Title: Quantum computing with 3-d surface codes

Date: Nov 14, 2018 04:00 PM

URL: http://pirsa.org/18110080

Abstract: <p>Quantum computers have the potential to significantly outperform classical computers at certain tasks. However, many applications of quantum computers require a fault-tolerant quantum computer. Such a device would function correctly even in the presence of noise. State-of-the art quantum computing architecture proposals require at least hundreds of thousands of high quality qubits to achieve fault-tolerance. These requirements are far beyond today's technology. In this talk, I will present results about a novel quantum computing architecture which is based on 3-d surface codes (a family of quantum error-correcting codes). This architecture may have smaller resource requirements than the current leading architectures for certain experimental parameters. I will show that the 3-d surface code has a transversal CCZ gate and I will discuss a cellular automaton which can be used to decode 3-d surface codes. Time permitting, I will also discuss the relationship between 3-d surface codes and 3-d colour codes. </p>

Quantum Computing with 3-d Surface Codes

Michael Vasmer

November 14, 2018

University College London

Motivation

- Quantum computers have the potential to significantly outperform classical computers at certain tasks
- To deal with imperfect control and decoherence, we must build a fault-tolerant quantum computer
- Current state-of-the art fault-tolerant quantum computing architecture proposals[1] require hundreds of thousands of high quality qubits
- This is far beyond the reach of current experimental systems

[1] Fowler & Gidney, arXiv:1808.06709

Motivation

- Topological codes have desirable properties such as high error threshold[2]
- No-go theorems limit 2-d topological codes[3]
- 3-d surface codes are a family of topological codes closely related to 3-d color codes[4]
- 3-d color codes have useful properties such as transversal non-Clifford gates[5] and single-shot error correction[6]

---

[2] Stephens, *Phys. Rev. A* 89, 022321 (2014)
[3] Bravyi & König, *Phys. Rev. Lett.* 110, 170503 (2013)
[4] Kubica et al, *New J. Phys.* 17, 083026 (2015)
[5] Bombín, *New J. Phys.* 17, 083002 (2015)
[6] Bombín, *Phys. Rev. X* 5, 031043 (2015)

Talk Overview

1. Background
2. 3-d Surface Code Architecture

   Transversal CCZ in 3-d surface codes

   Decoding 3-d Surface Codes
3. Relationship Between Surface Codes and Color Codes
4. Conclusion

Background

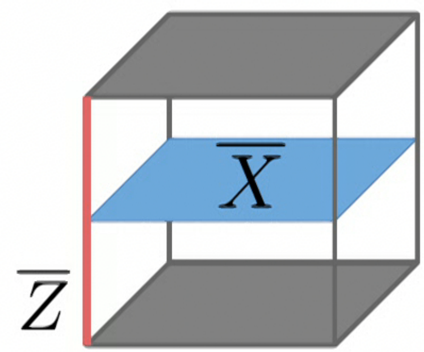
Stabilizer Codes

- Code described by stabilizer $\mathcal{S}$, an abelian subgroup of the Pauli group where $I \notin \mathcal{S}$
- For all codewords $|\psi\rangle$: $S\,|\psi\rangle = |\psi\rangle\ \forall S \in \mathcal{S}$[7]
- Detect errors by measuring generators of stabilizer

[7] Gottesman, PhD Thesis, Caltech (1997)

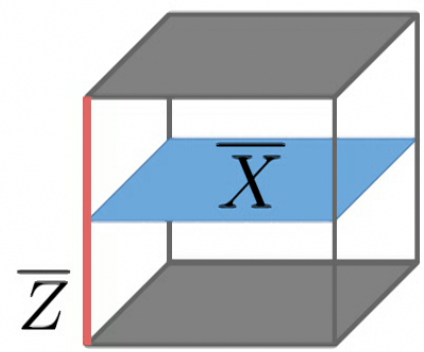
3-d Surface Code

- Code defined on some (Euclidean) 3-d lattice with boundaries
- Two types of boundaries: ‘rough’ and ‘smooth’
- We consider lattices with two smooth boundaries and four rough boundaries



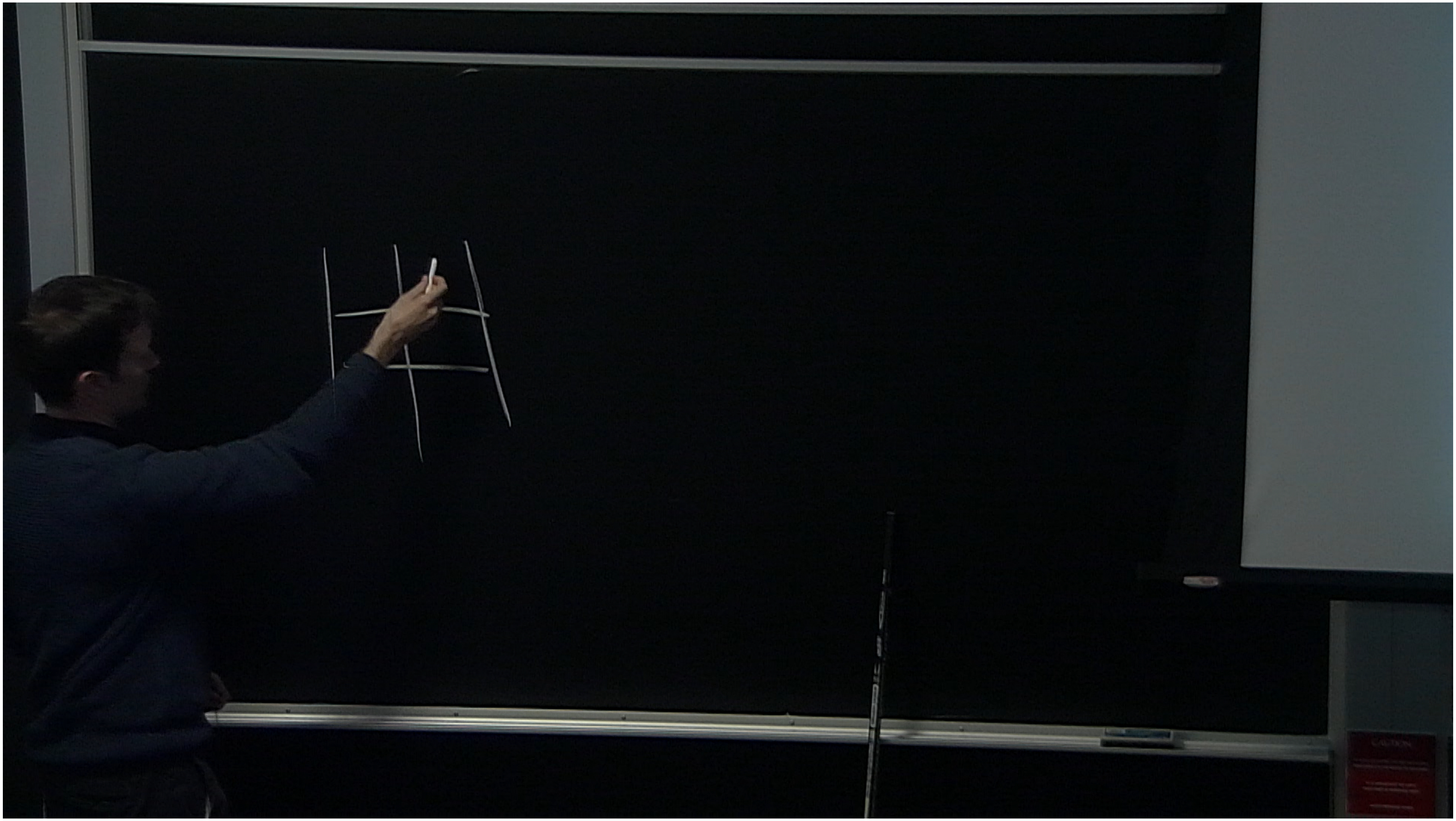
3-d Surface Code

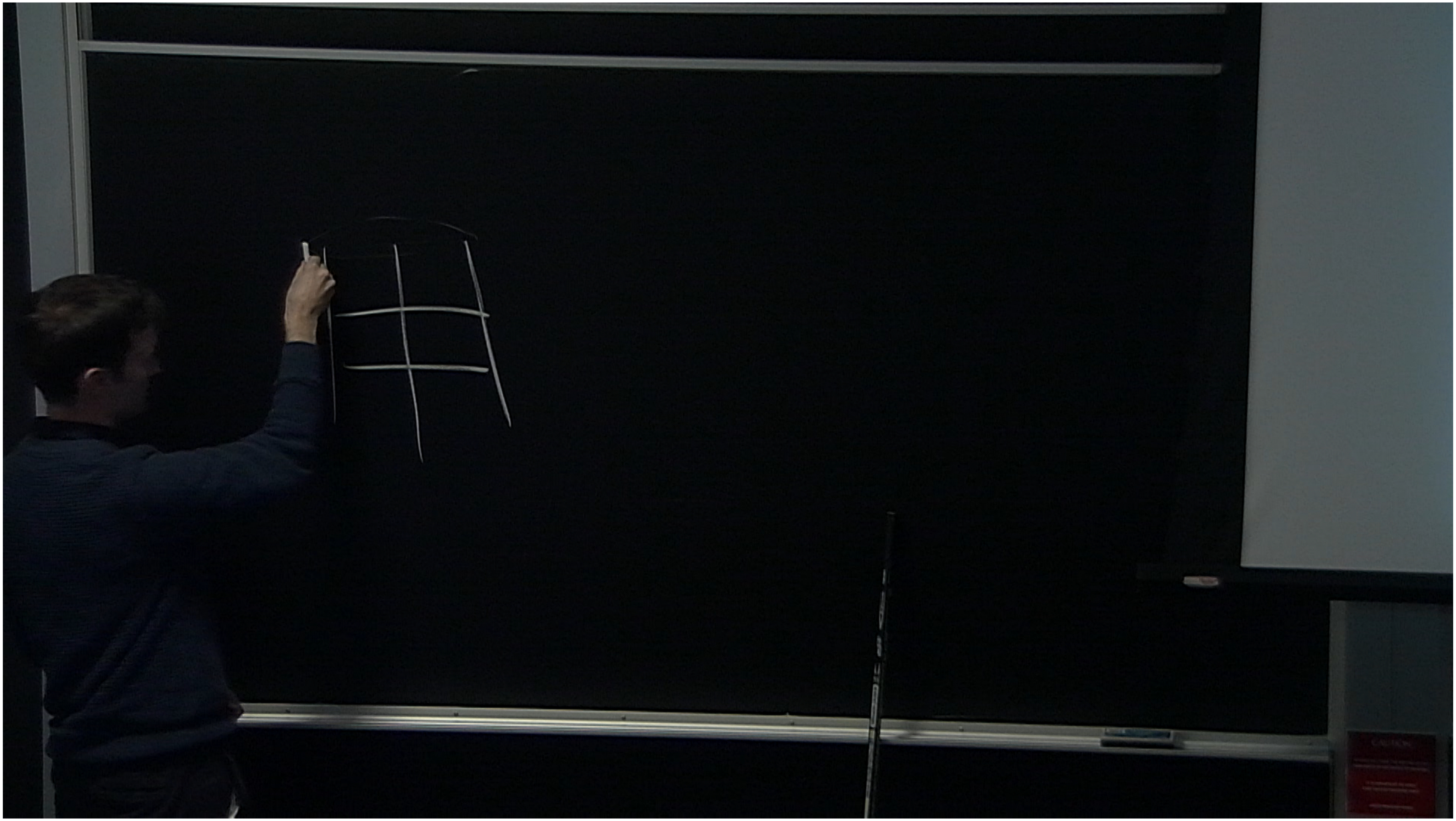
- Qubits on faces, X stabilizers associated with cells and Z stabilizers with edges
- One logical qubit
- Logical $\overline{Z}$ operators are strings of Z operators which terminate at opposite smooth boundaries
- Logical $\overline{X}$ operators are membranes of X operators whose boundary spans the four rough boundaries



3-d Surface Code

- In the bulk, each physical qubit is acted upon non-trivially by two X stabilizer generators and $m \geq 3$ Z stabilizer generators
- Every physical qubit on the smooth boundaries is acted upon by a single X stabilize generator
- Every physical qubit on the rough boundaries is acted upon by fewer than m Z stabilizer generators





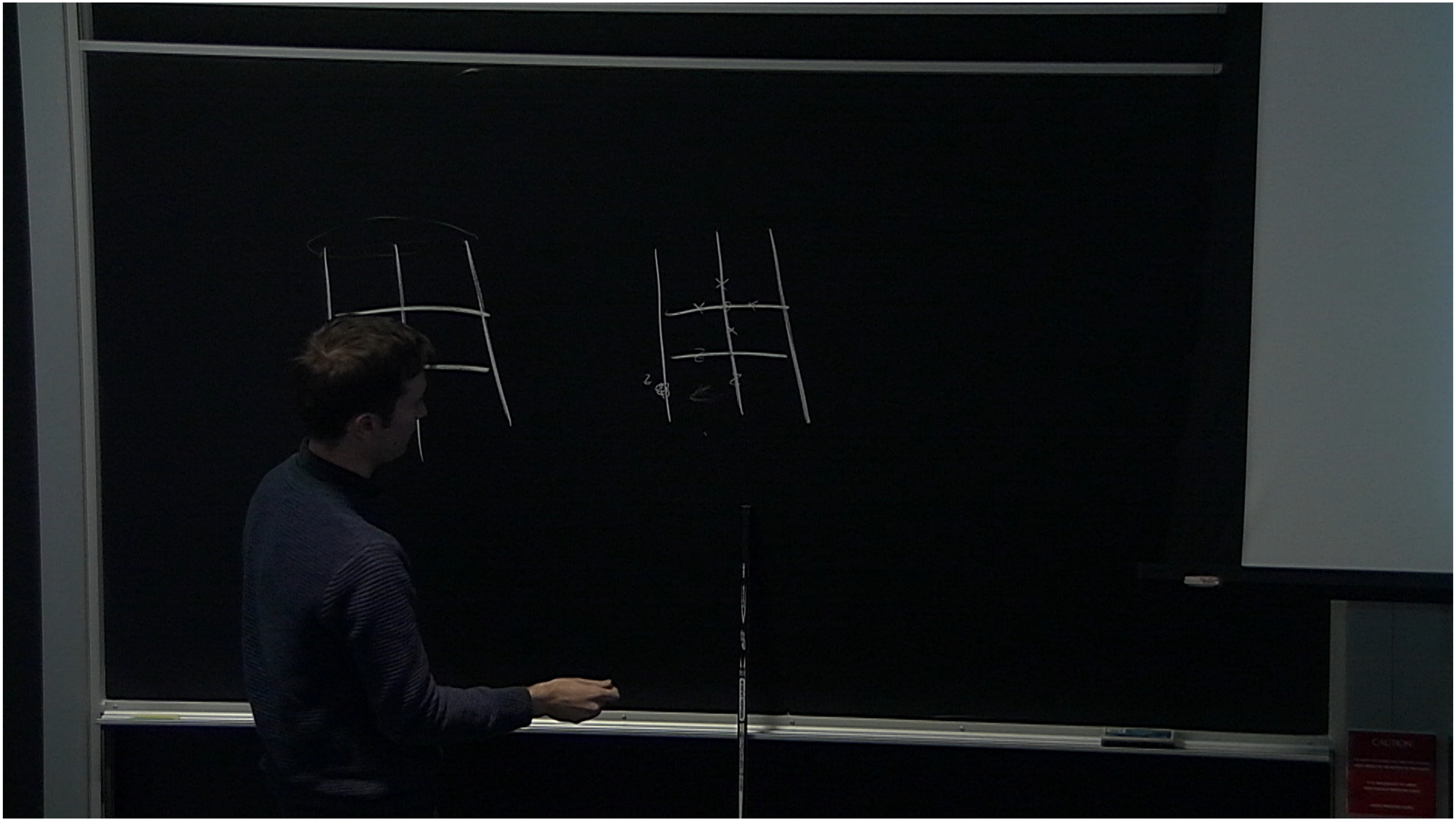
3-d Surface Code

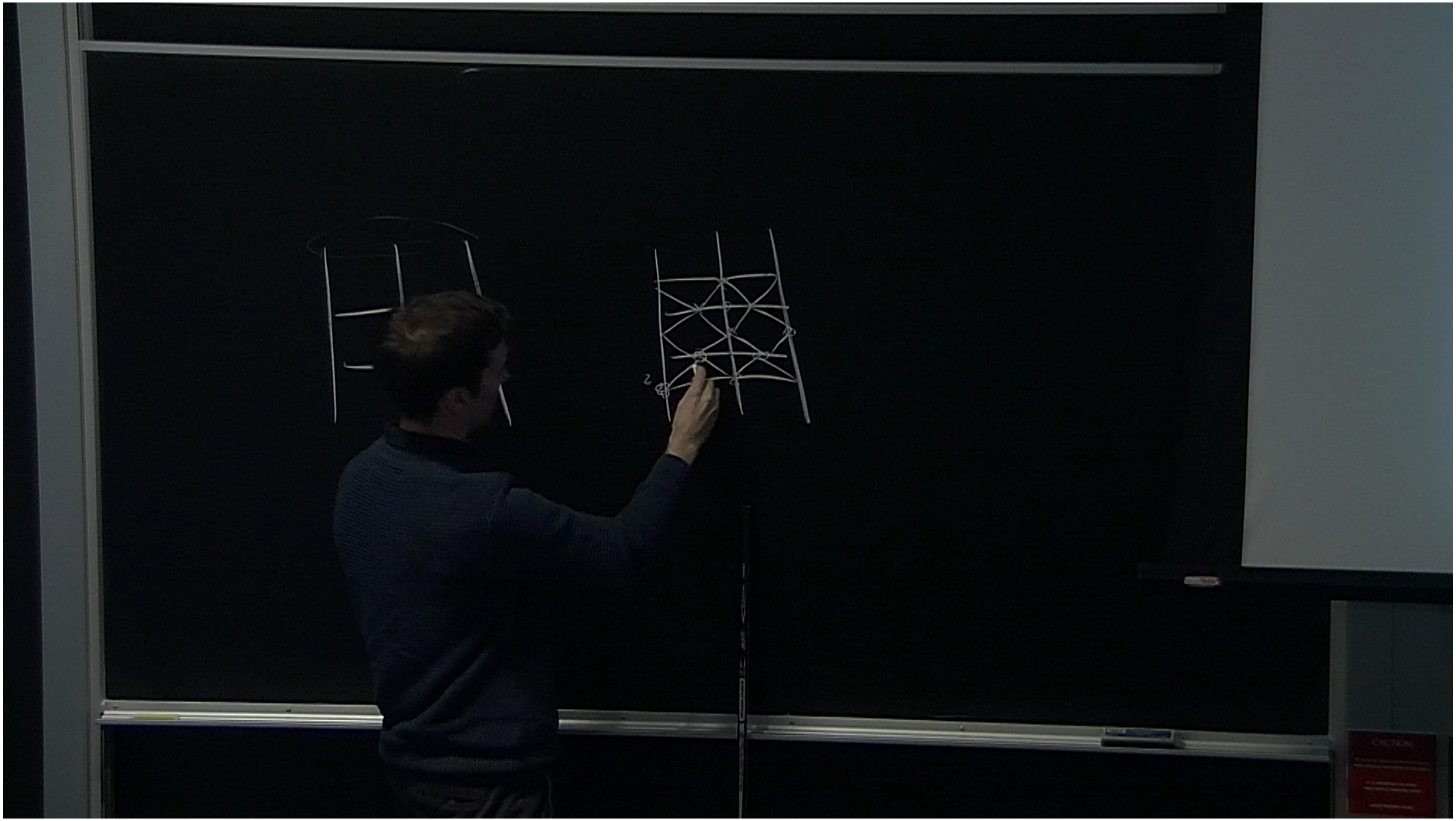
- In the bulk, each physical qubit is acted upon non-trivially by two X stabilizer generators and $m \geq 3$ Z stabilizer generators
- Every physical qubit on the smooth boundaries is acted upon by a single X stabilize generator
- Every physical qubit on the rough boundaries is acted upon by fewer than m Z stabilizer generators

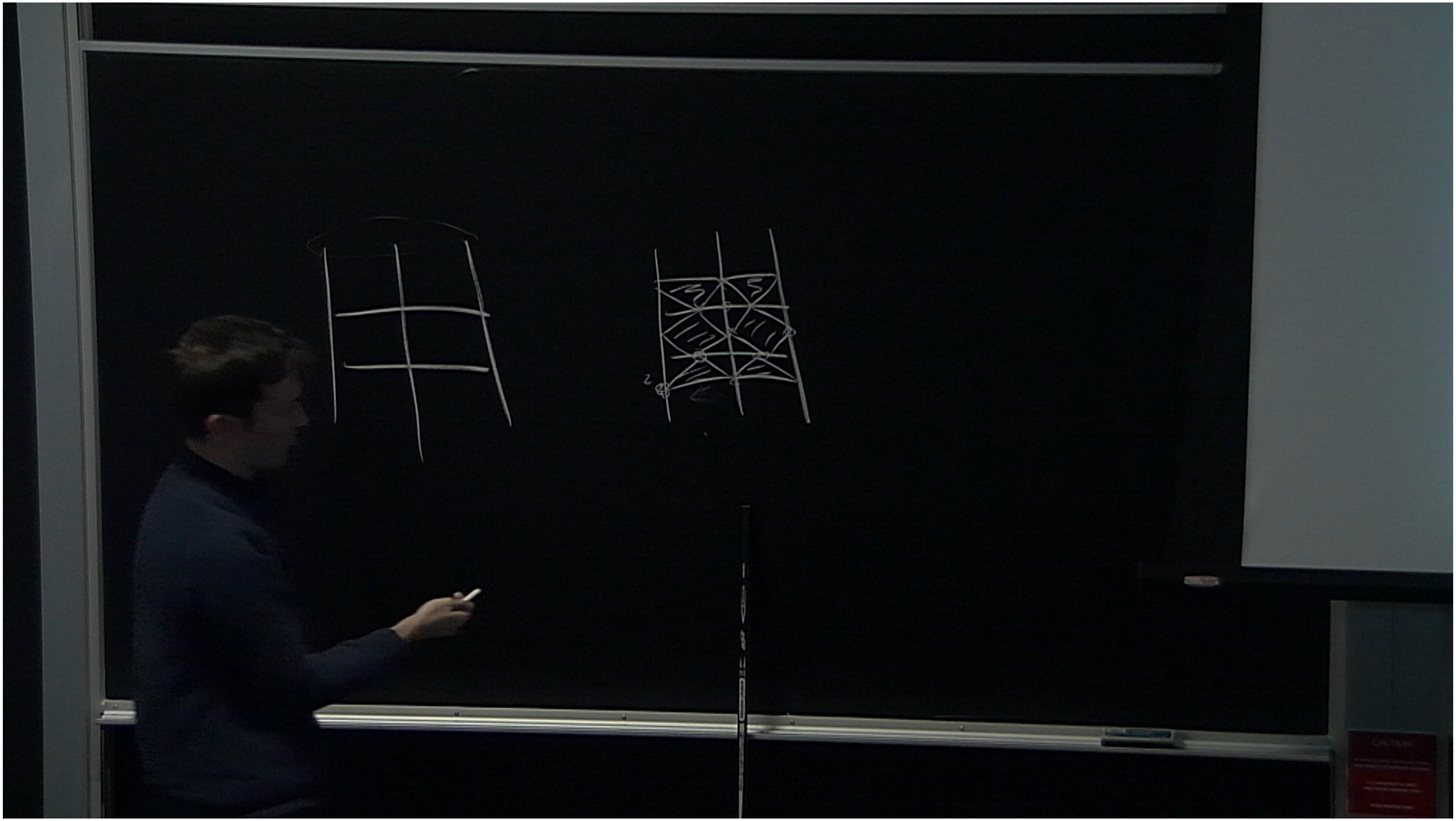
Rectified Picture

- Consider a stack of three 3-d surface codes
- Rectified picture[8] lattice which describes all three codes
- Qubits on vertices, Z stabilizers on faces, X stabilizers on cells
- Different colour cells & faces correspond to different codes

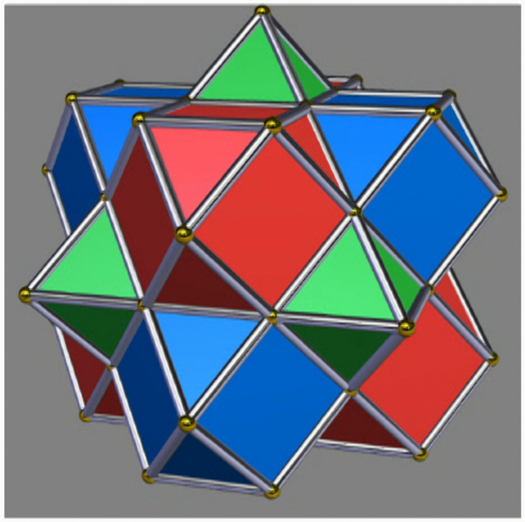
[8] Vasmer & Browne, arXiv:1801.04255





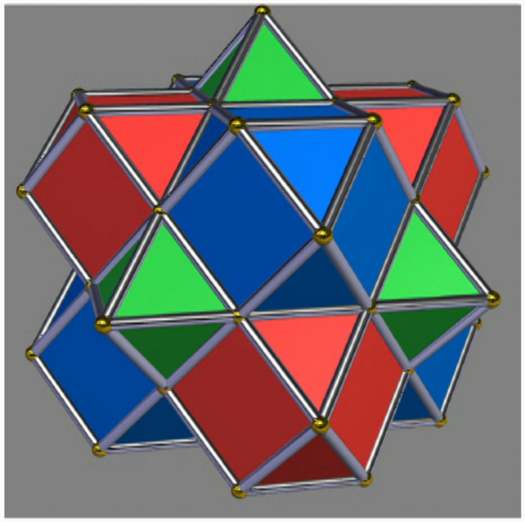


Rectified Picture

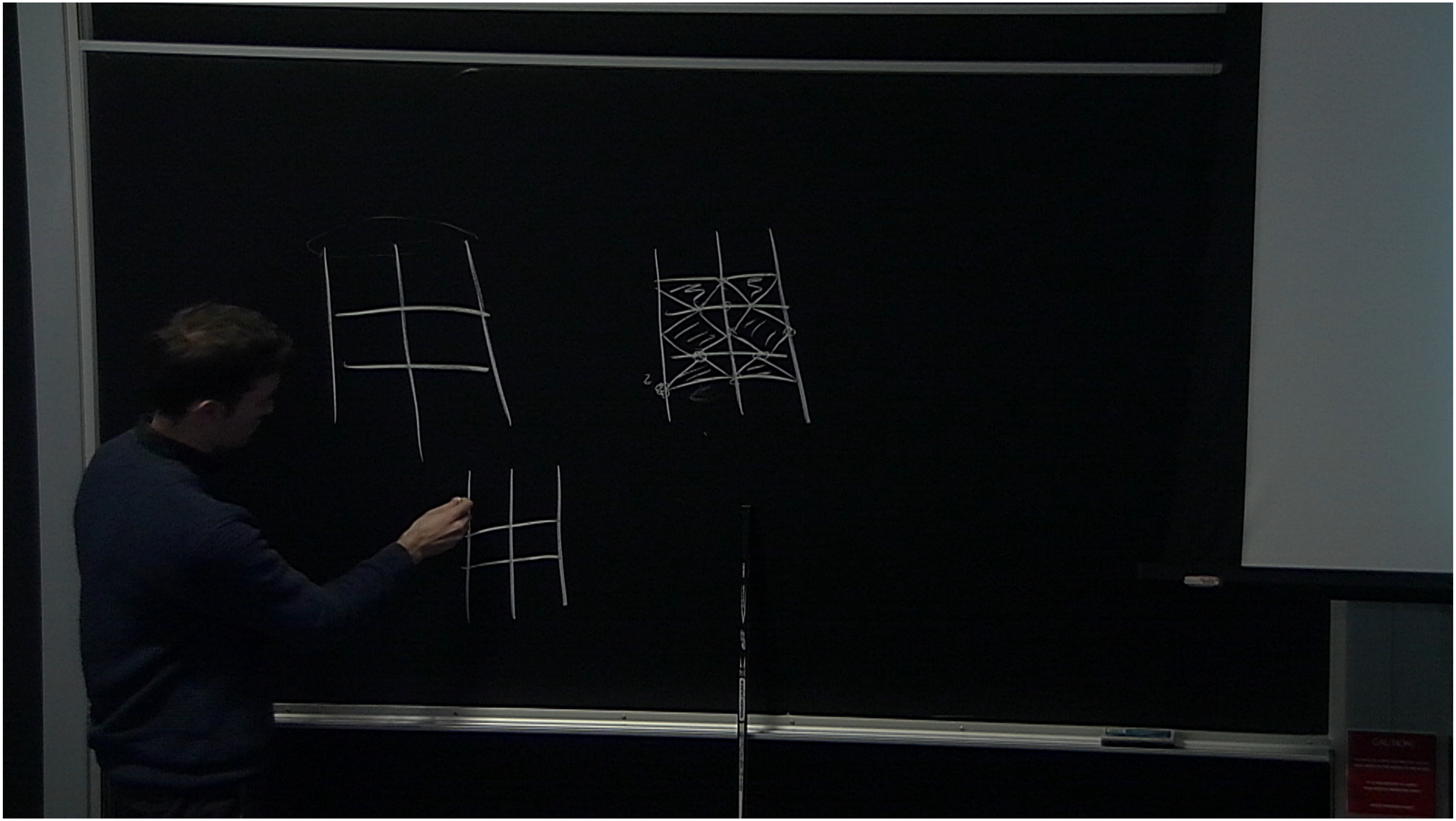


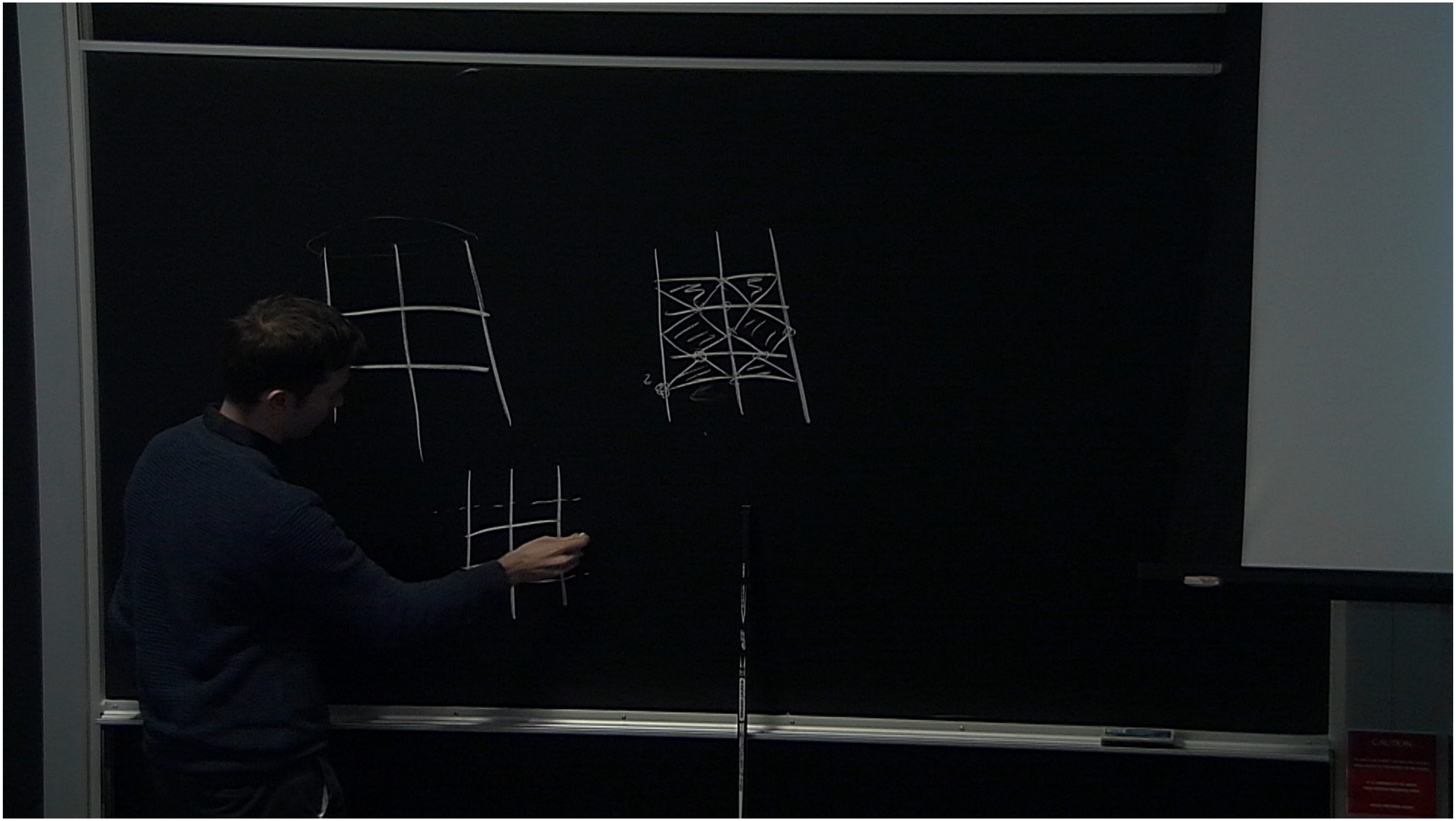
Kitaev Code	X checks	Z checks
Cubic	*g*-cells (octahedra)	*rb*-faces (squares)
Rhombic (B)	*b*-cells (cuboctahedra)	*rg*-faces (triangles)
Rhombic (R)	*r*-cells (cuboctahedra)	*bg*-faces (triangles)

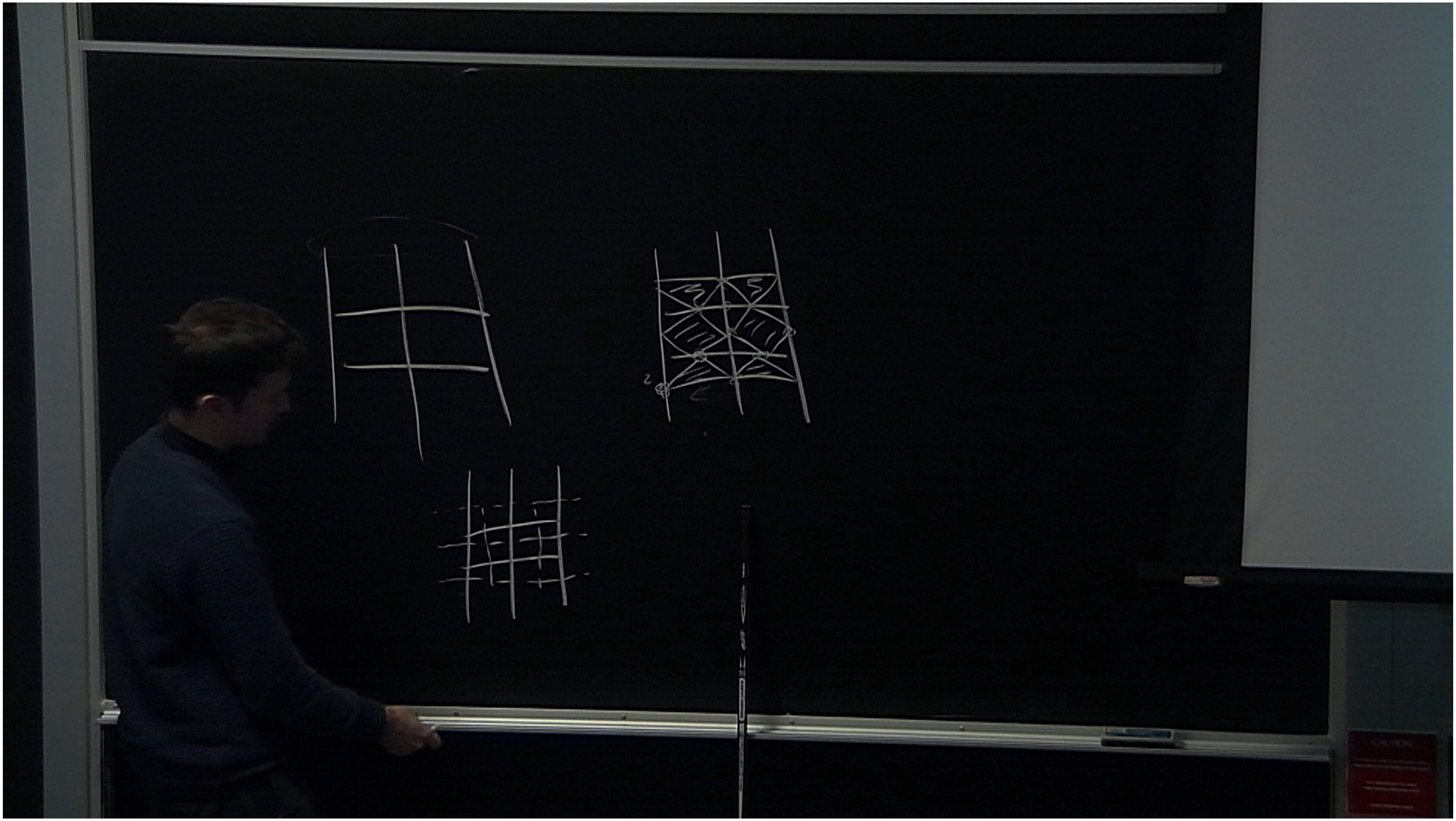
Rectified Picture

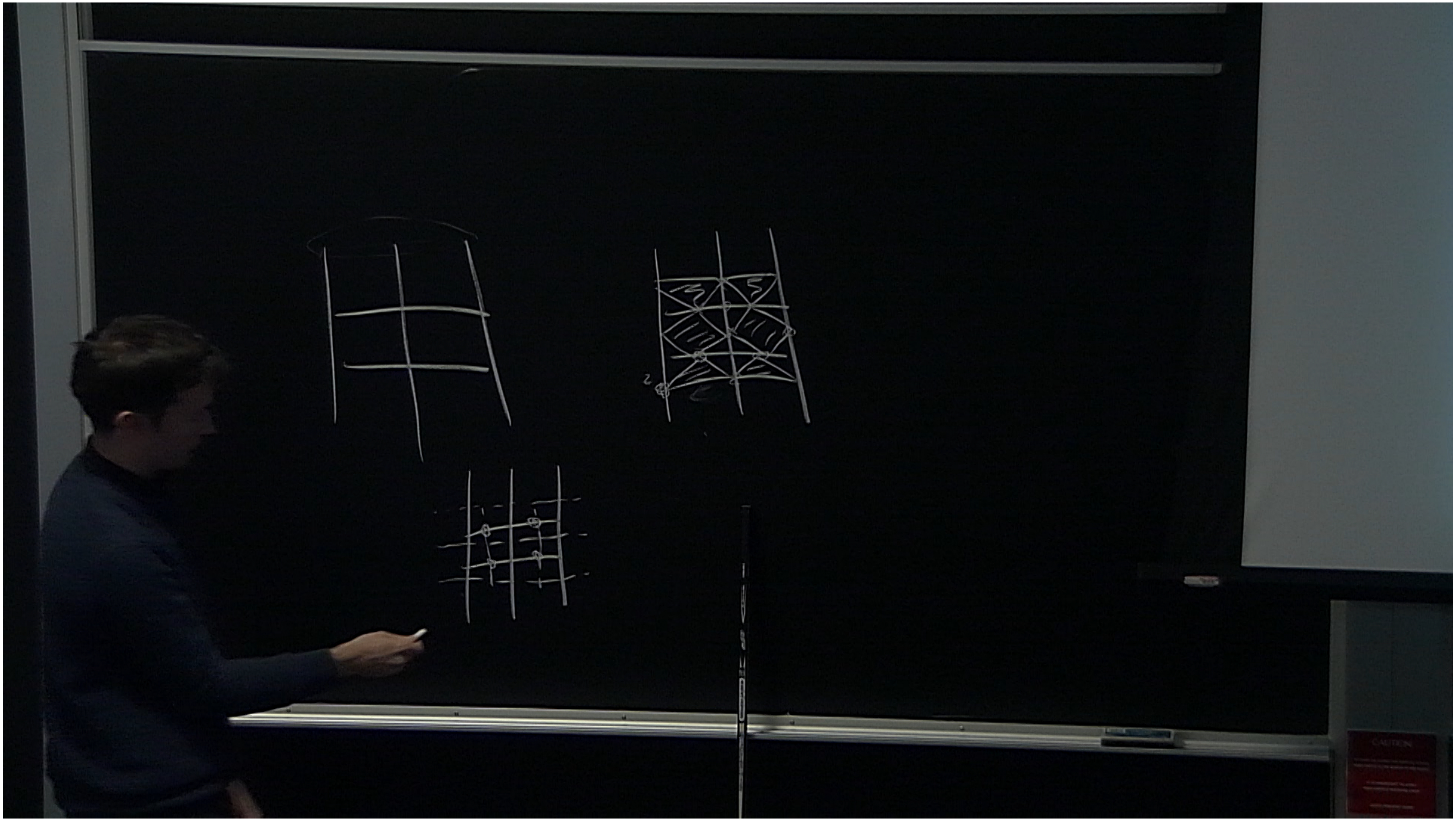


Kitaev Code	X checks	Z checks
Cubic	g-cells (octahedra)	rb-faces (squares)
Rhombic (B)	b-cells (cuboctahedra)	rg-faces (triangles)
Rhombic (R)	r-cells (cuboctahedra)	bg-faces (triangles)

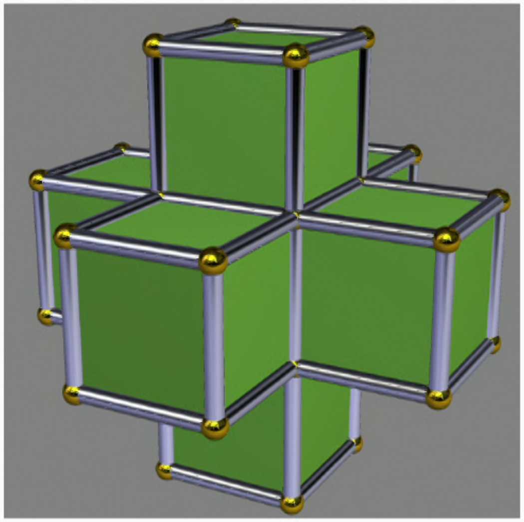




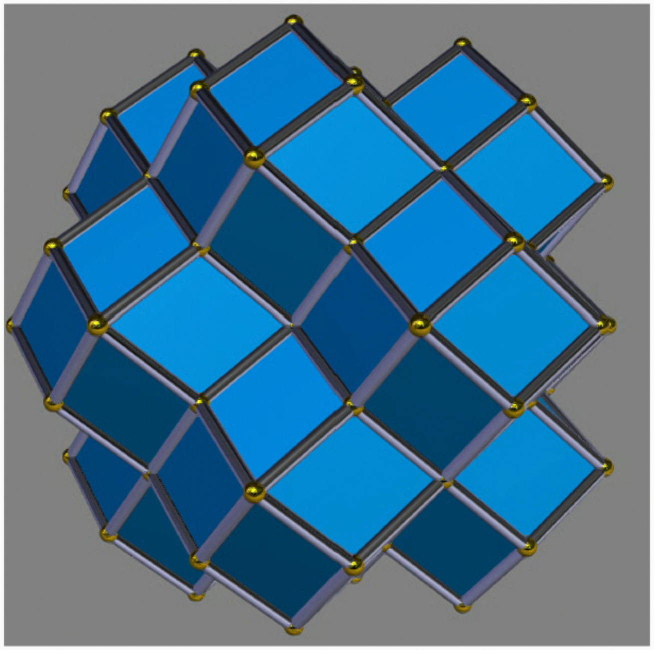




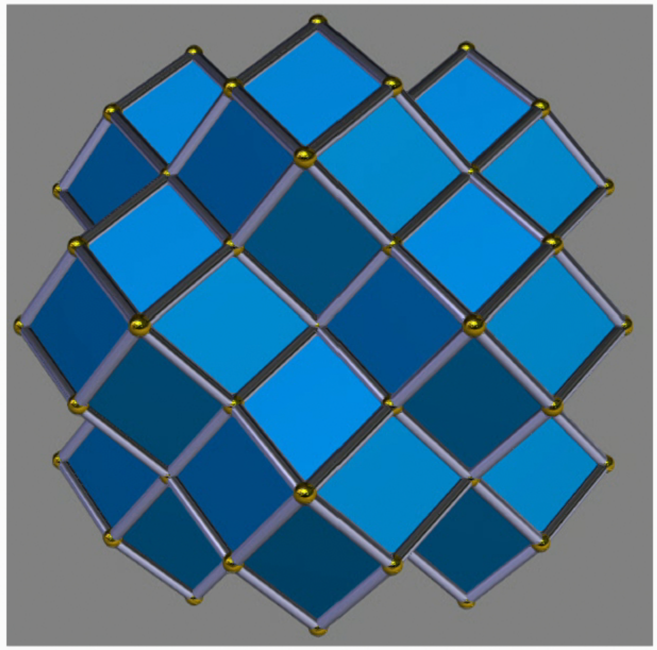
Cubic Lattice



Rhombic Dodecahedral Lattice

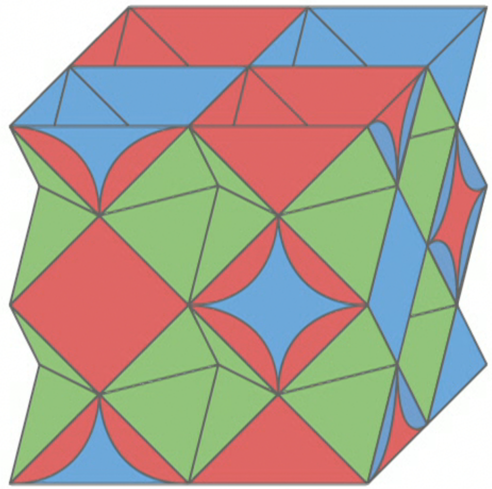


Rhombic Dodecahedral Lattice

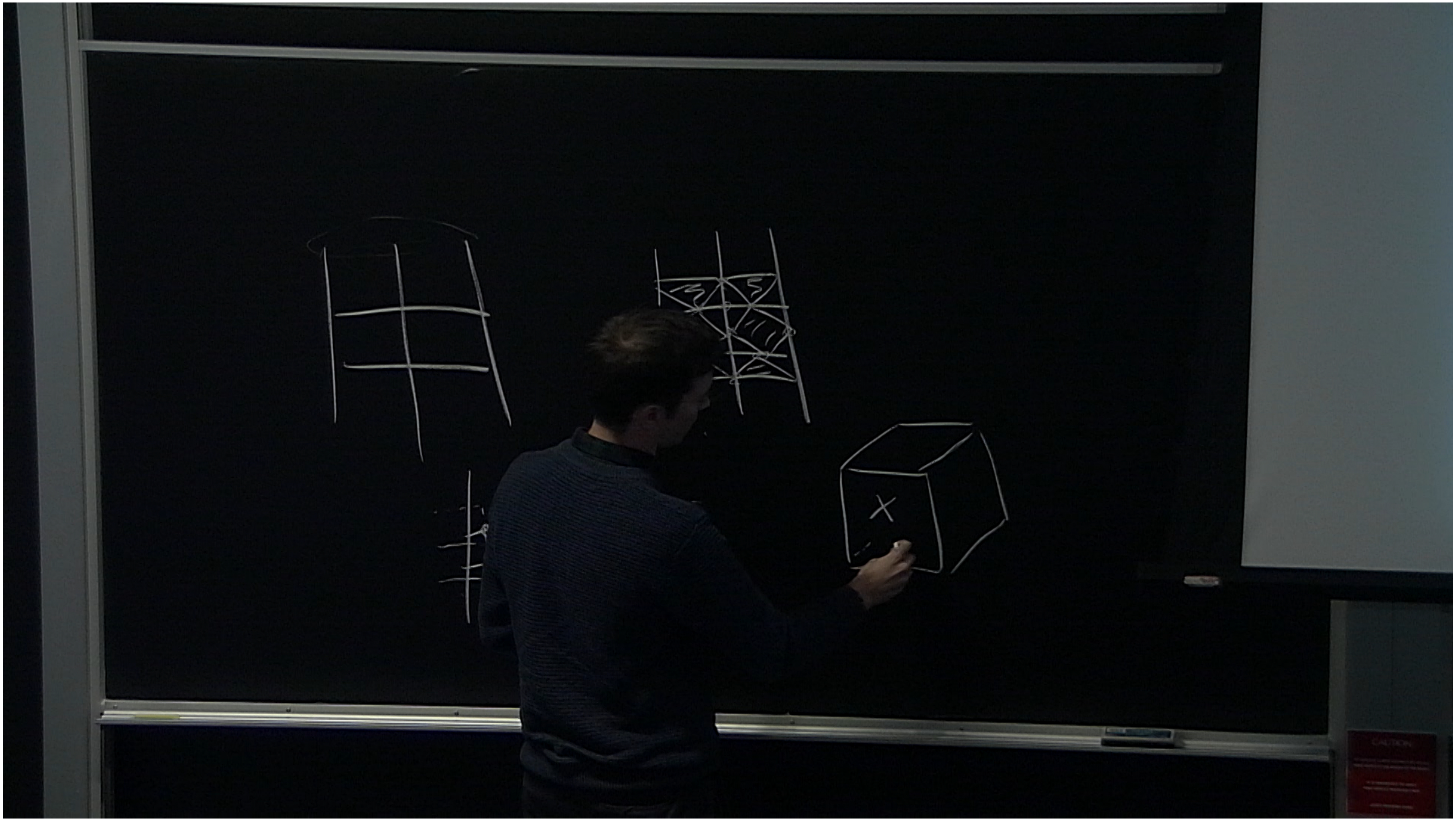


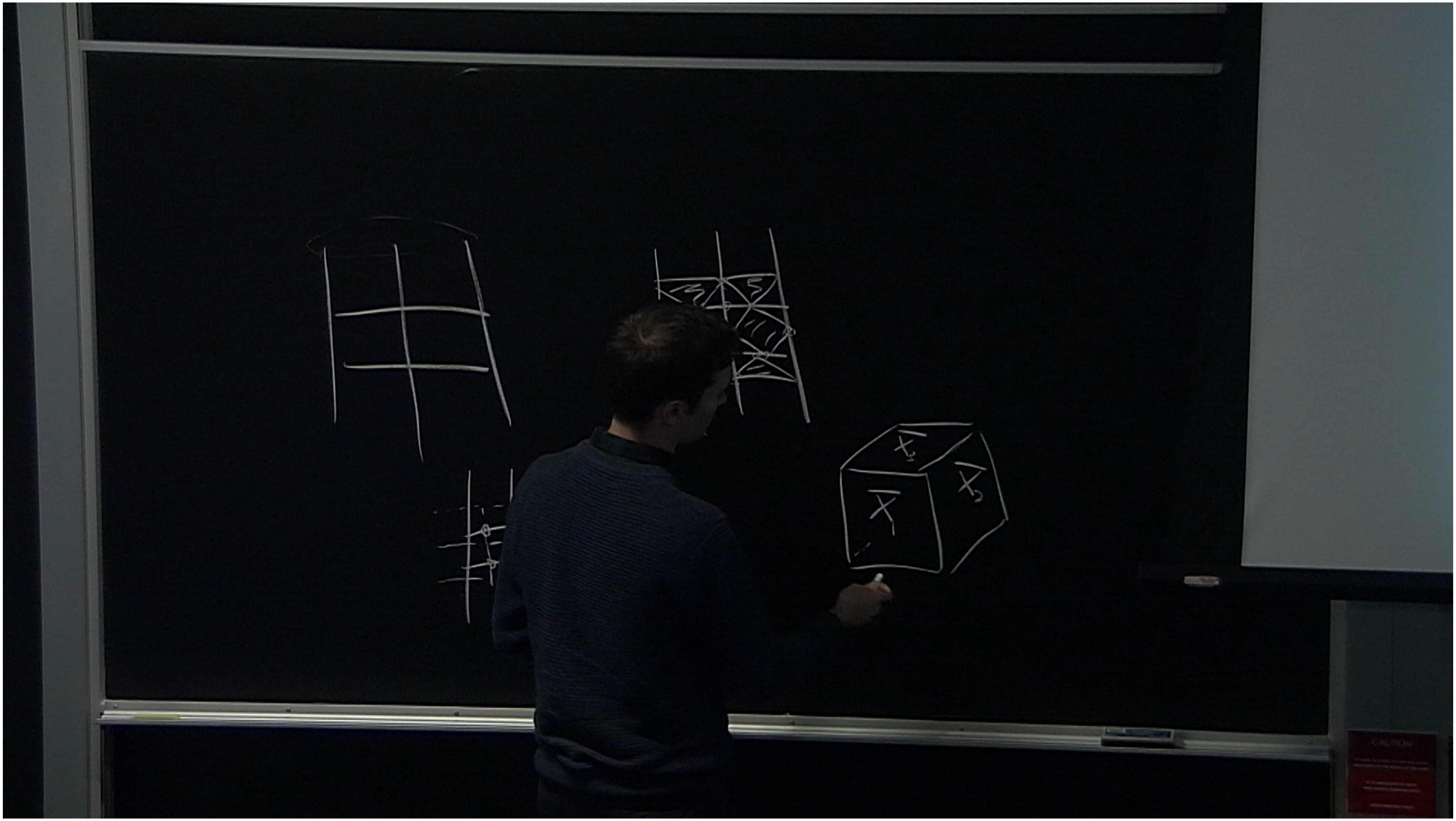
11

Rectified Lattice with Boundaries



12





Transversal CCZ Proof Sketch

- Can always write the encoded states of a $(k = 1)$ CSS code as follows:
  - $|\overline{0}\rangle = \sum_{S \in \mathcal{S}_X} S\,|0\rangle^{\otimes n}$,
  - $|\overline{1}\rangle = \overline{X}\,|\overline{0}\rangle = \sum_{S \in \mathcal{S}_X} S\overline{X}\,|0\rangle^{\otimes n}$
- The combined computational basis states of the three codes in the stack can be written:

$$|\overline{\alpha\beta\gamma}\rangle = \sum_{S_r, S_g, S_b} S_r \overline{X}_r^{\alpha} |0\rangle^{\otimes n} \otimes S_g \overline{X}_g^{\beta} |0\rangle^{\otimes n} \otimes S_b \overline{X}_b^{\gamma} |0\rangle^{\otimes n}, \quad (1)$$

where $\alpha, \beta, \gamma \in \{0, 1\}$.

Transversal CCZ Proof Sketch

- To prove CCZ is transversal, we compute the action of $CCZ^{\otimes n}$ on the computational basis states
- Consider the state $|\overline{000}\rangle$
- Apply $CCZ^{\otimes n}$:

$$CCZ^{\otimes n} |\overline{000}\rangle = \sum_{S_r, S_g, S_b} (-1)^{\mathcal{O}(S_r, S_g, S_b)} S_r |0\rangle^{\otimes n} \otimes S_g |0\rangle^{\otimes n} \otimes S_b |0\rangle^{\otimes n} \quad (2)$$

- $\mathcal{O}(S_r, S_g, S_b)$ counts the number of vertices (triples of qubits) where S_r, S_g and S_b all act non-trivially

Transversal CCZ Proof Sketch

- S_g are associated with g-cells, S_b are associated with b-cells
- S_g and S_b overlap (both act non-trivially) on gb-faces (faces shared by one g-cell and one b-cell)
- The Z stabilizers of $\mathcal{SC}_r$ are associated with gb-faces
- Therefore all S_r (X stabilizers of $\mathcal{SC}_r$) have even overlap with gb-faces

$\Rightarrow CCZ^{\otimes n} \left|\overline{000}\right\rangle = \left|\overline{000}\right\rangle$

Transversal CCZ Proof Sketch

- Consider the state $\left|\overline{100}\right\rangle$
- We need to compute $\mathcal{O}(S_r\overline{X}_r, S_g, S_b)$, the number of vertices (triples of qubits) where $S_r\overline{X}_r$, S_g and S_b all act non-trivially
- We already know that S_g and S_b overlap on faces which support $\mathcal{SC}_r$ Z stabilizers
- $S_r\overline{X}_r = X_r'$ commutes with all $\mathcal{SC}_r$ stabilizers

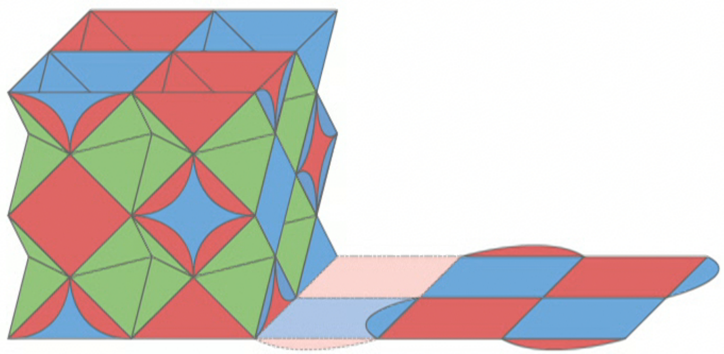
$\Rightarrow CCZ^{\otimes n}\left|\overline{100}\right\rangle = \left|\overline{100}\right\rangle$

Transversal CCZ Proof Sketch

- Other cases are similar
- Care needs to be taken when dealing with the boundaries
- End result is that only $|\overline{111}\rangle$ picks up a -1 phase, which implies that $CCZ^{\otimes n}$ implements a transversal $\overline{CCZ}$

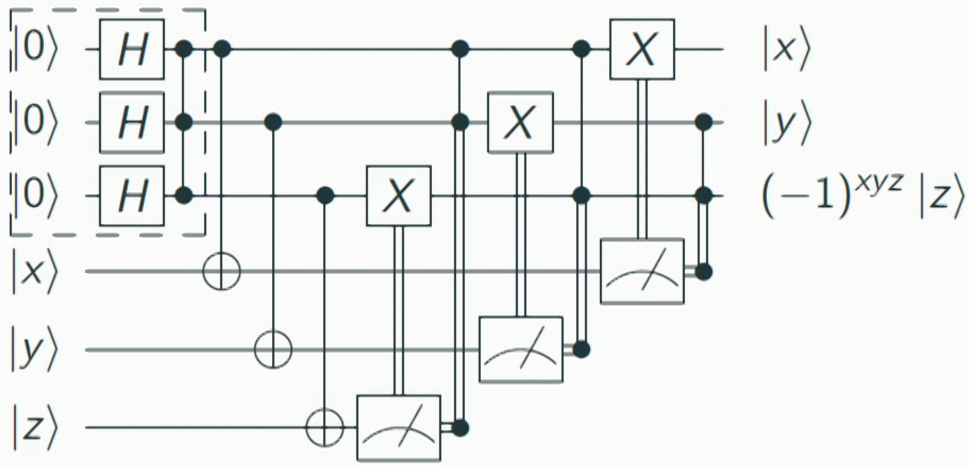
3-d Surface Code Architecture

- Use 3-d surface code to produce $|CCZ\rangle = CCZ\,|+++\rangle$ states
- Transfer $|CCZ\rangle$ states into 2-d surface code architecture using lattice surgery



3-d Surface Code Architecture

- Given a $|CCZ\rangle$ state, we can implement a CCZ gate using only Clifford gates and Pauli basis measurements (both fault-tolerant in 2-d surface code architecture)



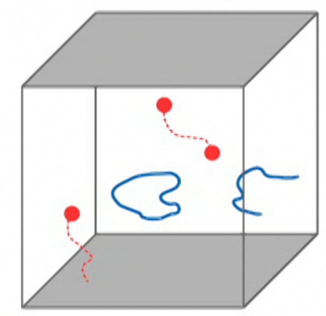
3-d Surface Code Architecture

- 3-d surface code eliminates the need for magic state distillation (substantial overhead in standard 2-d surface code architecture)
- Our architecture best suited to networked quantum computer e.g. ion-trap qubits connected with photonic links[9]

[9] Barrett & Kok, *Phys. Rev. A* 71, 060310 (2005)

Decoding 3-d Surface Codes

- To estimate architecture performance, we need a decoder
- Decoder estimates an error given an error syndrome (stabilizer measurement outcomes)
- Z errors: Matching
- X errors: cellular automaton
- Cellular automaton decoder attractive because it is local and single-shot



Previous Work

Decoder	Perfect Measurements	Measurement Errors
Renormalization[10]	$p_{th} = 17.2\%$	$p_{th} = 7.3\%$
Toom's Rule[11]	$p_{th} \approx 12\%$	?

- Above decoders only work for cubic surface codes.
- We need a decoder which works for rhombic dodecahedral surface codes as well.

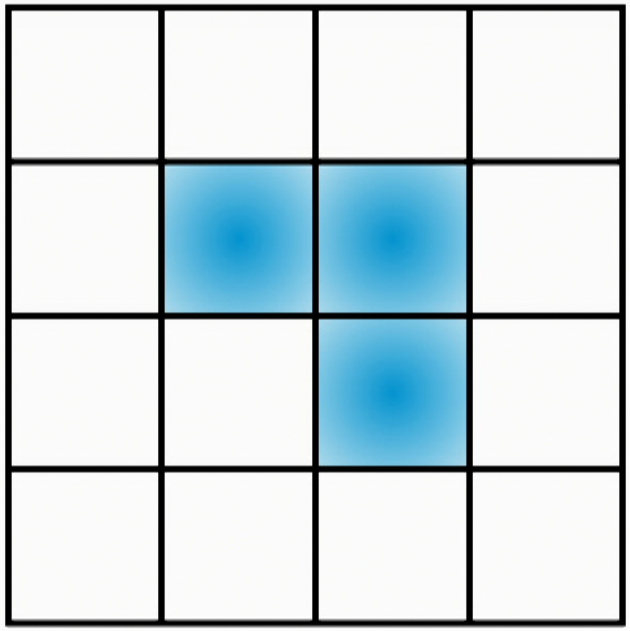
[10] Duivenvoorden et al, arXiv:1708.09286
[11] Kulkarni & Sarvepalli, arXiv:1808.03092

Toom's Rule in 2-d

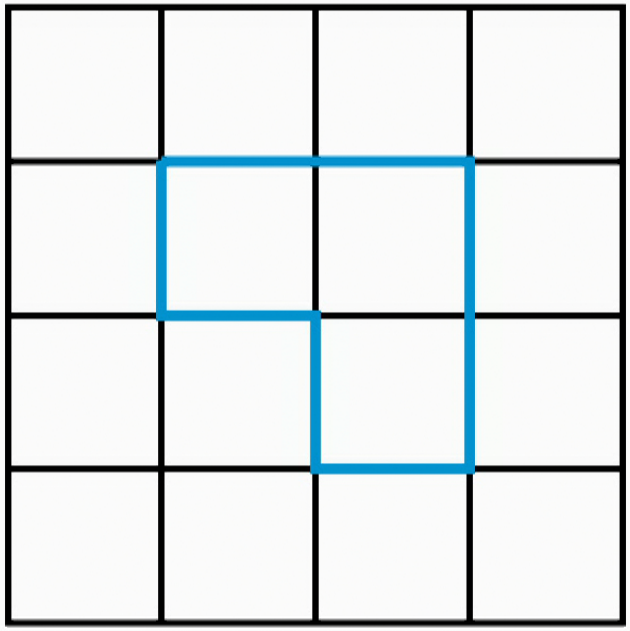
- Square lattice, periodic boundaries
- Qubits on faces, Z checks associated with edges
- Cellular automaton for each face[12]
- If N and E edges unsatisfied, then flip face
- Resilient to 'measurement noise'[13]

[12] Toom, *Multicomponent Syst.* 6, 549-575 (1980)
[13] Grinstein, *IBM J. Res. Dev.* 48, 5-12 (2004)

Toom's Rule in 2-d



Toom's Rule in 2-d



Toom's Rule in 3-d

- Cubic lattice, periodic boundaries
- Qubits on faces, Z checks associated with edges
- Cellular automaton for each face
- Apply Toom's rule in *xy*, *xz* and *yz* planes sequentially[14]

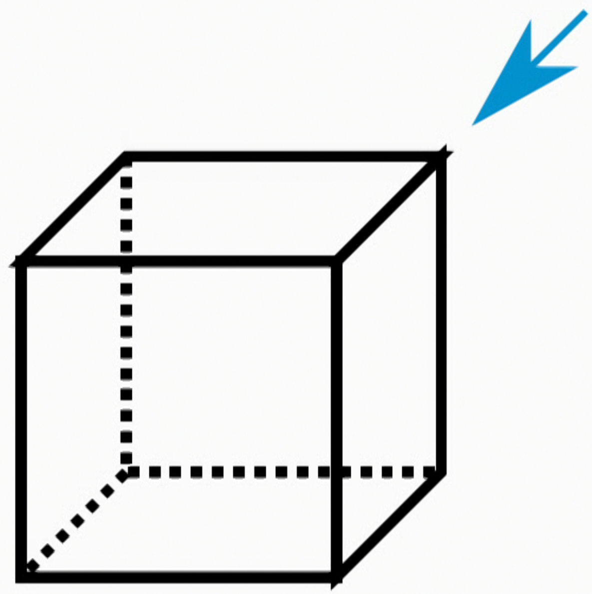
[14] Breuckmann et al, *Quantum Inf. Comput.* 17, 0181 (2017)

Sweep Rule

- Generalisation of Toom's rule to a wide range of lattices[15]
- Provable threshold (for perfect measurements)
- For 3-d toric code, numeric evidence of robustness to measurement errors, $p_{th} \approx 2\%$[16]

[15]Kubica & Preskill, arXiv:1809.10145
[16]Kubica, PhD Thesis, Caltech (2018)

Sweep Rule Decoder: 3-d Toric Code

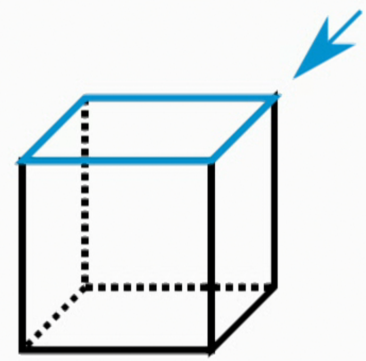


Cellular automata on vertices

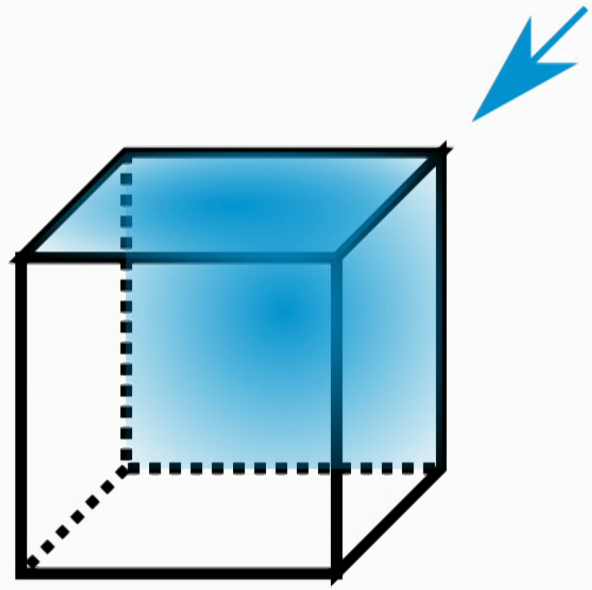
Sweep Rule Decoder: 3-d Toric Code

For each vertex v, flip face $f \ni v$ if:

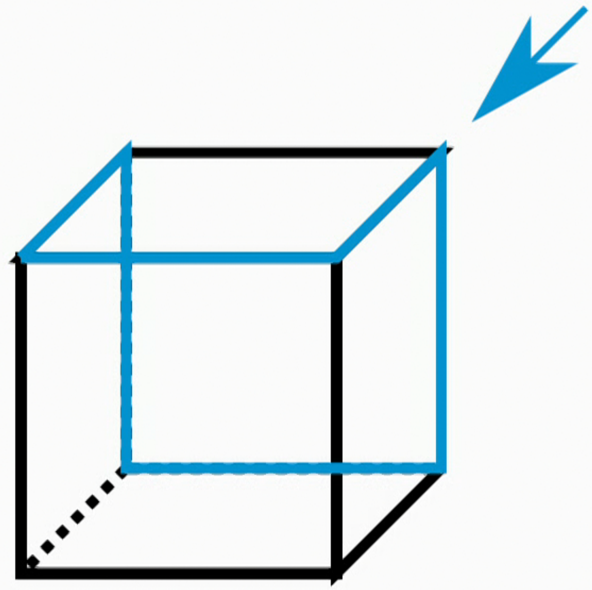
- Boundary of the face (restricted to the neighbourhood of v) matches non-zero syndrome (restricted to the neighbourhood of v)
- Above non-zero syndrome edges have positive inner product with sweep direction



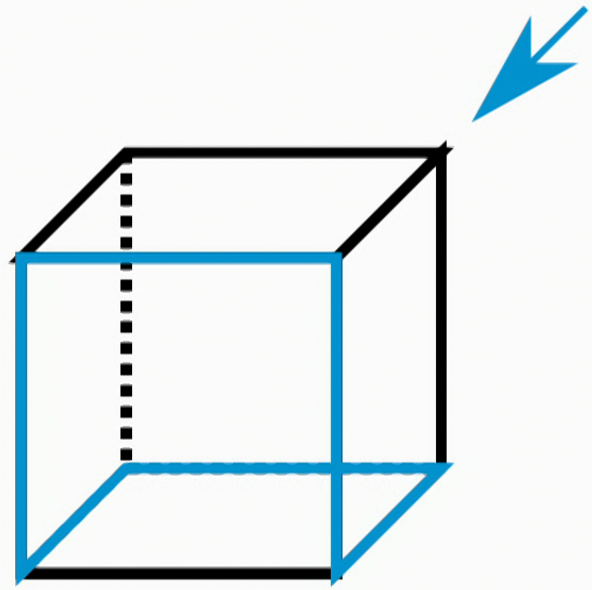
Sweep Rule Decoder: 3-d Toric Code



Sweep Rule Decoder: 3-d Toric Code

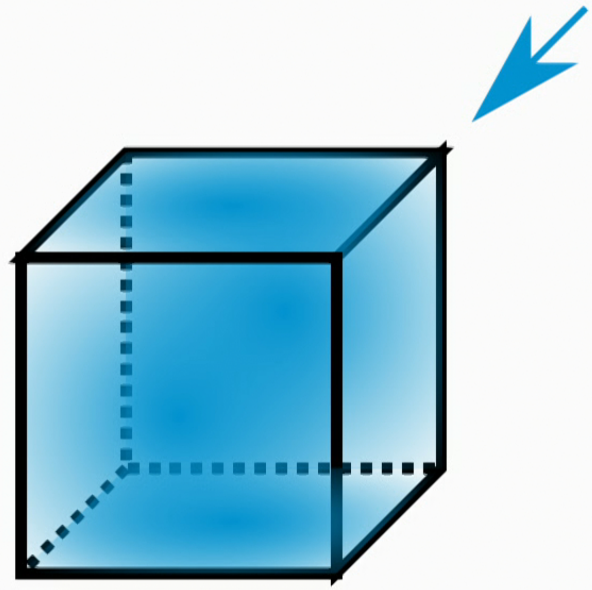


Sweep Rule Decoder: 3-d Toric Code



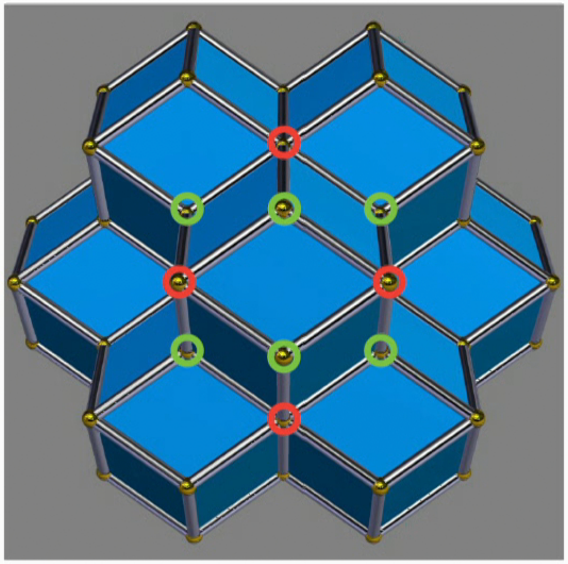
34

Sweep Rule Decoder: 3-d Toric Code



Sweep Rule Decoder: Rhombic Code

- Rhombic dodecahedral lattice is not vertex transitive
- Two types of vertex, sweep rule is different at each vertex

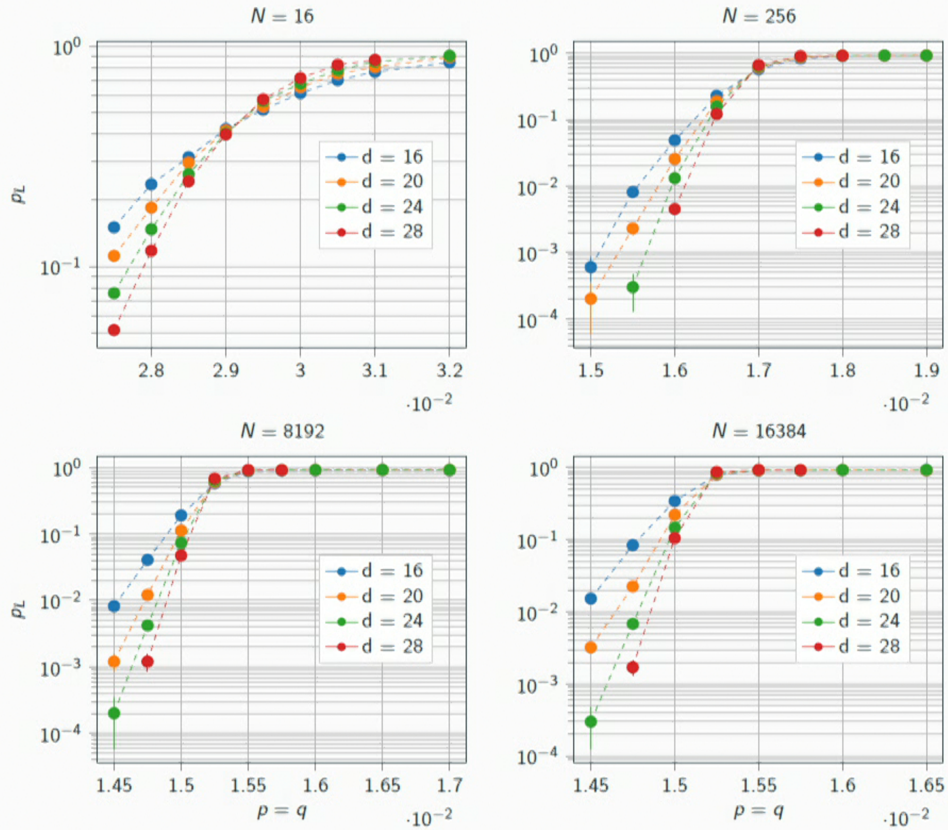


36

Sweep Rule Decoder: Rhombic Code

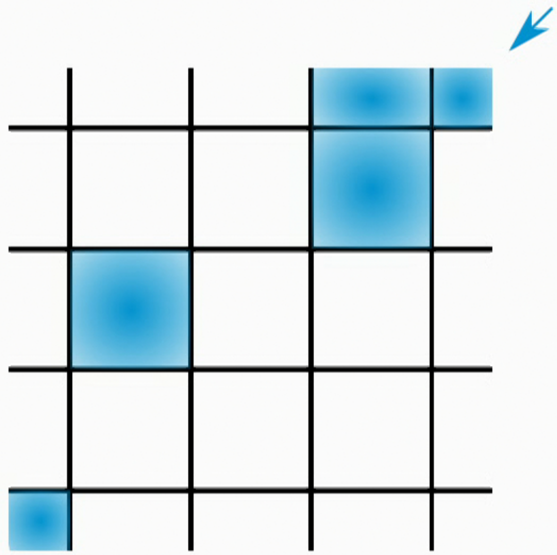
- Rhombic dodecahedral lattice with periodic boundaries
- Numeric evidence for threshold $p_{th} \approx 1.55\%$
- Investigated the sustainable threshold: threshold as a function of error correction rounds N
- In each round:
  - Qubit flips with probability p
  - Syndrome flips with probability q
  - One application of sweep rule to each vertex
- After N rounds, 'readout' (apply sweep rule with no errors $O(L)$ times), where L controls lattice size

Sweep Rule Decoder: Rhombic Code



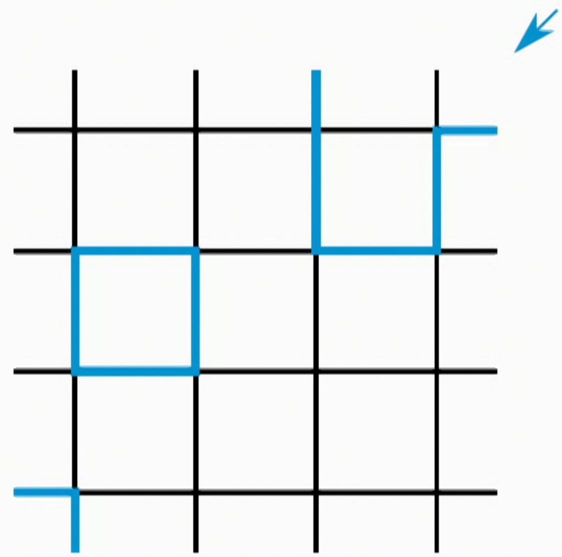
Example threshold plots

Sweep Rule with Boundaries



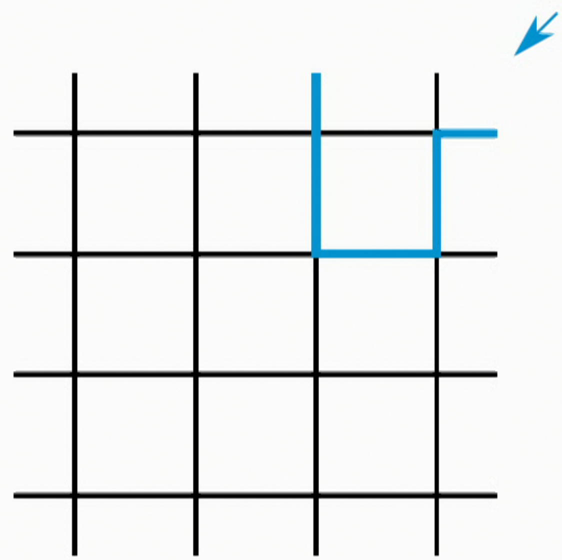
Problem: in lattices with boundaries, there are some persistent syndrome configurations which are not removed by the sweep rule

Sweep Rule with Boundaries



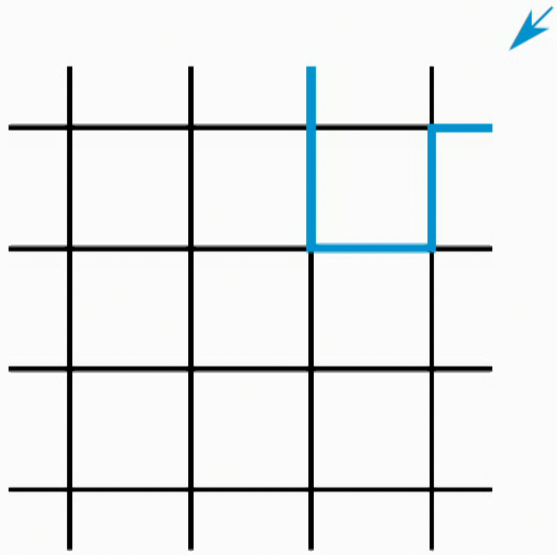
Problem: in lattices with boundaries, there are some persistent syndrome configurations which are not removed by the sweep rule

Sweep Rule with Boundaries



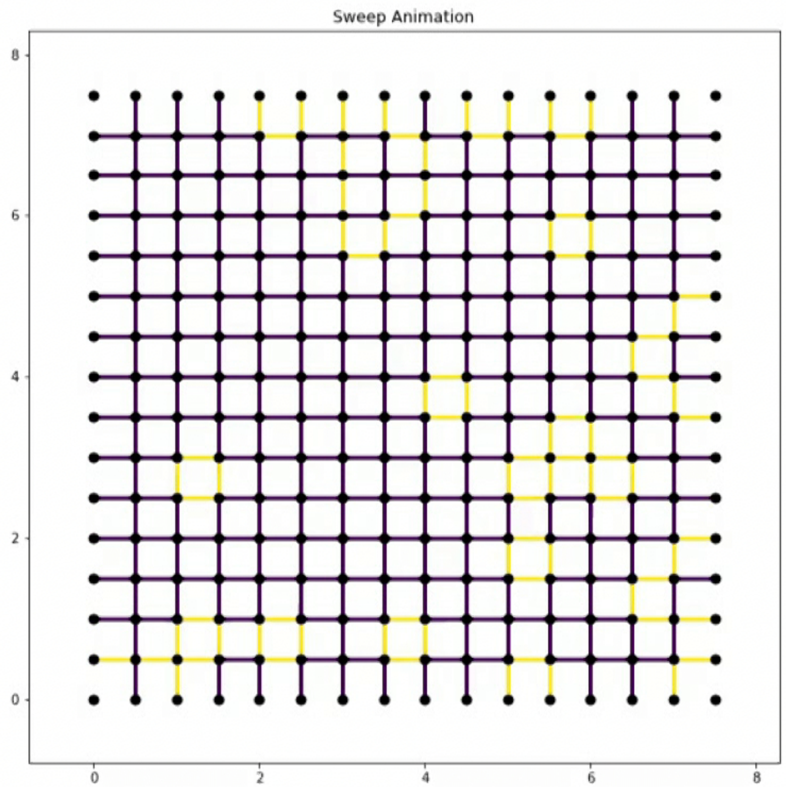
Problem: in lattices with boundaries, there are some persistent syndrome configurations which are not removed by the sweep rule

Sweep Rule with Boundaries

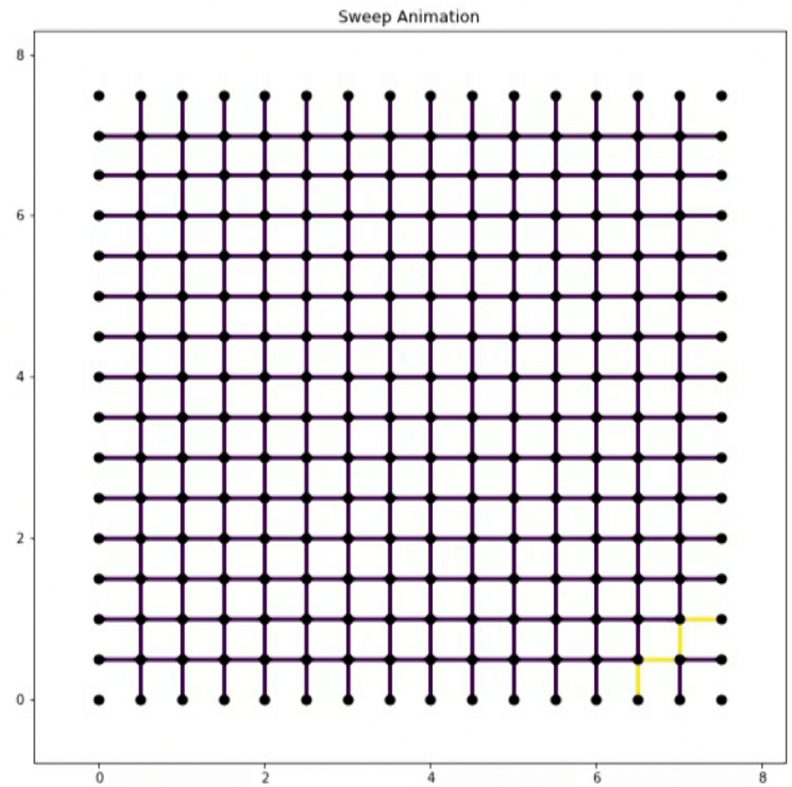


Solution: cycle the sweep direction

Sweep Rule with Boundaries

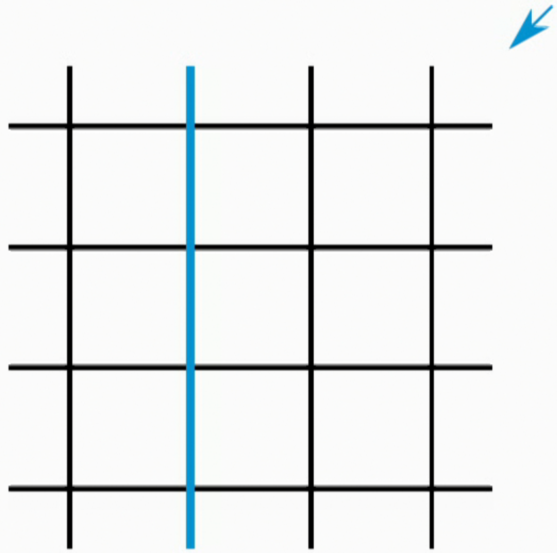


Sweep Rule with Boundaries



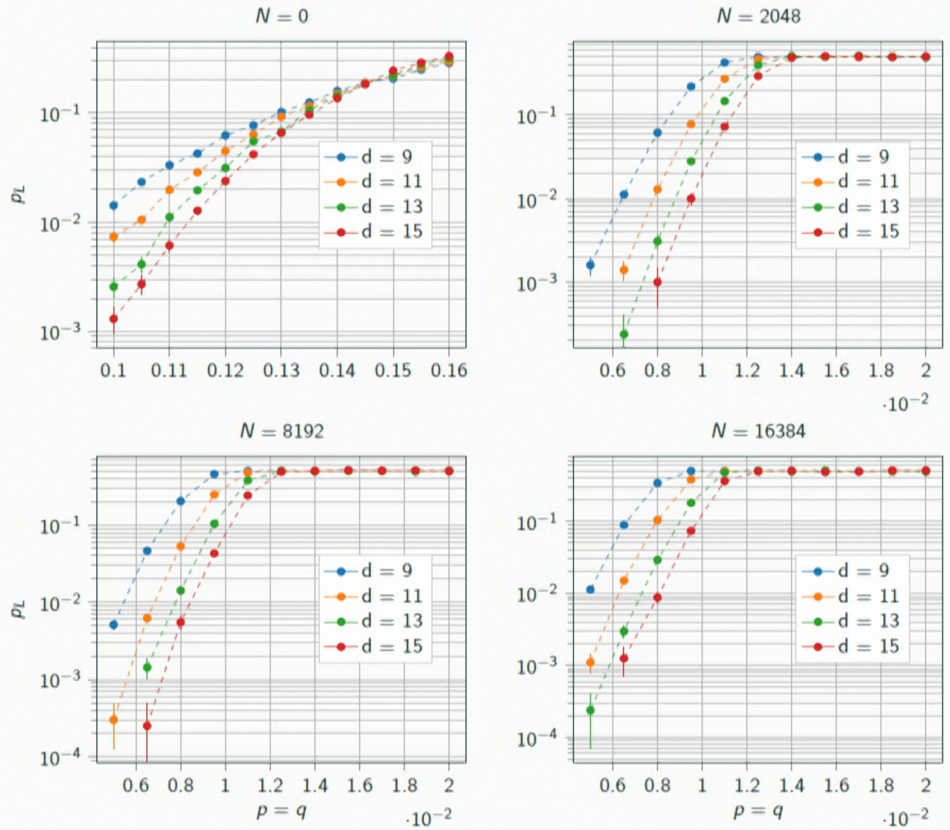
44

Sweep Rule Decoder: Cubic Code



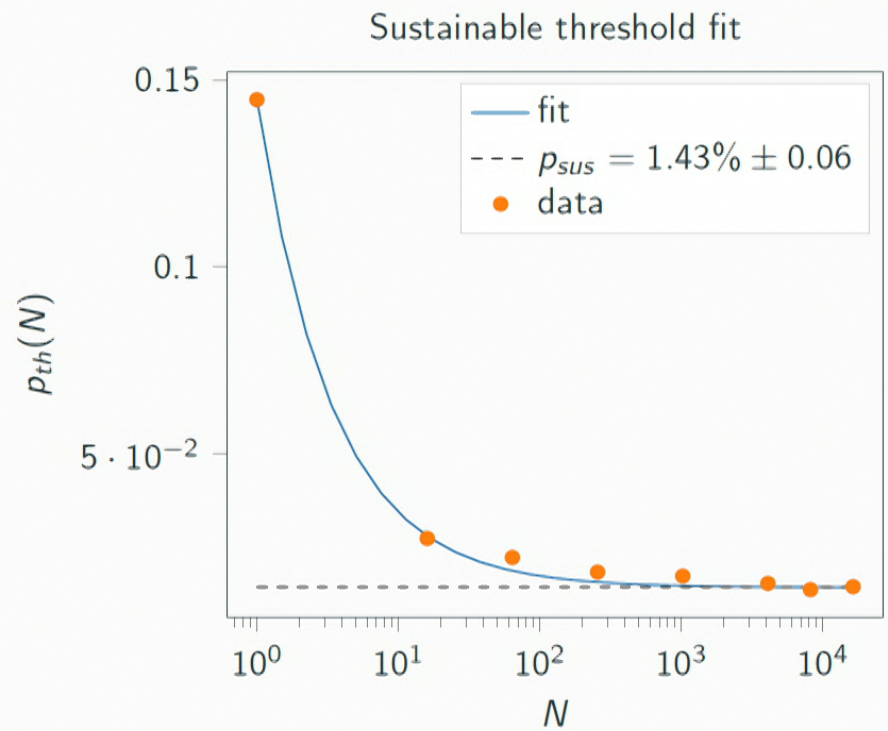
- Errors exist which sweep rule doesn't remove (unlike toric)
- Subroutine at the readout step can try to correct these errors
- Evidence of sustainable threshold $p_{sus} \approx 1.43\%$
- Need to go to larger lattice sizes to have more confidence

Sweep Rule Decoder: Cubic Code



Example threshold plots

Sweep Rule Decoder: Cubic Code



$$p_{th}(N) = p_{sus}\left(1 - \left(1 - \frac{p_{th}(1)}{p_{sus}}\right) N^{-\gamma}\right)$$

Sweep Rule Decoder: Rhombic Code

- Boundaries even trickier to deal with than the cubic case
- Some ‘faces’ only contain one edge
- Currently working out the correct sweep rule for the boundaries

Relationship Between Surface Codes and Color Codes

Previous Work

- Color codes are another family of topological stabilizer codes
- Color codes can be transformed into multiple surface codes by local Clifford unitaries[17][18][19]
- In 2-d, transformation can be understood as code concatenation[20] (concatenate two 2-d surface codes with the [[4,2,2]] error-detecting code to get a 2-d color code)

[17] Bombín et al, *New J. Phys.* 14, 073048 (2012)
[18] Delfosse, *Phys. Rev. A* 89, 912317 (2014)
[19] Kubica et al, *New J. Phys.* 17, 083026 (2015)
[20] Criger & Terhal, *Quantum Inf. Comput.* 16, 1261 (2016)

Concatenation Transformation in 3-d

- We showed that we can transform three 3-d surface codes into a 3-d color code by concatenating the three surface codes with the [[8,3,2]] error-detecting code[21]
- This transformation could be potentially realised in a near-term experiment (for small codes)

[21] Vasmer & Browne, arXiv:1801.04255

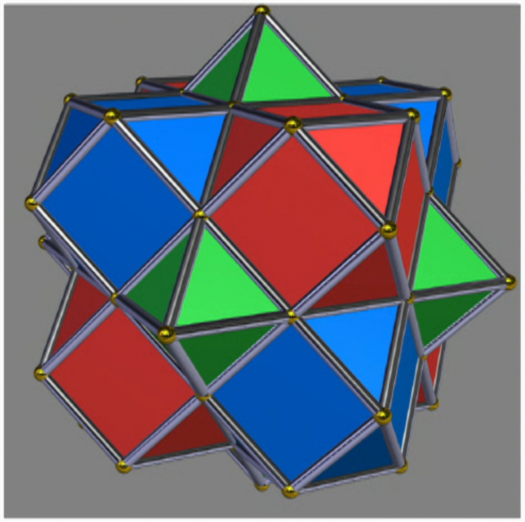
Concatenation Transformation in 3-d

- The [[8,3,2]] code is the smallest example of a 3-d color code
- Qubits on the vertices of a cube
- Single X stabilizer $X^{\otimes 8}$
- Z stabilizers associated with faces of the cube
- Non-Clifford T-gate ($T = diag(1, e^{i\pi/4})$) is transversal in this code

51

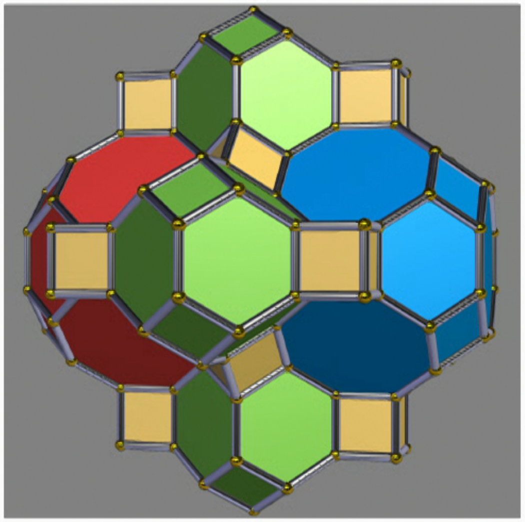
Concatenation Transformation in 3-d

The transformation between three surface codes and one color code has a nice geometric interpretation in the rectified picture



Concatenation Transformation in 3-d

The transformation between three surface codes and one color code has a nice geometric interpretation in the rectified picture



Conclusion

Summary & Open Questions

In this talk:

- Proposed 3-d surface code architecture which may have advantages over current state-of-the art architectures
- Adapted sweep rule decoder to work for codes with boundaries

Open questions:

- How does the resource overhead of our architecture compare with current state-of-the-art architectures?
- Can renormalization group decoders be designed for the rhombic dodecahedral lattice?

Acknowledgements

Joint work with Dan Browne (UCL) and Alex Kubica (PI)

EPSRC
Engineering and Physical Sciences
Research Council