Title: The ABCs of color codes

Date: Dec 12, 2016 04:00 PM

URL: http://pirsa.org/16120028

Abstract: <p>To build a fully functioning quantum computer, it is necessary to encode quantum information to protect it from noise. Topological codes, such as the color code, naturally protect against local errors and represent our best hope for storing quantum information. Moreover, a quantum computer must also be capable of processing this information. Since the color code has many computationally valuable transversal logical gates, it is a promising candidate for a future quantum computer architecture.</p>

<p>In the talk, I will provide an overview of the color code. First, I will establish a connection between the color code and a well-studied model - the toric code. Then, I will explain how one can implement a universal gate set with the subsystem and the stabilizer color codes in three dimensions using techniques of code switching and gauge fixing. Next, I will discuss the problem of decoding the color code. Finally, I will explain how one can find the optimal error correction threshold by analyzing phase transitions in certain statistical-mechanical models.</p>

<p>The talk is based on http://arxiv.org/abs/1410.0069, http://arxiv.org/abs/1503.02065 and recent works with M. Beverland, F. Brandao, N. Delfosse, J. Preskill and K. Svore.</p>

THE ABC OF COLOR CODES

Aleksander Kubica

INSTITUTE FOR QUANTUM INFORMATION AND MATTER

arXiv:1410.0069, 1503.02065

ongoing work w/ M. Beverland, F. Brandao, N. Delfosse, J. Preskill, K. Svore

1

TOWARDS QUANTUM COMPUTER

- Promises of quantum computation:
  - simulations of quantum many-body systems,
  - quantum algorithms.

TOWARDS QUANTUM COMPUTER

- Promises of quantum computation:
  - simulations of quantum many-body systems,
  - quantum algorithms.

- Interactions with environment lead to errors.

TOWARDS QUANTUM COMPUTER

- Promises of quantum computation:
  - simulations of quantum many-body systems,
  - quantum algorithms.
- Interactions with environment lead to errors.
- Scalable quantum computation possible given sufficiently weak noise!

2

TOWARDS QUANTUM COMPUTER

- Promises of quantum computation:
  - simulations of quantum many-body systems,
  - quantum algorithms.

- Interactions with environment lead to errors.

- Scalable quantum computation possible given sufficiently weak noise!

- Need for methods of controlling and correcting errors.

TOWARDS QUANTUM COMPUTER

- Promises of quantum computation:
  - simulations of quantum many-body systems,
  - quantum algorithms.

- Interactions with environment lead to errors.

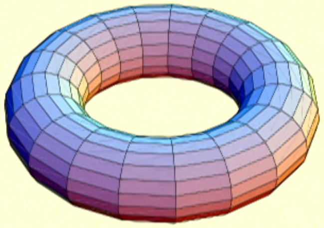
- Scalable quantum computation possible given sufficiently weak noise!

- Need for methods of controlling and correcting errors.

- Want to perform computation - logical operations should be fault-tolerant and easy to implement.

TOPOLOGICAL QUANTUM CODES

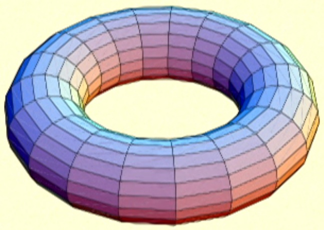
- Topological quantum codes - (geometrically) local generators, encode information in non-local degrees of freedom.



3

TOPOLOGICAL QUANTUM CODES

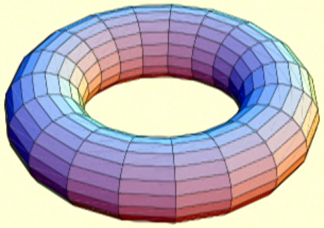
- Topological quantum codes - (geometrically) local generators, encode information in non-local degrees of freedom.

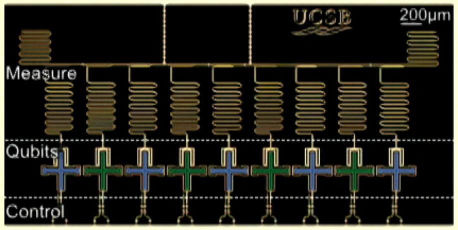


- Examples: toric code, color code.

TOPOLOGICAL QUANTUM CODES

- Topological quantum codes - (geometrically) local generators, encode information in non-local degrees of freedom.
- Examples: toric code, color code.
- Experimentally realizable: 2-dim, scalable…

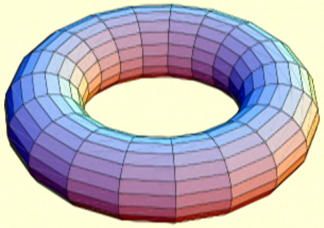


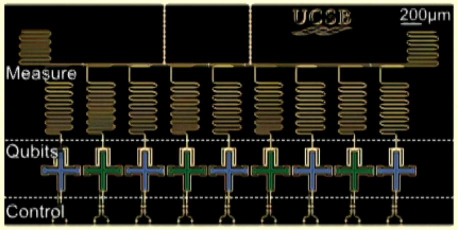


Kelly et al., Nature 519, 66–69 (2015)

TOPOLOGICAL QUANTUM CODES

- Topological quantum codes - (geometrically) local generators, encode information in non-local degrees of freedom.
- Examples: toric code, color code.
- Experimentally realizable: 2-dim, scalable…
- Important properties:
  - low-weight measurements
  - efficient decoders
  - high threshold
  - fault-tolerant logical gates

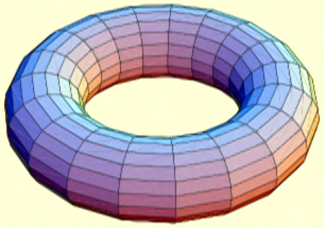


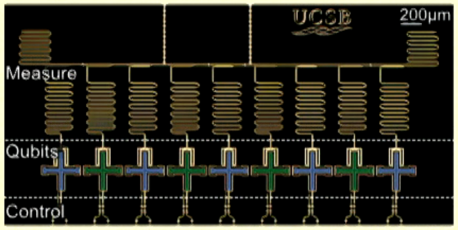


Kelly et al., Nature 519, 66–69 (2015)

TOPOLOGICAL QUANTUM CODES

- Topological quantum codes - (geometrically) local generators, encode information in non-local degrees of freedom.
- Examples: toric code, color code.
- Experimentally realizable: 2-dim, scalable…
- Important properties:
  - low-weight measurements
  - efficient decoders
  - high threshold
  - fault-tolerant logical gates





Kelly et al., Nature 519, 66–69 (2015)

- Side remark: topological codes as (exactly solvable) toy models for classification of quantum phases.

OUTLINE

COLOR CODE

4

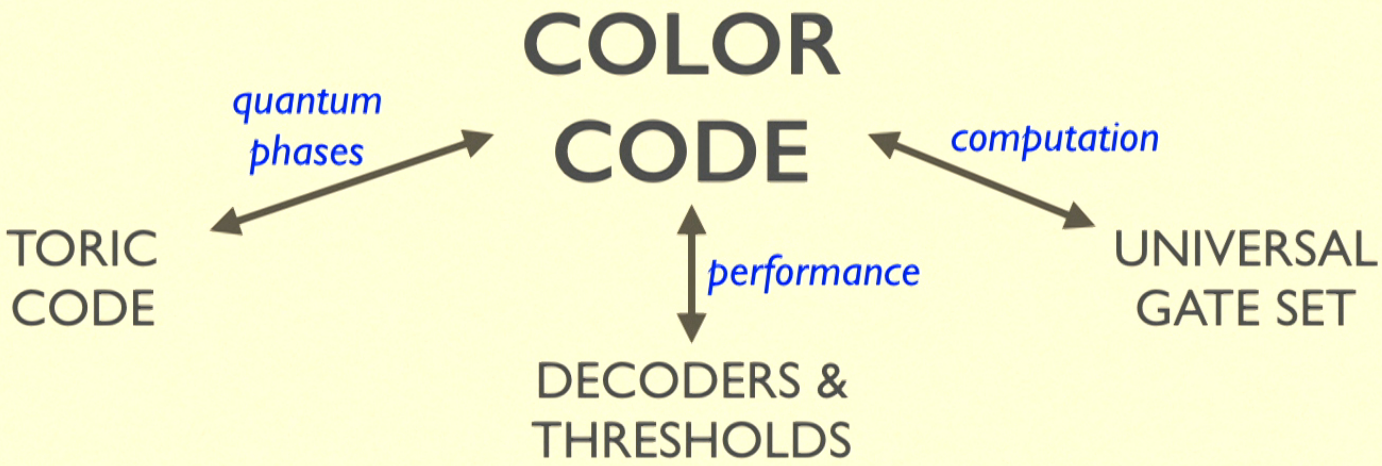
OUTLINE

COLOR CODE

TORIC CODE ⟷ *quantum phases* ⟷ COLOR CODE

COLOR CODE ⟷ *computation* ⟷ UNIVERSAL GATE SET

OUTLINE


COLOR
CODE
quantum
phases
TORIC
CODE
performance
DECODERS &
THRESHOLDS
computation
UNIVERSAL
GATE SET

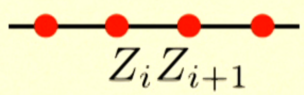
STABILIZER CODES

- Stabilizer code - stabilizers are products of Pauli operators and the code space is +1 eigenspace of stabilizers.

STABILIZER CODES

- Stabilizer code - stabilizers are products of Pauli operators and the code space is +1 eigenspace of stabilizers.
- Stabilizer Hamiltonian - commuting terms are products of Pauli operators; ground space corresponds to the code space.

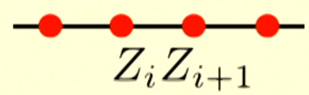
$$H = -\sum_{i=1}^{n} Z_i Z_{i+1}$$



STABILIZER CODES

- Stabilizer code - stabilizers are products of Pauli operators and the code space is +1 eigenspace of stabilizers.
- Stabilizer Hamiltonian - commuting terms are products of Pauli operators; ground space corresponds to the code space.

$$H = -\sum_{i=1}^{n} Z_i Z_{i+1}$$

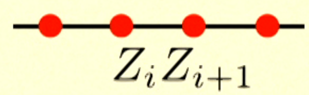


- Repeated stabilizer measurements needed to perform error correction.

STABILIZER CODES

- Stabilizer code - stabilizers are products of Pauli operators and the code space is +1 eigenspace of stabilizers.
- Stabilizer Hamiltonian - commuting terms are products of Pauli operators; ground space corresponds to the code space.

$$H = -\sum_{i=1}^{n} Z_i Z_{i+1}$$

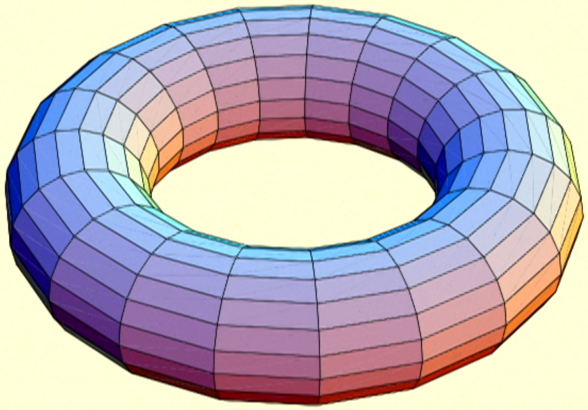


- Repeated stabilizer measurements needed to perform error correction.
- Generalization: subsystem code specified by a gauge group (might be non-Abelian).

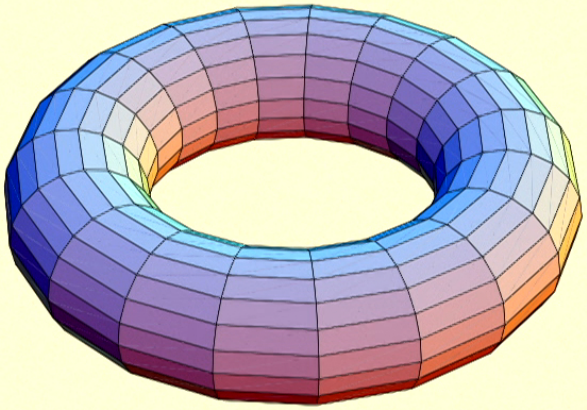
TOPOLOGICAL QUANTUM CODE: 2D TORIC CODE

6

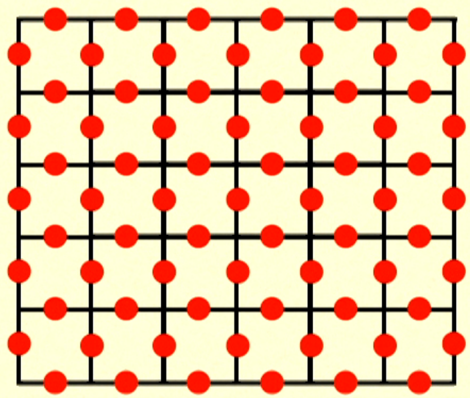
TOPOLOGICAL QUANTUM CODE: 2D TORIC CODE



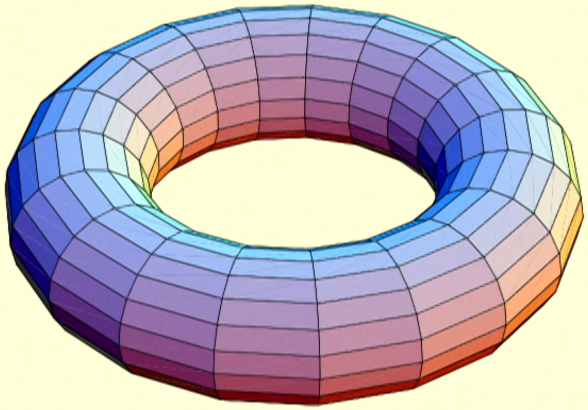
TOPOLOGICAL QUANTUM CODE: 2D TORIC CODE



- qubits on edges
- X-vertex and Z-plaquette terms



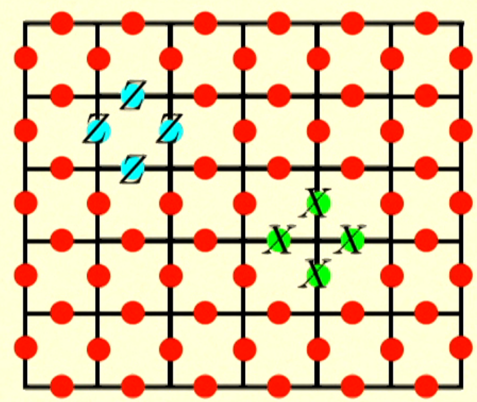
TOPOLOGICAL QUANTUM CODE: 2D TORIC CODE



- qubits on edges
- X-vertex and Z-plaquette terms

$$H = -\sum_v X(v) - \sum_p Z(p)$$

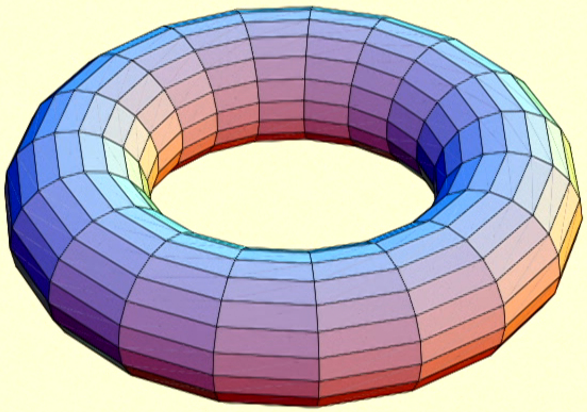
$$\forall v, p : \ [X(v), Z(p)] = 0$$



- code space C = ground space of H

$$\prod_v X(v) = \prod_p Z(p) = I$$

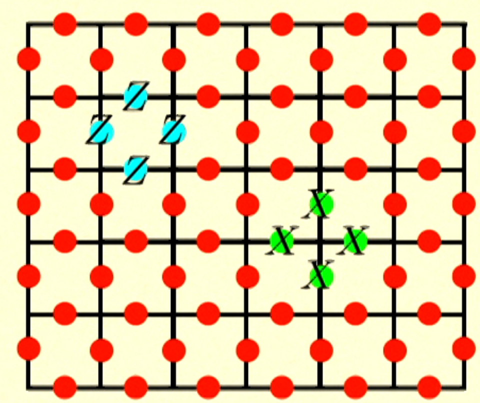
TOPOLOGICAL QUANTUM CODE: 2D TORIC CODE



- qubits on edges
- X-vertex and Z-plaquette terms

$$H = -\sum_v X(v) - \sum_p Z(p)$$

$$\forall v, p: \; [X(v), Z(p)] = 0$$



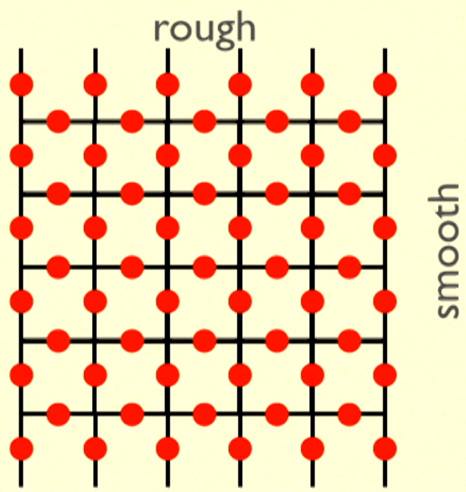
- code space C = ground space of H

$$\prod_v X(v) = \prod_p Z(p) = I$$

- degeneracy(C) = 2^2

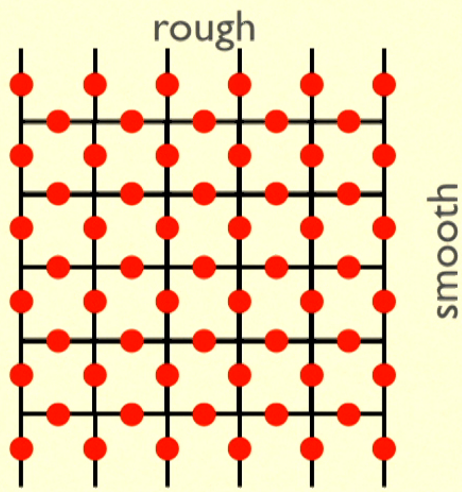
2D TORIC CODE WITH BOUNDARIES

- 2 dim toric code with boundaries: rough and smooth



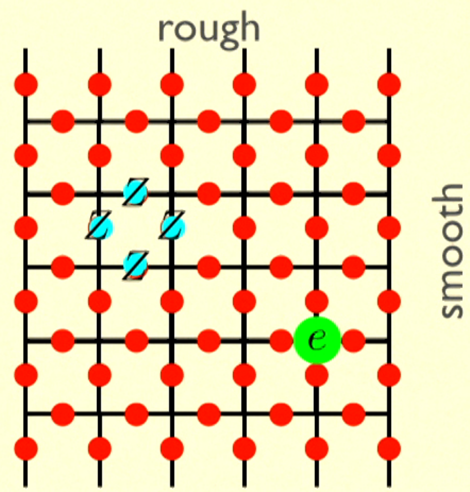
2D TORIC CODE WITH BOUNDARIES

- 2 dim toric code with boundaries: rough and smooth
- excitations (violated stabilizers): electric *e* and magnetic *m*



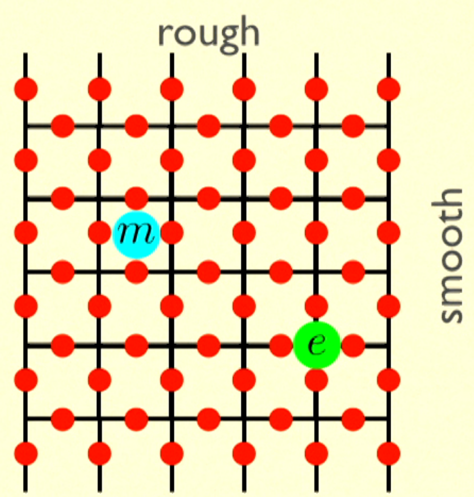
2D TORIC CODE WITH BOUNDARIES

- 2 dim toric code with boundaries: rough and smooth
- excitations (violated stabilizers): electric *e* and magnetic *m*



2D TORIC CODE WITH BOUNDARIES

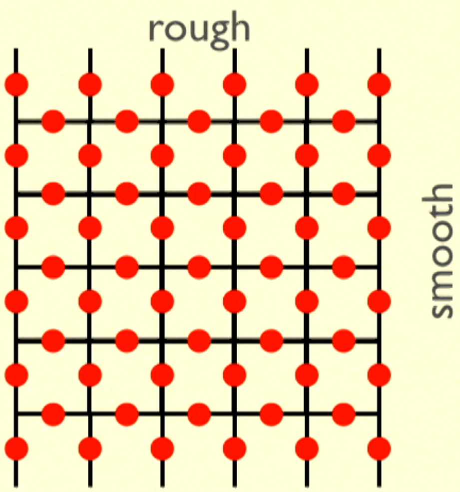
- 2 dim toric code with boundaries: rough and smooth
- excitations (violated stabilizers): electric *e* and magnetic *m*
- bulk fusion rules: $e \times e = I$ and $m \times m = I$.



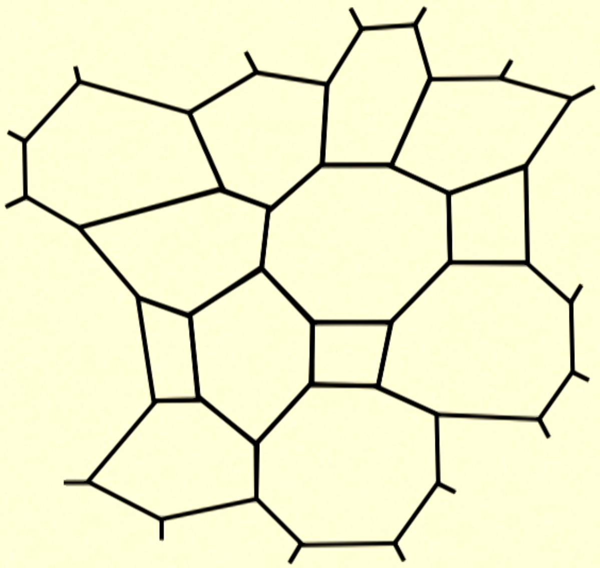
7

2D TORIC CODE WITH BOUNDARIES

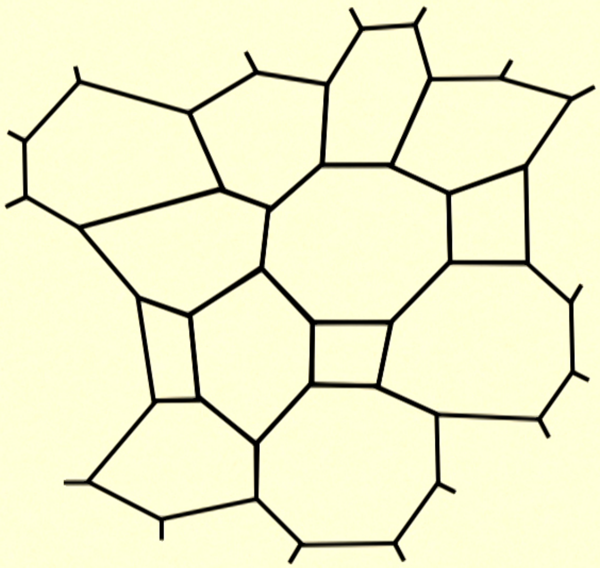
- 2 dim toric code with boundaries: rough and smooth
- excitations (violated stabilizers): electric *e* and magnetic *m*
- bulk fusion rules: $e \times e = 1$ and $m \times m = 1$.
- 1 logical qubit and string-like logical operators



TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE

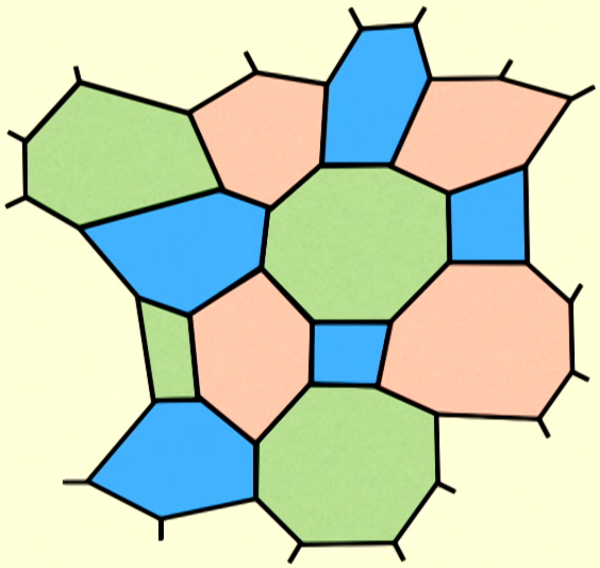


TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE



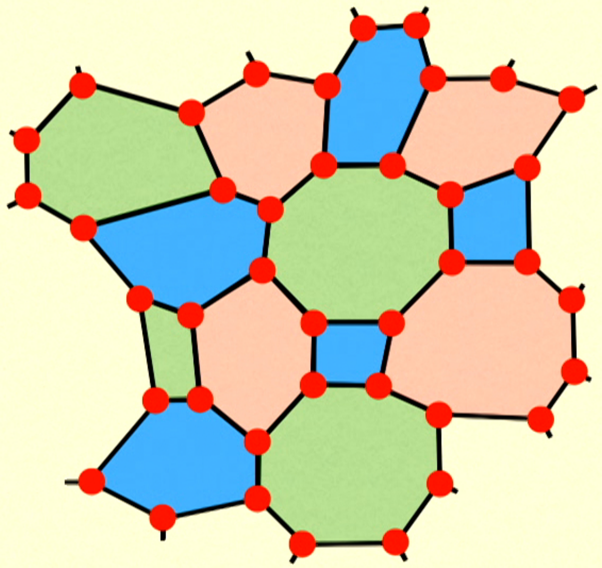
- 2 dim lattice:
  - 3-valent
  - 3-colorable

TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE



- 2 dim lattice:
  - 3-valent
  - 3-colorable

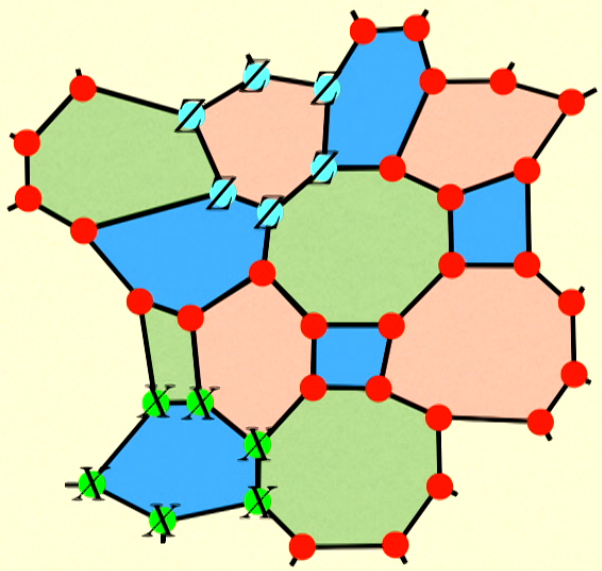
TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE



- 2 dim lattice:
  - 3-valent
  - 3-colorable
- qubits on vertices
- plaquette terms

8

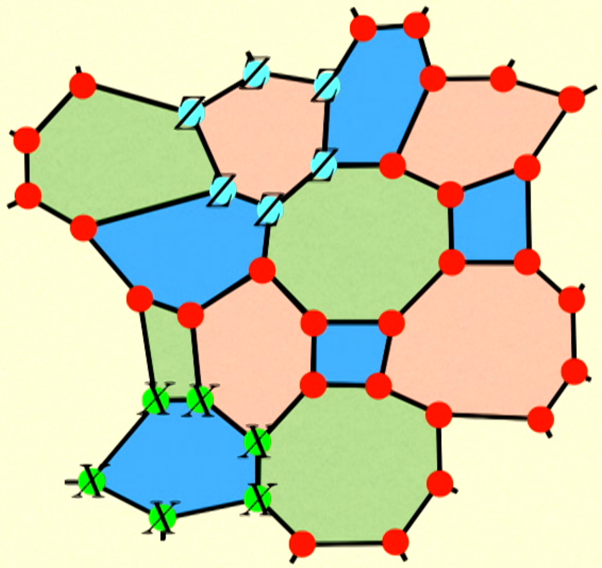
TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE



- 2 dim lattice:
  - 3-valent
  - 3-colorable
- qubits on vertices
- plaquette terms

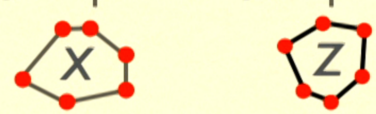
$$H = -\sum_p X(p) - \sum_p Z(p)$$

TOPOLOGICAL QUANTUM CODE: 2D COLOR CODE



- 2 dim lattice:
  - 3-valent
  - 3-colorable
- qubits on vertices
- plaquette terms

$$H = -\sum_p X(p) - \sum_p Z(p)$$



$$\forall p, p' : \; [X(p), Z(p')] = 0$$

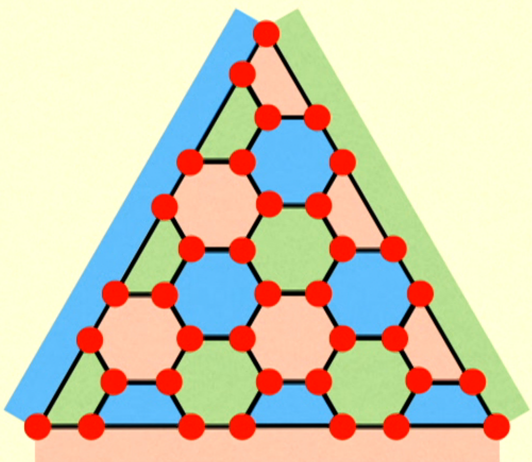
- code space C = ground space of H
- degeneracy(C) = 2^{4g}, where g - genus

2D COLOR CODE WITH BOUNDARIES

- 2 dim color code with boundaries: red, green and blue

2D COLOR CODE WITH BOUNDARIES

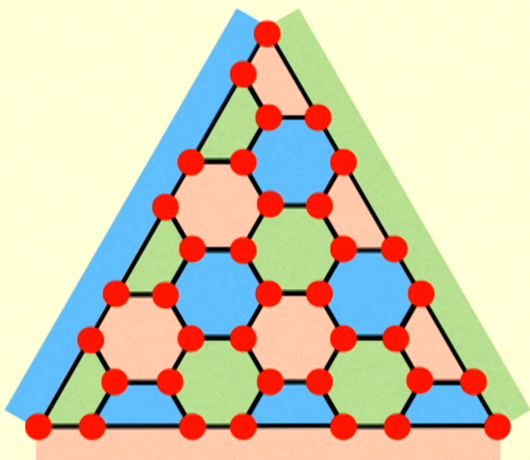
- 2 dim color code with boundaries: red, green and blue



9

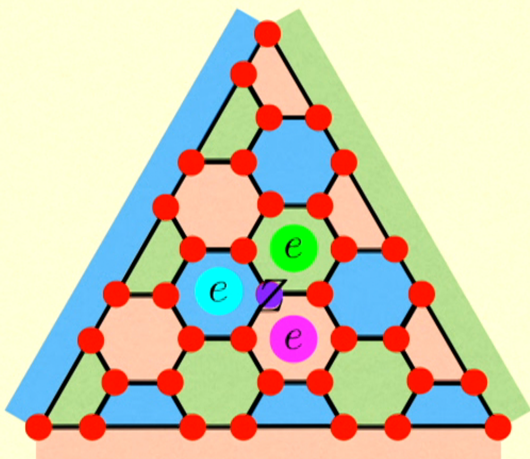
2D COLOR CODE WITH BOUNDARIES

- 2 dim color code with boundaries: red, green and blue
- red/green/blue excitations of X-type e_R, e_G, e_B, and Z-type m_R, m_G, m_B.



2D COLOR CODE WITH BOUNDARIES

- 2 dim color code with boundaries: red, green and blue
- red/green/blue excitations of X-type e_R, e_G, e_B, and Z-type m_R, m_G, m_B.
- bulk fusion rules: $e_R \times e_G \times e_B = I$ and $m_R \times m_G \times m_B = I$.



- **Main results (Kubica, Yoshida, Pastawski'15):**
  - in $d \geq 2$ dim, color code equivalent to toric code
  - explicit construction of unitary mapping

CLASSIFICATION OF QUANTUM PHASES OF MATTER

arXiv:1503.02065

CLASSIFICATION OF QUANTUM PHASES OF MATTER

- Classification of quantum phases as one of the central problems in many-body physics. Question of finding a self-correcting quantum memory.

arXiv:1503.02065

CLASSIFICATION OF QUANTUM PHASES OF MATTER

- Classification of quantum phases as one of the central problems in many-body physics. Question of finding a self-correcting quantum memory.
- Two gapped ground states belong to the same phase iff they are related by a local unitary evolution (**Hastings&Wen'05, Chen et al.'10**).

arXiv:**1503.02065**

CLASSIFICATION OF QUANTUM PHASES OF MATTER

- Classification of quantum phases as one of the central problems in many-body physics. Question of finding a self-correcting quantum memory.
- Two gapped ground states belong to the same phase iff they are related by a local unitary evolution (**Hastings&Wen'05, Chen et al.'10**).
- Equivalence of codes = ground states of stabilizer Hamiltonians in the same quantum phase.

arXiv:**1503.02065**

CLASSIFICATION OF QUANTUM PHASES OF MATTER

- Classification of quantum phases as one of the central problems in many-body physics. Question of finding a self-correcting quantum memory.
- Two gapped ground states belong to the same phase iff they are related by a local unitary evolution (**Hastings&Wen'05, Chen et al.'10**).
- Equivalence of codes = ground states of stabilizer Hamiltonians in the same quantum phase.
- **Yoshida'10, Bombin'11**: 2D stabilizer Hamiltonians w/ local interactions, translation and scale symmetries are equivalent to toric code.

arXiv:**1503.02065**

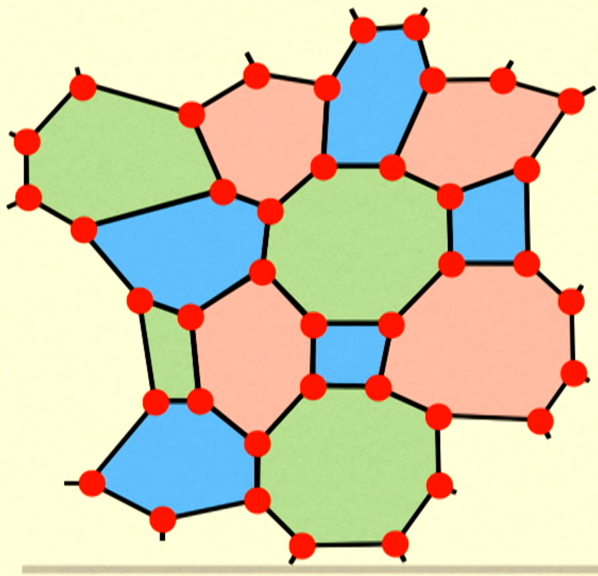
CLASSIFICATION OF QUANTUM PHASES OF MATTER

- Classification of quantum phases as one of the central problems in many-body physics. Question of finding a self-correcting quantum memory.
- Two gapped ground states belong to the same phase iff they are related by a local unitary evolution (**Hastings&Wen'05, Chen et al.'10**).
- Equivalence of codes = ground states of stabilizer Hamiltonians in the same quantum phase.
- **Yoshida'10, Bombin'11**: 2D stabilizer Hamiltonians w/ local interactions, translation and scale symmetries are equivalent to toric code.
- **Our results**: beyond 2D, any lattice, include boundaries, explicit mapping!

arXiv:**1503.02065**

TRANSFORMATION IN 2D

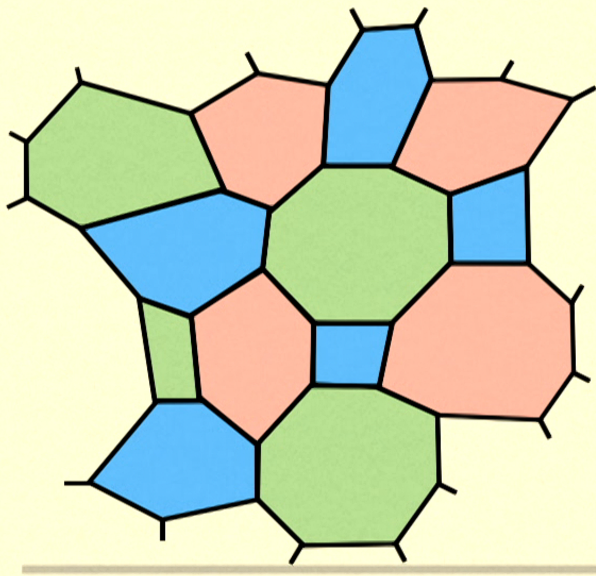
- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.



arXiv: 1503.02065

TRANSFORMATION IN 2D

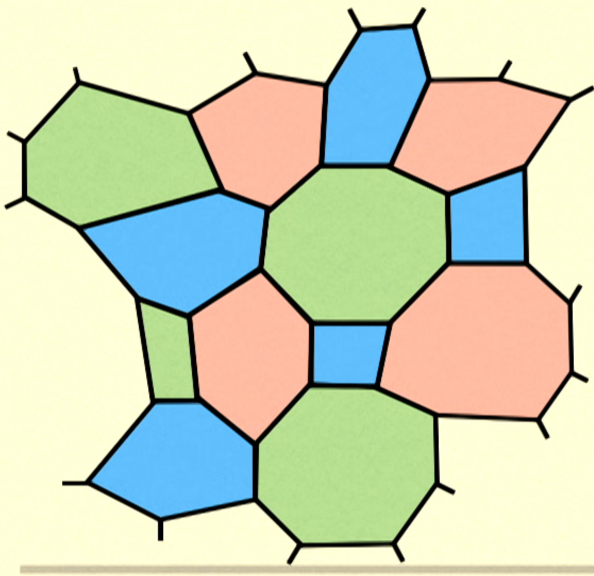
- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.
- We can construct two TC lattices from the starting CC lattice.

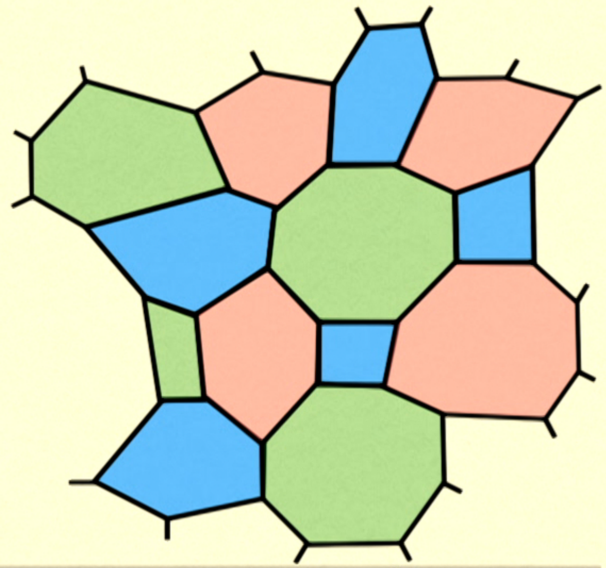


arXiv:**1503.02065**

TRANSFORMATION IN 2D

- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.
- We can construct two TC lattices from the starting CC lattice.

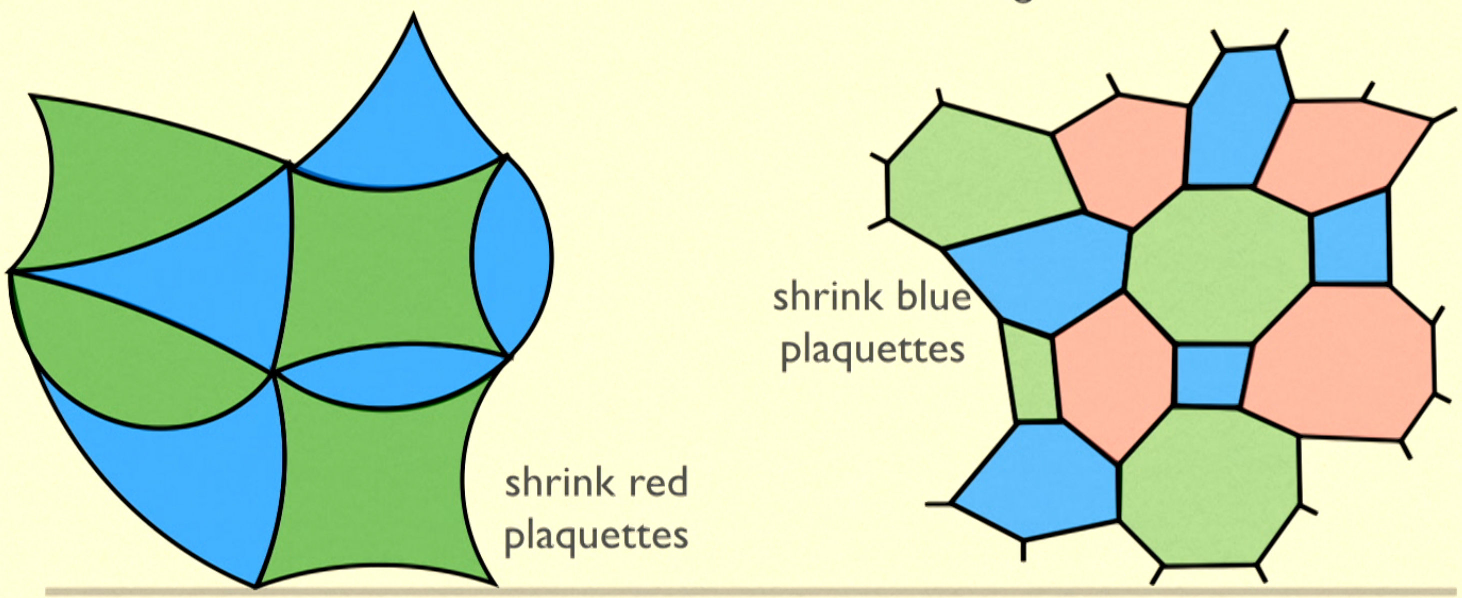




arXiv:1503.02065

TRANSFORMATION IN 2D

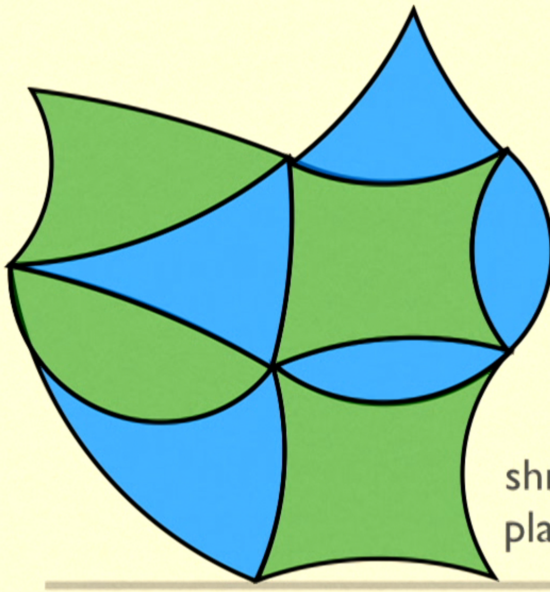
- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.
- We can construct two TC lattices from the starting CC lattice.



arXiv:**1503.02065**

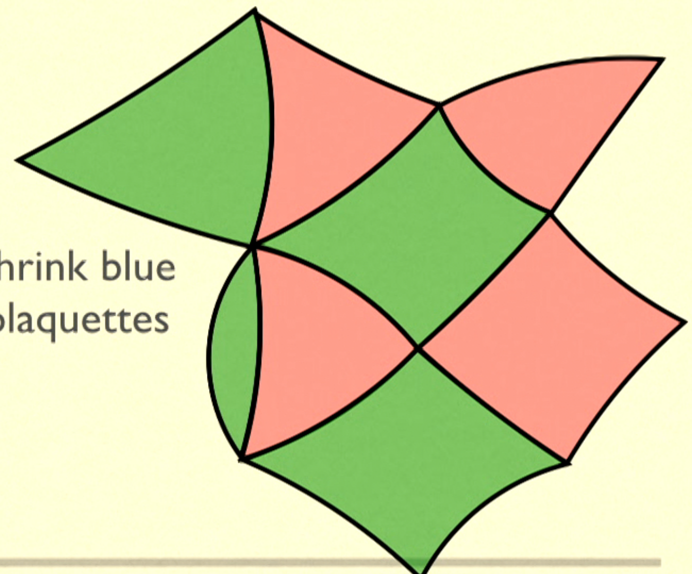
TRANSFORMATION IN 2D

- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.
- We can construct two TC lattices from the starting CC lattice.



shrink red plaquettes

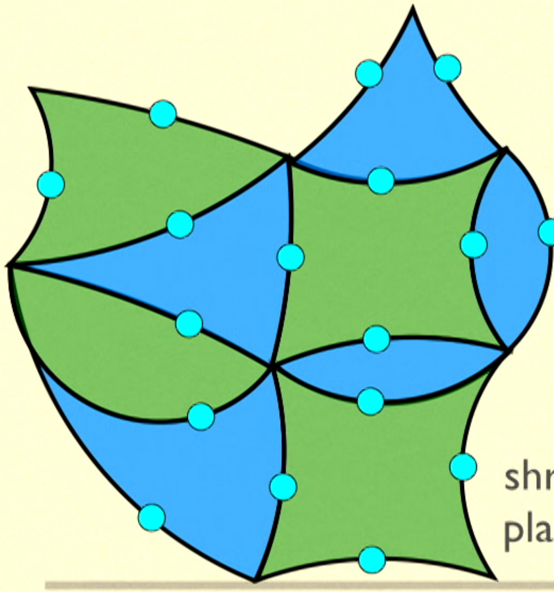
shrink blue plaquettes

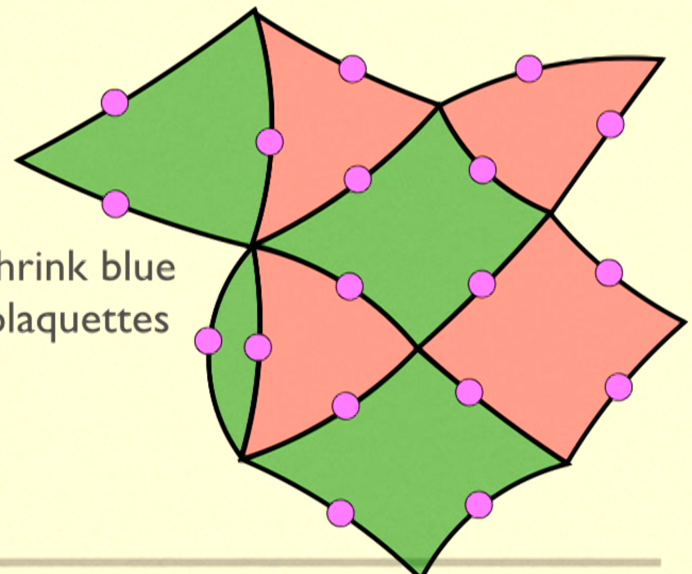


arXiv:**1503.02065**

TRANSFORMATION IN 2D

- Color code encodes twice as many logical qubits as toric code. This suggests that CC = 2 x TC.
- We can construct two TC lattices from the starting CC lattice.





arXiv:1503.02065

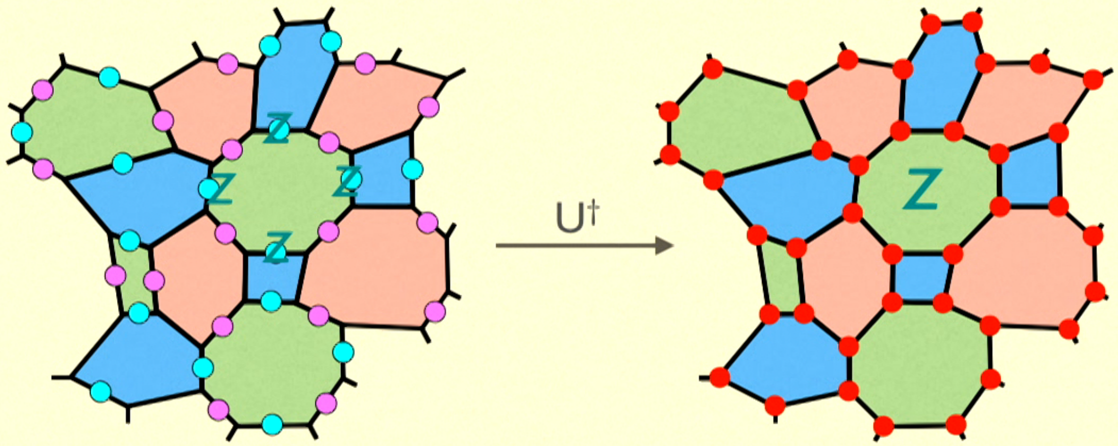
TRANSFORMATION IN 2D

- We want a local unitary $U^\dagger$ transforming X-vertex and Z-plaquette terms of two toric codes into X- and Z-plaquette terms of color code.

arXiv:1503.02065

TRANSFORMATION IN 2D

- We want a local unitary $U^\dagger$ transforming X-vertex and Z-plaquette terms of two toric codes into X- and Z-plaquette terms of color code.

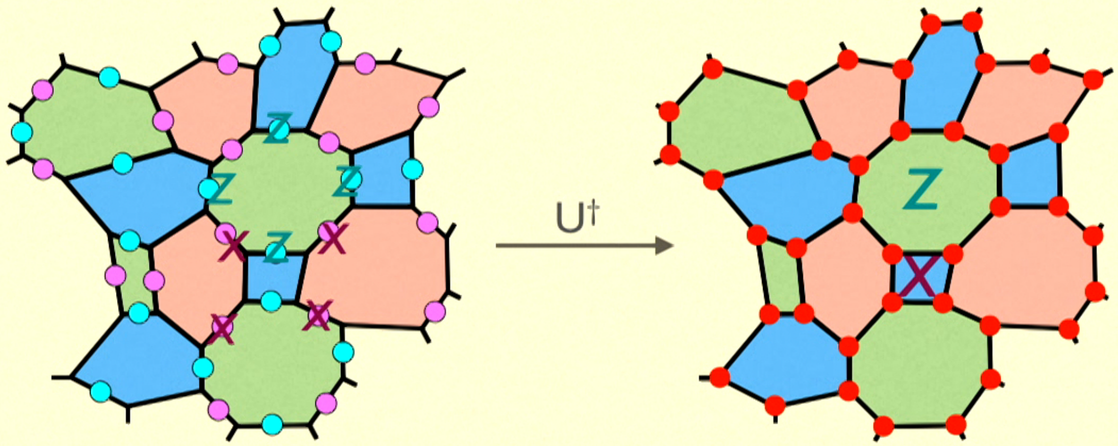


13

arXiv:**1503.02065**

TRANSFORMATION IN 2D

- We want a local unitary $U^\dagger$ transforming X-vertex and Z-plaquette terms of two toric codes into X- and Z-plaquette terms of color code.

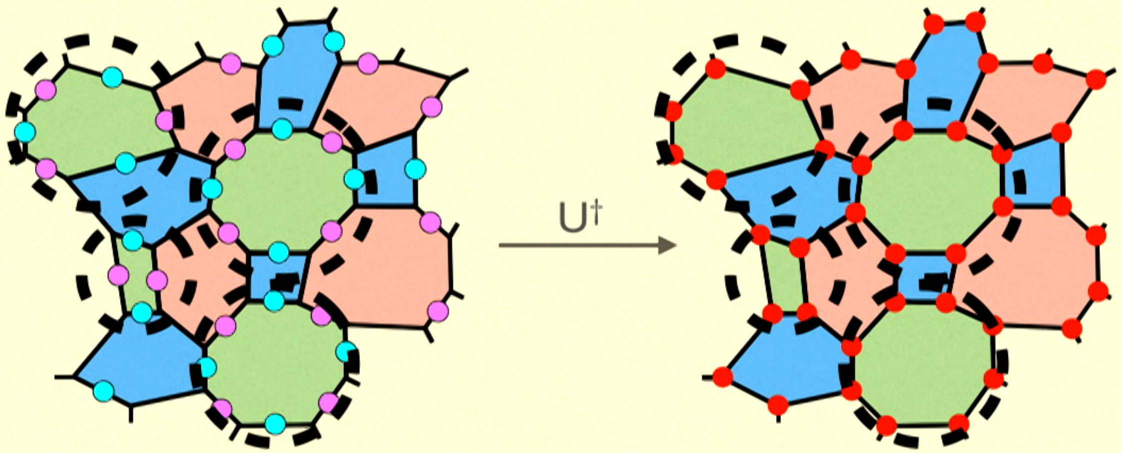


- Unitary U exists = tensor product of unitaries U_p on green plaquettes.

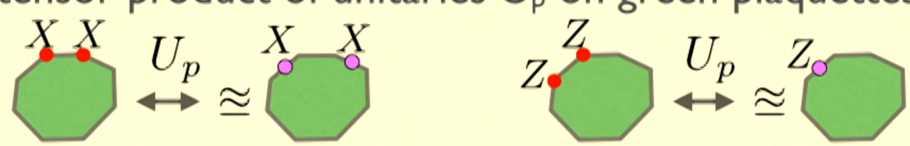
arXiv:**1503.02065**

TRANSFORMATION IN 2D

- We want a local unitary $U^\dagger$ transforming X-vertex and Z-plaquette terms of two toric codes into X- and Z-plaquette terms of color code.



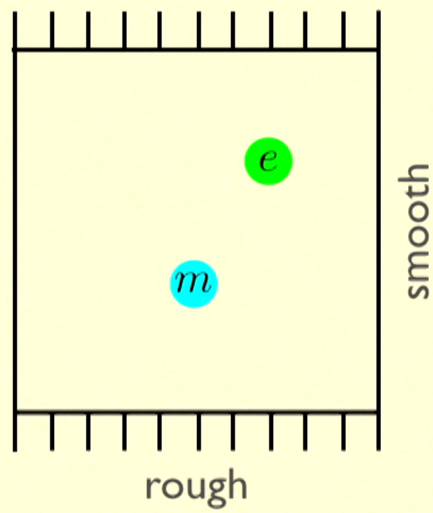
- Unitary U exists = tensor product of unitaries U_p on green plaquettes.
- U specified by :



arXiv:1503.02065

BOUNDARIES AND CONDENSATION OF ANYONS

- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.

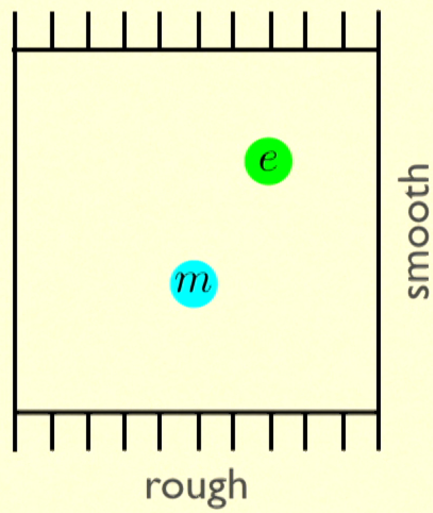


14

arXiv:**1503.02065**

BOUNDARIES AND CONDENSATION OF ANYONS

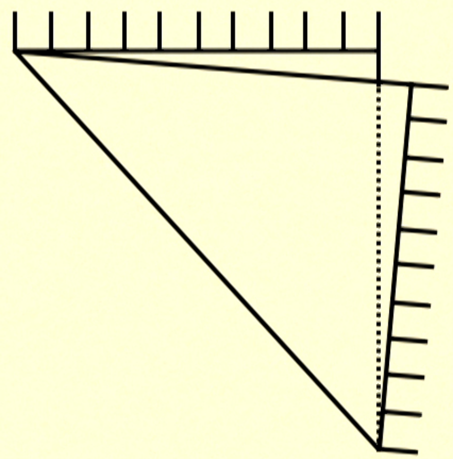
- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.
- **Toric code:** e - rough, m - smooth.



arXiv:1503.02065

BOUNDARIES AND CONDENSATION OF ANYONS

- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.
- **Toric code:** e - rough, m - smooth.
- **Folded toric code:**

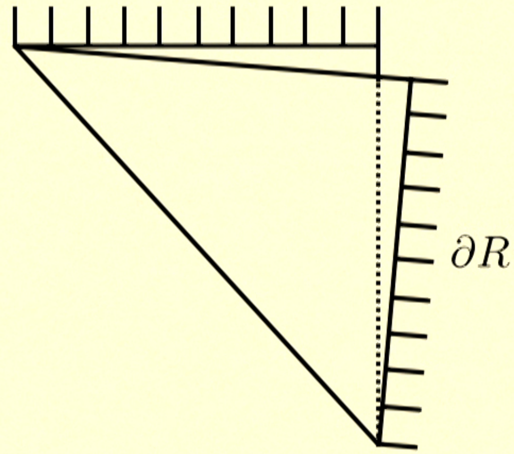


arXiv:**1503.02065**

BOUNDARIES AND CONDENSATION OF ANYONS

- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.
- **Toric code:** e - rough, m - smooth.
- **Folded toric code:**

$\partial R = \{1, e_1, m_2, e_1 m_2\}$ rough/smooth

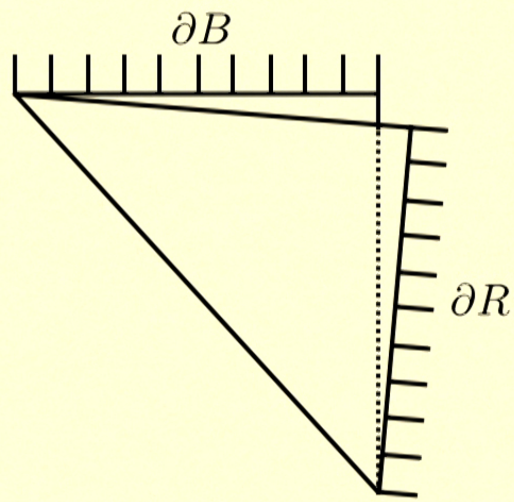


arXiv:**1503.02065**

BOUNDARIES AND CONDENSATION OF ANYONS

- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.
- **Toric code:** e - rough, m - smooth.
- **Folded toric code:**

$\partial R = \{1, e_1, m_2, e_1 m_2\}$ rough/smooth
$\partial B = \{1, e_2, m_1, e_2 m_1\}$ smooth/rough



arXiv:**1503.02065**

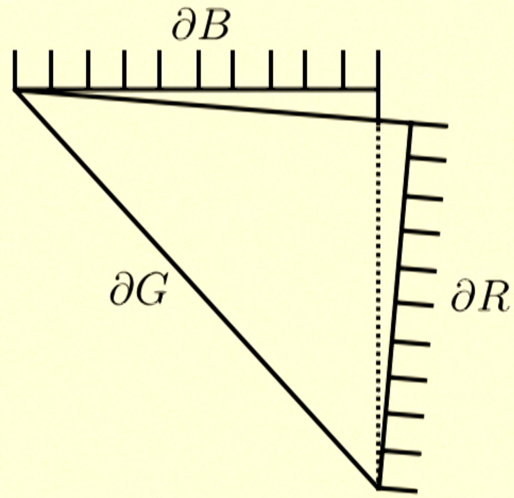
BOUNDARIES AND CONDENSATION OF ANYONS

- **Fact**: anyons condensing onto a gapped boundary have mutually trivial statistics.
- **Toric code:** e - rough, m - smooth.
- **Folded toric code:**

$\partial R = \{1, e_1, m_2, e_1 m_2\}$ rough/smooth
$\partial B = \{1, e_2, m_1, e_2 m_1\}$ smooth/rough
$\partial G = \{1, e_1 e_2, m_1 m_2, \epsilon_1 \epsilon_2\}$ fold

- Correspondence between anyonic excitations in toric code and color code:

$e_1 \equiv R_X \qquad e_2 \equiv B_X$
$m_2 \equiv R_Z \qquad m_1 \equiv B_Z$



arXiv:**1503.02065**

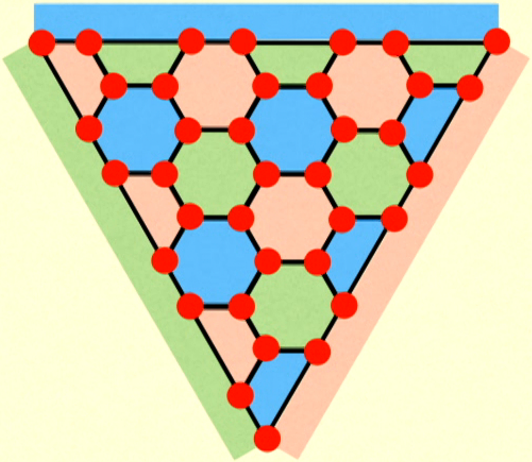
COLOR CODE UNFOLDED

- Correspondence between excitations suggests that color code is equivalent to folded toric code!

arXiv:**1503.02065**

COLOR CODE UNFOLDED

- Correspondence between excitations suggests that color code is equivalent to folded toric code!

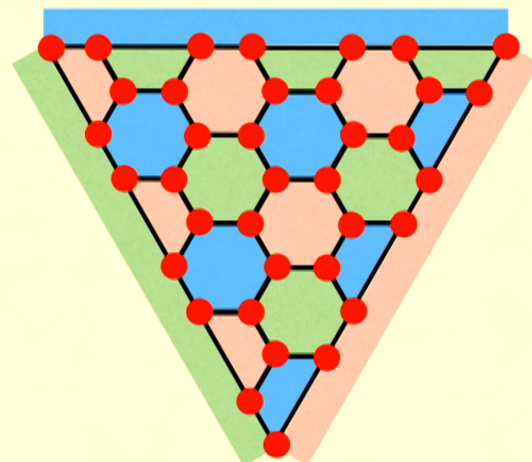


15

arXiv:**1503.02065**

COLOR CODE UNFOLDED

- Correspondence between excitations suggests that color code is equivalent to folded toric code!

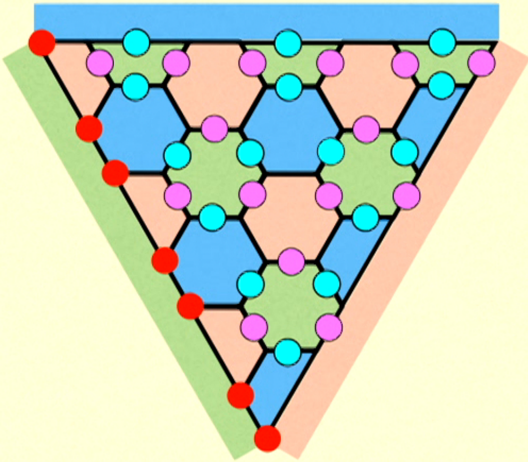


- Apply local unitaries on green faces to map CC stabilizers into TC term.

arXiv:1503.02065

COLOR CODE UNFOLDED

- Correspondence between excitations suggests that color code is equivalent to folded toric code!

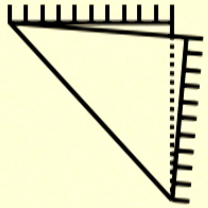


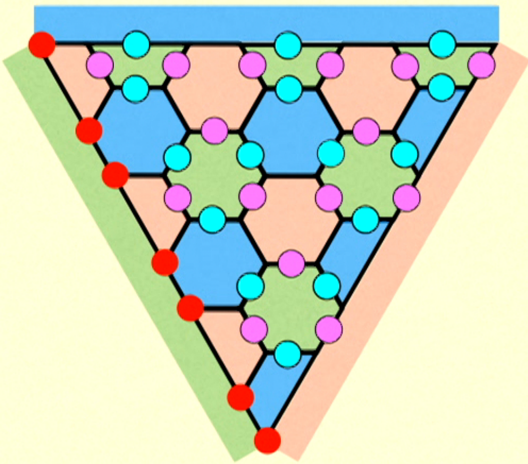
- Apply local unitaries on green faces to map CC stabilizers into TC term.
- Note: two copies of toric code not decoupled along the green boundary! 15

arXiv: **1503.02065**

COLOR CODE UNFOLDED

- Correspondence between excitations suggests that color code is equivalent to folded toric code!





- Apply local unitaries on green faces to map CC stabilizers into TC term.
- Note: two copies of toric code not decoupled along the green boundary!

arXiv: **1503.02065**

EQUIVALENCE IN D DIMENSIONS WITH OR WITHOUT BOUNDARIES

Theorem: there exists a unitary $U = \bigotimes_\delta U_\delta$, which is a tensor product of local terms with disjoint support, such that U transforms the color code into $n = \binom{d}{k-1}$ decoupled copies of the toric code.

$$U\left[CC_k(\mathcal{L})\right]U^\dagger = \bigotimes_{i=1}^{n} TC_{k-1}(\mathcal{L}_i)$$

arXiv:**1503.02065**

EQUIVALENCE IN D DIMENSIONS WITH OR WITHOUT BOUNDARIES

Theorem: there exists a unitary $U = \bigotimes_\delta U_\delta$, which is a tensor product of local terms with disjoint support, such that U transforms the color code into $n = \binom{d}{k-1}$ decoupled copies of the toric code.

$$U\left[CC_k(\mathcal{L})\right]U^\dagger = \bigotimes_{i=1}^{n} TC_{k-1}(\mathcal{L}_i)$$

CC_k(L): qubits - 0-cells
X stabilizers - (d+2-k)-cells
Z stabilizers - k-cells

arXiv:**1503.02065**

EQUIVALENCE IN D DIMENSIONS WITH OR WITHOUT BOUNDARIES

Theorem: there exists a unitary $U = \bigotimes_\delta U_\delta$, which is a tensor product of local terms with disjoint support, such that U transforms the color code into $n = \binom{d}{k-1}$ decoupled copies of the toric code.

$$U\left[CC_k(\mathcal{L})\right]U^\dagger = \bigotimes_{i=1}^{n} TC_{k-1}(\mathcal{L}_i)$$

obtained from $\mathcal{L}$ by local deformations

CC_k(L): qubits - 0-cells
X stabilizers - (d+2-k)-cells
Z stabilizers - k-cells

TC_k(L): qubits - k-cells
X stabilizers - (k-1)-cells
Z stabilizers - (k+1)-cells

arXiv:**1503.02065**

EQUIVALENCE IN D DIMENSIONS WITH OR WITHOUT BOUNDARIES

Theorem: there exists a unitary $U = \bigotimes_\delta U_\delta$, which is a tensor product of local terms with disjoint support, such that U transforms the color code into $n = \binom{d}{k-1}$ decoupled copies of the toric code.

$$U\left[CC_k(\mathcal{L})\right]U^\dagger = \bigotimes_{i=1}^{n} TC_{k-1}(\mathcal{L}_i)$$

obtained from $\mathcal{L}$ by local deformations

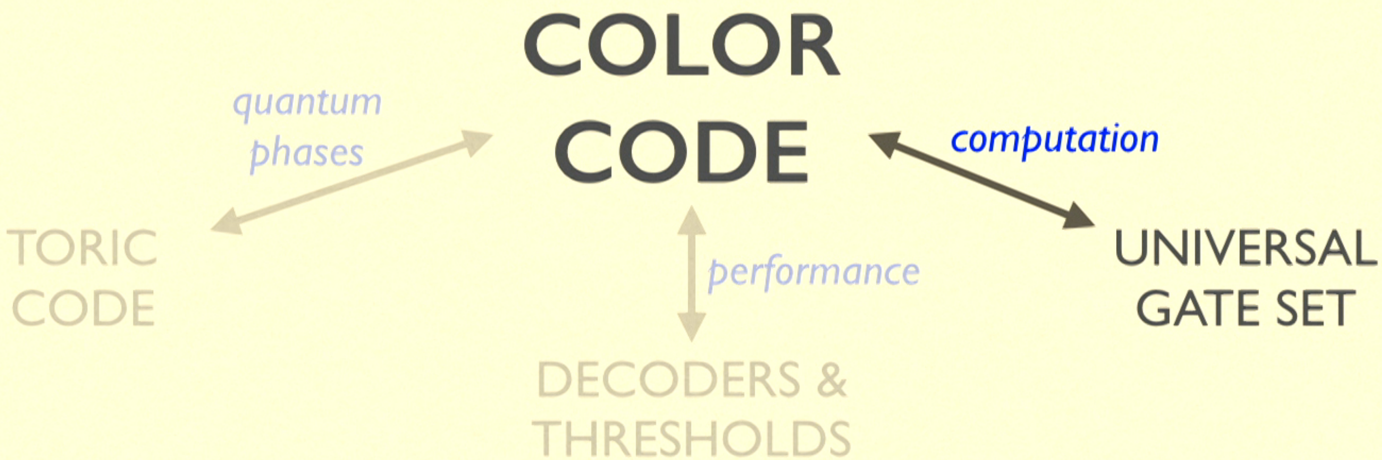
CC$_k$(L): qubits - 0-cells
X stabilizers - (d+2-k)-cells
Z stabilizers - k-cells

TC$_k$(L): qubits - k-cells
X stabilizers - (k-1)-cells
Z stabilizers - (k+1)-cells

Theorem: color code with point-like excitations in d dim with d+1 boundaries is equivalent to d copies of toric code attached along (d-1)-dim boundary.

$$U\left[CC(\mathcal{L})\right]U^\dagger = TC(\#_{i=1}^{d}\mathcal{L}_i)$$

arXiv:**1503.02065**



- **Main results:**
  - **Kubica, Beverland'15:** explicit construction of logical R_d gate; universality with without magic-state distillation
  - **Kubica, Yoshida, Pastawski'15:** fault-tolerant implementation of d-qubit control-Z gate in toric code
  - **Beverland, Kubica, Brandao, Svore (in prep.):** estimating the cost of universality by switching between color codes

17

PERFORMING COMPUTATION

- When performing computation, we do not want to spread errors and corrupt encoded information!

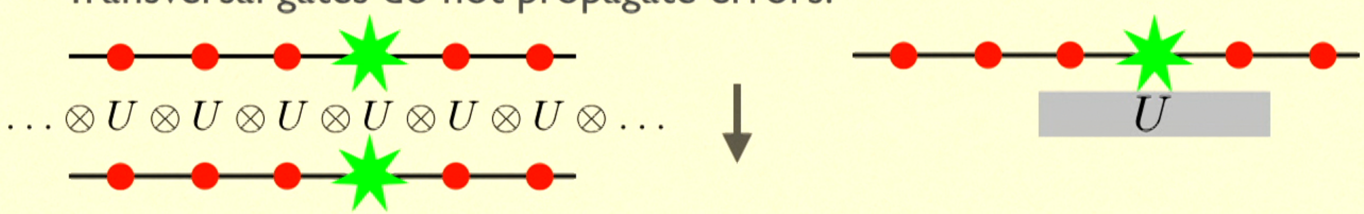
arXiv:1410.0069

PERFORMING COMPUTATION

- When performing computation, we do not want to spread errors and corrupt encoded information!
- **Transversal gates** - tensor product of single-qubit unitaries.

$$\overline{U} = \ldots \otimes U_{i-1} \otimes U_i \otimes U_{i+1} \otimes \ldots$$

- Transversal gates do not propagate errors.



arXiv:**1410.0069**

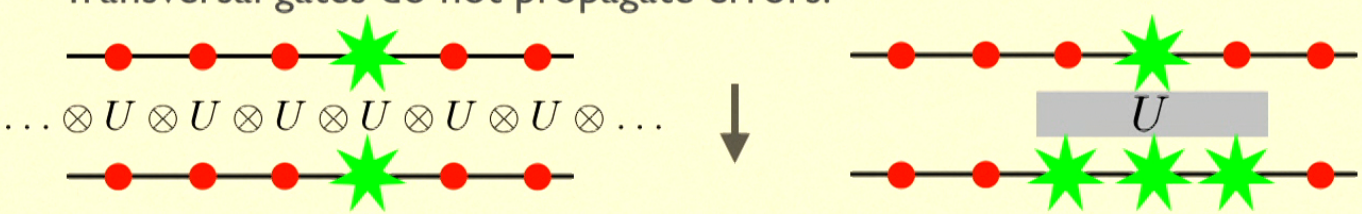
PERFORMING COMPUTATION

- When performing computation, we do not want to spread errors and corrupt encoded information!

- **Transversal gates** - tensor product of single-qubit unitaries.

$$\overline{U} = \ldots \otimes U_{i-1} \otimes U_i \otimes U_{i+1} \otimes \ldots$$

- Transversal gates do not propagate errors.



18

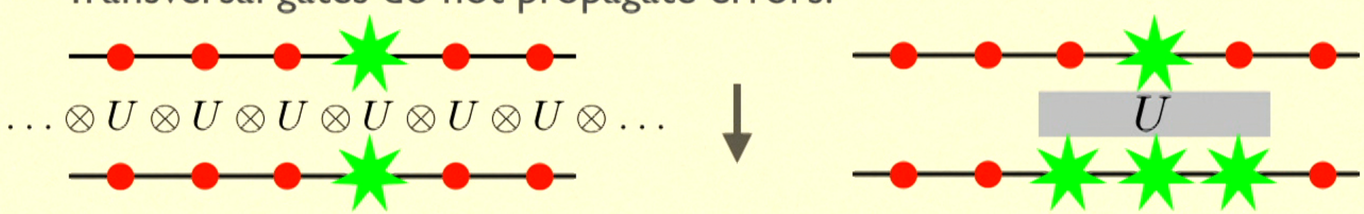
arXiv:**1410.0069**

PERFORMING COMPUTATION

- When performing computation, we do not want to spread errors and corrupt encoded information!
- **Transversal gates** - tensor product of single-qubit unitaries.

$$\overline{U} = \ldots \otimes U_{i-1} \otimes U_i \otimes U_{i+1} \otimes \ldots$$

- Transversal gates do not propagate errors.



- **Zeng et al.'07, Eastin&Knill'09**: for any nontrivial local-error-detecting quantum code, the set of transversal logical unitaries is not universal.

arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the d^{th} level of the Clifford hierarchy.

arXiv:**1410.0069**

NO-GO RESULTS

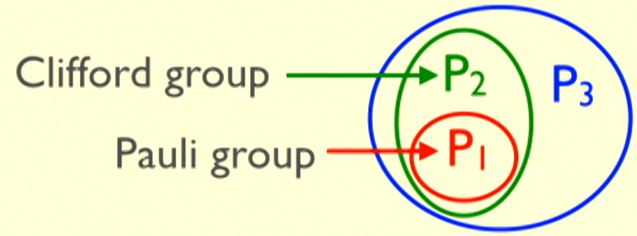
- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the dth level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.

$$\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1} \; \forall P \in \mathcal{P}_1\}$$

arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the dth level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.
  $\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1}\ \forall P \in \mathcal{P}_1\}$

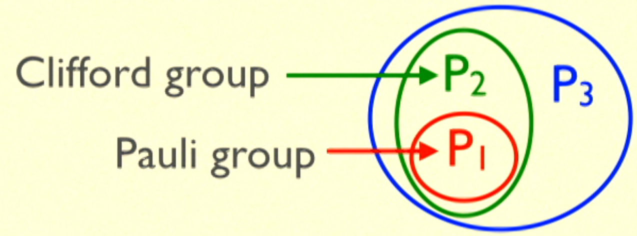


arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the dth level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.

$$\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1} \ \forall P \in \mathcal{P}_1\}$$



- Gates in dth level of Clifford hierarchy

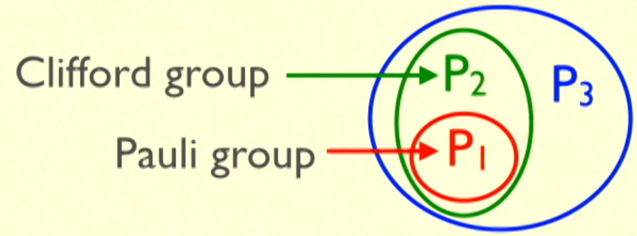
$$R_d|x\rangle = e^{2\pi i x/2^d}|x\rangle \qquad C^{d-1}Z|x_1,\ldots,x_d\rangle = (-1)^{x_1\ldots x_d}|x_1,\ldots,x_d\rangle$$

19

arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the d^{th} level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.
  $\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1}\ \forall P \in \mathcal{P}_1\}$



- Gates in d^{th} level of Clifford hierarchy

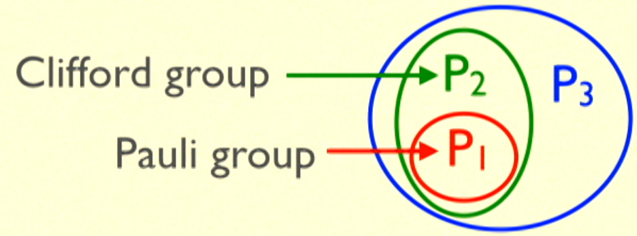
$$R_d|x\rangle = e^{2\pi i x/2^d}|x\rangle \qquad C^{d-1}Z|x_1,\ldots,x_d\rangle = (-1)^{x_1\ldots x_d}|x_1,\ldots,x_d\rangle$$

- **Question:** are there codes saturating the hierarchy?

arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the d^{th} level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.
  $\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1} \ \forall P \in \mathcal{P}_1\}$



- Gates in d^{th} level of Clifford hierarchy

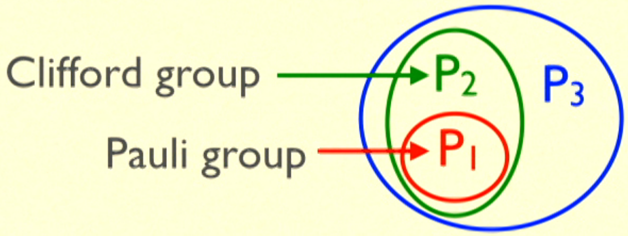
$$R_d|x\rangle = e^{2\pi i x/2^d}|x\rangle \qquad C^{d-1}Z|x_1,\ldots,x_d\rangle = (-1)^{x_1\ldots x_d}|x_1,\ldots,x_d\rangle$$

- **Question:** are there codes saturating the hierarchy?
- **Bombin'13**: color code saturates the hierarchy and achieves universality.

arXiv:**1410.0069**

NO-GO RESULTS

- **Bravyi&König'13**: for a topological stabilizer code in d dim, local unitaries can't implement gates from outside the d^{th} level of the Clifford hierarchy.
- Clifford hierarchy defined recursively.
  $\mathcal{P}_j = \{\text{unitary } U | UPU^\dagger \in \mathcal{P}_{j-1}\ \forall P \in \mathcal{P}_1\}$



- Gates in d^{th} level of Clifford hierarchy

$$R_d|x\rangle = e^{2\pi i x/2^d}|x\rangle \qquad C^{d-1}Z|x_1,\ldots,x_d\rangle = (-1)^{x_1\ldots x_d}|x_1,\ldots,x_d\rangle$$

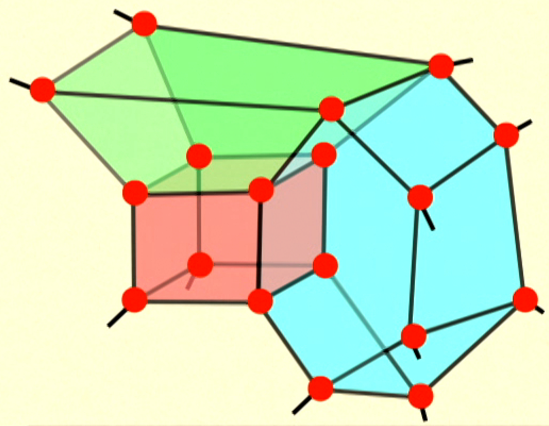
- **Question:** are there codes saturating the hierarchy?
- **Bombin'13**: color code saturates the hierarchy and achieves universality.
- **Our results:** simplification of the argument by constructing explicitly a logical R_d gate.

19

arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!

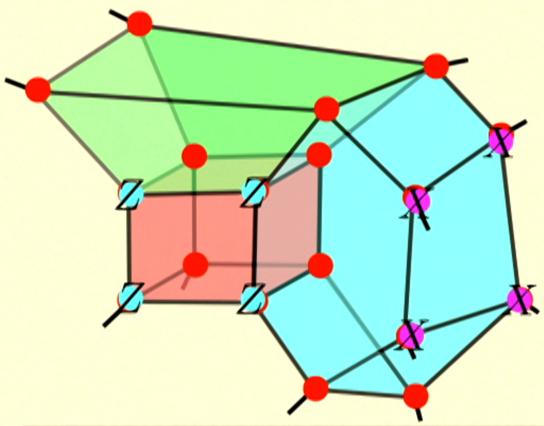


20

arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

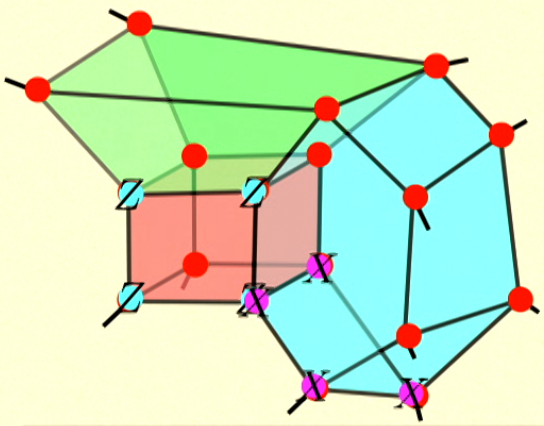
- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!
- Subsystem CC in 3D: gauge generators on **faces**.



arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!
- Subsystem CC in 3D: gauge generators on **faces**.
- In d dim: X and Z gauge generators on x-cells and z-cells, where $x,z \geq 2$ and $x+z \leq d+2$.

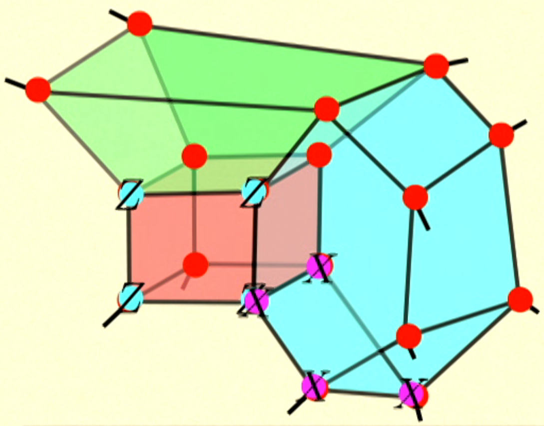


arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!
- Subsystem CC in 3D: gauge generators on **faces**.
- In d dim: X and Z gauge generators on x-cells and z-cells, where $x,z \geq 2$ and $x+z \leq d+2$.

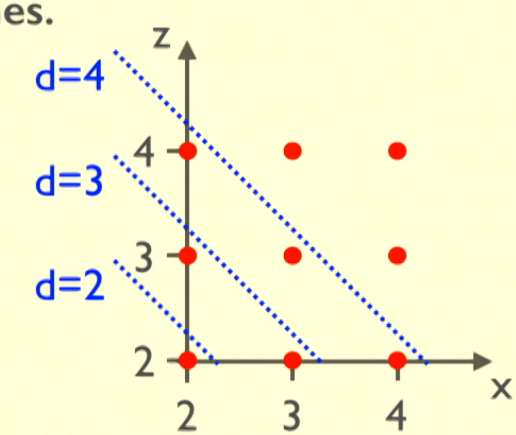
stabilizer iff "="

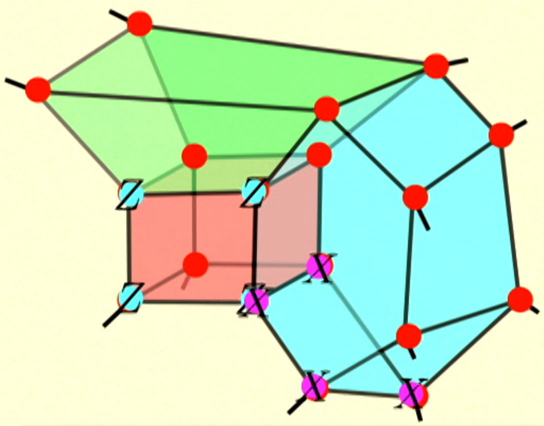


arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!
- Subsystem CC in 3D: gauge generators on **faces**.
- In d dim: X and Z gauge generators on x-cells and z-cells, where $x,z \geq 2$ and $x+z \leq d+2$.





stabilizer iff "="

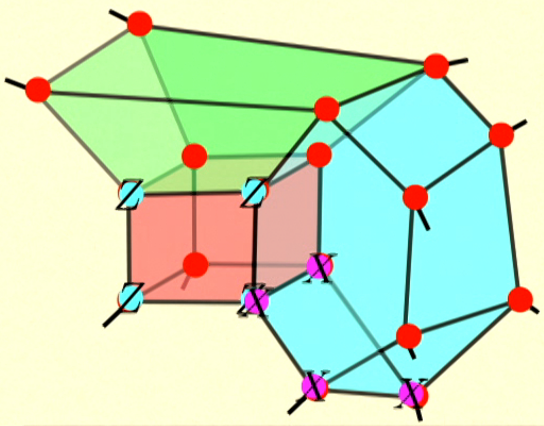
20

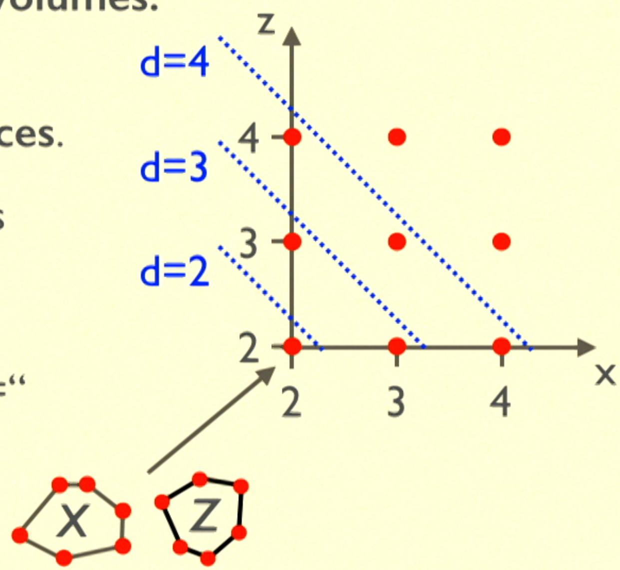
arXiv:**1410.0069**

SUBSYSTEM COLOR CODES

- Stabilizer CC in 3D: stabilizers on **faces** and **volumes.** Difficult to measure due to high weight!
- Subsystem CC in 3D: gauge generators on **faces**.
- In d dim: X and Z gauge generators on x-cells and z-cells, where x,z ≥ 2 and x+z ≤ d+2.

stabilizer iff "="

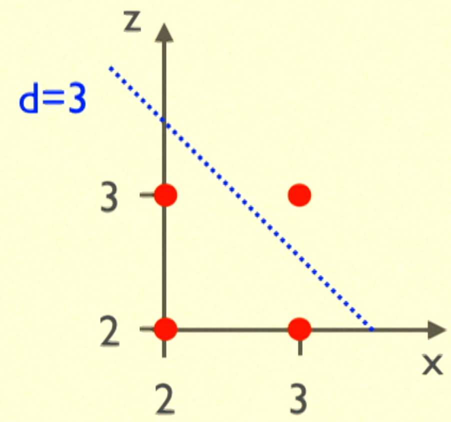




20

arXiv:**1410.0069**

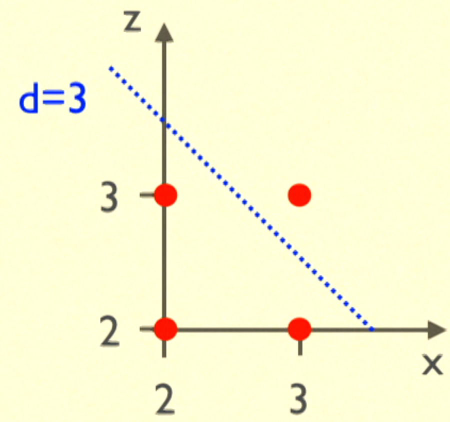
UNIVERSAL GATE SET WITH COLOR CODES IN 3D



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

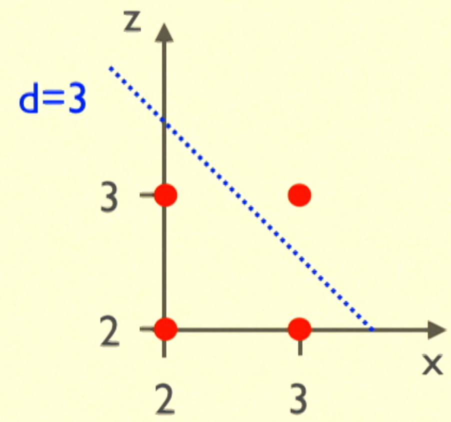
- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

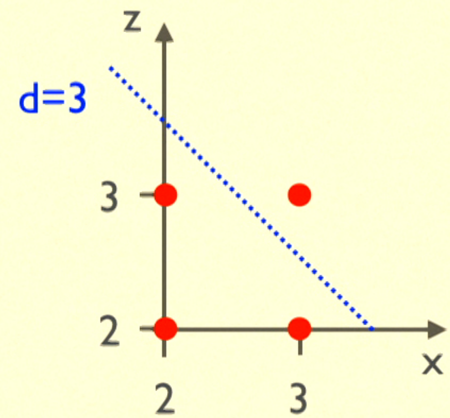
- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.
- Universal gate set = CNOT, H, R_3.



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

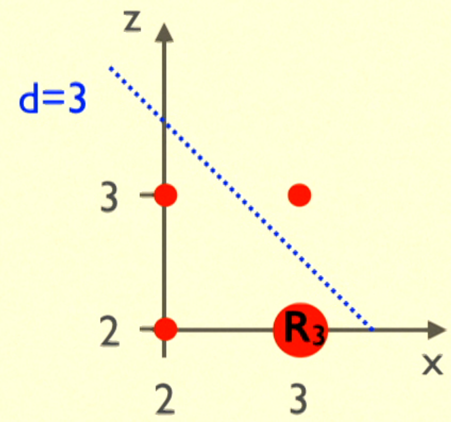
- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.
- Universal gate set = CNOT, H, R_3.
- CNOT - transversal in any color code (CSS).



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

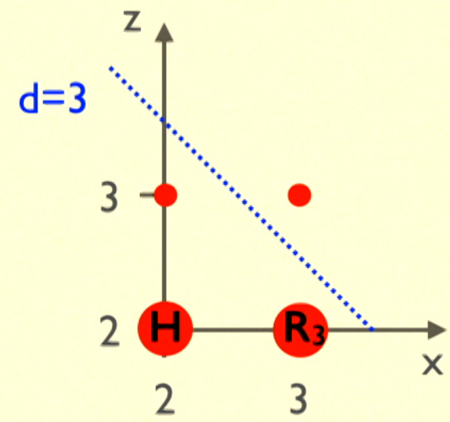
- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.
- Universal gate set = CNOT, H, R_3.
- CNOT - transversal in any color code (CSS).
- R_3 - transversal in stabilizer $CC_3(3,2)$.



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

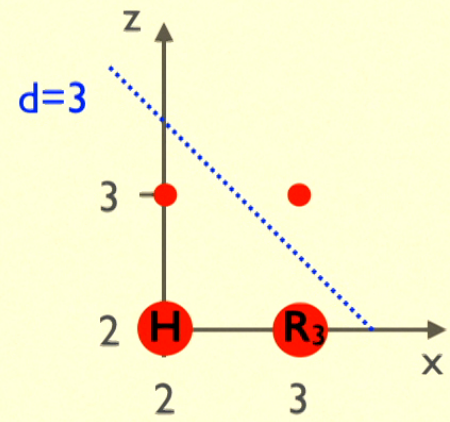
- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.
- Universal gate set = CNOT, H, R_3.
- CNOT - transversal in any color code (CSS).
- R_3 - transversal in stabilizer $CC_3(3,2)$.
- H - transversal in subsystem $CC_3(2,2)$ (self-dual).



arXiv:1410.0069

UNIVERSAL GATE SET WITH COLOR CODES IN 3D

- Tetrahedron-like lattice w/ 1 logical qubit. Smallest instance = 15-qubit Reed-Muller.
- Universal gate set = CNOT, H, R_3.
- CNOT - transversal in any color code (CSS).
- R_3 - transversal in stabilizer $CC_3(3,2)$.
- H - transversal in subsystem $CC_3(2,2)$ (self-dual).
- Code switching between $CC_3(x=2, z=2)$ and $CC_3(x=3, z=2)$ possible in a fault-tolerant way. Universal gate set in 3D!



arXiv:1410.0069

COST OF UNIVERSALITY IN 3D COLOR CODES

- Universality with 3D color code without magic-state distillation (the main source of overhead).

work w/ **M. Beverland, F. Brandao, K. Svore**

COST OF UNIVERSALITY IN 3D COLOR CODES

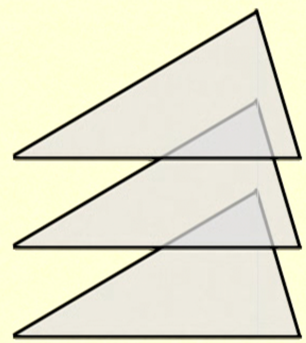
- Universality with 3D color code without magic-state distillation (the main source of overhead).
- **Question:** what is the cost of code switching?
- 3D color code challenging: layout, CNOT.

22

work w/ **M. Beverland, F. Brandao, K. Svore**

COST OF UNIVERSALITY IN 3D COLOR CODES

- Universality with 3D color code without magic-state distillation (the main source of overhead).
- **Question:** what is the cost of code switching?
- 3D color code challenging: layout, CNOT.
- More realistic: 2D architecture and stack of codes. 2D color code - transversal logical Clifford group.

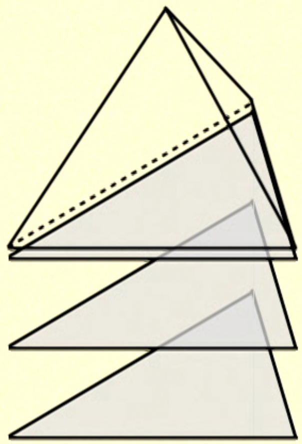


22

work w/ **M. Beverland, F. Brandao, K. Svore**

COST OF UNIVERSALITY IN 3D COLOR CODES

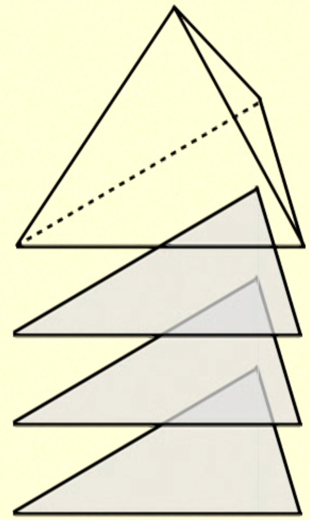
- Universality with 3D color code without magic-state distillation (the main source of overhead).
- **Question:** what is the cost of code switching?
- 3D color code challenging: layout, CNOT.
- More realistic: 2D architecture and stack of codes. 2D color code - transversal logical Clifford group.
- When logical R_3 needed, switch to 3D color code - dimensional jump (**Anderson et al.'14, Bombin'14**).



work w/ **M. Beverland, F. Brandao, K. Svore**

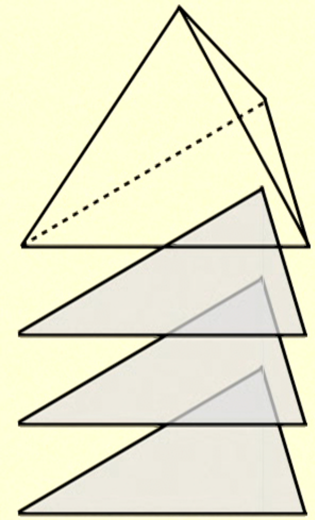
COST OF UNIVERSALITY IN 3D COLOR CODES

- Universality with 3D color code without magic-state distillation (the main source of overhead).
- **Question:** what is the cost of code switching?
- 3D color code challenging: layout, CNOT.
- More realistic: 2D architecture and stack of codes. 2D color code - transversal logical Clifford group.
- When logical R_3 needed, switch to 3D color code - dimensional jump (**Anderson et al.'14, Bombin'14**).



work w/ **M. Beverland, F. Brandao, K. Svore**

COST OF UNIVERSALITY IN 3D COLOR CODES



- Universality with 3D color code without magic-state distillation (the main source of overhead).
- **Question:** what is the cost of code switching?
- 3D color code challenging: layout, CNOT.
- More realistic: 2D architecture and stack of codes. 2D color code - transversal logical Clifford group.
- When logical R_3 needed, switch to 3D color code - dimensional jump (**Anderson et al.'14, Bombin'14**).
- **Our results:** qubit overhead of code switching compares unfavourably to magic-state distillation!

work w/ **M. Beverland, F. Brandao, K. Svore**

$C^{d-1}Z$ GATE IN COLOR CODE

23

arXiv:**1503.02065**

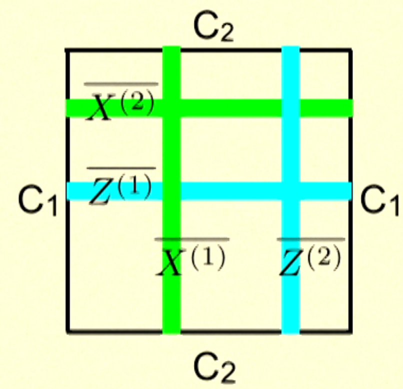
$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.

arXiv:**1503.02065**

$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.
- Color code w/ boundaries on a d-dim hypercube - d logical qubit. What about transversal logical gates?
- Consider 2D case and check how a transversal unitary R_2 transforms logical operators:

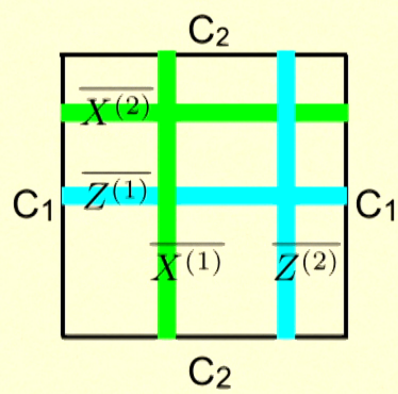


arXiv:1503.02065

$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.
- Color code w/ boundaries on a d-dim hypercube - d logical qubit. What about transversal logical gates?
- Consider 2D case and check how a transversal unitary R_2 transforms logical operators:

$\overline{Z^{(1)}} \rightarrow \overline{Z^{(1)}}$ $\overline{Z^{(2)}} \rightarrow \overline{Z^{(2)}}$



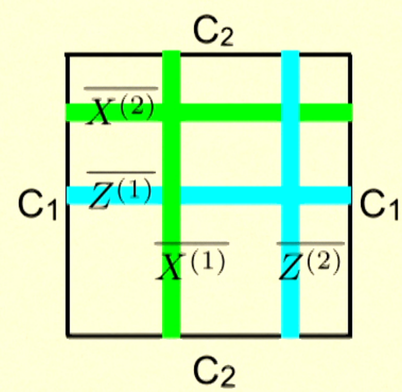
arXiv:1503.02065

$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.
- Color code w/ boundaries on a d-dim hypercube - d logical qubit. What about transversal logical gates?
- Consider 2D case and check how a transversal unitary R_2 transforms logical operators:

$$\overline{Z^{(1)}} \rightarrow \overline{Z^{(1)}} \qquad \overline{Z^{(2)}} \rightarrow \overline{Z^{(2)}}$$

$$\overline{X^{(1)}} \rightarrow \overline{X^{(1)}}\ \overline{Z^{(2)}} \qquad \overline{X^{(2)}} \rightarrow \overline{X^{(2)}}\ \overline{Z^{(1)}}$$



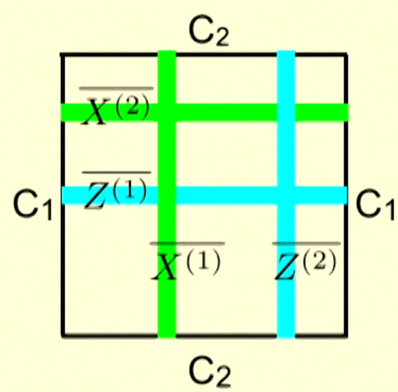
arXiv:1503.02065

$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.
- Color code w/ boundaries on a d-dim hypercube - d logical qubit. What about transversal logical gates?
- Consider 2D case and check how a transversal unitary R_2 transforms logical operators:

$$\overline{Z^{(1)}} \rightarrow \overline{Z^{(1)}} \qquad \overline{Z^{(2)}} \rightarrow \overline{Z^{(2)}}$$

$$\overline{X^{(1)}} \rightarrow \overline{X^{(1)}}\ \overline{Z^{(2)}} \qquad \overline{X^{(2)}} \rightarrow \overline{X^{(2)}}\ \overline{Z^{(1)}}$$



- Transversal unitary R_d implements logical $C^{d-1}Z$ gate.

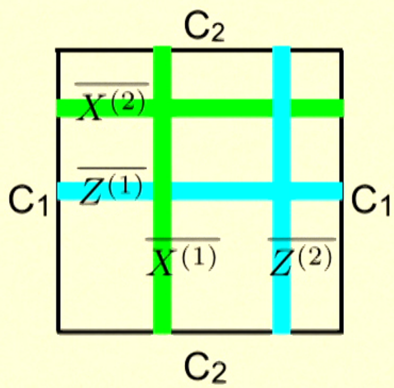
arXiv:1503.02065

$C^{d-1}Z$ GATE IN COLOR CODE

- Color code w/ boundaries on a d-simplex-like lattice: 1 logical qubit and transversal logical R_d gate.
- Color code w/ boundaries on a d-dim hypercube - d logical qubit. What about transversal logical gates?
- Consider 2D case and check how a transversal unitary R_2 transforms logical operators:

$$\overline{Z^{(1)}} \to \overline{Z^{(1)}} \qquad \overline{Z^{(2)}} \to \overline{Z^{(2)}}$$

$$\overline{X^{(1)}} \to \overline{X^{(1)}}\ \overline{Z^{(2)}} \qquad \overline{X^{(2)}} \to \overline{X^{(2)}}\ \overline{Z^{(1)}}$$

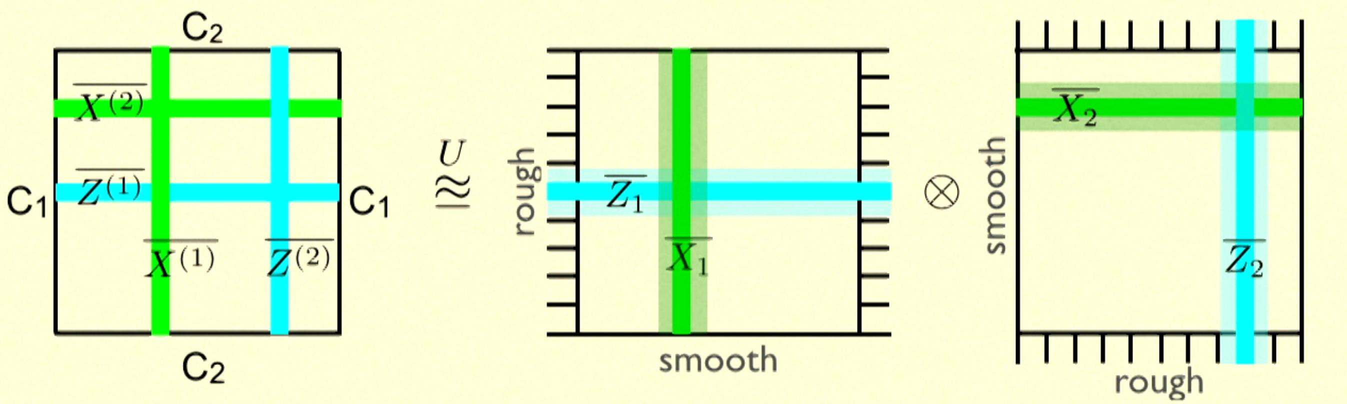


- Transversal unitary R_d implements logical $C^{d-1}Z$ gate.
- Smallest interesting instance, the [[8, 3, 2]] code, enables to distill 8 noisy T-states into less noisy CCZ magic state!

arXiv:1503.02065

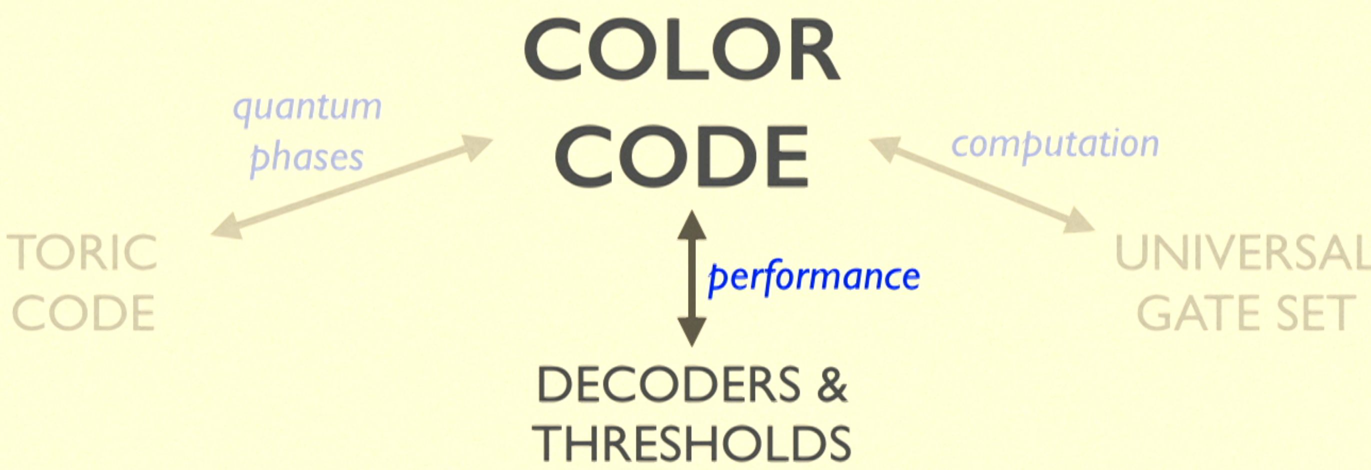
$C^{d-1}Z$ GATE IN TORIC CODE

- Recall the local unitary mapping U - the color code on a d-dim hypercube equivalent to d decoupled copies of the d-dim toric code.



- Can verify that a local unitary $UR_dU^\dagger$ implements logical $C^{d-1}Z$ gate in d copies of the toric code.

arXiv:1503.02065



- **Main results (Kubica, Delfosse, in prep.):**
  developing and benchmarking decoders of color code in $d \geq 3$ dim

25

NEED FOR DECODERS

- Color code: many transversal gates, universality without magic-state distillation. Promising quantum computer architecture.

work w/ **N. Delfosse**

NEED FOR DECODERS

- Color code: many transversal gates, universality without magic-state distillation. Promising quantum computer architecture.
- To prevent errors from accumulation one performs active error correction, which requires efficient decoders!

work w/ **N. Delfosse**

NEED FOR DECODERS

- Color code: many transversal gates, universality without magic-state distillation. Promising quantum computer architecture.
- To prevent errors from accumulation one performs active error correction, which requires efficient decoders!
- **Question:** how to decode color code?
- **Idea:** map color code to toric code with a local unitary and use existing toric code decoders.

work w/ **N. Delfosse**

NEED FOR DECODERS

- Color code: many transversal gates, universality without magic-state distillation. Promising quantum computer architecture.
- To prevent errors from accumulation one performs active error correction, which requires efficient decoders!
- **Question:** how to decode color code?
- **Idea:** map color code to toric code with a local unitary and use existing toric code decoders.
- **Problem:** uncorrelated noise becomes correlated!

26

work w/ **N. Delfosse**

NEED FOR DECODERS

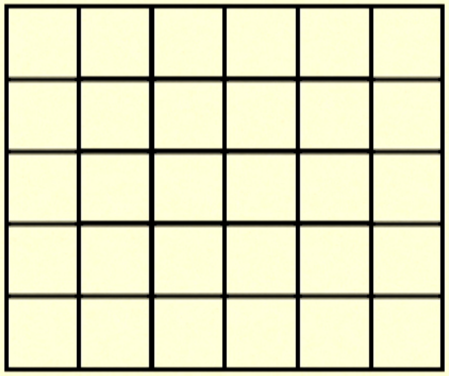
- Color code: many transversal gates, universality without magic-state distillation. Promising quantum computer architecture.
- To prevent errors from accumulation one performs active error correction, which requires efficient decoders!
- **Question:** how to decode color code?
- **Idea:** map color code to toric code with a local unitary and use existing toric code decoders.
- **Problem:** uncorrelated noise becomes correlated!
- **Our results:** developing a decoder for color code in $d \geq 3$ dim and analyzing its performance.

26

work w/ **N. Delfosse**

EXAMPLE: DECODING OF TORIC CODE

- CSS codes - can decode X and Z separately (might reduce performance!).

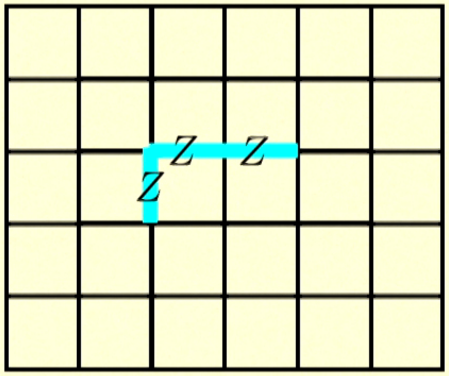


27

work w/ **N. Delfosse**

EXAMPLE: DECODING OF TORIC CODE

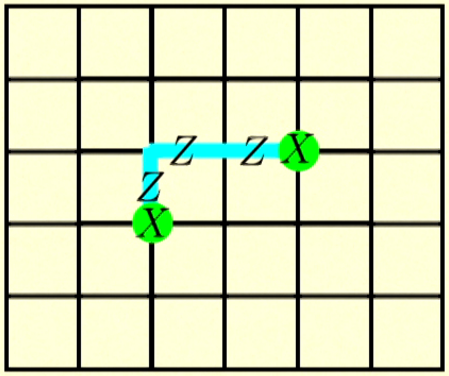
- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.



work w/ **N. Delfosse**

EXAMPLE: DECODING OF TORIC CODE

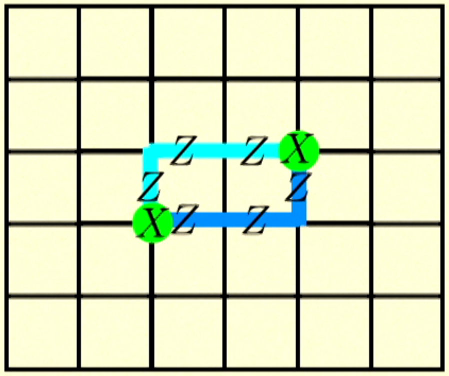
- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.
- **Input:** s = violated X-star stabilizers.



work w/ **N. Delfosse**

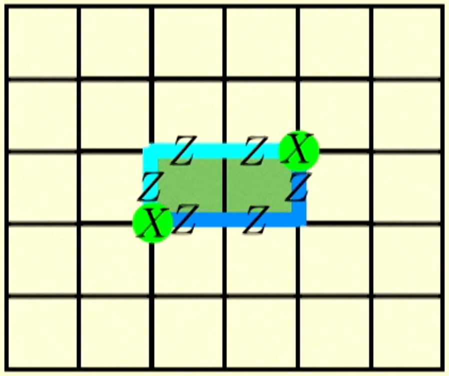
EXAMPLE: DECODING OF TORIC CODE

- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.
- **Input:** s = violated X-star stabilizers.
- **Output:** e' = Z-errors which might have happened.



work w/ **N. Delfosse**

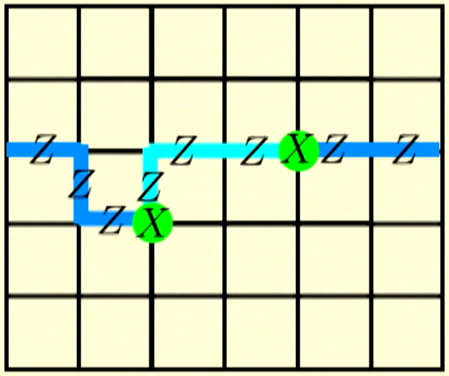
EXAMPLE: DECODING OF TORIC CODE



- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.
- **Input:** s = violated X-star stabilizers.
- **Output:** e' = Z-errors which might have happened.
- **Success:** e and e' differ only by stabilizer b (product of Z-plaquettes).

work w/ **N. Delfosse**

EXAMPLE: DECODING OF TORIC CODE

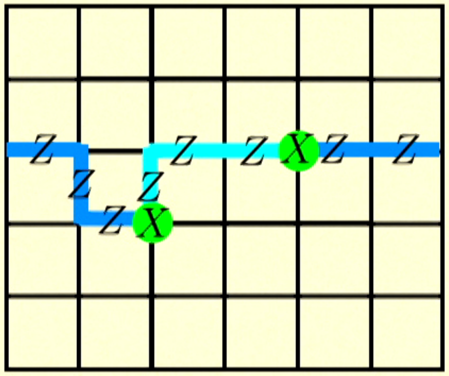


- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.
- **Input:** s = violated X-star stabilizers.
- **Output:** e' = Z-errors which might have happened.
- **Success:** e and e' differ only by stabilizer b (product of Z-plaquettes).
- **Best strategy**: if syndrome s observed, pick the most probable equivalence class of errors consistent with s (errors equivalent iff differ by stabilizer b).

work w/ **N. Delfosse**

EXAMPLE: DECODING OF TORIC CODE

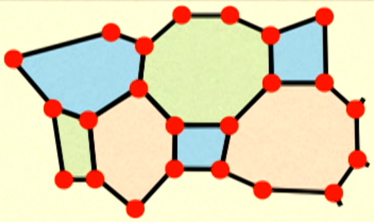
- CSS codes - can decode X and Z separately (might reduce performance!).
- Z-errors appear at locations e.
- **Input:** s = violated X-star stabilizers.
- **Output:** e' = Z-errors which might have happened.



- **Success:** e and e' differ only by stabilizer b (product of Z-plaquettes).
- **Best strategy**: if syndrome s observed, pick the most probable equivalence class of errors consistent with s (errors equivalent iff differ by stabilizer b).
- Minimum weight perfect matching - efficient and close to optimal.

work w/ **N. Delfosse**

DELFOSSE'S DECODER: 2D COLOR CODE

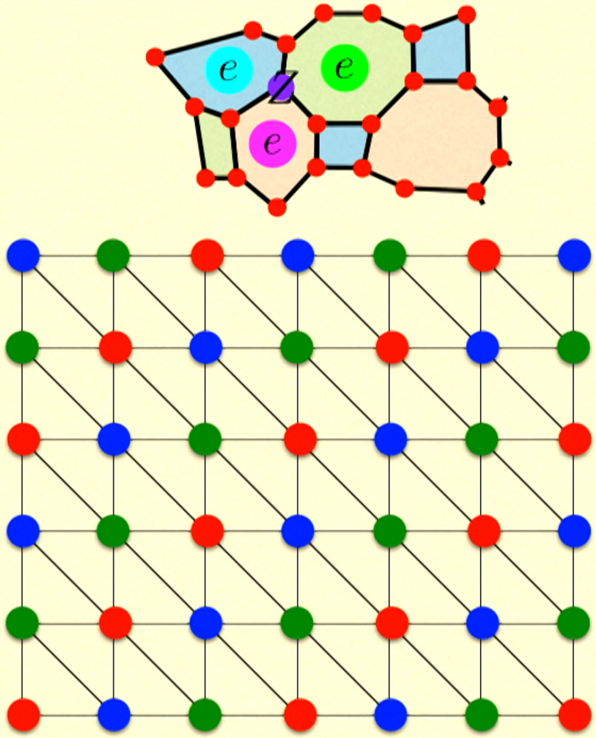


28

work w/ **N. Delfosse**

DELFOSSE'S DECODER: 2D COLOR CODE

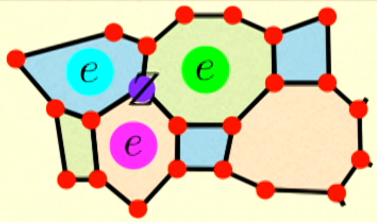
- Decoding of color code more challenging: excitations can be created in triples!
- Switch to dual lattice:



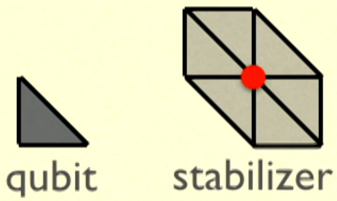
work w/ **N. Delfosse**

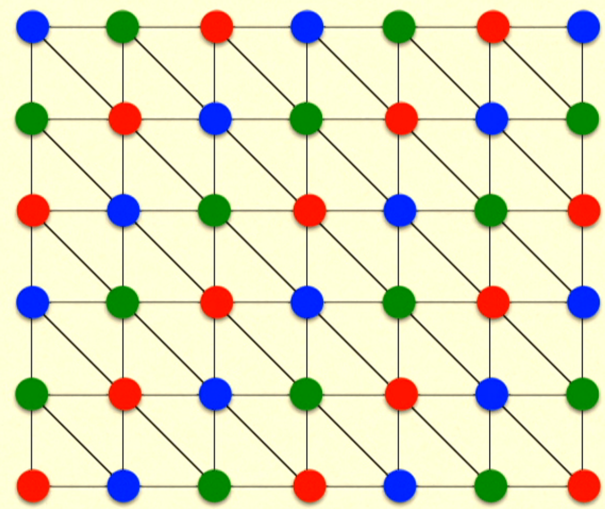
DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!



- Switch to dual lattice:

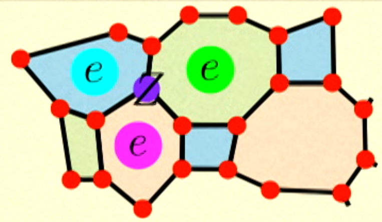




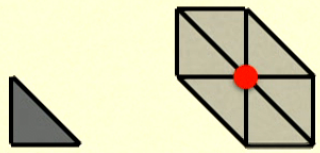
work w/ **N. Delfosse**

DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!



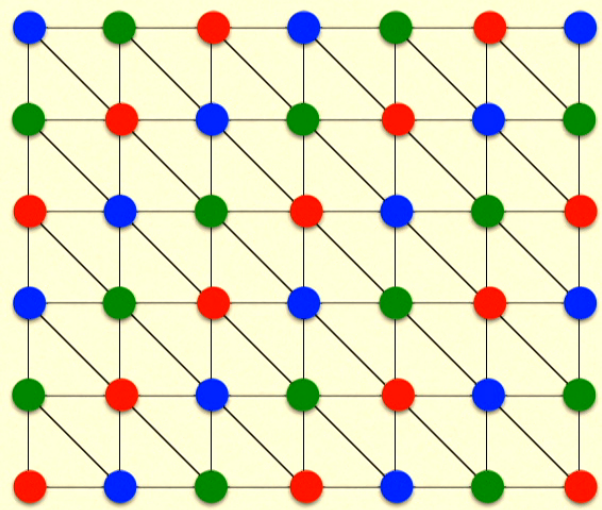
- Switch to dual lattice:



- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.

error ← syndrome

2D ← 0D

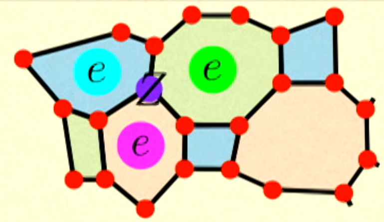


28

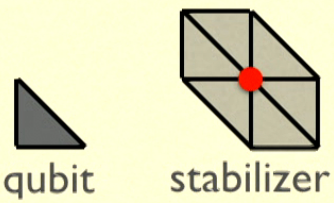
work w/ **N. Delfosse**

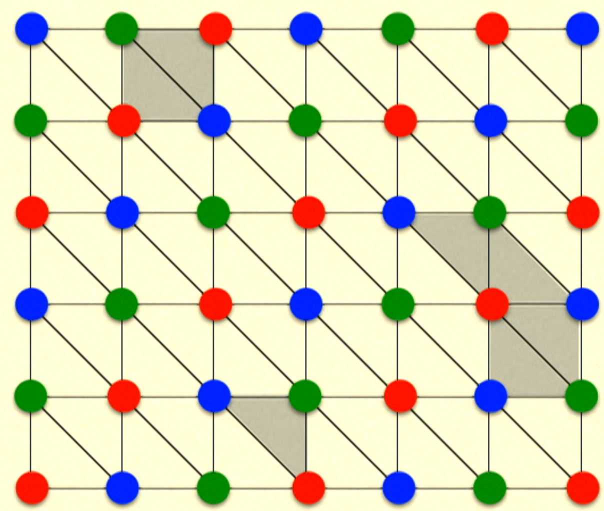
DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!

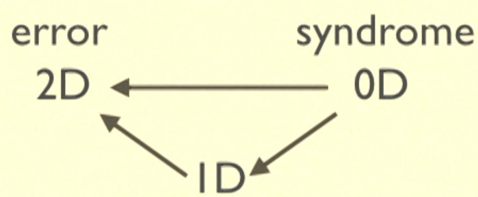


- Switch to dual lattice:





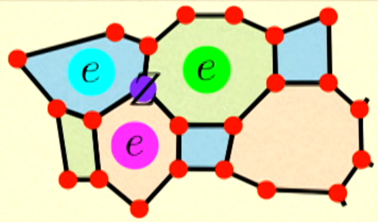
- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.



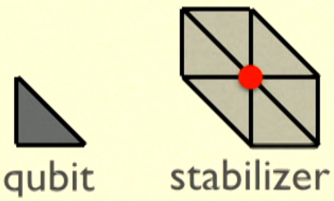
work w/ **N. Delfosse**

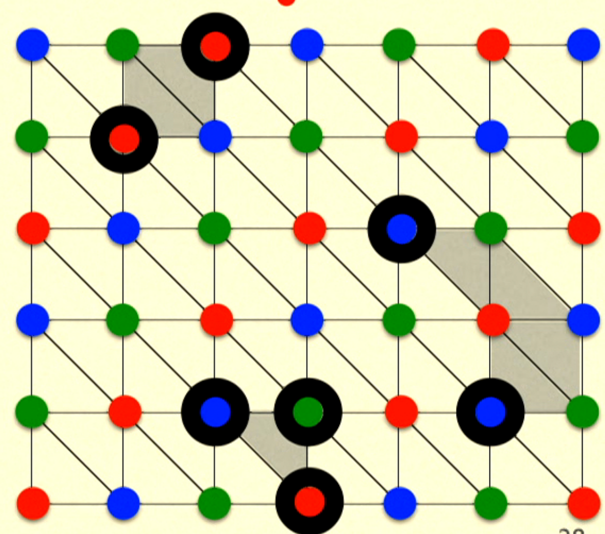
DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!

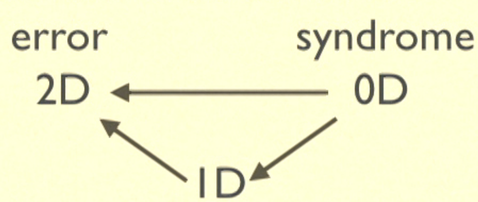


- Switch to dual lattice:





- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.

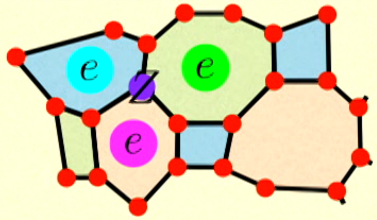


28

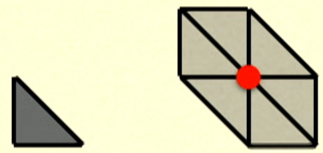
work w/ **N. Delfosse**

DELFOSSE'S DECODER: 2D COLOR CODE

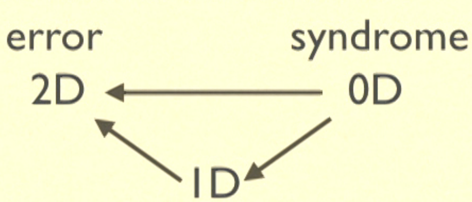
- Decoding of color code more challenging: excitations can be created in triples!

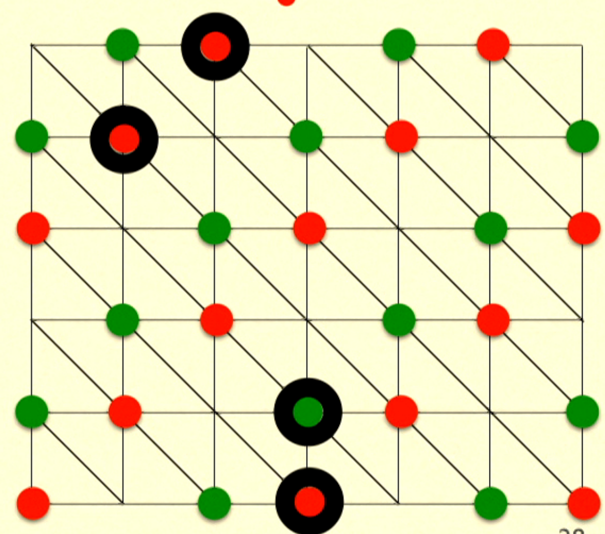


- Switch to dual lattice:



- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.

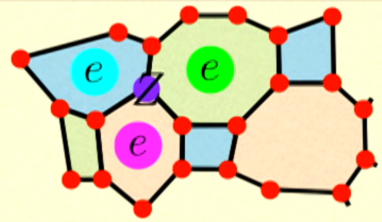




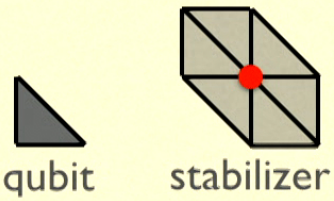
work w/ **N. Delfosse**

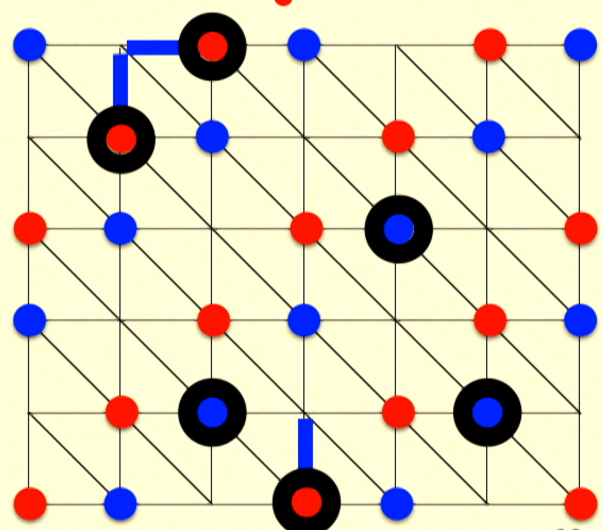
DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!

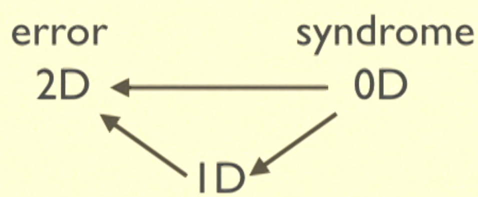


- Switch to dual lattice:





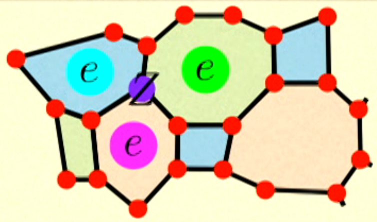
- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.



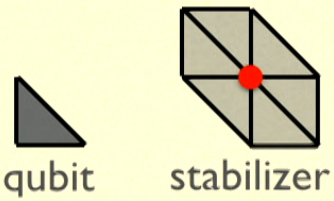
work w/ **N. Delfosse**

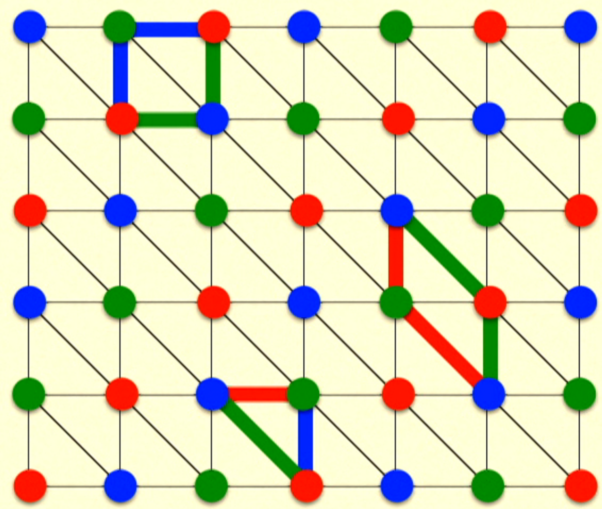
DELFOSSE'S DECODER: 2D COLOR CODE

- Decoding of color code more challenging: excitations can be created in triples!

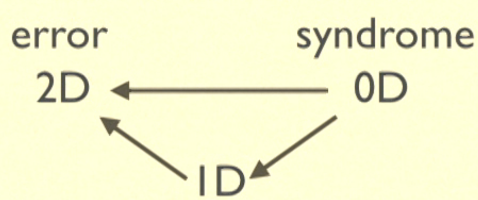


- Switch to dual lattice:





- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.

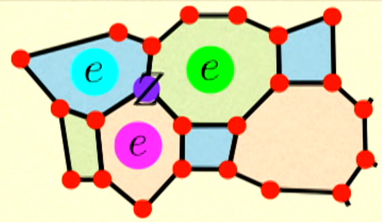


28

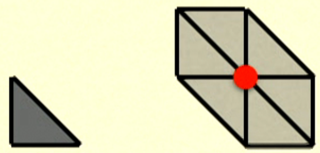
work w/ **N. Delfosse**

DELFOSSE'S DECODER: 2D COLOR CODE

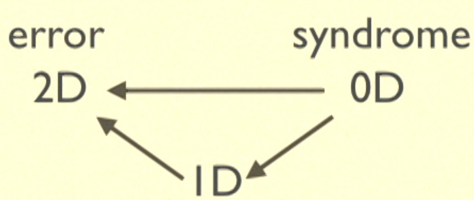
- Decoding of color code more challenging: excitations can be created in triples!

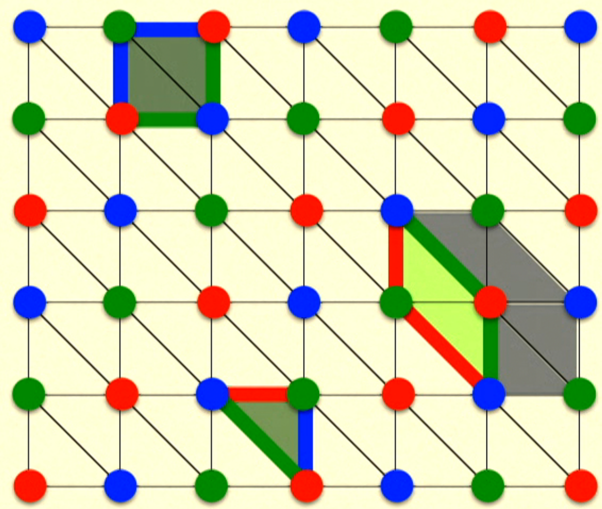


- Switch to dual lattice:



- **Delfosse'13:** project onto 3 sublattices and use toric code decoder.



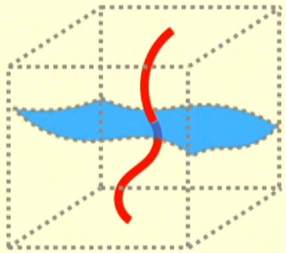


28

work w/ **N. Delfosse**

DECODING OF COLOR CODE IN 3D

- 3D color code has string-like (1-dim) and sheet-like (2-dim) logical operators.

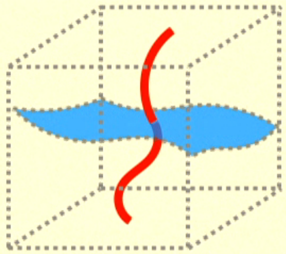


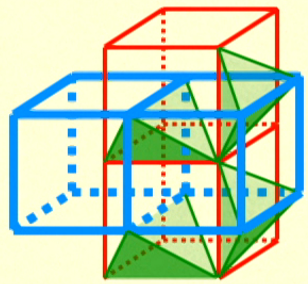
29

work w/ **N. Delfosse**

DECODING OF COLOR CODE IN 3D

- 3D color code has string-like (1-dim) and sheet-like (2-dim) logical operators.
- Qubits = tetrahedra, stabilizers = vertices/edges.



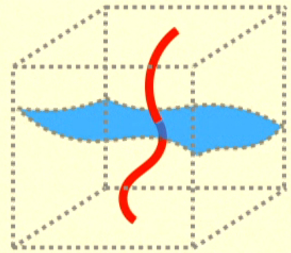


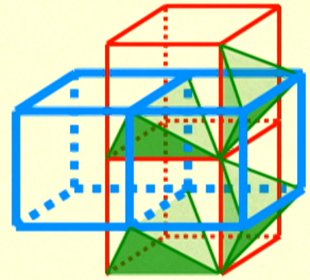
29

work w/ **N. Delfosse**

DECODING OF COLOR CODE IN 3D

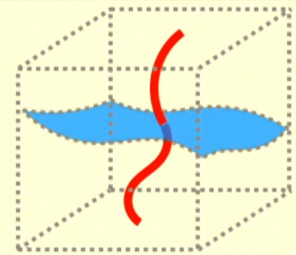
- 3D color code has string-like (1-dim) and sheet-like (2-dim) logical operators.
- Qubits = tetrahedra, stabilizers = vertices/edges.
- **Goal:** error ⟵ syndrome
  3D   1D   0D





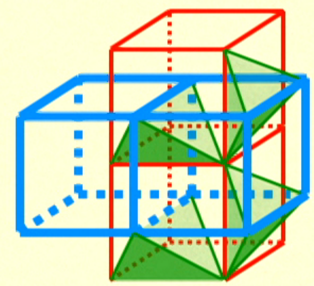
work w/ **N. Delfosse**

DECODING OF COLOR CODE IN 3D



- 3D color code has string-like (1-dim) and sheet-like (2-dim) logical operators.
- Qubits = tetrahedra, stabilizers = vertices/edges.
- **Goal:** error ⟵ syndrome

  3D ↖ 2D ⟵ 1D ↙ 0D; 1D ↙ 2D

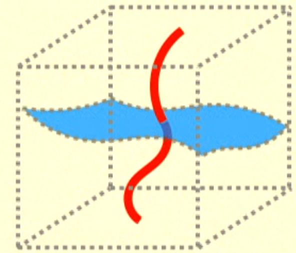


- **Decoder:** project onto 6 (pairs of colors for 0D) or 4 sublattices (triples of colors for 1D)and use toric code decoder!

29

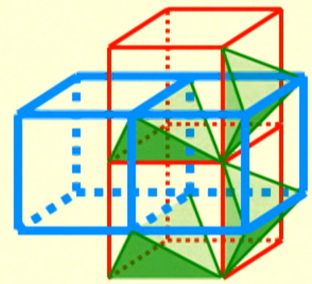
work w/ **N. Delfosse**

DECODING OF COLOR CODE IN 3D



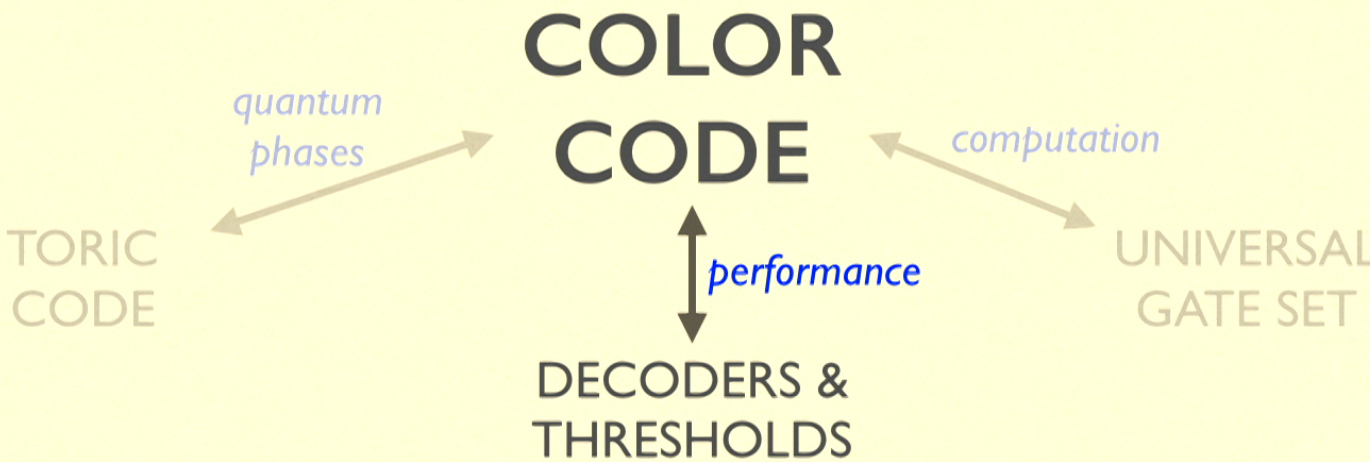
- 3D color code has string-like (1-dim) and sheet-like (2-dim) logical operators.
- Qubits = tetrahedra, stabilizers = vertices/edges.
- **Goal:** error ⟵ syndrome

  3D ↖ 2D ⟵ 1D ↙ 0D ; 1D ↙ 2D



- **Decoder:** project onto 6 (pairs of colors for 0D) or 4 sublattices (triples of colors for 1D)and use toric code decoder!
- Beyond perfect matching - finding min area k-chain w/ given boundary, i.e. (k-1)-chain. Efficient for (n-1)-chains in n dim (**Sullivan PhD thesis'94**).

work w/ **N. Delfosse**



- **Main results (Kubica, Beverland, Brandao, Preskill, Svore, in prep.):**
  - investigation of phase diagrams of new statistical-mechanical models,
  - optimal error correction thresholds of 3D color code.

30

THRESHOLDS FROM STATISTICAL MECHANICS

- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).

31

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

THRESHOLDS FROM STATISTICAL MECHANICS

- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).
- Thresholds relevant for: comparing codes and decoders, guiding experiments, resource analysis.

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

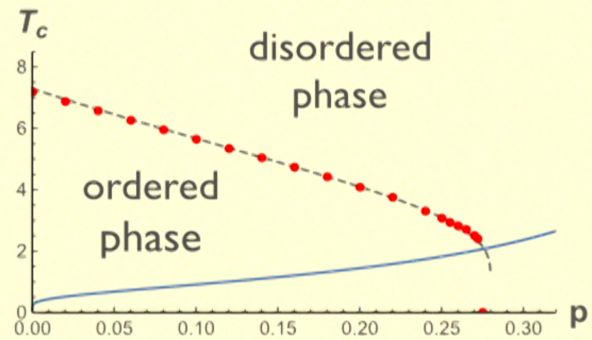
THRESHOLDS FROM STATISTICAL MECHANICS

- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).
- Thresholds relevant for: comparing codes and decoders, guiding experiments, resource analysis.
- **Dennis et al.'02:** to find optimal threshold of toric code one can study a classical spin model (random-bond Ising model).

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

THRESHOLDS FROM STATISTICAL MECHANICS

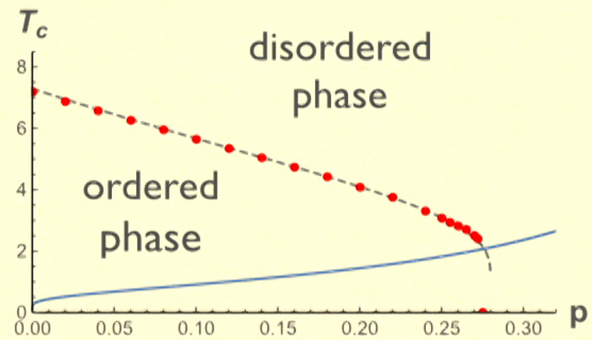
- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).
- Thresholds relevant for: comparing codes and decoders, guiding experiments, resource analysis.
- **Dennis et al.'02:** to find optimal threshold of toric code one can study a classical spin model (random-bond Ising model).



work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

THRESHOLDS FROM STATISTICAL MECHANICS

- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).
- Thresholds relevant for: comparing codes and decoders, guiding experiments, resource analysis.
- **Dennis et al.'02:** to find optimal threshold of toric code one can study a classical spin model (random-bond Ising model).

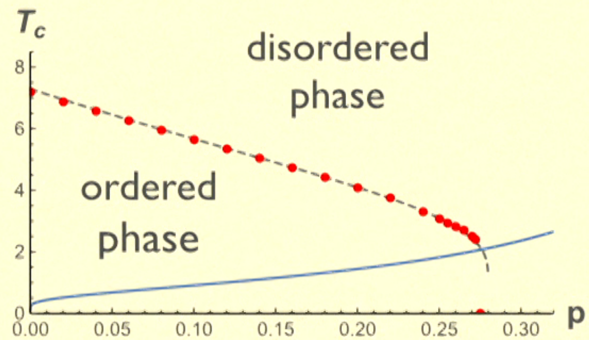


- Mapping allows to use tools and method from statistical mechanics!

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

THRESHOLDS FROM STATISTICAL MECHANICS

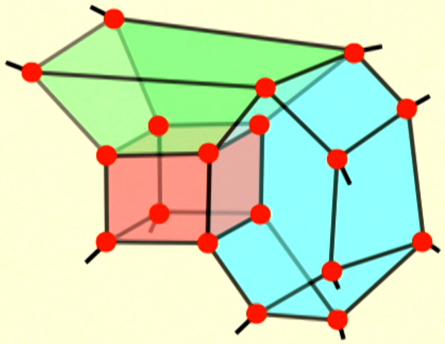
- Error correction threshold p_{th} = maximum error rate the code can tolerate. Decoding succeeds if $p < p_{th}$ (in the limit of infinite system size).
- Thresholds relevant for: comparing codes and decoders, guiding experiments, resource analysis.
- **Dennis et al.'02:** to find optimal threshold of toric code one can study a classical spin model (random-bond Ising model).



- Mapping allows to use tools and method from statistical mechanics!
- **Our results:** introducing new stat-mech models relevant for 3D color code, studying their phase diagrams, finding thresholds of 3D color code.

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE AND RANDOM COUPLING ISING MODEL



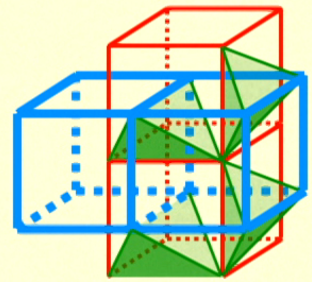
- 3D color code - stabilizer and subsystem:
  - stabilizers on faces (A) and volumes (B),
  - gauge generators on faces (C).
- Two stat-mech models to analyze:
  - 4-body random coupling Ising model (A),

$$H = -\sum \varkappa_{abcd} s_a s_b s_c s_d$$

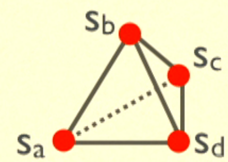
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE AND RANDOM COUPLING ISING MODEL

- 3D color code - stabilizer and subsystem:
  - stabilizers on faces (A) and volumes (B),
  - gauge generators on faces (C).
- Two stat-mech models to analyze:
  - 4-body random coupling Ising model (A),



$$H = -\sum \varkappa_{abcd} s_a s_b s_c s_d$$

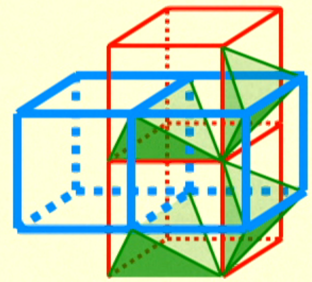


$$\varkappa_{abcd} = \pm 1$$

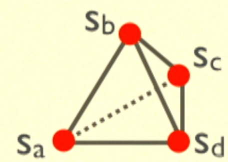
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE AND RANDOM COUPLING ISING MODEL

- 3D color code - stabilizer and subsystem:
  - stabilizers on faces (A) and volumes (B),
  - gauge generators on faces (C).
- Two stat-mech models to analyze:
  - 4-body random coupling Ising model (A),



$$H = -\sum \varkappa_{abcd} s_a s_b s_c s_d$$



$$\varkappa_{abcd} = \pm 1$$

  - local order parameter - fairly easy to analyze!

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

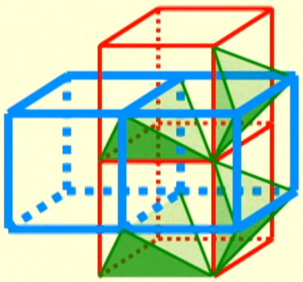
3D COLOR CODE AND RANDOM COUPLING ISING MODEL

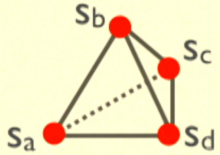
- 3D color code - stabilizer and subsystem:
  - stabilizers on faces (A) and volumes (B),
  - gauge generators on faces (C).
- Two stat-mech models to analyze:
  - 4-body random coupling Ising model (A),

$$H = -\sum \varkappa_{abcd} s_a s_b s_c s_d$$

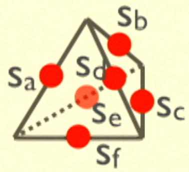
  - local order parameter - fairly easy to analyze!
  - 6-body random coupling Ising model (B & C),

$$H = -\sum \varkappa_{abcdef} s_a s_b s_c s_d s_e s_f$$





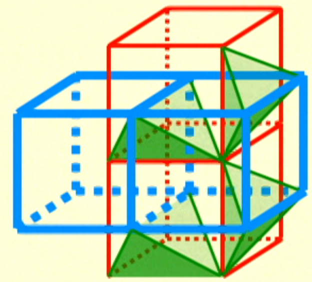
$\varkappa_{abcd} = \pm 1$



$\varkappa_{abcdef} = \pm 1$

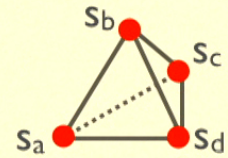
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE AND RANDOM COUPLING ISING MODEL



- 3D color code - stabilizer and subsystem:
  - stabilizers on faces (A) and volumes (B),
  - gauge generators on faces (C).
- Two stat-mech models to analyze:
  - 4-body random coupling Ising model (A),

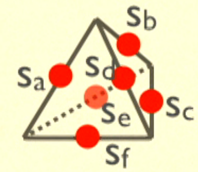
$$H = -\sum \varkappa_{abcd} s_a s_b s_c s_d$$



$$\varkappa_{abcd} = \pm 1$$

  - local order parameter - fairly easy to analyze!
  - 6-body random coupling Ising model (B & C),

$$H = -\sum \varkappa_{abcdef} s_a s_b s_c s_d s_e s_f$$



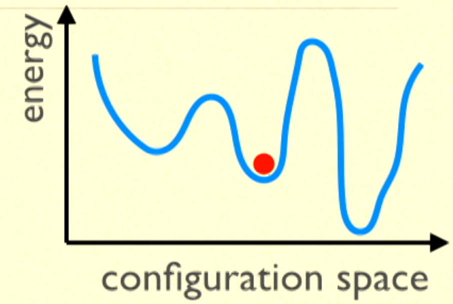
$$\varkappa_{abcdef} = \pm 1$$

  - challenging due to gauge symmetries (no local order parameter).

32

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

CHALLENGES

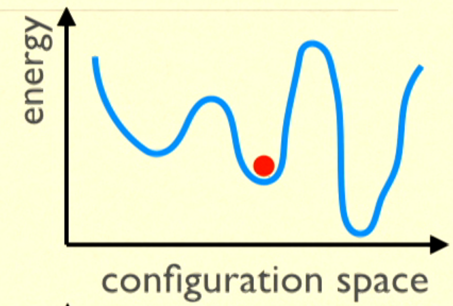


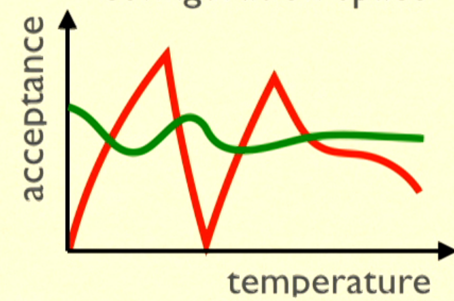
- Simulation - Monte Carlo w/ parallel tempering (replica exchange) to avoid local minima.

work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

CHALLENGES

- Simulation - Monte Carlo w/ parallel tempering (replica exchange) to avoid local minima.
- How to choose temperatures for parallel tempering? Flat acceptance rate (not optimal).



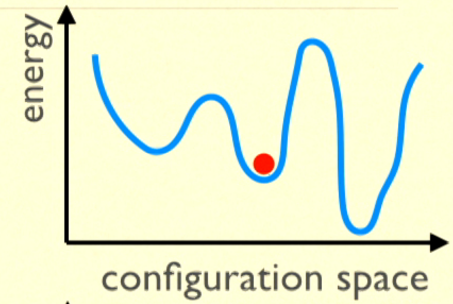


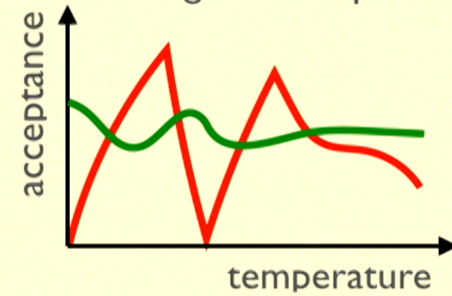
33

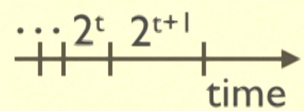
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

CHALLENGES

- Simulation - Monte Carlo w/ parallel tempering (replica exchange) to avoid local minima.
- How to choose temperatures for parallel tempering? Flat acceptance rate (not optimal).
- When does the system thermalize? Logarithmic binning of data. Critical slowing down!





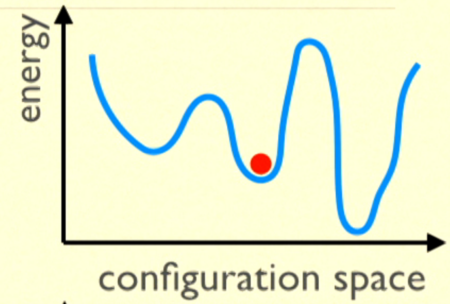


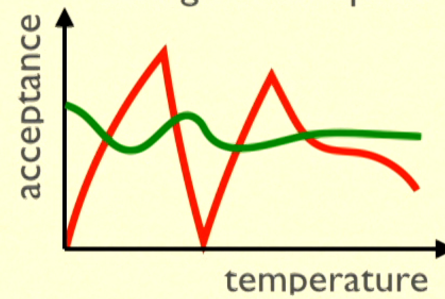
33

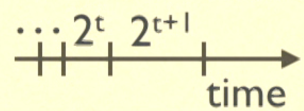
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

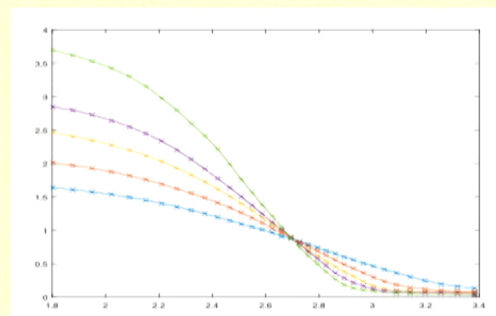
CHALLENGES

- Simulation - Monte Carlo w/ parallel tempering (replica exchange) to avoid local minima.
- How to choose temperatures for parallel tempering? Flat acceptance rate (not optimal).
- When does the system thermalize? Logarithmic binning of data. Critical slowing down!
- Suitable order parameters to detect phase transition: correlation length and Wilson loops.





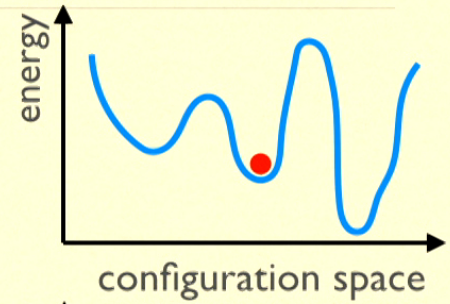


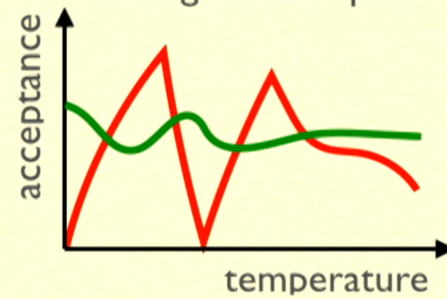


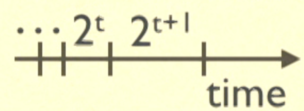
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

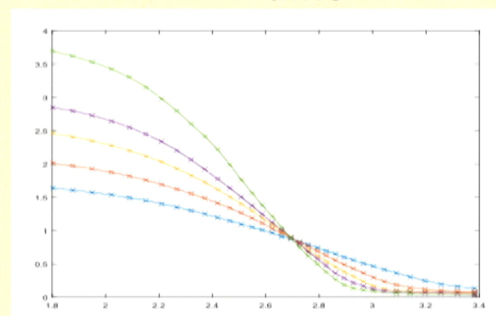
CHALLENGES

- Simulation - Monte Carlo w/ parallel tempering (replica exchange) to avoid local minima.
- How to choose temperatures for parallel tempering? Flat acceptance rate (not optimal).
- When does the system thermalize? Logarithmic binning of data. Critical slowing down!
- Suitable order parameters to detect phase transition: correlation length and Wilson loops.
- Finite-size scaling effects!
- How to analyze data? Bootstrap methods to estimate statistical errors.



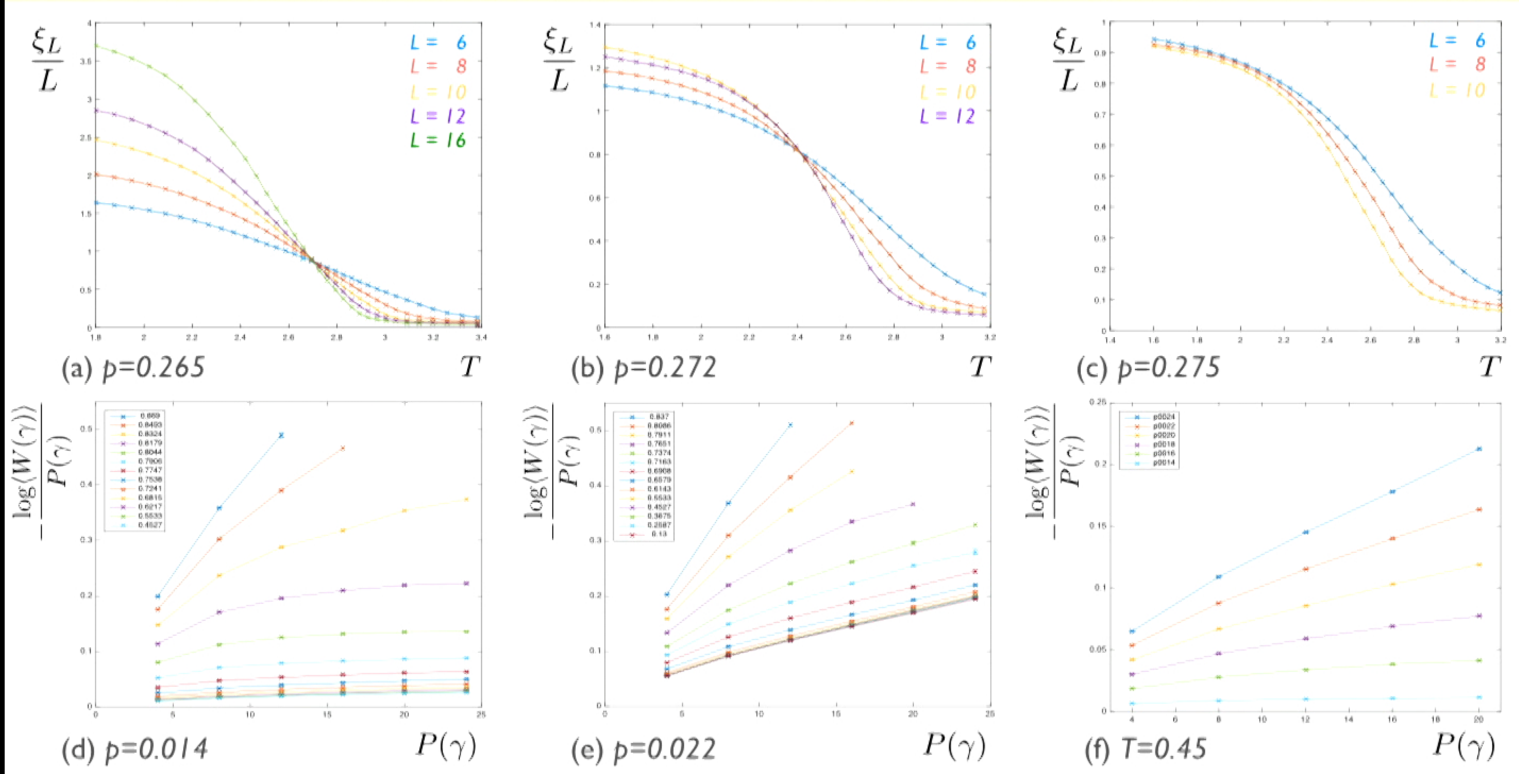






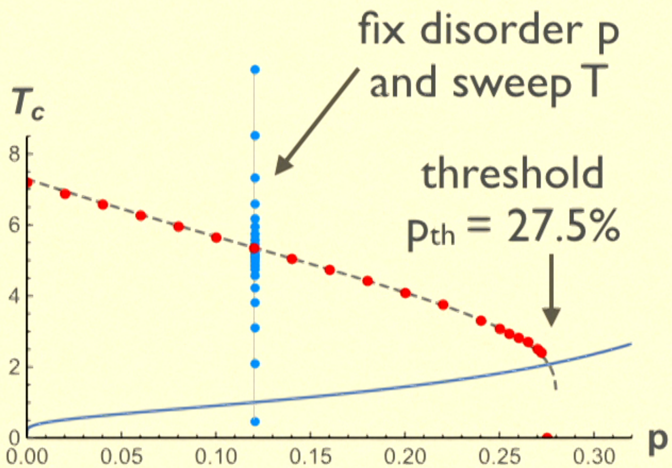
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

NUMERICS

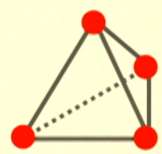


work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE THRESHOLDS FROM PHASE DIAGRAMS



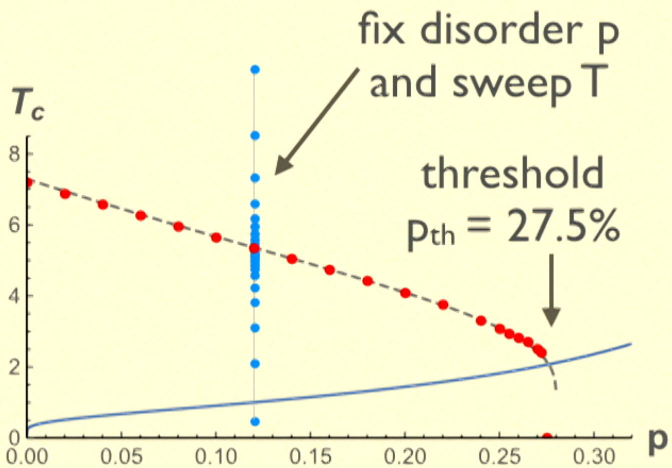
- 4-body RCIM

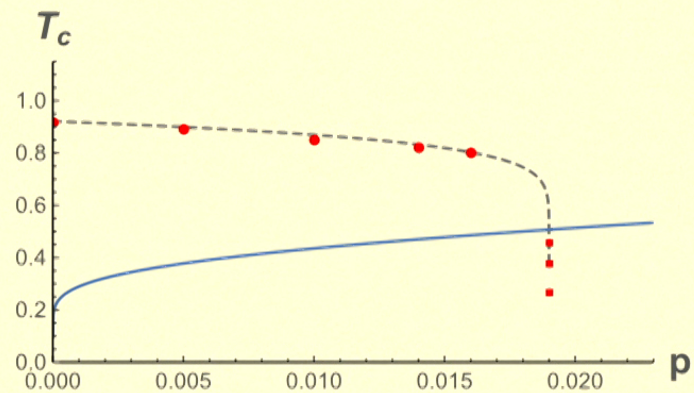


35

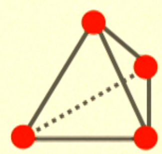
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE THRESHOLDS FROM PHASE DIAGRAMS

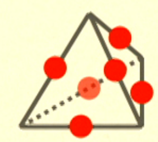




- 4-body RCIM
- threshold for sheet-like (2D) logical operators



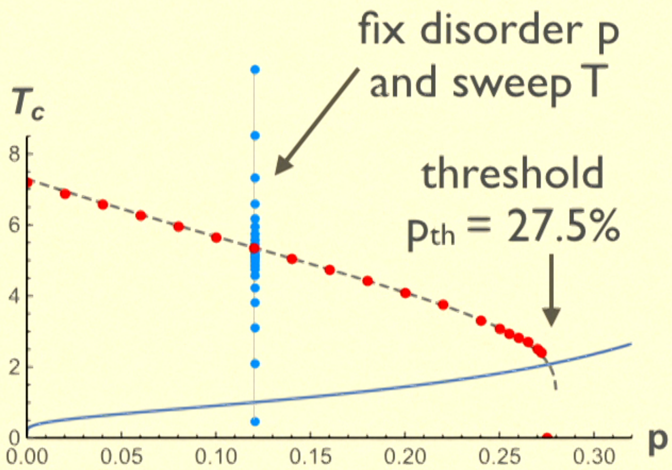
- 6-body RCIM

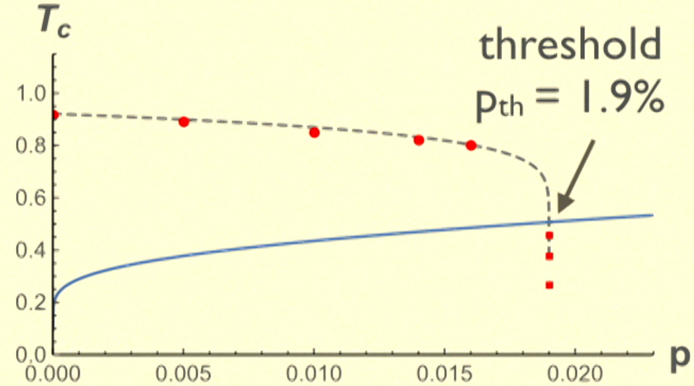


35

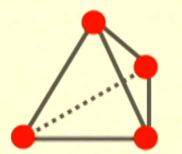
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE THRESHOLDS FROM PHASE DIAGRAMS

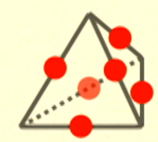




- 4-body RCIM
- threshold for sheet-like (2D) logical operators



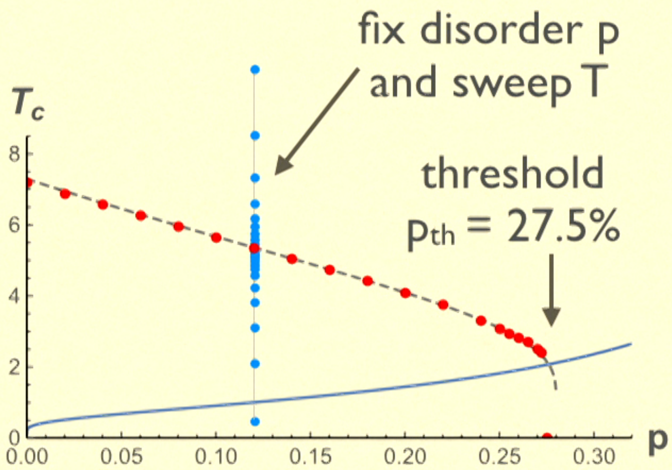
- 6-body RCIM

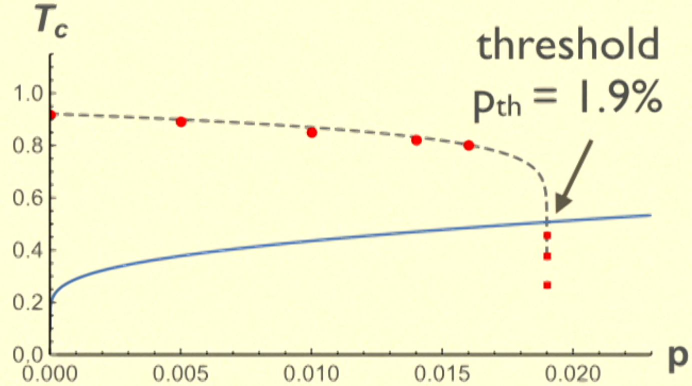


35

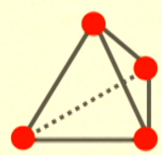
work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

3D COLOR CODE THRESHOLDS FROM PHASE DIAGRAMS

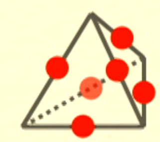




- 4-body RCIM
- threshold for sheet-like (2D) logical operators

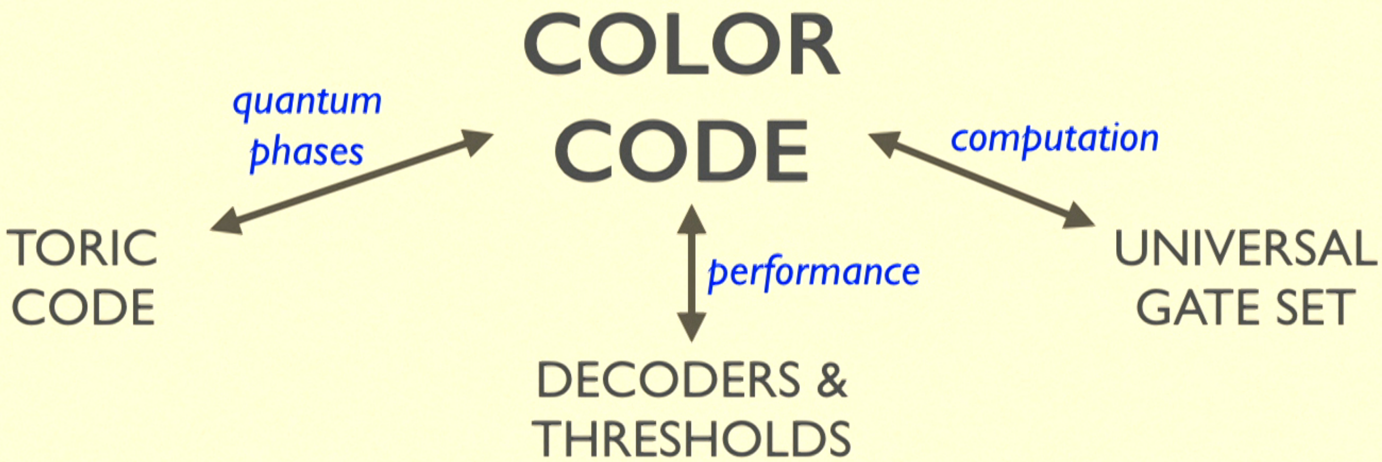


- 6-body RCIM
- threshold for string-like (1D) logical operators



work w/ **M. Beverland, F. Brandao, J. Preskill, K. Svore**

SUMMARY


COLOR CODE
quantum phases
TORIC CODE
computation
UNIVERSAL GATE SET
performance
DECODERS & THRESHOLDS

36