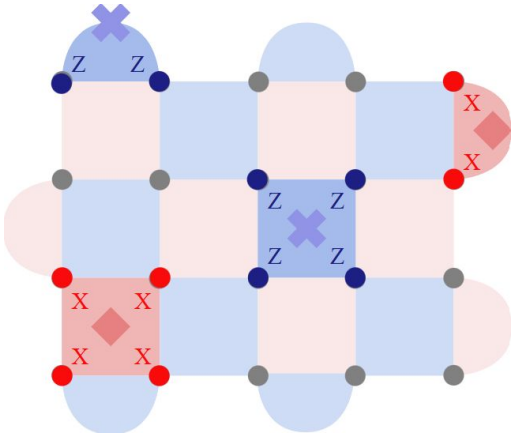
Concatenating Algebraic Codes over High-Rate Quantum LDPC Codes

Adam Wills, Michael E. Beverland, Lev S. Bishop, Jay M. Gambetta, Patrick Rall, Vikesh Siddhu, Andrew W. Cross, Theodore J. Yoder, Isaac Chuang

arXiv:2605.21898

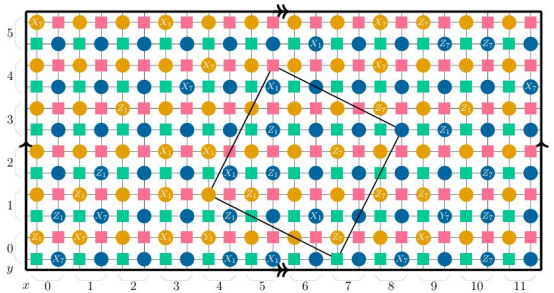
Companion Papers: arXiv:2605.20346, arXiv:2605.18981

Surface Codes versus High-Rate qLDPC Codes



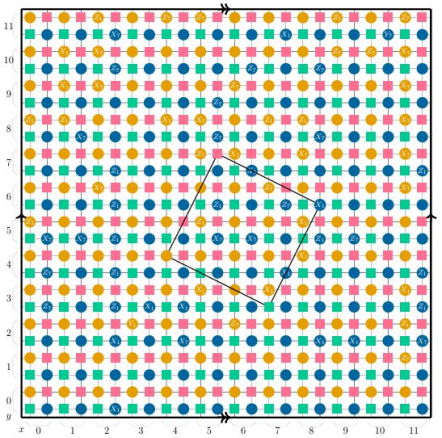
$[[d^2, 1, d]]$

Surface Code



$[[144, 12, 12]]$

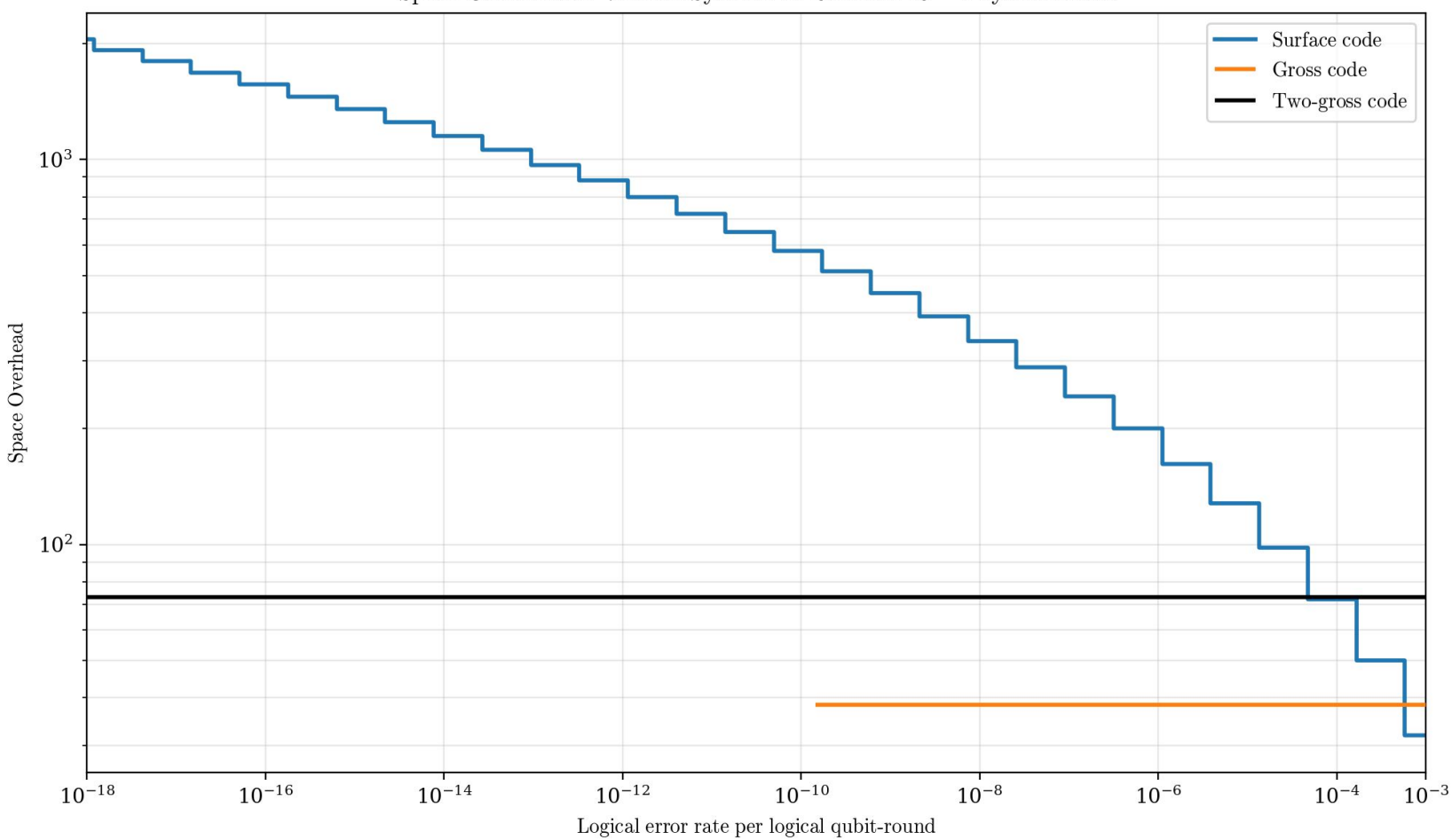
Gross Code



$[[288, 12, 18]]$

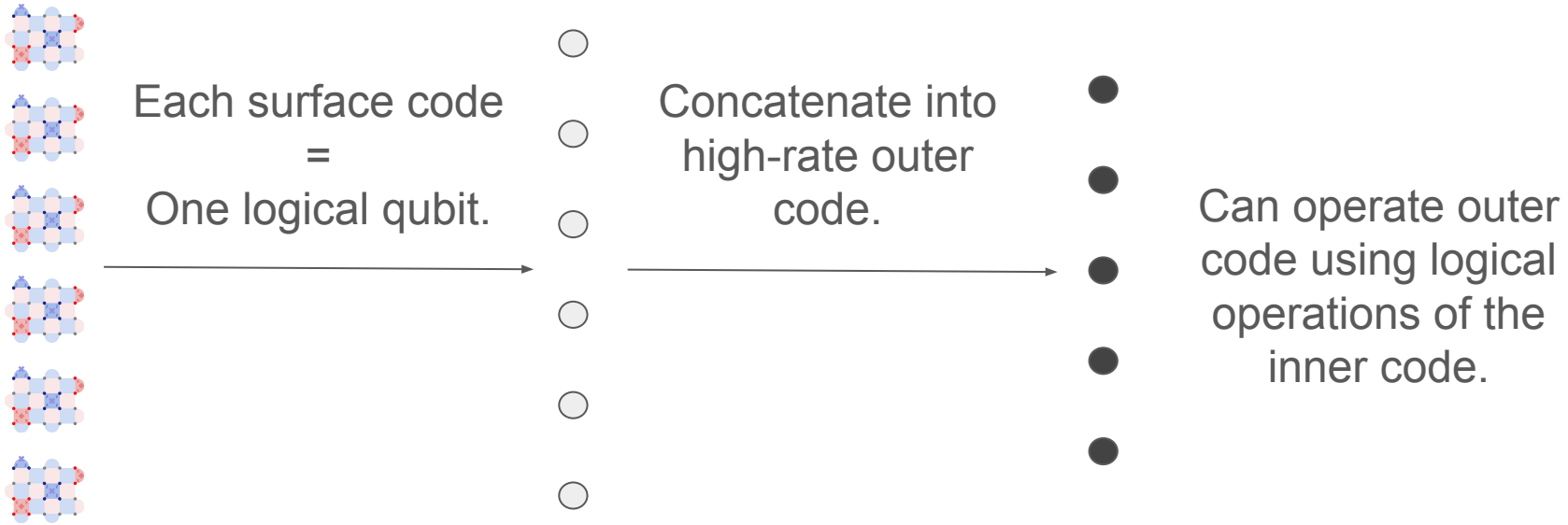
Two-Gross Code

Figures from the Error Correction Zoo and arXiv:2506.03094


Space Overhead of Various Systems at Uniform 10^{-3} Physical Noise
Surface code
Gross code
Two-gross code
10^3
10^2
Space Overhead
10^{-18}
10^{-16}
10^{-14}
10^{-12}
10^{-10}
10^{-8}
10^{-6}
10^{-4}
10^{-3}
Logical error rate per logical qubit-round

Concatenated Codes (1 of 2)

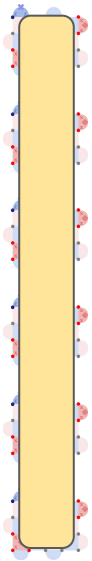
Imagine concatenating an outer code over the surface code.



Concatenated Codes (2 of 2)

E.g. $[[n, n-2, 2]]$ outer code (even n)

Two Checks: $X^{\otimes n}$ and $Z^{\otimes n}$



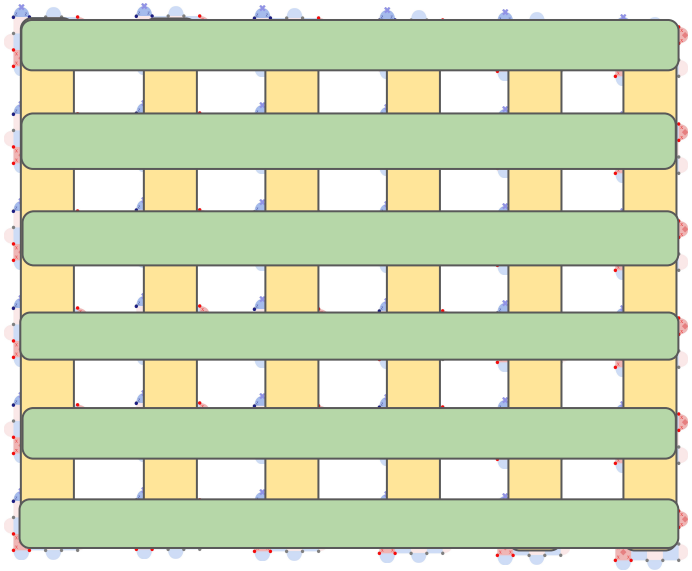
Extract outer code syndrome using logical operations of surface codes

References:

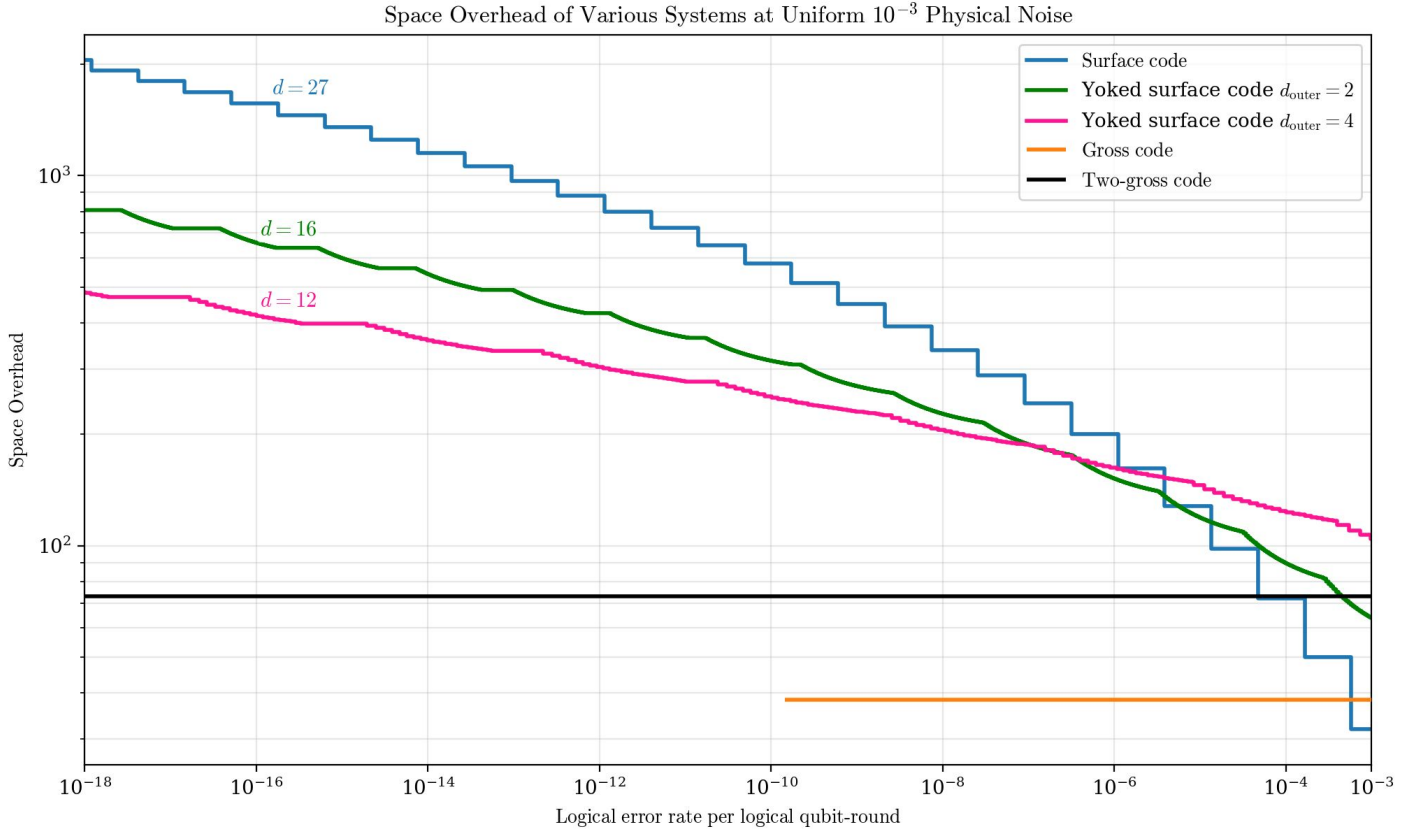
Yoked Surface Codes (Gidney et al.)

Simulating qLDPC codes with local gates (Pattison et al.)

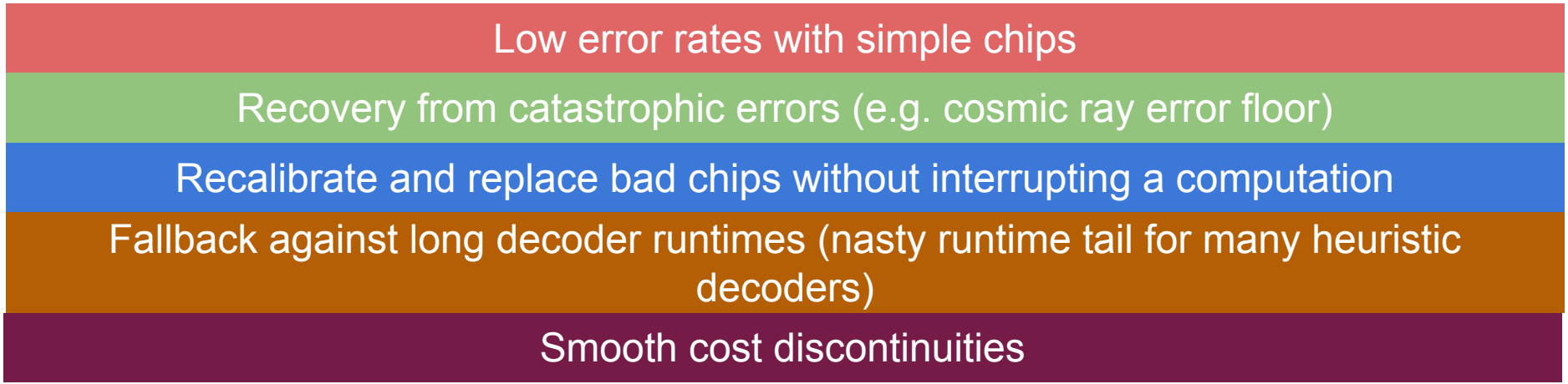
E.g. $[[n, n-4\sqrt{n}+2, 4]]$ outer code (square n)



Concatenation: A Free Lunch?



Engineering Benefits of Concatenation



Concatenation over High-Rate Codes: The Challenge

Key Challenge:

In a single high-rate codeblock, the multiple logical qubits fail in a correlated way

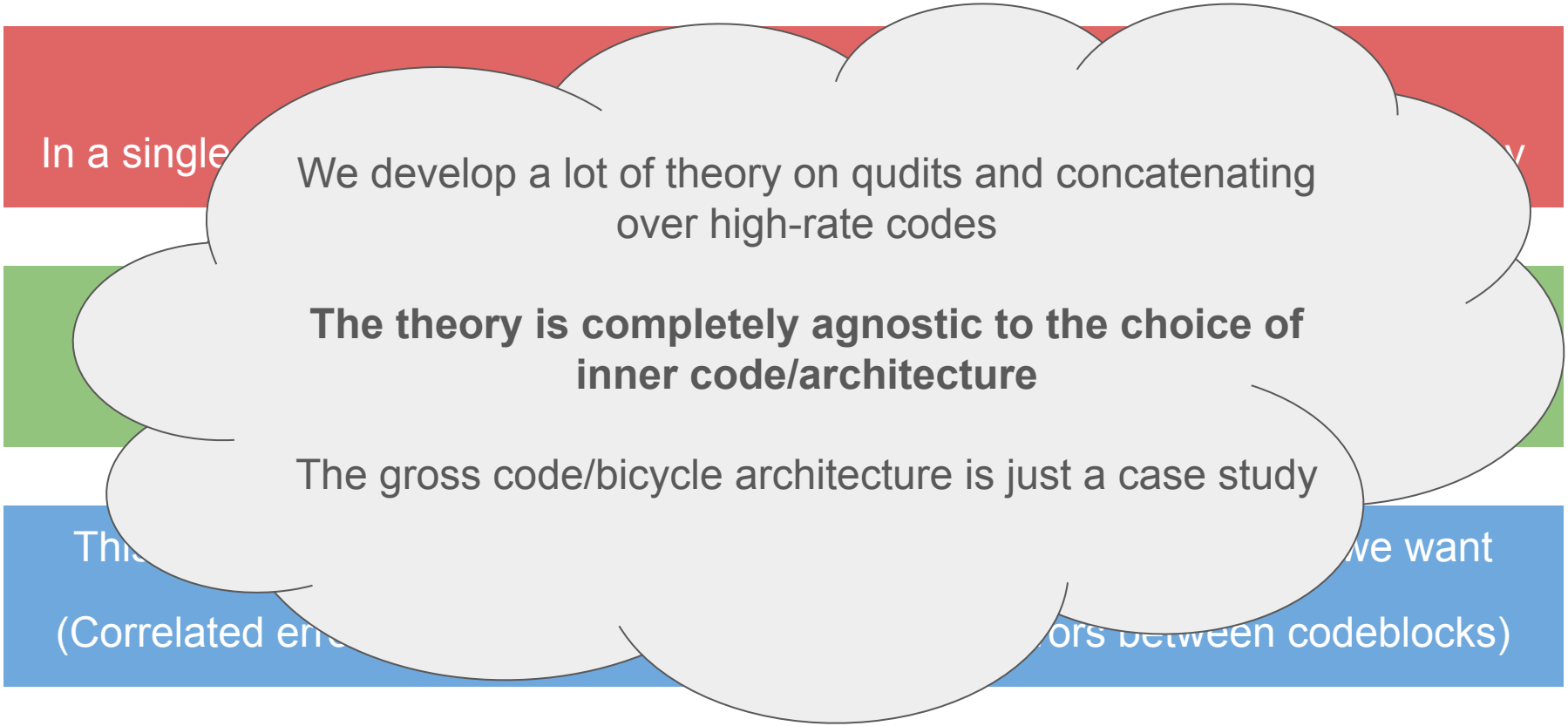
Solution:

Package the logical qubits into one larger qudit, and use qudit codes

This works because the qudit codes treat the exact error distribution we want

(Correlated errors in each codeblock, independent errors between codeblocks)

Concatenation over High-Rate Codes: The Challenge



What are Galois Qudits?

A q-dimensional Galois qudit is a qudit encoding a finite field $\mathbb{F}_q$

- Hilbert Space: $\mathbb{C}^q$ and computational basis states $\{|\eta\rangle : \eta \in \mathbb{F}_q\}$
- Paulis: For $\gamma, \beta \in \mathbb{F}_q$, $X^\gamma |\eta\rangle = |\eta + \gamma\rangle$ and $Z^\beta |\eta\rangle = (-1)^{\mathrm{tr}(\beta\eta)} |\eta\rangle$ where $\mathrm{tr} : \mathbb{F}_q \to \mathbb{F}_2$ is a linear map

A $q=2^s$-dimensional Galois qudit is the same thing as a set of s qubits

(Hilbert space, Pauli group, Clifford group, Clifford hierarchy)

"Qudit-to-qubit mapping" given by basis B for F_q over F_2 — gives consistent way to map all states, Paulis, Cliffords, stabiliser codes, etc.

See arXiv:2605.18981 for full mathematical preliminaries

Quantum Reed-Solomon Codes

We use quantum Reed-Solomon codes (build from two classical Reed-Solomon codes)

Qudit codes with parameters $[[n, n-2, 2]]_q$, $[[n, n-4, 3]]_q$, $[[n, n-6, 4]]_q$ and so on for any $n \leq q$

Qudit dimension $q = 2^{\#\text{ Logical qubits in one block}}$

Optimal parameters (quantum Singleton bound) — only available to qudits/high-rate inner codes

List Decoding (1 of 3)

There are efficient list decoders for classical Reed-Solomon codes (Sudan `97; Guruswami-Sudan `98)

List decoders can list all possible errors up to a certain weight (generally beyond half the distance)

These directly apply to quantum Reed-Solomon codes because they are built from two classical Reed-Solomon codes (one for X, one for Z)

The structure of RS codes means that under random errors the list is usually very short (e.g. only one error beyond half the distance)

List Decoding (2 of 3)

In the usual binary (qubit) setting, we are used to

Failure Probability $\approx C \cdot p^t$

Minimum number of errors causing failure

(usually thinking of $t = d/2$)

Combinatorial factor (positive integer)

For large qudits, C can be a fraction!

The structure of RS codes means that under random errors the list is usually very short (e.g. only one error beyond half the distance)

List Decoding (3 of 3)

Example: $[[n, n-8, 5]]_q$ code. Minimum number of errors causing failure is $t = 3$

You can calculate failure probability $\approx \frac{n^6}{36(q-1)} \cdot p^3$

List decoding becomes stronger as q gets much larger than n (q is exponential in the number of logical qubits)

Spacetime Error Mixing (1 of 3): How to Benchmark the Scheme with Limited Information

- Typically, when we simulate the performance of the gross code, we would usually just estimate the ``block error rate''

$$p_{\text{block}} = \mathbb{P}\left[\text{Gross code has any logical error}\right]$$

- In principle, to calculate the performance of our scheme, we need access to much more fine-grained information:

$$p_P = \mathbb{P}\left[\text{Gross code has the logical Pauli error P}\right]$$

for every non-trivial Pauli P

This is a problem, because we do not have knowledge of every p_P

Spacetime Error Mixing (2 of 3)

How can we calculate the performance of our scheme even given limited information?

- Consider a well-chosen parameter of the protocol (“mixing parameter”), randomise it, calculate the probability of failure in expectation over that randomness
- These calculations can be done only using knowledge of aggregate statistics like p_{block}

- Key Example: Qudit-to-qubit mappings B. This randomises the qudit error to which each qubit error corresponds

Spacetime Error Mixing (3 of 3)

- Using only the knowledge of aggregate statistics, we are able to calculate upper bounds that look like $\mathbb{E}_B\left\{\mathbb{P}\left[\text{Failure Mode}\right]\right\} \leq V$
- We can combine these calculations using a union bound and linearity of expectation:

$$\begin{aligned}\mathbb{E}_B\left\{\mathbb{P}\left[\text{Failure Mode 1 or Failure Mode 2}\right]\right\} &\leq \mathbb{E}_B\left\{\mathbb{P}\left[\text{Failure Mode 1}\right]\right\} + \mathbb{E}_B\left\{\mathbb{P}\left[\text{Failure Mode 2}\right]\right\} \\ &\leq V_1 + V_2\end{aligned}$$

- You end up with some statement like $\mathbb{E}_B\left\{\mathbb{P}\left[\text{Outer Code Failure}\right]\right\} \leq V_{\text{Tot}}$
- Regardless of what the true error distribution is, there must exist some B making the failure probability at most V_{Tot} (in practice, you would take B random and simulate)

Operating the Outer Code Fault Tolerantly: Galois Qudit Shor Error Correction

- Operations are noisy — need FT syndrome extraction for outer codes
- Shor error correction is well-known for qubits
- The qubit stabiliser $X^{\otimes n}$ can be measured by preparing and checking the cat state $\frac{|0\rangle^{\otimes n} + |1\rangle^{\otimes n}}{\sqrt{2}}$ offline
- The qudit stabiliser $X^{\gamma_1} X^{\gamma_2} \dots X^{\gamma_n}$ can be measured by preparing and checking the qudit cat state

$$\mathrm{Cat}(\gamma_1, \gamma_2, \dots, \gamma_n) := \frac{1}{\sqrt{q}} \sum_{\eta \in \mathbb{F}_q} |\gamma_1 \eta\rangle |\gamma_2 \eta\rangle \dots |\gamma_n \eta\rangle$$

Lightweight Fault-Tolerant Syndrome Extraction

- Traditionally, in order to protect against measurement errors, we would repeat each syndrome extraction ~d times

- The distance d RS code has d-1 (qudit) X checks and d-1 (qudit) Z checks
- Naively, you would have to extract each of these ~d times
- We consider d ~ 3 to 9 → ~d^2 X and Z checks is very slow and creates a lot of noise

- Because of effects unique to qudits, we show how one can do perform much lower-overhead fault-tolerant syndrome extraction
- We measure d - 1 + M X and Z checks, where M = 1 or 2

Time-Like Reed-Solomon Codes

- Suppose the X checks of our code are written in the rows of a parity-check matrix $H_X \in \mathbb{F}_q^{(d-1)\times n}$. Consider a full-rank matrix $\beta^{(X)} \in \mathbb{F}_q^{M\times(d-1)}$
- Start by measuring the $d-1$ ``regular'' X checks given by the rows of $H_X$. Then measure the $M$ ``overcomplete'' checks given by the rows of $\beta^{(X)} \cdot H_X$
- Suppose you measure $y \in \mathbb{F}_q^{d-1}$and $z \in \mathbb{F}_q^M$. We check that we have $z = \beta^{(X)} \cdot y$. If so, accept y as the syndrome, if not, re-attempt X-syndrome extraction
- We then consider the classical code with parity-check matrix

$$H_{X,\text{time}} = \begin{pmatrix} \beta^{(X)} & I_M \end{pmatrix} \in \mathbb{F}_q^{M\times(d-1+M)}$$

We choose this to be a classical (``time-like'') Reed-Solomon code (of distance M+1 and length d-1+M)

Qudit Fault Tolerance: A Different Theory

- Our scheme performs well, despite being ``0-fault-tolerant'' in the strict sense

$$\mathbb{P}\left[\text{Syndrome Extraction Fails to One Error}\right] \sim \frac{n(d-1-M)}{(q-1)^M}\, p$$

Inner code failure probability

- We need a notion of spacetime error mixing for our fault-tolerant syndrome extraction, again to make calculations feasible.
- Here, we use a different mixing parameter $\nu \in \left(\mathbb{F}_q^*\right)^{d-1}$ which changes the time-like code as $H_{X,\text{time}} = \begin{pmatrix} \beta^{(X)} & I_M \end{pmatrix} \mapsto \begin{pmatrix} \beta^{(X)} * \nu & I_M \end{pmatrix}$ where $\beta^{(X)} * \nu$ is the matrix $\beta^{(X)}$ with its columns multiplied by the entries of ν

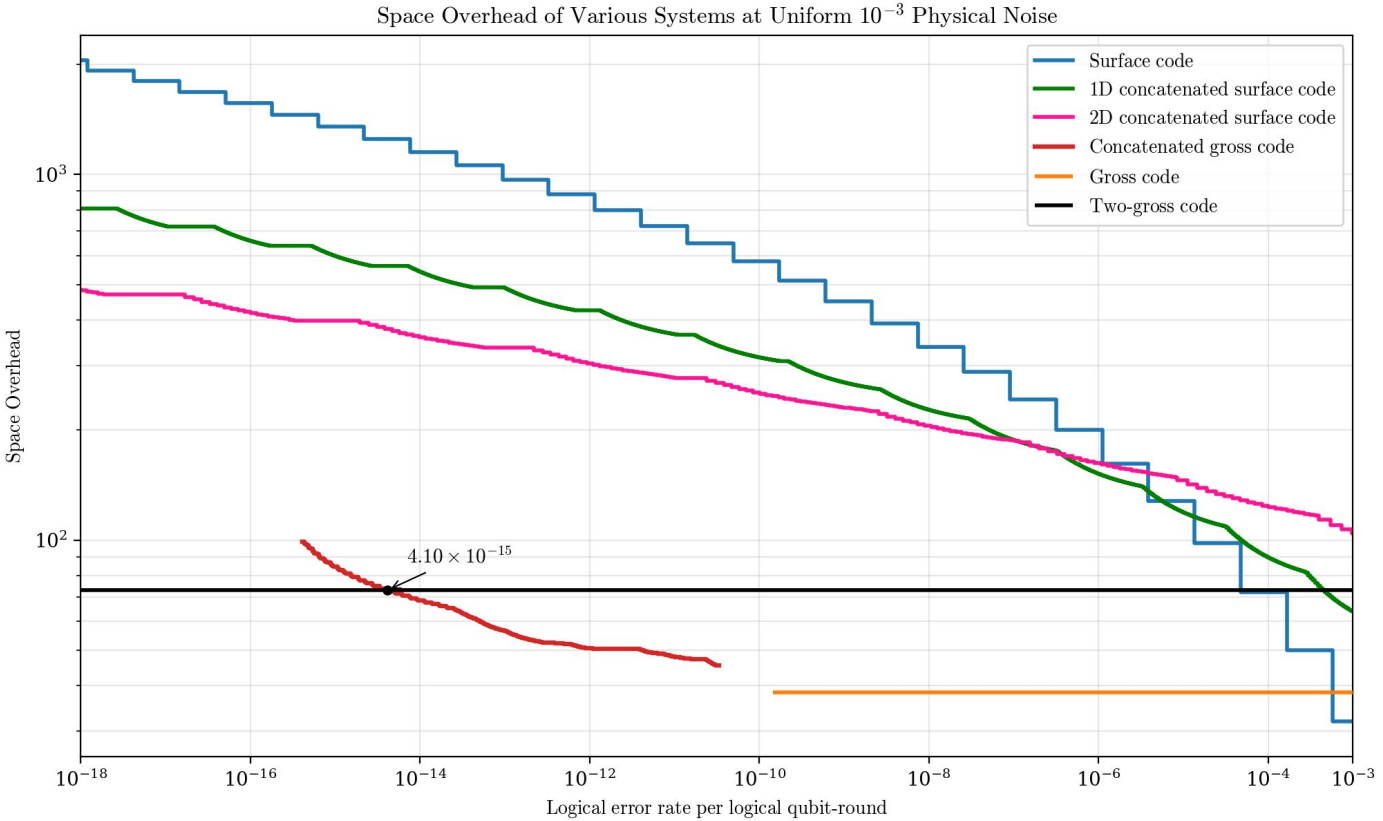
Just as for codes, building fault tolerance schemes for qudits is likely to offer genuinely new effects, motivating the development of theory that is unexplored, even classically

What did I Miss?

- There are many pieces to the construction — I don’t have time to go into detail on all here
- Significant improvements to inner code logical instruction performance via belief propagation-based decoder tuning
- Faster compilation with surgery gadgets to speed up cat state compilation
- High-performance and hardware-friendly post-selection strategies for rejecting suspicious cat states (see companion paper arXiv:2605.20346)

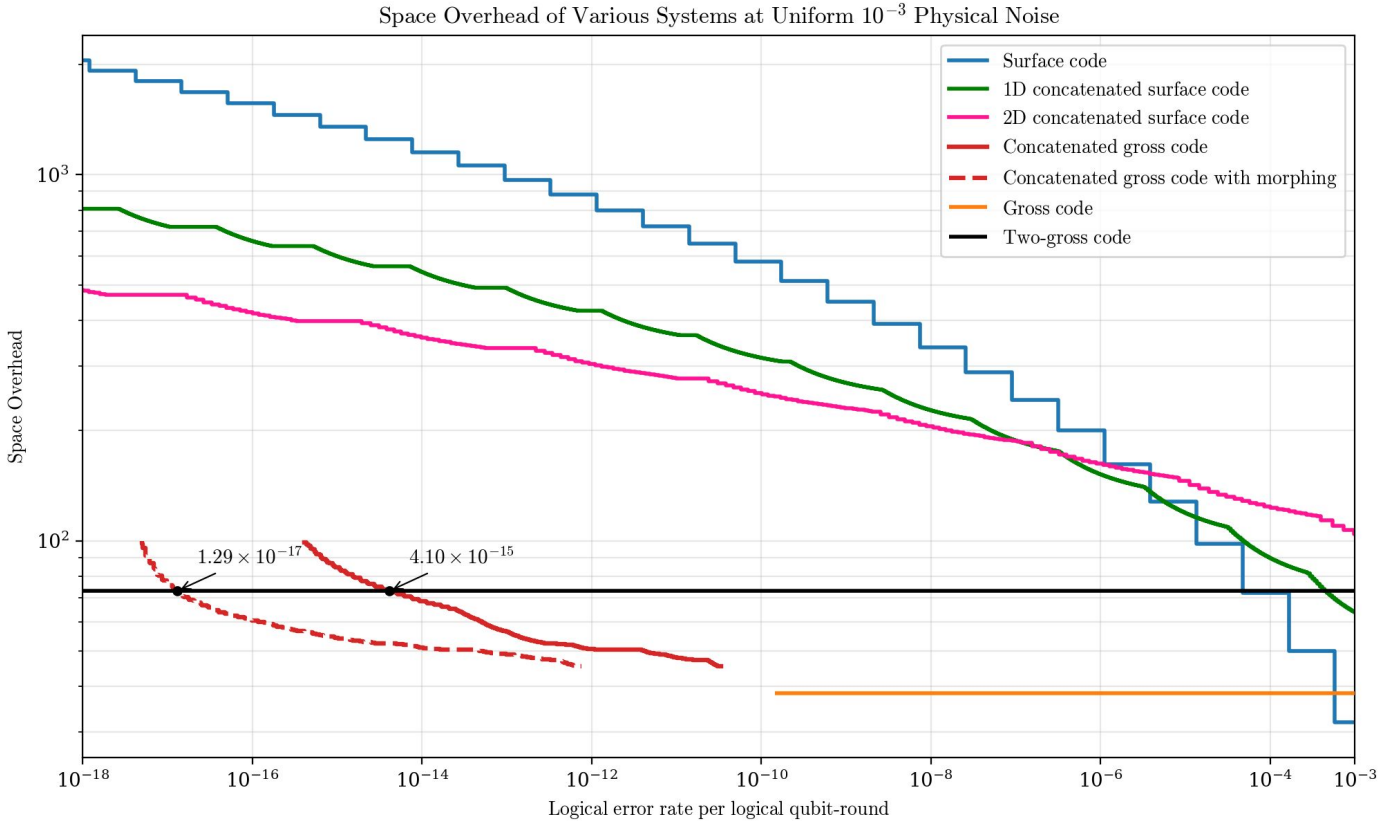
Results of Case Study on the Gross Code

(*Picking optimal system with up to 500,000 physical qubits)



New idling circuit constructed using circuit morphing; see Shaw-Terhal `25

One Future Improvement: Morphing Circuits



More Future Directions

- Concatenation over other high-rate architectures
- Soft information list decoding for our scheme
- List decoding for fault-tolerant circuits, not only for codes
- Computation on the concatenated system

Thank you!

Correspondence Welcome:

a_wills@mit.edu

Main Paper:

Companion Papers: