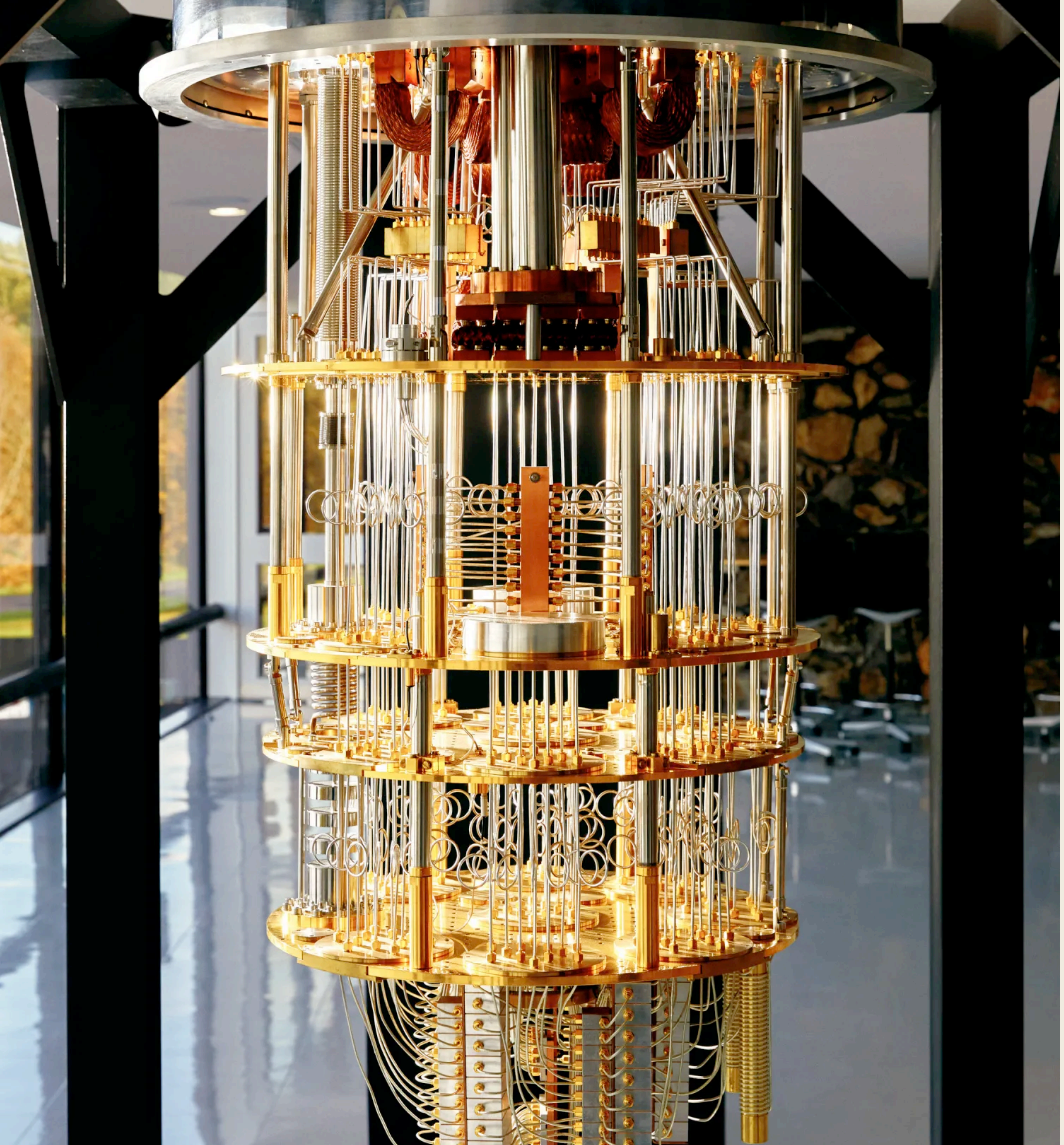
Quantum LDPC Codes

A New Frontier of Quantum Error Correction

Zhiyang He (Sunny), March 12, at UCLA

Outline

1. The Basic Principles of QEC
2. Stabilizer Codes, CSS, and the Greatest of Them All
3. Quantum LDPC Codes, Finally!
   A. Theory in Asymptopia
   B. Towards QLDPC Hardware
4. Our Path Forward



The Basic Principles of QEC

First of all: What is QEC?

> Quantum error correction is a fundamental building block of large-scale quantum computing.

– Start of almost every single talk on QEC.

Quantum Computing Power as a Resource

Discovery

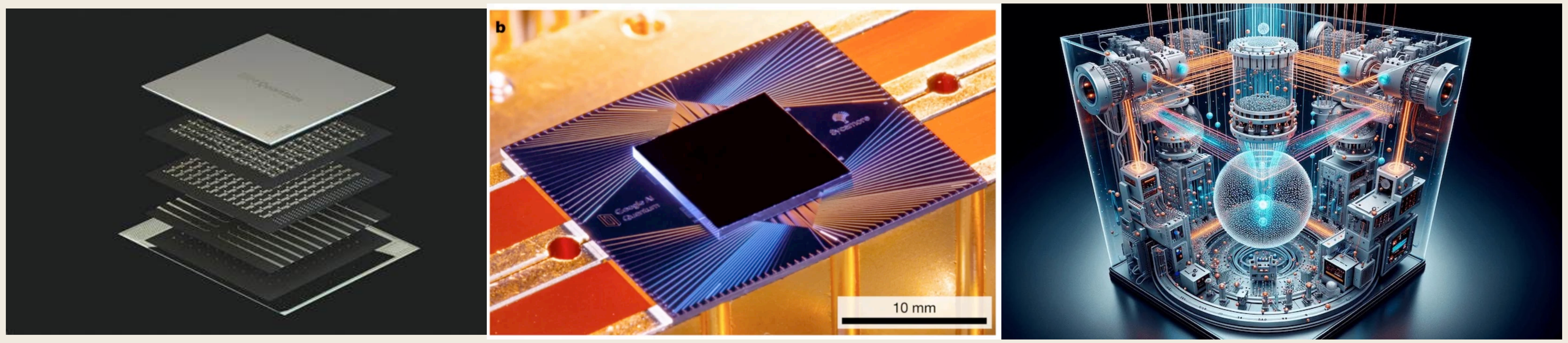
Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

Acquisition of raw resource



Left to Right: IBM Eagle, Google Sycamore, QuEra Neutral Atoms.
Many more: photonics, fermionics, ion traps...

Quantum Error Correction

noun. The procedure of processing noisy quantum computing power into logical quantum computing power.

Application

Algorithms for Quantum Computation: Discrete Logarithms and Factoring

Peter W. Shor
AT&T Bell Labs

... and three (!) decades of amazing research on Algorithms, Simulations, Cryptography, Communications, and many more.

How is it even possible?

Qubits can be lost, superposition can collapse, and we are suffering from a continuous spectrum of errors.

But, projective measurements can collapse the continuous spectrum of errors into discrete generating sets!

To protect quantum information, suffice for us to correct from this discrete set of errors.

... and our story begins here.

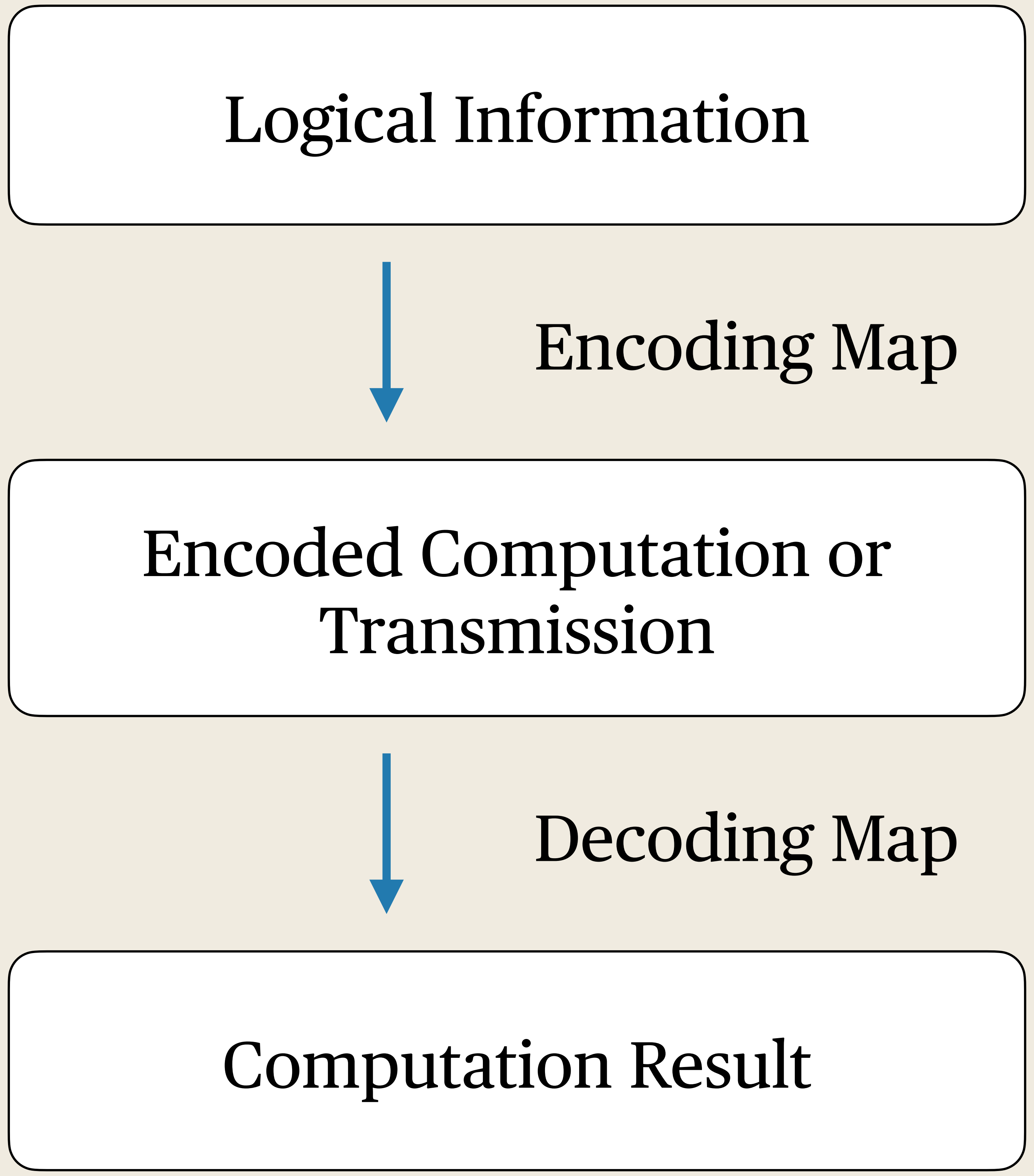
Rolf Landauer:
We have, essentially, returned to analog computers as foreseen by [Asher] Peres in remarkably perceptive comments made in 1985. Analog computers can do far more per step than digital computers, but they cannot take very many steps before the accumulated errors have derailed the computation.

Summary of his objection:
Quantum computers are essentially analog, and analog computers can't correct errors.

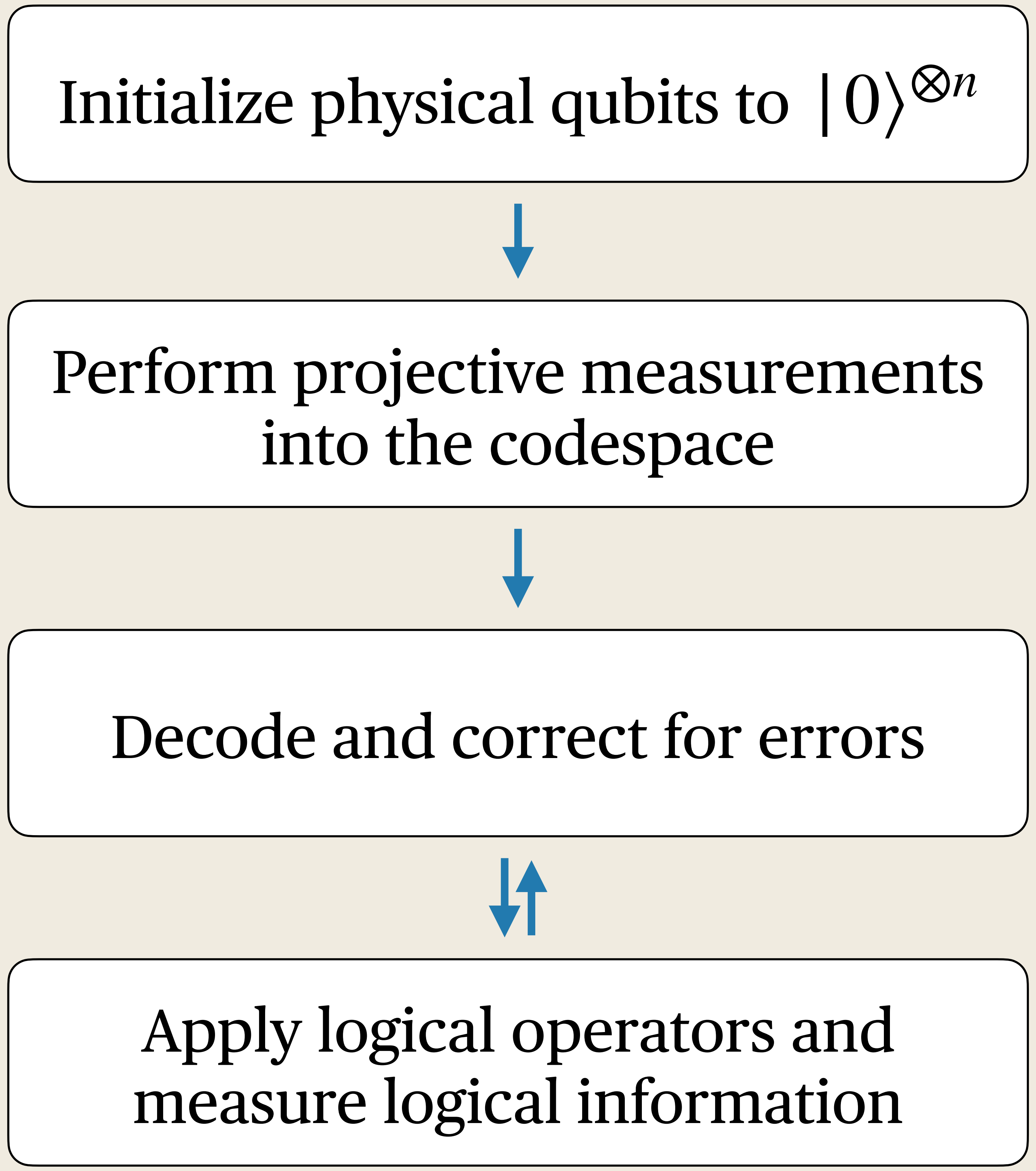
Peter's ~~My~~ rebuttal (in retrospect): Just like quantum objects are both waves and particles, quantum computers are both analog and digital.

From Peter Shor's Green Family Lecture at IPAM, UCLA. Available on Youtube.

The Procedure of QEC



Common Setting in Classical EC



Common Setting in Quantum EC

Question: Why is the encoding circuit less discussed in the quantum setting?

Challenges and wishlist of QEC

Initialize physical qubits to $|0\rangle^{\otimes n}$

↓

Perform projective measurements into the codespace

↓

Decode and correct for errors

⇅

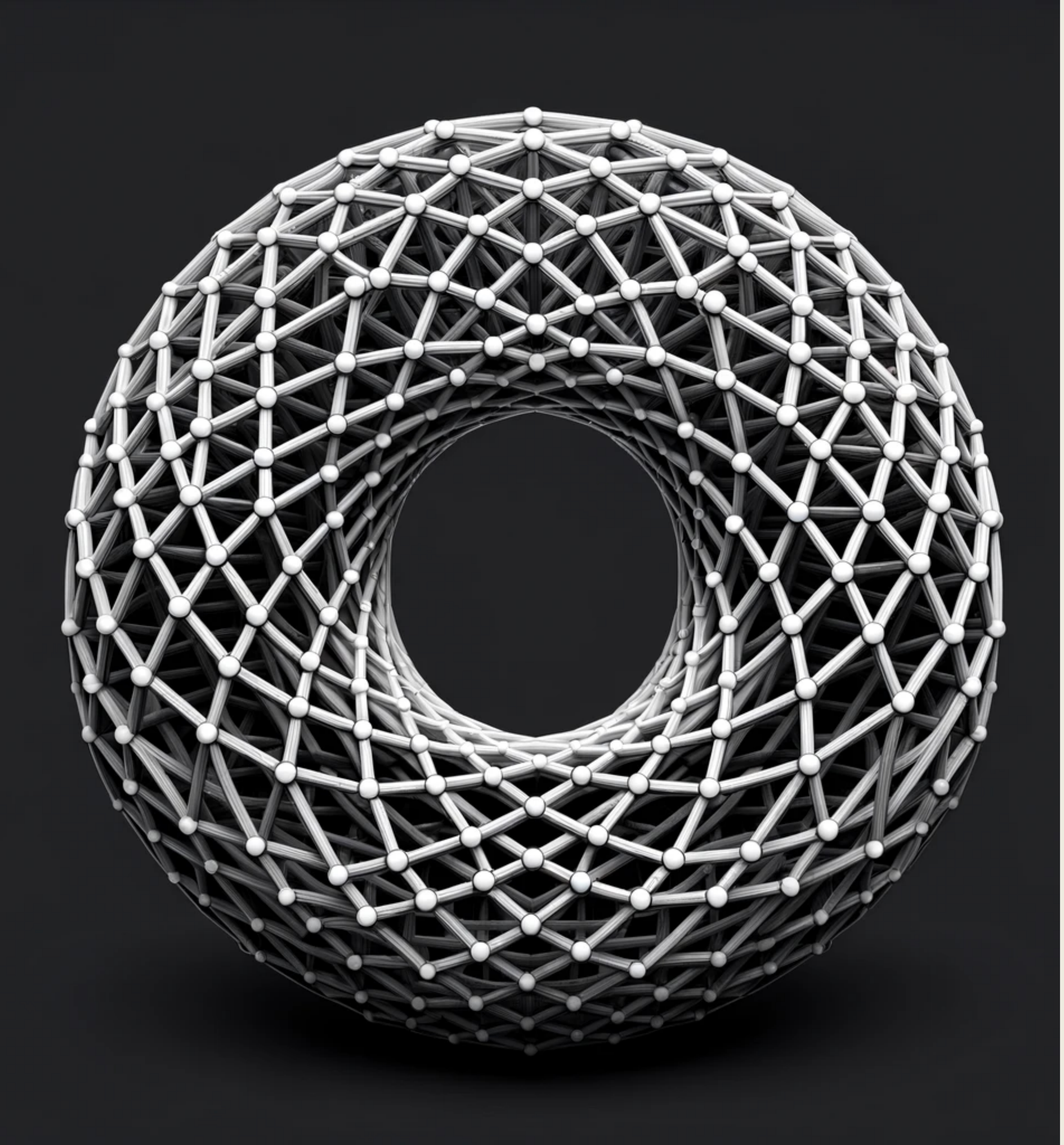
Apply logical operators and measure logical information

Challenges

1. Hardwares are difficult to build and malleate;
2. Errors can occur everywhere;
3. More qubits or more runtime are both expensive;
4. Errors may generate faster than we can decode;
5. If error accumulates, logical information could be lost;
6. Logical gates are difficult to perform without error.

Wishlist

1. All operations are hardware-friendly;
2. Every step should be fault-tolerant;
3. The space and time overheads should be minimized;
4. Efficient and high-performance decoding algorithm;
5. Low logical error rate;
6. Effective ways of implementing logical gates.



Stabilizer Codes, CSS, and the Greatest of Them All

Stabilizer Codes

Definition. Consider a commuting subgroup S of the n-qubit Pauli group G. They define a codespace as follows

$$\mathscr{C} = \{\, |\psi\rangle : s|\psi\rangle = |\psi\rangle, \forall s \in S \,\}\,.$$

1. S is called the stabilizer group, they are the projective measurements.
2. To correct from errors, we measure stabilizers s, and obtain syndromes.
3. The continuous errors collapse into Pauli errors.
4. We run a decoding algorithm (often classical) to find and apply Pauli corrections.
5. If S has $(n-k)$ independent generators (as a group), the codespace $\mathscr{C}$ encodes k logical qubits.

Initialize physical qubits to $|0\rangle^{\otimes n}$

$\downarrow$

Perform projective measurements into the codespace

$\downarrow$

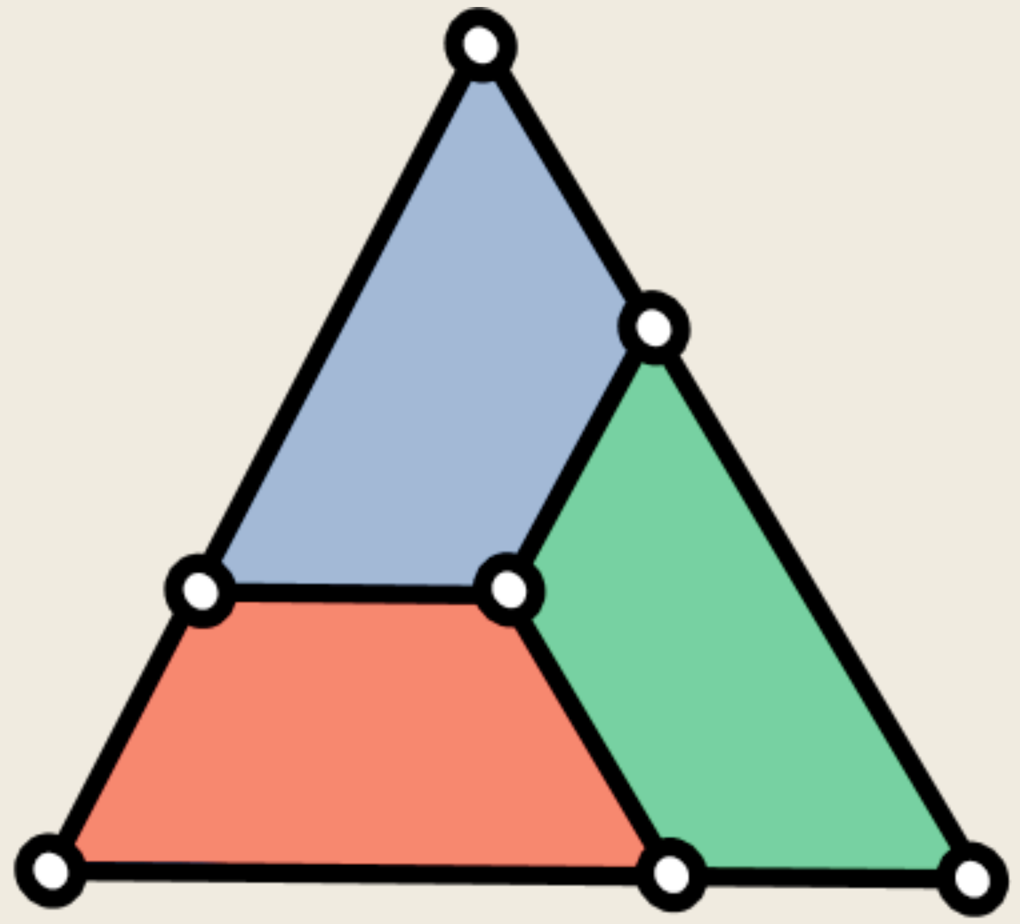
Decode and correct for errors

$\downarrow\uparrow$

Apply logical operators and measure logical information

Stabilizer Codes, Continued...

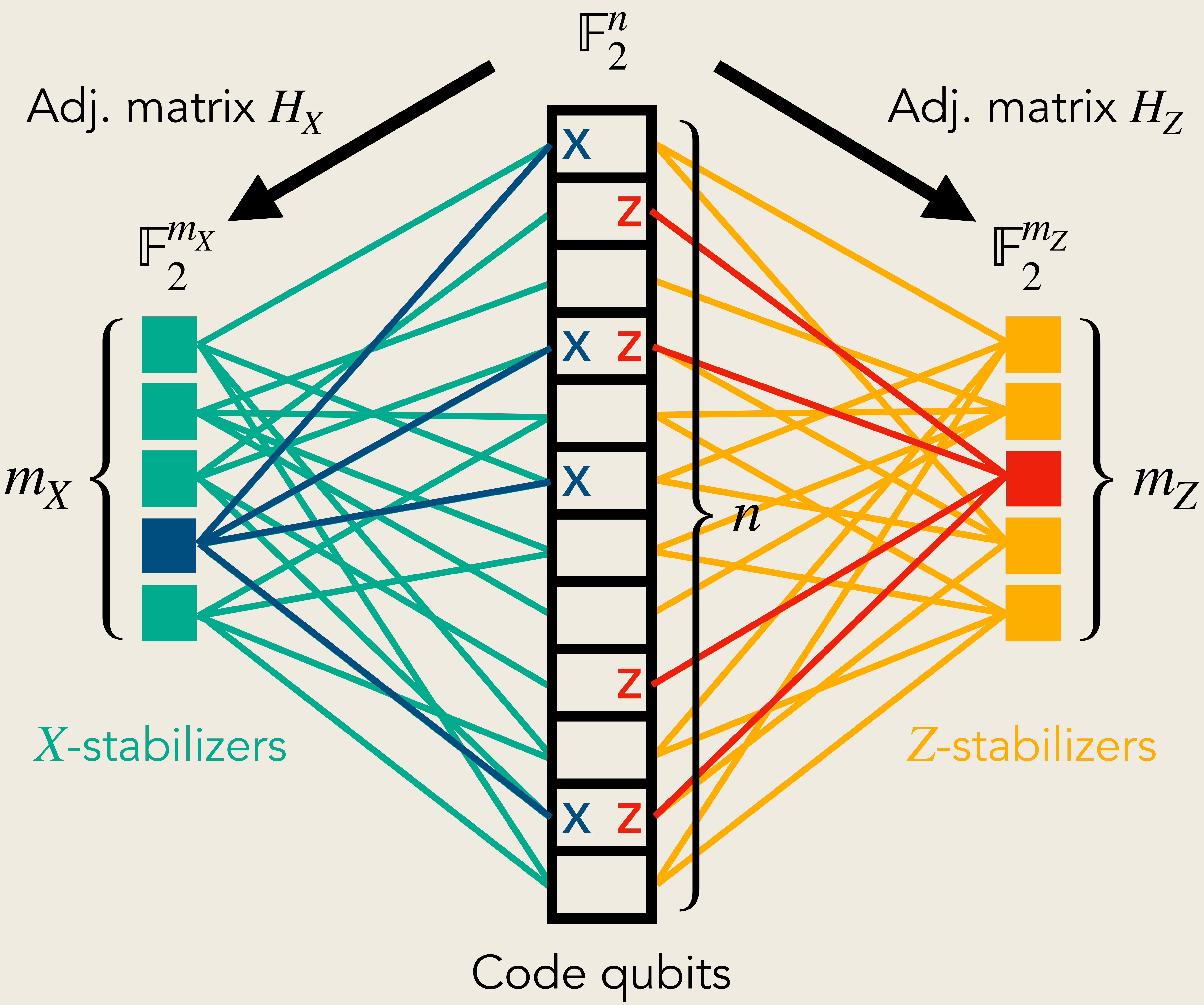
1. How can we perform encoded/logical computation? We want operations that:
   a. triggers no syndrome, so they preserve the codespace $\mathscr{C}$;
   b. is not in S, so they map logical states to logical states.
2. These operators are called logical Pauli operators.
3. A code has 2k independent logical Pauli operators, which acts as logical X or Z on the k logical qubits.
4. Question: Which state in $\mathscr{C}$ is the logical $|0\rangle^{\otimes k}$ state now?
5. If all logical Pauli operators has weight at least d, we say that the code has distance d.
6. What about gates beyond Pauli?



S_1	X	X	X	X	X	X	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_2	X	X	X	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	X	X	X
S_3	Z	Z	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_4	$\mathbb{1}$	Z	Z	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_5	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z	Z	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_6	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z	Z	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_7	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z	Z	$\mathbb{1}$
S_8	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z	Z
$\overline{Z}$	Z	Z	Z	Z	Z	Z	Z	Z	Z
$\overline{X}$	X	X	X	X	X	X	X	X	X

CSS Codes

– 90% of stabilizer codes we study.



Suppose we restrict our stabilizers, so that they can only be all X or all Z.

We can group the X-checks and Z-checks into two separate adjacency matrices.

Question: What are the dimensions of H_X and H_Z?

Stabilizers commute: $H_Z H_X^\top = 0$

Classical codes: $C_X = \ker H_X$, $C_Z = \ker H_Z$

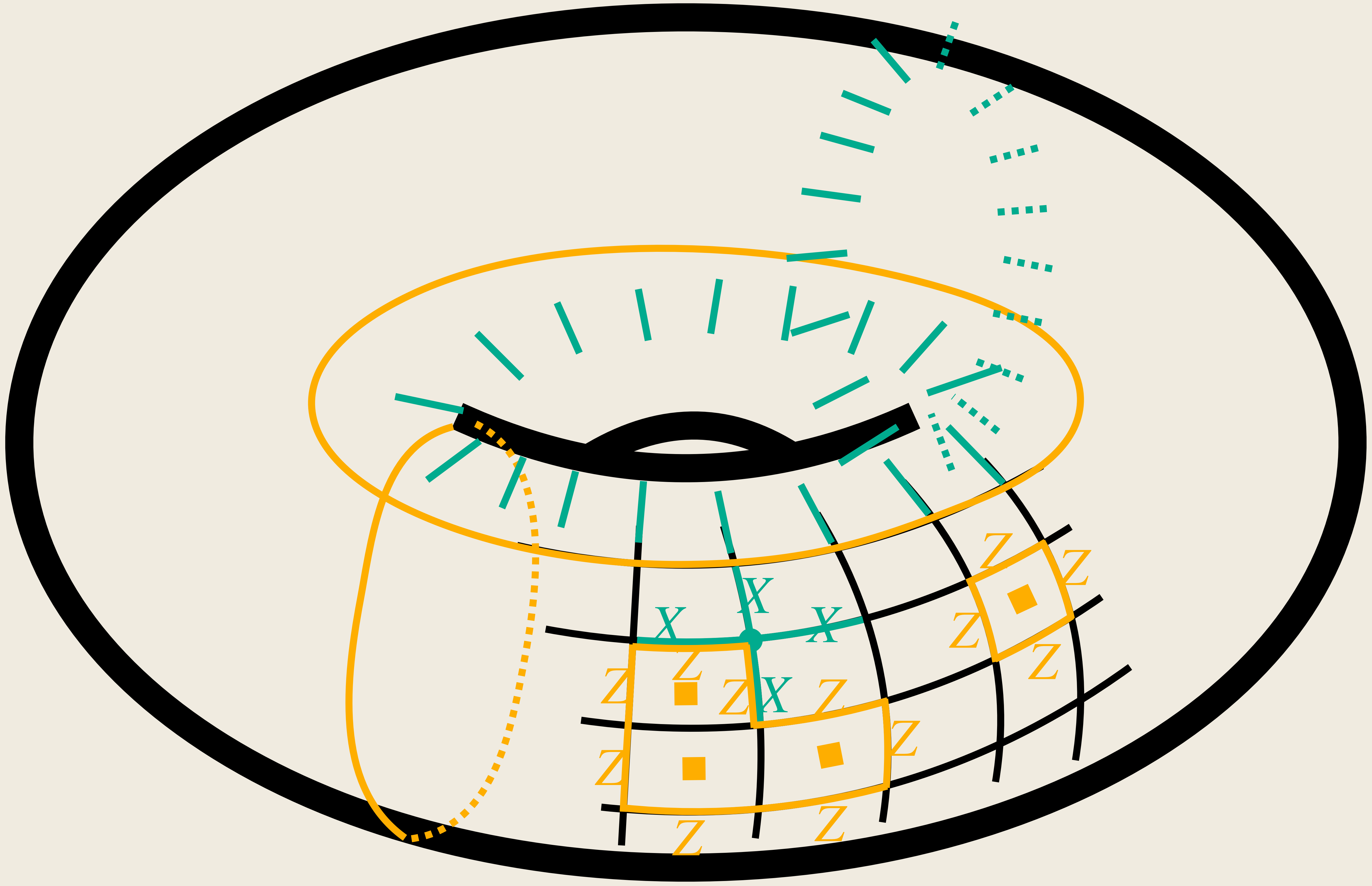
This formalism enables us to bring decades of classical coding theory knowledge into QEC!

Named after their inventors: Calderbank, Shor, and Steane. I once had to explain CSS codes in front of Peter Shor.*

* He fell asleep.

Toric code

– the greatest code of them all.



Edges = qubits

Commute! $\begin{cases} \text{Vertices} = X\text{-stabilizers} \\ \text{Squares} = Z\text{-stabilizers} \end{cases}$

Logical Z operator:

Commutes with all X-stabilizers... (cycle)

but is not a product of Z-stabilizers

(not a boundary)

Logical X operator:

Commutes with all Z-stabilizers

but is not a product of X-stabilizers

N-qubit Toric code:

Dimension $k = 2$, Distance $d = \Theta(\sqrt{N})$

Why is Toric Code the Greatest?

- For Practical QEC: Toric code admits many desirable properties (next slide!). Decades of academic and industrial research have been dedicated to building quantum architectures based on surface codes (later!).*
- For Theoretical QEC: Toric code can be studied using tools from:
  - Mathematics: Algebraic topology (later!)
  - Physics: Condensed matter physics (2D Ising Model), led to topological QEC.
  - Computer Science: Graph algorithms and optimizations.
- For Hamiltonian complexity: Toric code serves as a great model to instantiate theories on.
- And many more...

The Destined Toric Theorem (Shor, He)

Given any conversation on QEC parametrized by time t, as $t \to \infty$, the conversation will eventually talk about the toric code.

* QEC was solved in 2001! – Researchers in QEC, after getting drunk.

Why is Toric Code the Greatest

... from a practical perspective

... so far?

Toric/Surface Code

1. All checks are spatially local on a 2D grid.
2. Following decades of research, many engineering and theoretical techniques to control noise in surface codes.
3. Toric code achieves the best parameters, assuming hardware is 2D and connections are local. (Later!)
4. The Minimum-Weight Perfect Matching decoder works quite well in practice, highly optimized.
5. Threshold: By using larger surface codes, logical error rate will decrease (*kind of* shown by experiments).
6. Many proposals for fault-tolerant gates, notably lattice surgery.

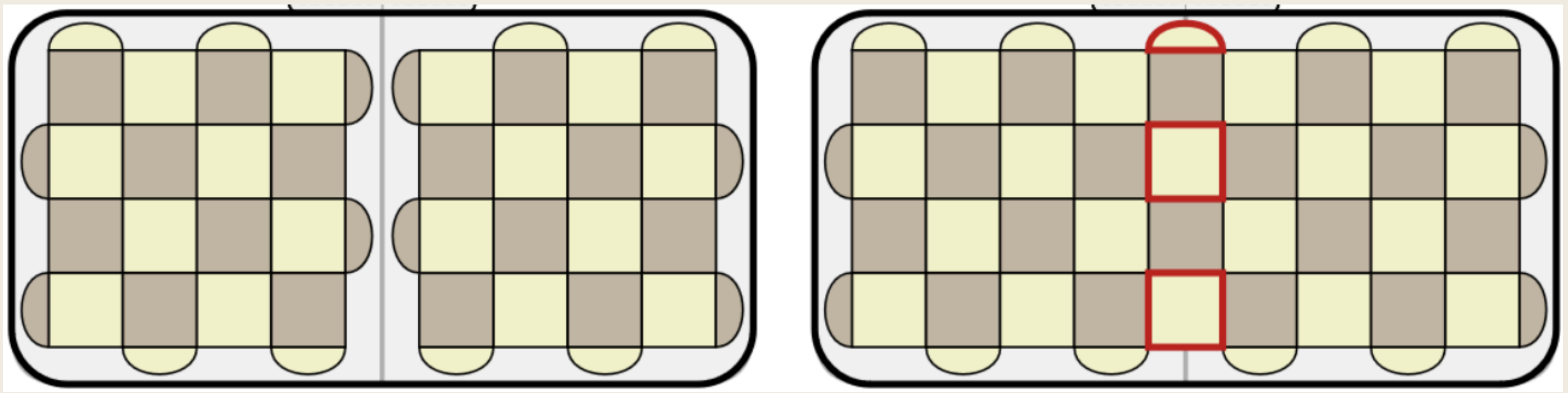
Our Wishlist

1. All operations are hardware-friendly;
2. Every step should be fault-tolerant;
3. The space and time overheads should be minimized;
4. Efficient and high-performance decoding algorithm;
5. Low logical error rate;
6. Effective ways of implementing logical gates.

Question: If Toric code is so great… Why are we here again?

The Daunting Space Overhead

To perform logical measurements or gates on logical qubits, we often need to use a technique called lattice surgery.



Daniel Litinski, A Game of Surface Codes: Large-Scale Quantum Computing with Lattice Surgery, 2019

"Assuming a physical error rate of 10^{-4} and a code cycle time of 1 μs, a classically intractable 100-qubit quantum computation with a T count of 10^8 and a T depth of 10^6 can be executed in 4 hours using 55,000 qubits, in 22 minutes using 120,000 qubits, or in 1 second using 330,000,000 qubits."

Craig Gidney & Martin Ekerå, How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, 2021

Fundamental Limitation

Theorem. The BPT (Barvyi-Poulin-Terhal) Bound[1]

For a quantum code, if its qubits can be placed in 2D such that all checks are spatially local, then its parameters must satisfy:

$$O(n) \geq k \cdot d^2$$

This theorem revealed the path beyond surface codes, and marked a fundamental bottleneck.

To get better parameters (both practically and theoretically), our checks must be non-local, i.e., reaching beyond nearest neighbors;

However, it is very difficult to build hardwares with non-local connectivity.

Question: What if we give up some distance?

[1] Tradeoffs for reliable quantum information storage in 2D systems

The Era of NISQs

– Noisy Intermediate Scale Quantum Devices

We are now building quantum computers on tens to hundreds of physical qubits,

through many different physical hardware, codes, and architectures.

Article | Open access | Published: 22 February 2023

Suppressing quantum errors by scaling a surface code logical qubit

Google Quant

Nature 614,

103k Access

IonQ Achieves New Performance Milestone of 29 Algorithmic Qubits (#AQ) on

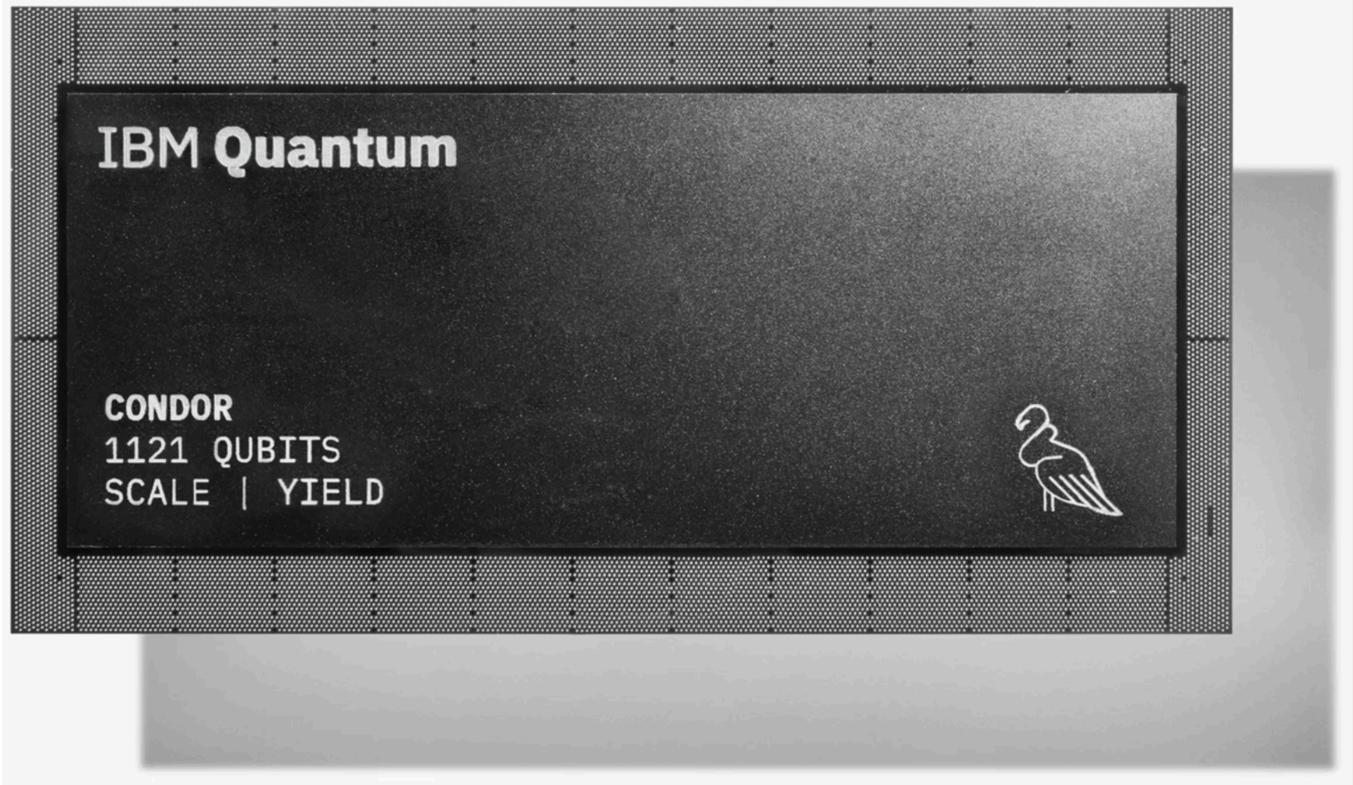
Article | Open access | Published: 06 December 2023

Logical quantum processor based on reconfigurable atom arrays

Dolev Bluvstein, Simon J. Evered, Alexandra A. Geim, Sophie H. Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, J. Pablo Bonilla Ataides, Nishad Maskara, Iris Cong, Xun Gao, Pedro Sales Rodriguez, Thomas Karolyshyn, Giulia Semeghini, Michael J. Gullans, Markus Greiner, Vladan Vuletić & Mikhail D. Lukin

Nature **626**, 58–65 (2024) | Cite this article

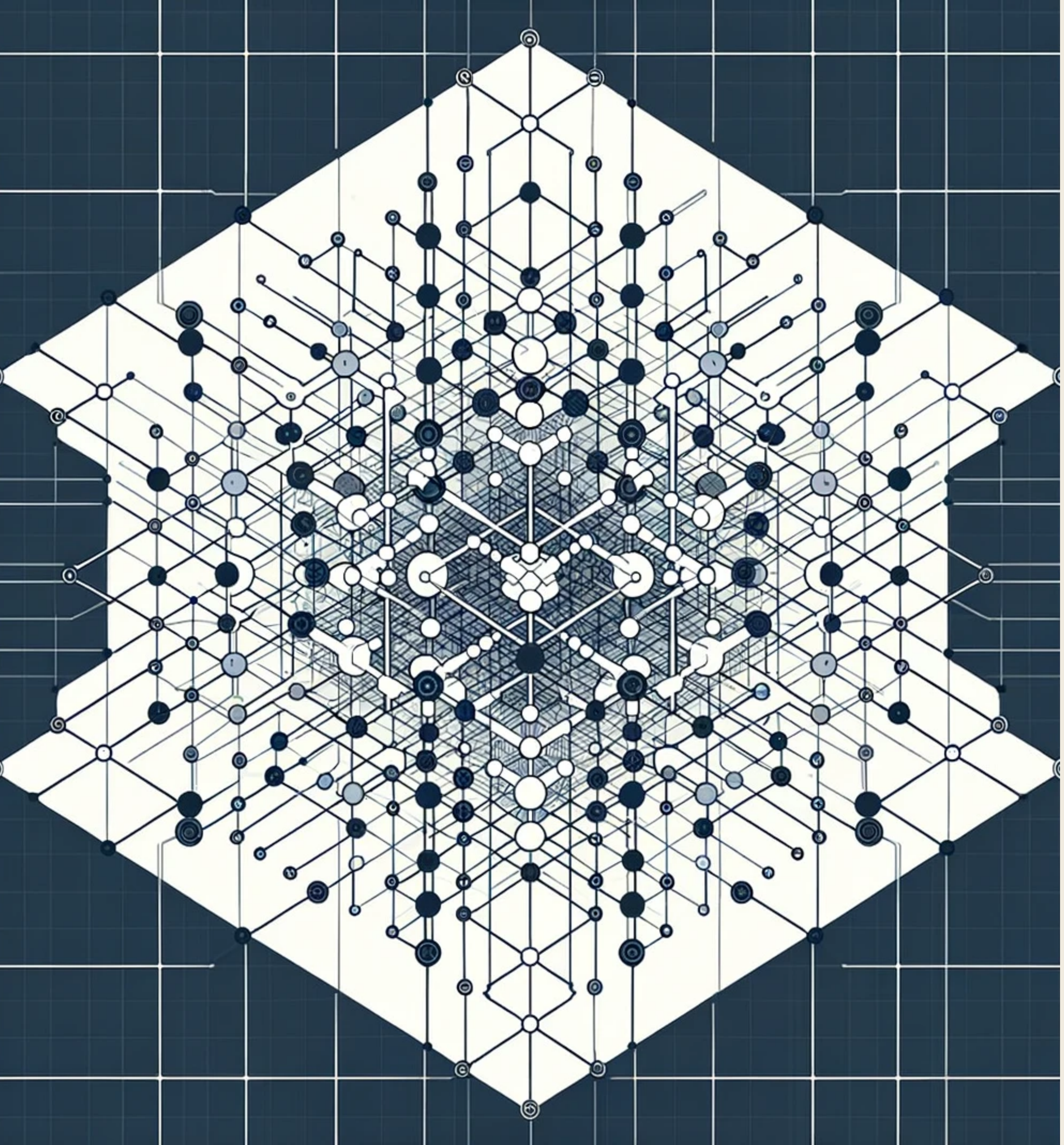
84k Accesses | **7** Citations | **866** Altmetric | Metrics



With decades of hardware developments, we arrived at a turning point

where we can finally imagine building non-local connections.

What world lies ahead?



QLDPC Codes, Finally!

Definitions

Quantum Low Density Parity Check (LDPC) Codes

A family of stabilizer code is LDPC if every check acts on a constant number of qubits, and every qubit is involved in a constant number of checks.[1]

We sometimes say that locality is constant.

There are two important measures of hardware connectivity:

1. Qubit degree: how many other qubits is a qubit connected to?
2. Connection range: after embedding qubits into space, how long is the longest connection?

The central question is, therefore:

How much connectivity do we need, and how much better parameters do we get?

[1] Technically speaking, toric code is LDPC as well.

The Landscape

Theoretical Constructions in Asymptopia

We want to construct families of codes such that as n grows, k and d grows with n while locality stays as $O(1)$. If possible, we want $k, d = \Omega(n)$.

2009 - 2021

Practical Constructions for Hardware

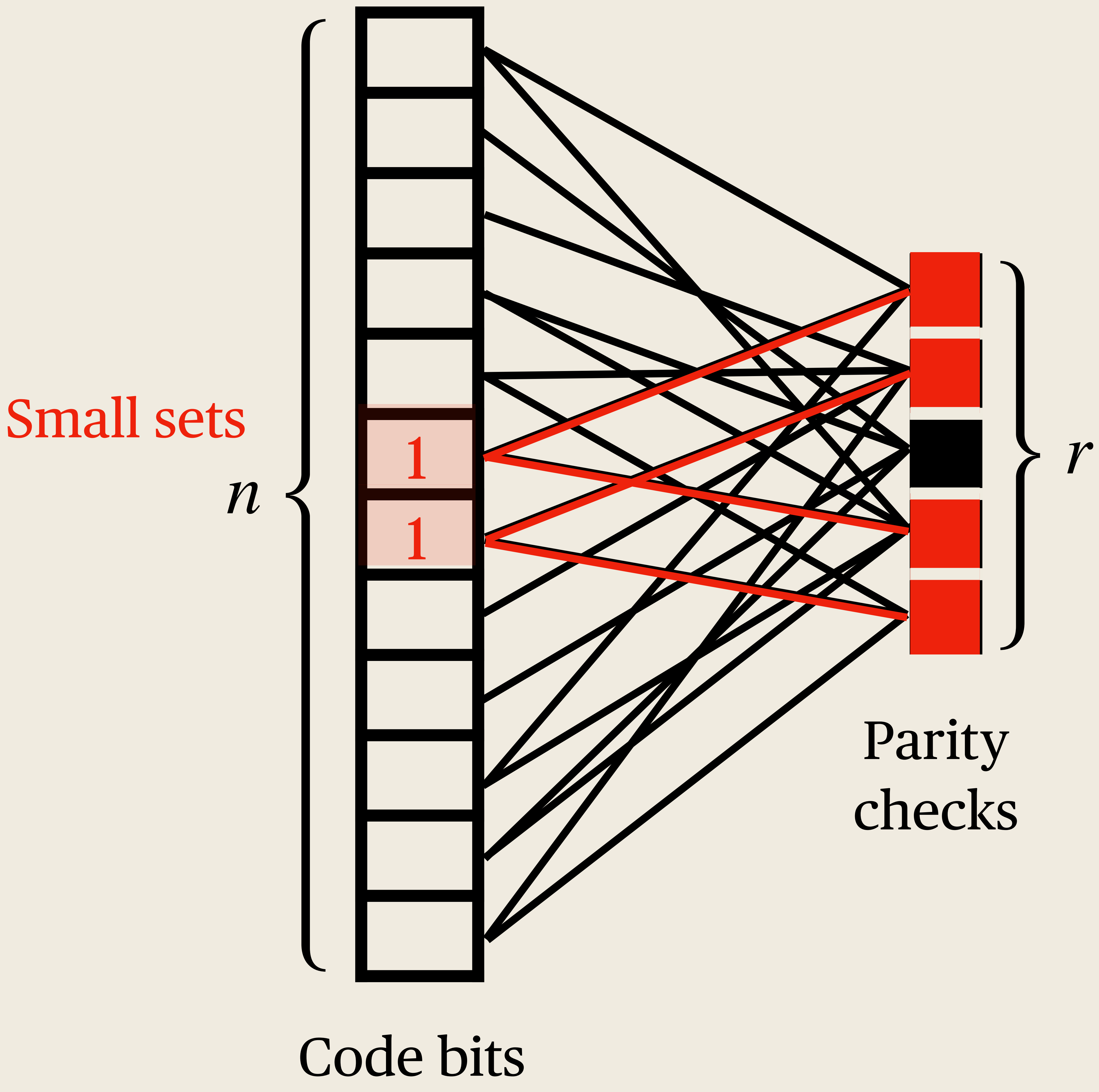
We want to construct find codes on the scale of hundreds to thousands of qubits, with parameters better than surface code and connectivity requirement as low as possible.

2019+

There are three important ingredients in most of modern theoretical and practical constructions of QLDPC codes. They come from three sources:

Classical wisdom, the Toric code, and Algebraic Topology.

Invoking Classical Wisdom



Definition. Expander graphs

A degree d bipartite graph is an expander graph if any small set S of vertices on the left has a large neighborhood on the right.

Small: $|S| \leq O(n)$ Large: $|N(S)| \geq O(d|S|)$

Classical LDPC Codes are often built with these expander graphs.

Have large neighborhoods

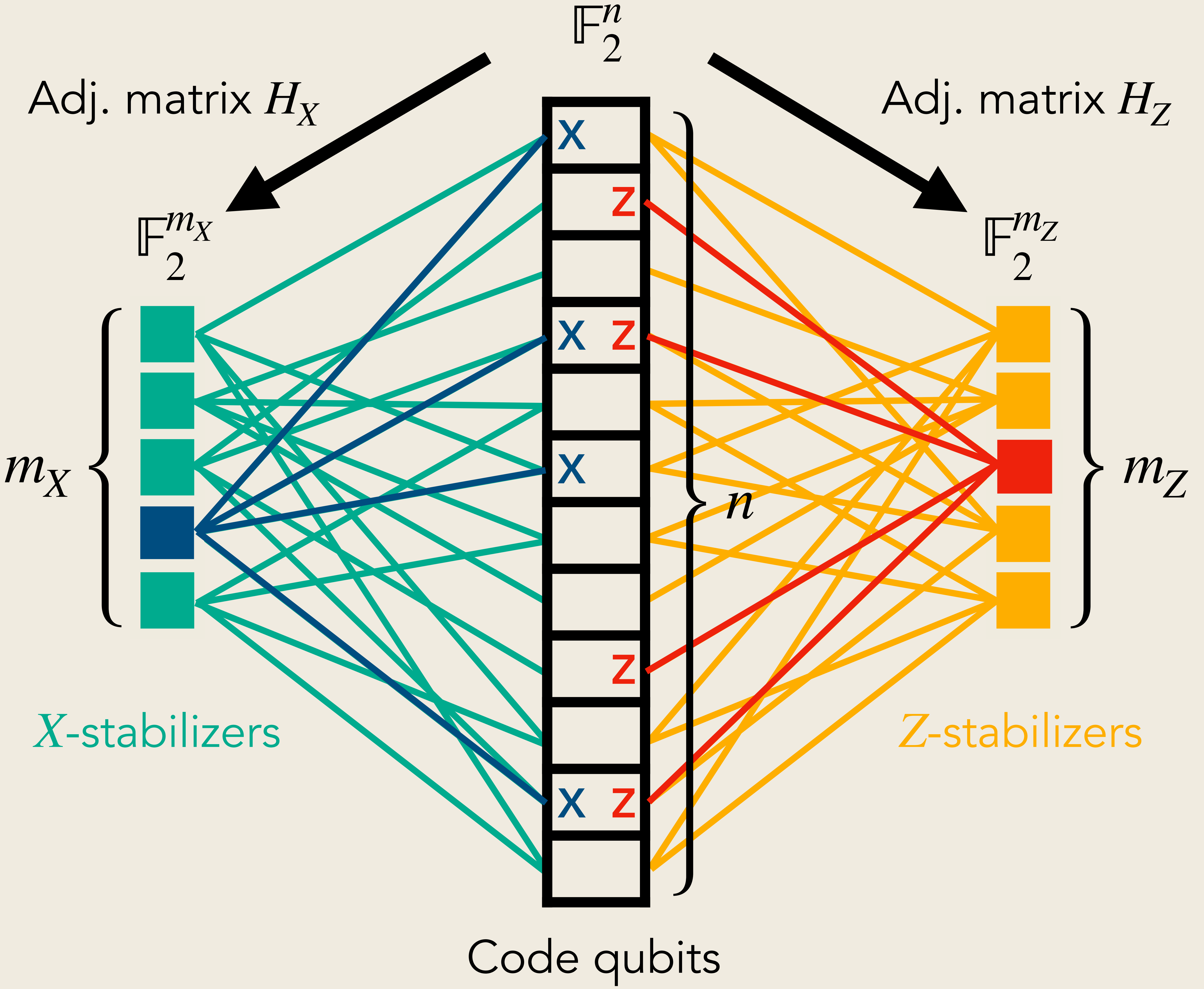
$\Downarrow$

Many violated parity checks!

Asymptotically good cLDPCs: $k, d = \Omega(n)$.

Invoking Classical Wisdom

... What now?



Question: If classical codes are so great, why don't we use two of them to build a CSS code?

Question: OK, they may not commute… what if they do?

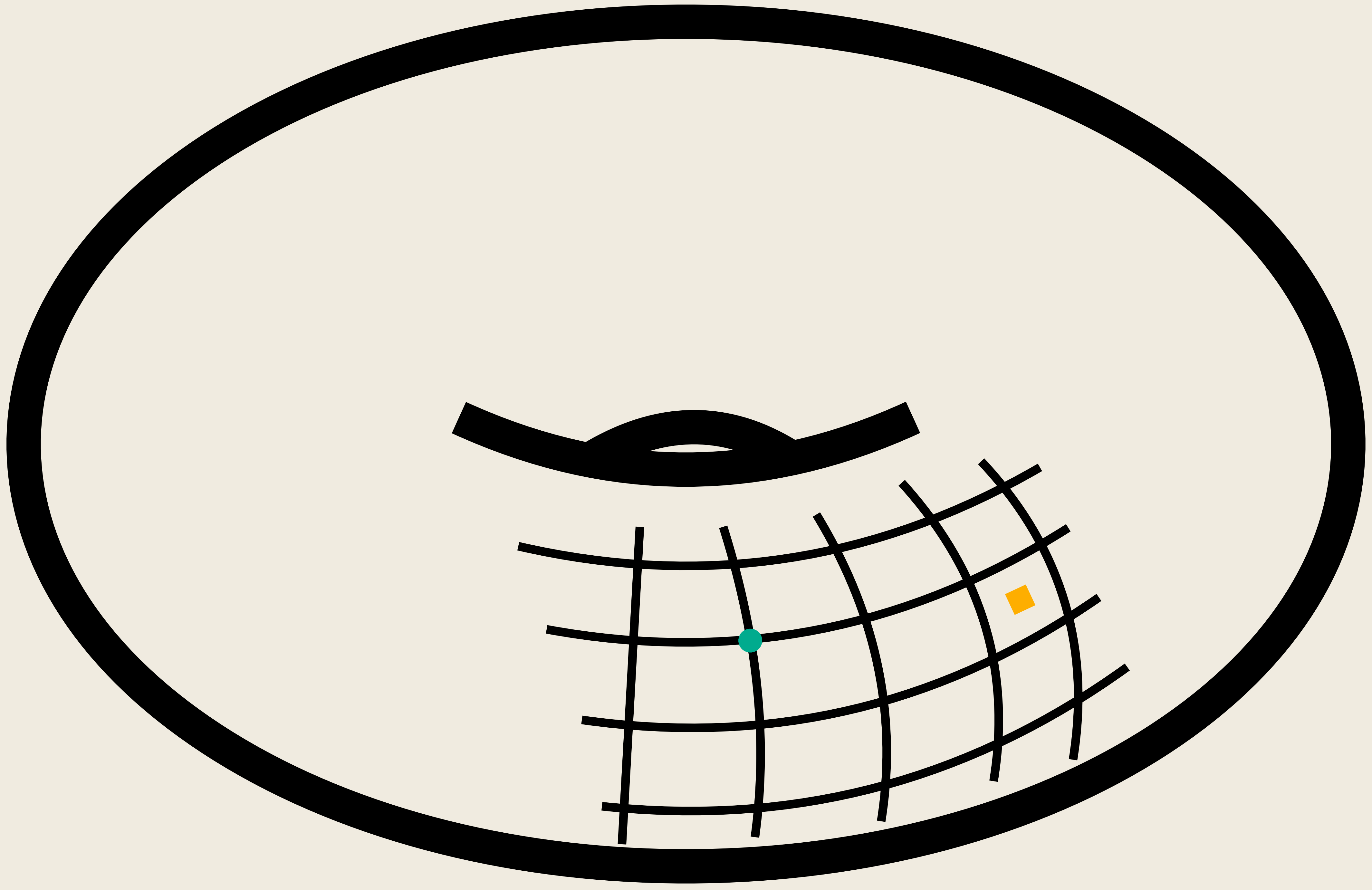
Let $z \in \mathbb{F}_2^n$ be a vector that indicates the qubits involved in a particular Z-check. Then

$$H_X z = 0.$$

What does this mean for the code C_X?

Torus in Topology

... Where is the code?



Edges = 1 Dimensional Objects

Vertices = 0 Dimensional Objects

Squares = 2 Dimensional Objects

Question: What is the boundary of a square?

Boundary maps ∂_2, ∂_1:

A 2D object is mapped to all 1D objects on its boundary;

A 1D object is mapped to all 0D objects on its boundary;

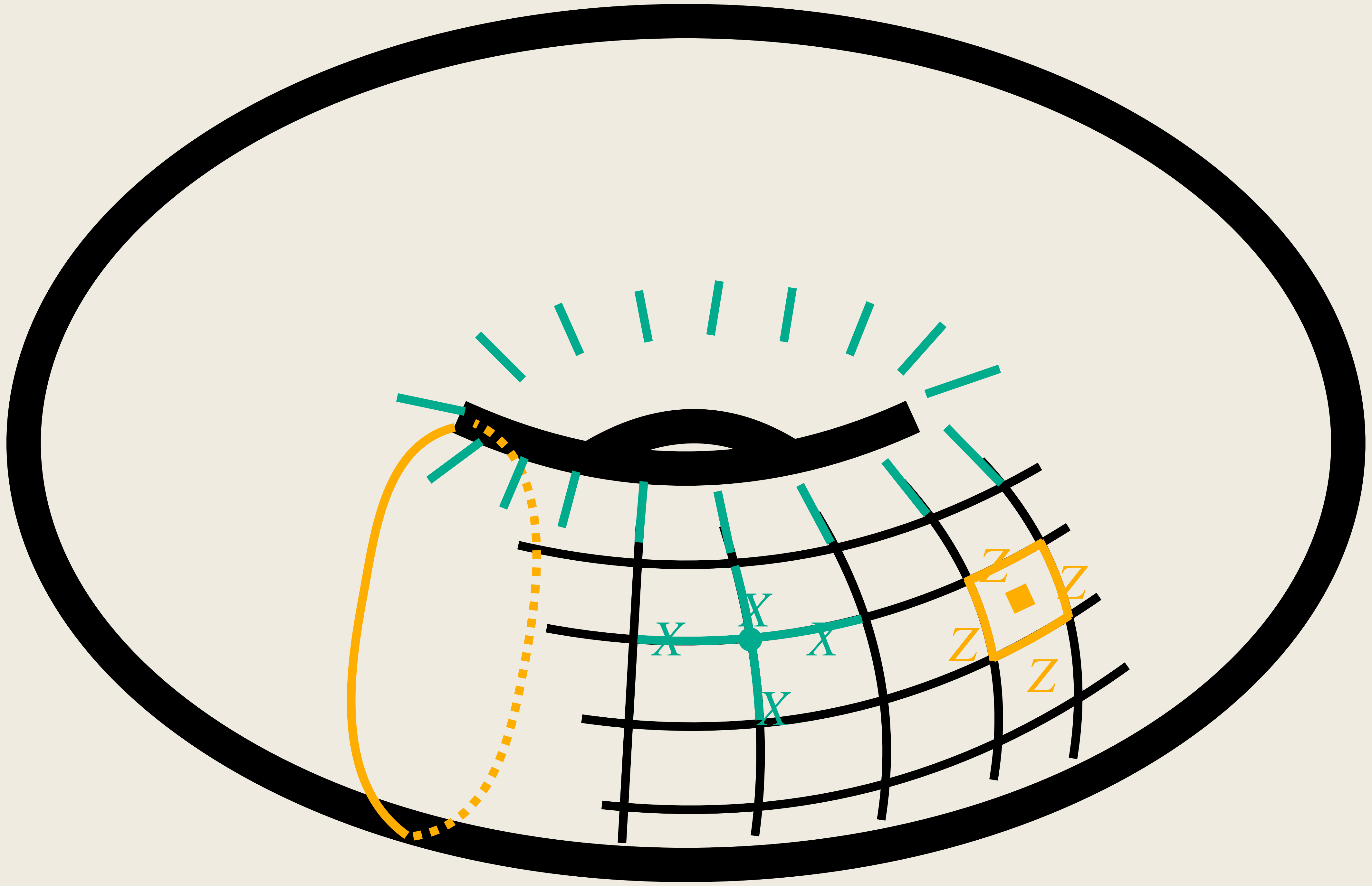
$$\text{Torus} = \text{Squares} \xrightarrow{\partial_2} \text{Edges} \xrightarrow{\partial_1} \text{Vertices}$$

Observe: Boundary of a set of squares have no boundary vertex!

In mathematical terms, $\partial_1 \partial_2 = 0$.

Where is the code?

$$\text{Toric Code} = \mathbb{F}_2^{\text{Z-checks}} \xrightarrow{\partial_2} \mathbb{F}_2^{\text{Qubits}} \xrightarrow{\partial_1} \mathbb{F}_2^{\text{X-checks}}$$



A Z-check is mapped to all qubits on its boundary;

Question: What about X-checks?

A X-check is mapped to all edges, whose boundary contains the X-check.

Logical Z operator:

Commutes with all X-stabilizers = cycle with no boundary;

but is not a product of Z-stabilizers = not a boundary.

Toric code is CSS: let $\partial_2 = H_Z^T, \partial_1 = H_X$.

$\partial_1 \partial_2 = 0 \Leftrightarrow H_X H_Z^T = 0.$

Takeaway: everything about the Toric code is described by its topology. *

* Sunny: "Qubits are holes." Peter: "Yes?" The student who worked with us: "???"

Quantum Codes and Topology

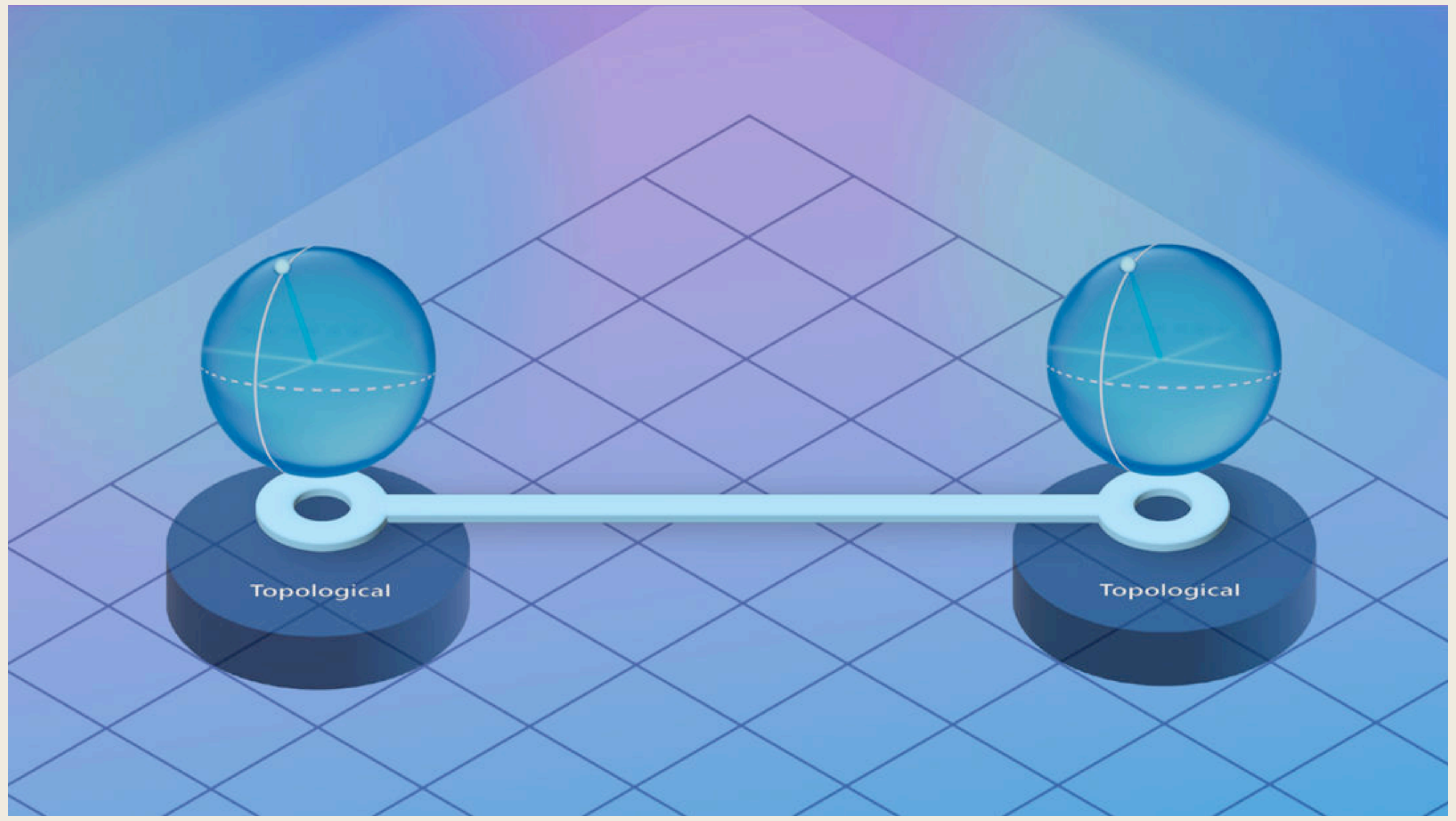
How does that help us build QLDPC codes?

On one hand, every CSS code can be understood from topology.

$$\mathrm{CSS}(H_X, H_Z) = \mathbb{F}_2^{m_Z} \xrightarrow{\partial_2 = H_Z^T} \mathbb{F}_2^{n} \xrightarrow{\partial_1 = H_X} \mathbb{F}_2^{m_X}$$

On the other hand, many topological objects can be studied as CSS codes.

This inspiration from Toric code led to the entire field of topological QEC.



March 14, 2022

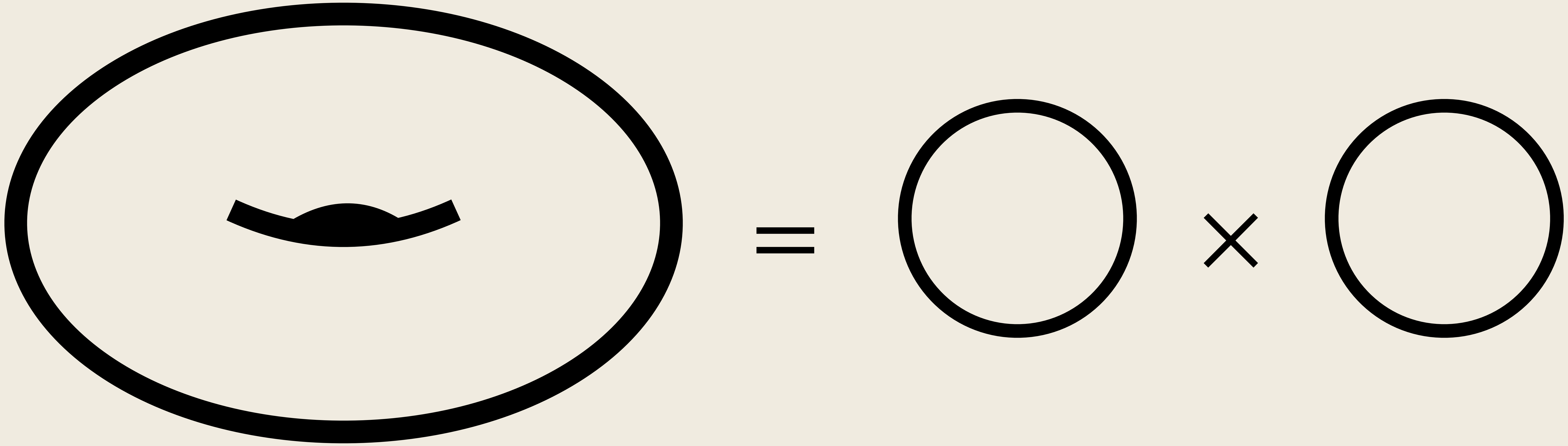
In a historic milestone, Azure Quantum demonstrates formerly elusive physics needed to build scalable topological qubits

A little bit of Algebraic Topology

The Toric Code, Again.

Question: What is the product of a circle with a circle?

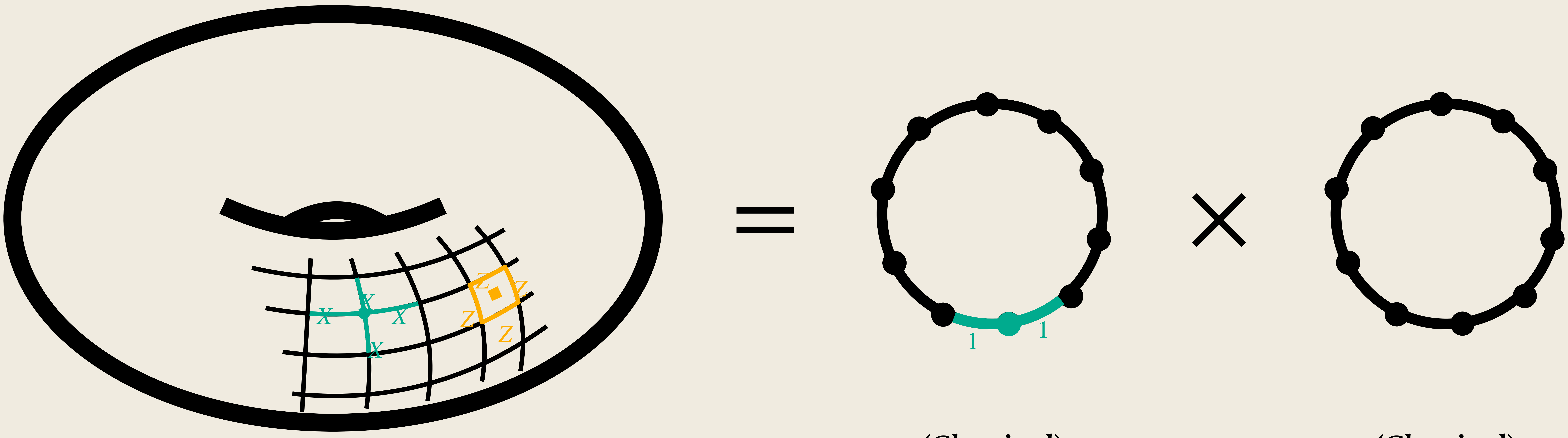
(I didn't define product for you, so just use intuition)



The Infinite Toric Theorem (Shor, He)

An infinitely long conversation on QEC will talk about the toric code an infinite number of times.

Where is the code now?



Toric code

(Classical)
Repetition code

(Classical)
Repetition code

See the board for mathematical formulation.

Quantum Hypergraph Product Code

Quantum hypergraph product code		**Classical code**		**Classical code**
$\mathscr{Q}$	$=$	$\mathscr{C}_1$	$\otimes$	$\mathscr{C}_2$
$[[\Theta(n^2), \Theta(n^2), d = \Theta(n)]]$	$\Longleftarrow$	$[n, \Theta(n), \Theta(n)]$		$[n, \Theta(n), \Theta(n)]$
Locality 2ℓ		Locality ℓ		Locality ℓ

(Notation: [block length, dimension, distance])

Tillich & Zémor, Quantum LDPC codes with positive rate and minimum distance proportional to $n^{1/2}$, 2009

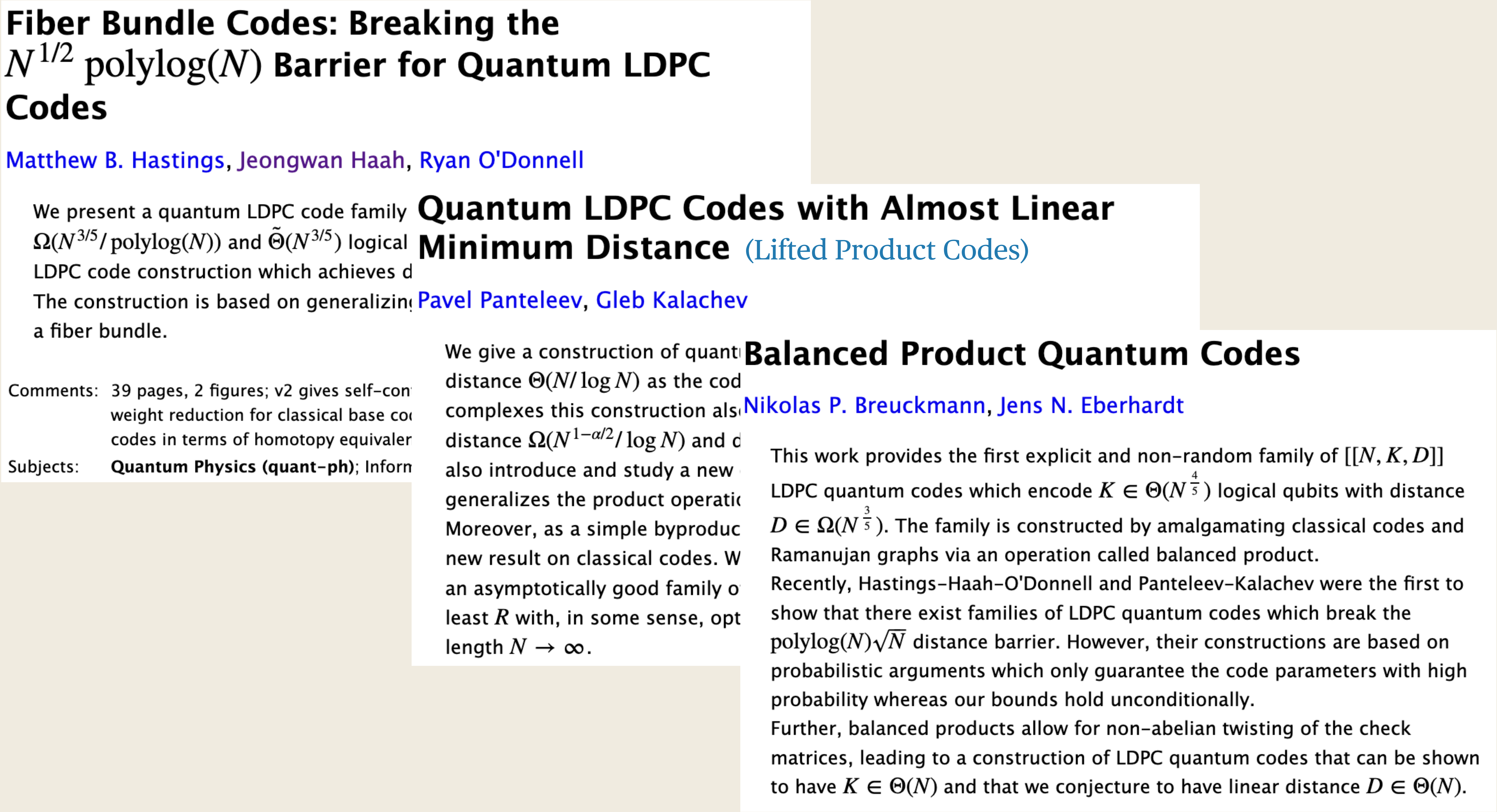
Quantum Hypergraph Product Code

Hypergraph product codes to QLDPC is the same as toric code to QEC.

- A decade of research dedicated to constructions, decoding, logical gates, and more.
- There are small instances of HP codes, such as [[356, 36, 6]] or [[832, 64, 8]][1]. Viable examples usually require a few hundred to a few thousand qubits. We'll discuss more about small codes later.
- Theoretical barrier: like Toric code, HP codes cannot have distance beyond $n^{1/2}$.
- We got stuck here for a decade, until…

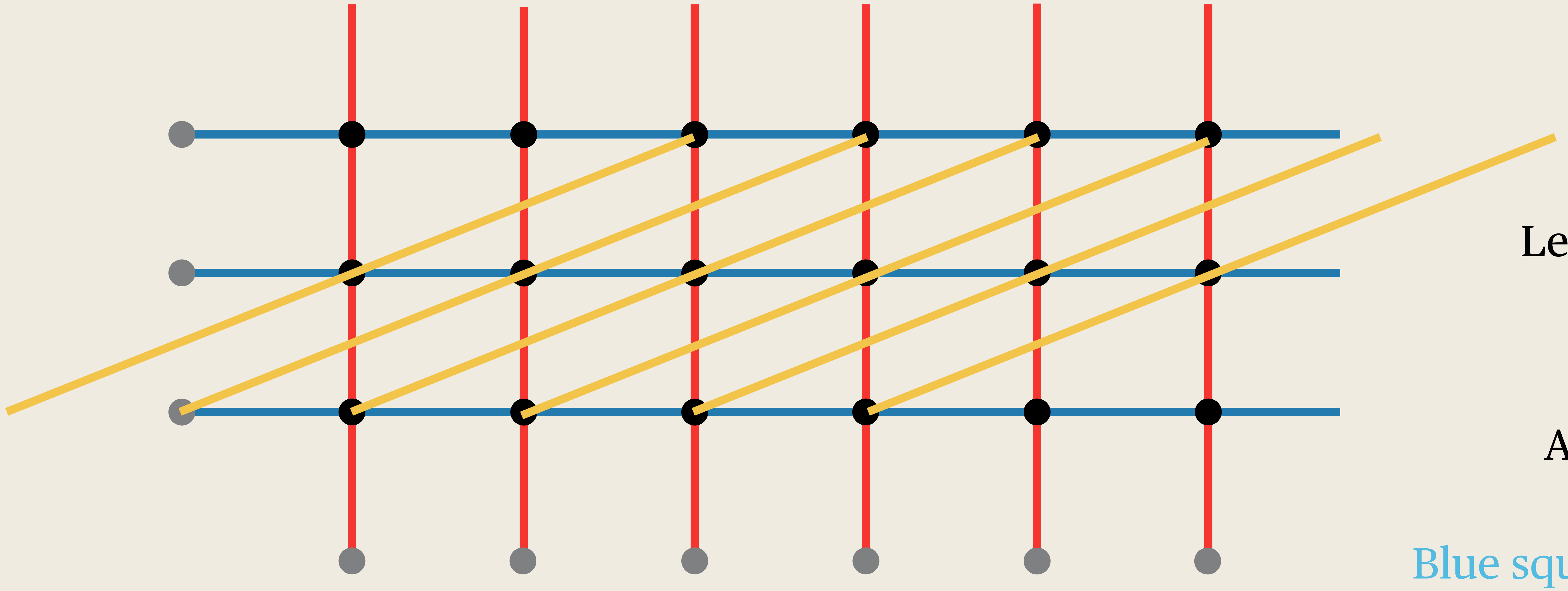
[1] Numerical and analytical bounds on threshold error rates for hypergraph-product codes

... and a bit of Algebraic Topology



Quotient by Group Symmetry

Let's play a game!

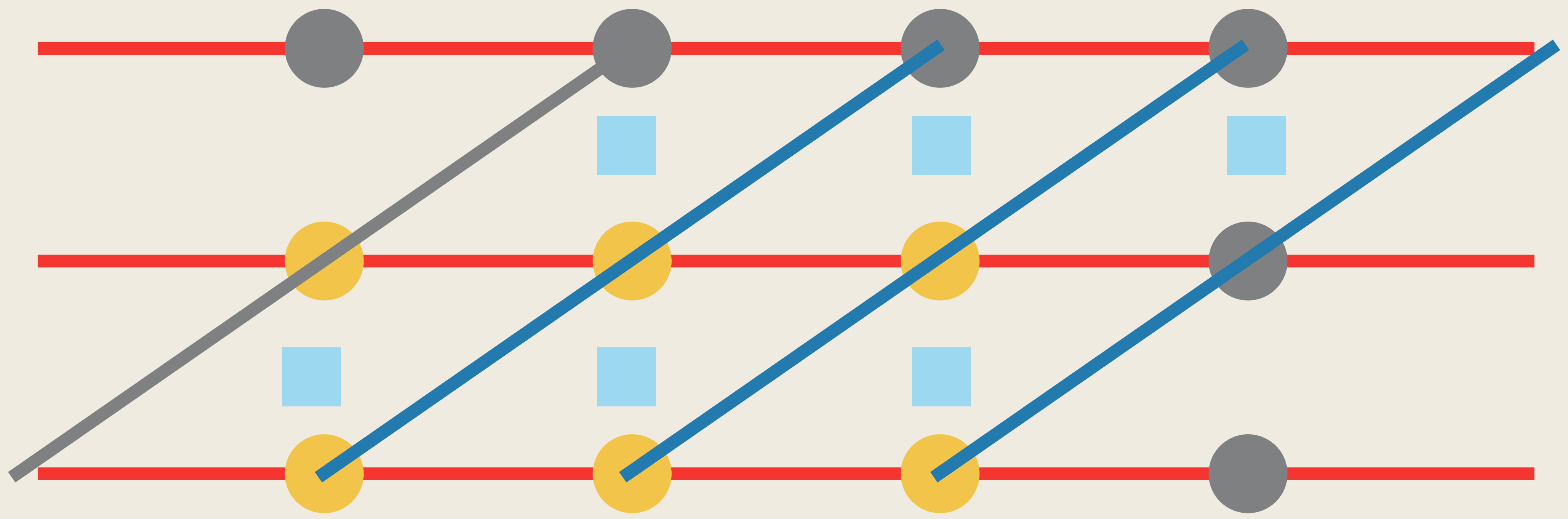


Here is a 3 by 6 Toric code. [[36, 2, 3]]

Let's identify following vertices as one group

$$(x, y) = (x + 2, y + 1) = (x + 4, y + 2)$$

And redraw according to this equivalence.



Blue squares are Z-checks, yellow vertices are X-checks.

This is now a [[12, 2, 3]] code.

Question: Wait… what just happened?

This is $(\mathbb{Z}_6 \times \mathbb{Z}_3)/\mathbb{Z}_3$.

Quotient by Group Symmetry

Asymptotically Good Quantum and Locally Testable Classical LDPC Codes

Pavel Panteleev, Gleb Kalachev

We study classical and quantum LDPC codes of constant rate obtained by the lifted product construction over non-abelian groups. We show that the obtained families of quantum LDPC codes are asymptotically good, which proves the qLDPC conjecture. Moreover, we show that the produced classical LDPC codes are also asymptotically good and locally testable with constant query and soundness parameters, which proves a well-known conjecture in the field of locally testable codes.

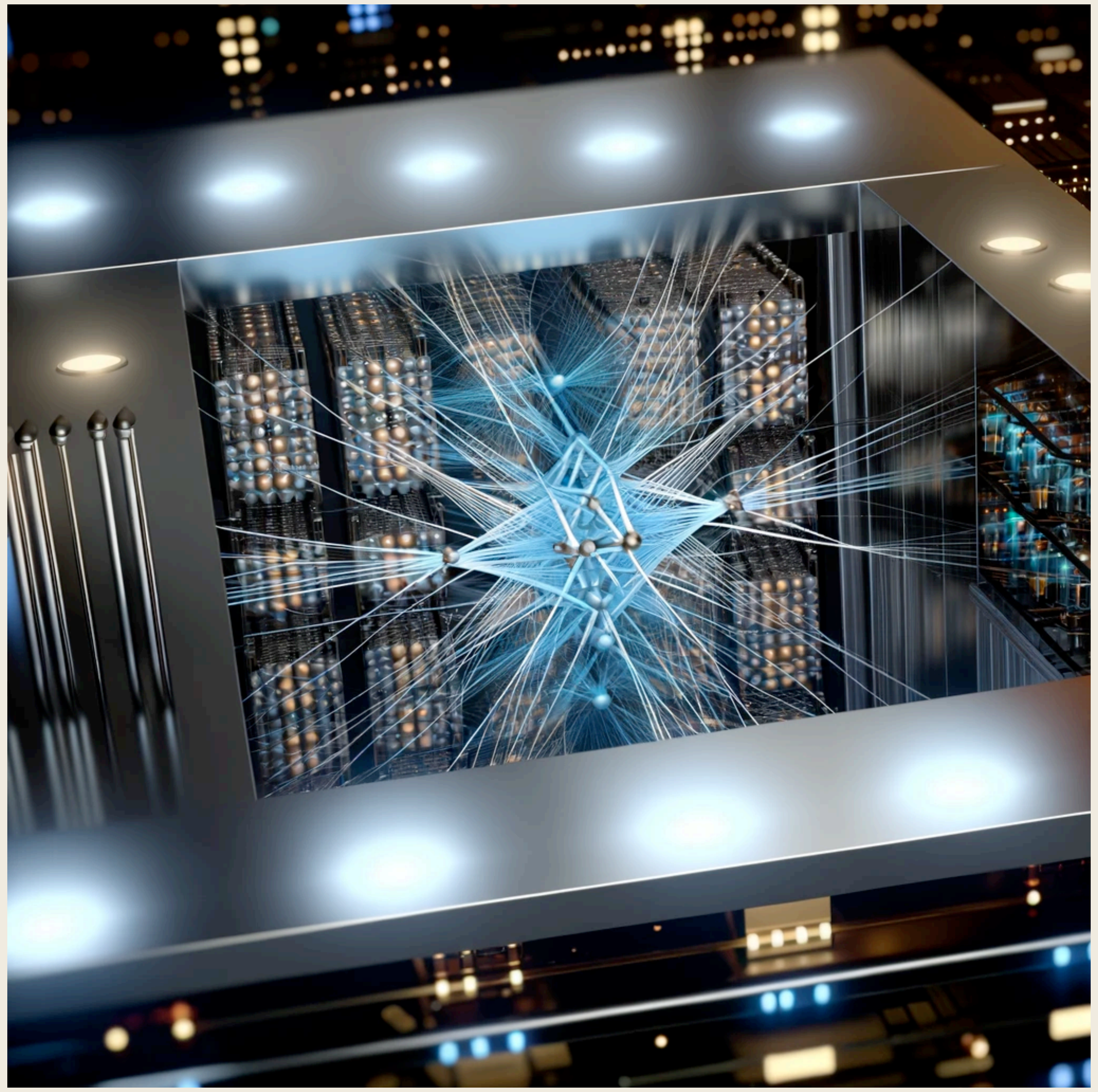
Comments: Updated the introduction, corrected some misprints, clarified some proofs, added some new bibliography including arXiv:2005.01045 containing an independent construction of good LTCs
Subjects: **Information Theory (cs.IT)**; Quantum Physics (quant-ph)
Cite as: arXiv:2111.03654 **[cs.IT]**
(or arXiv:2111.03654v2 **[cs.IT]** for this version)
https://doi.org/10.48550/arXiv.2111.03654

$$\left.\begin{array}{r}\text{Finite groups} \Longrightarrow \text{Expander graphs}\\ \text{`Expanding' local codes}^{1}\end{array}\right\} \text{Classical LDPC codes} \Longrightarrow \text{HP codes} \Longrightarrow \text{Lifted/Balanced Product codes.}$$

A tremendous breakthrough, lead to many progress in QEC and complexity theory, including the NLTS Theorem.

[1] A critical ingredient, ask me about it later.

QLDPC Codes, for Hardware



Quantum hardware based on QLDPC codes
By GPT4

Lifted/balanced product codes lets us encode a lot more logical qubits using the same number of physical qubits.

$$[[625, 25, 6]] \text{ vs. } [[544, 80, \leq 12]]$$

$$[[1225, 49, 8]] \text{ vs. } [[1020, 136, \leq 20]]$$

But, the fundamental challenge has not been addressed:

Hardware connectivity is hard to improve!

A New Chapter

Article | Open access | Published: 06 December 2023

Logical quantum processor based on reconfigurable atom arrays

Dolev Bluvstein, Simon J. Evered, Alexandra A. Geim, Sophie H. Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, J. Pablo Bonilla Ataides, Nishad Maskara, Iris Cong, Xun Gao, Pedro Sales Rodriguez, Thomas Karolyshyn, Giulia Semeghini, Michael J. Gullans, Markus Greiner, Vladan Vuletić & Mikhail D. Lukin

Nature **626**, 58–65 (2024) | Cite this article

Constant-Overhead Fault-Tolerant Quantum Computation with Reconfigurable Atom Arrays

Qian Xu,[1,*] J. Pablo Bonilla Ataides,[2,*] Christopher A. Pattison,[3] Nithin Raveendran,[4] Dolev Bluvstein,[2] Jonathan Wurtz,[5] Bane Vasić,[4] Mikhail D. Lukin,[2] Liang Jiang,[1,†] and Hengyun Zhou[2,5,‡]

[1]*Pritzker School of Molecular Engineering, The University of Chicago, Chicago 60637, USA*
[2]*Department of Physics, Harvard University, Cambridge, Massachusetts 02138, USA*
[3]*Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA 91125*
[4]*Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721, USA*
[5]*QuEra Computing Inc., 1284 Soldiers Field Road, Boston, MA, 02135, US*

Quantum low-density parity-check (qLDPC) codes can achieve high encoding rates and good code distance scaling, providing a promising route to low-overhead fault-tolerant quantum computing. However, the long-range connectivity required to implement such codes makes their physical realization challenging. Here, we propose a hardware-efficient scheme to perform fault-tolerant quantum computation with high-rate qLDPC codes on reconfigurable atom arrays, directly compatible with recently demonstrated experimental capabili-

What if we can move our physical qubits around?

Neutral atoms hardware allows us to physically move qubits, enabling long-range connections and higher qubit degree.

This paper gives a proposal of how to implement lifted product codes with neutral atom arrays.

A New Chapter

High-threshold and low-overhead fault-tolerant quantum memory

Sergey Bravyi, Andrew W. Cross, Jay M. Gambetta, Dmitri Maslov, Patrick Rall, Theodore J. Yoder

Quantum error correction becomes a practical possibility only if the physical error rate is below a threshold value that depends on a particular quantum code, syndrome measurement circuit, and decoding algorithm. Here we present an end-to-end quantum error correction protocol that implements fault-tolerant memory based on a family of LDPC codes with a high encoding rate that achieves an error threshold of 0.8% for the standard circuit-based noise model. This is on par with the surface code which has remained an uncontested leader in terms of its high error threshold for nearly 20 years. The full syndrome measurement cycle for a length-n code in our family requires n ancillary qubits and a depth-7 circuit composed of nearest-neighbor CNOT gates. The required qubit connectivity is a degree-6 graph that consists of two edge-disjoint planar subgraphs. As a concrete example, we show that 12 logical qubits can be preserved for nearly one million syndrome cycles using 288 physical qubits in total, assuming the physical error rate of 0.1%. We argue that achieving the same level of error suppression on 12 logical qubits with the surface code would require nearly 3000 physical qubits. Our findings bring demonstrations of a low-overhead fault-tolerant quantum memory within the reach of near-term quantum processors.

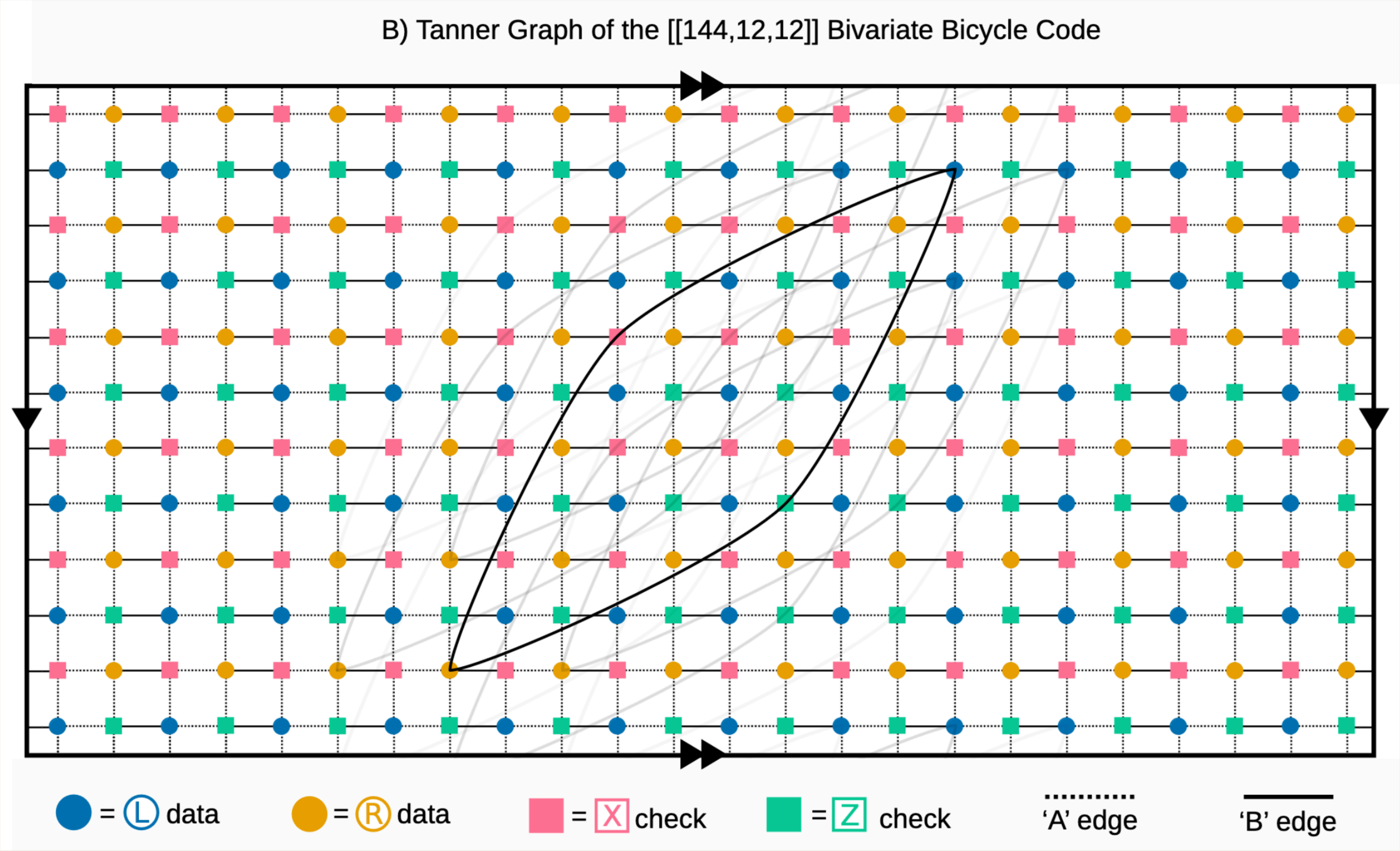
[[144, 12, 12]] code with lots of good properties!

Degenerate Quantum LDPC Codes With Good Finite Length Performance

Pavel Panteleev and Gleb Kalachev

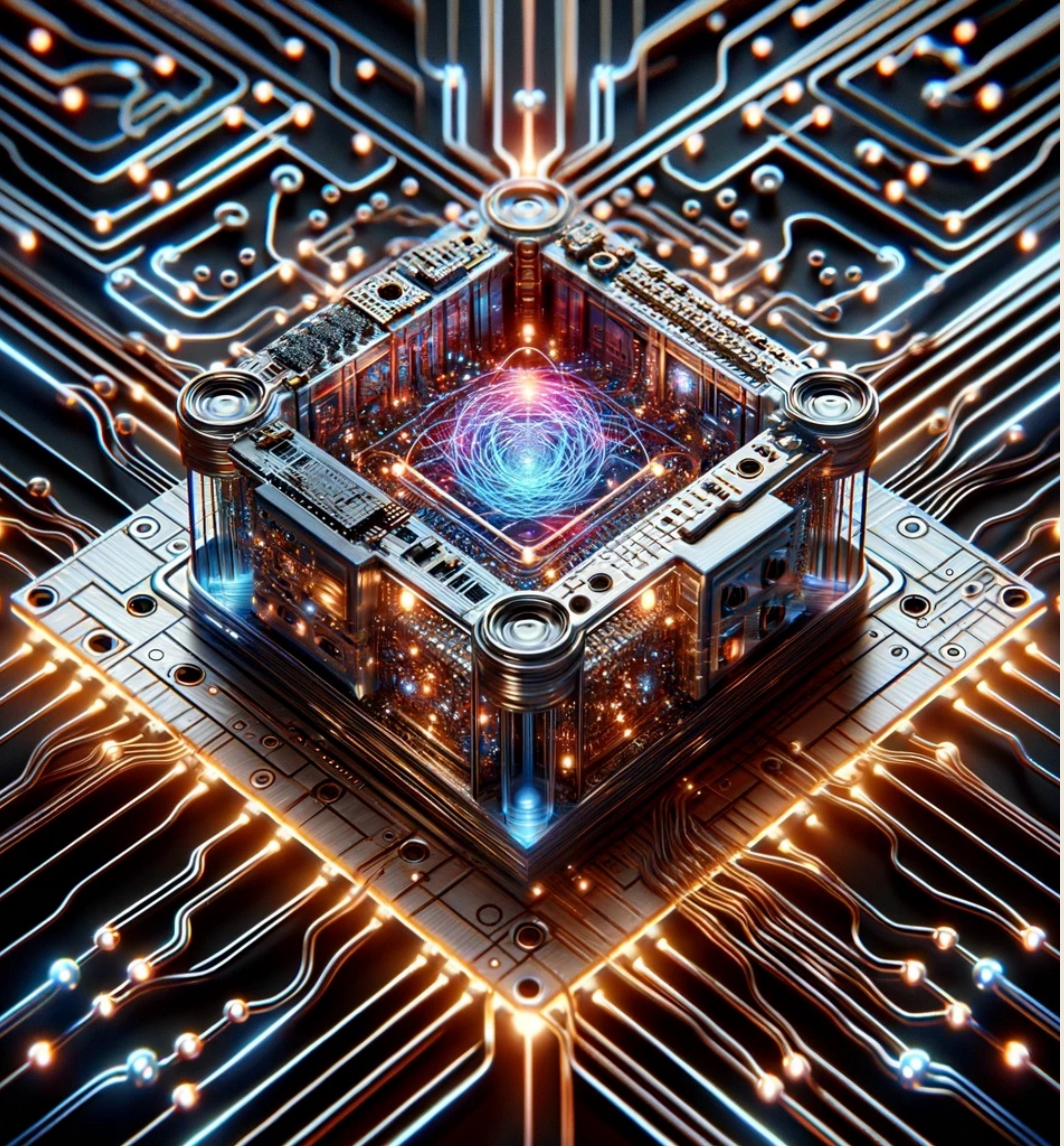
A similar [[126, 12, 10]] code.

Bivariate Bicycle Codes



Properties

1. [[144, 12, 12]] very good parameters on a very practical scale.
2. Qubit degree 6, only 2 more than surface codes!
3. Topologically close to a Torus, a rich example for future studies.
4. Can be embedded into 2 planar layers, IBM's bi-layer architecture.
5. And many, many more. Check the paper!



Our Path Forward

QLDPC in 2024+

Where we are

1. Hardware & QEC codesign.
2. Overall, we are still lacking fault-tolerant ways to implement universal gates on QLDPC codes.
3. QLDPC code shows a 10x space overhead over surface codes. We lacks data on time overhead;
4. Currently, we mostly use Belief-Propagation + Order Statistic Decoding. We lack a truly quantum decoder.
5. As a memory, QLDPC codes have comparable logical error rate to surface codes. But what about in computation?

Our Wishlist

1. QEC operations need to be viable on hardware;
2. Logical operations need to be fault-tolerant;
3. The space and time overheads should be minimized;
4. Efficient and high-performance decoding algorithm;
5. Low logical error rate;

The Central Open Problem

How can we do low-overhead, fault-tolerant logical gates on a practical QLDPC code?

An Exciting Time at A New Frontier

Two Talks I strongly recommend:

Dolev Bluvstein, Logical quantum processor based on reconfigurable atom arrays, available on Youtube;

Jeongwan Haah, What is Your Logical Qubit?, will be available on Youtube.

Many good talks here:

QEC 2023, 6th International Conference on Quantum Error Correction, Sydney, available on Youtube;

Many good works are being done. Surface code is not the end of practical QEC, but the beginning!

Acknowledgements

Thanks to Louis Golowich, for some important figures in this talk.

Thanks to GPT 4 + Dall-E 3, for some artistic illustrations.

Quantum LDPC Codes

A New Frontier of Quantum Error Correction

Zhiyang He (Sunny), March 12, at UCLA