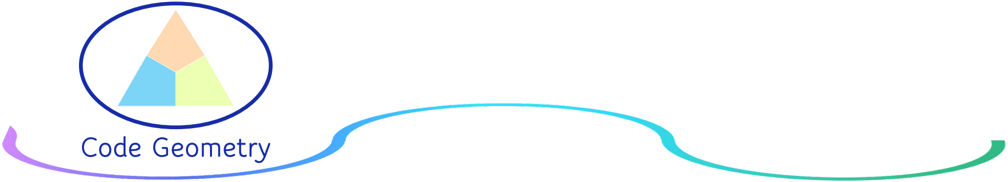
Graphical CSS Code Transformation Using ZX Calculus

Jiaxin Huang[1] Sarah Meng Li[2,3] Lia Yeh[4,5]
Aleks Kissinger[4] Michele Mosca [2,3,6] Michael Vasmer [2,6]

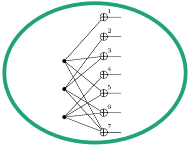
[1]QICI Quantum Information and Computation Initiative, Department of Computer Science, The University of Hong Kong.
[2]Department of Combinatorics & Optimization, [3]Institute for Quantum Computing, University of Waterloo
[4]Department of Computer Science, University of Oxford [5]Quantinuum
[6]Perimeter Institute for Theoretical Physics

IQC Student Seminar 2023

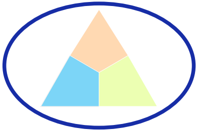
Can we unify the representations of a CSS code in one graphical language?



Can we unify the representations of a CSS code in one graphical language?

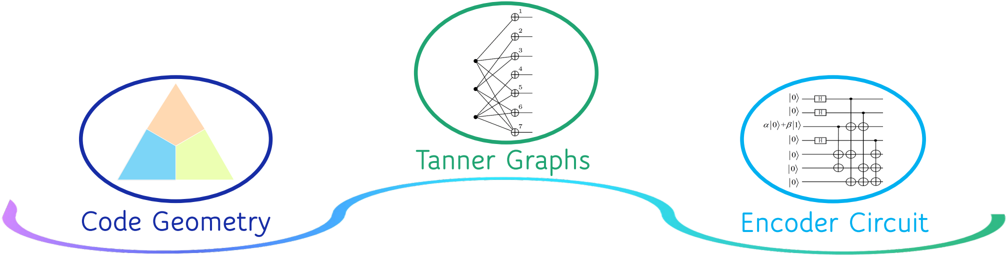


Tanner Graphs

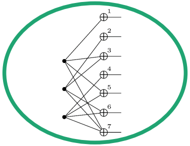


Code Geometry

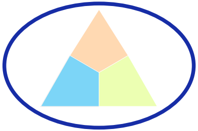
Can we unify the representations of a CSS code in one graphical language?



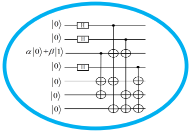
Can we unify the representations of a CSS code in one graphical language?



Tanner Graphs



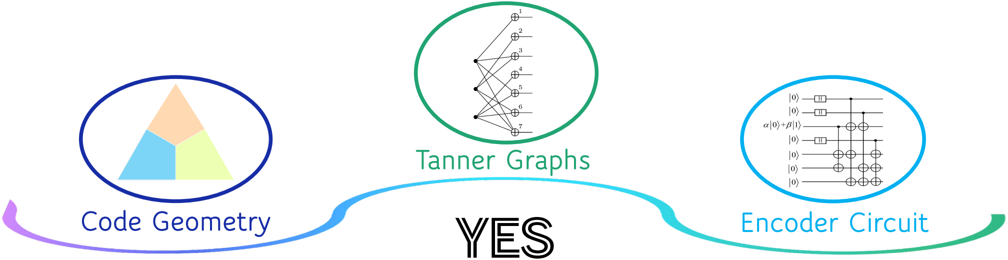
Code Geometry



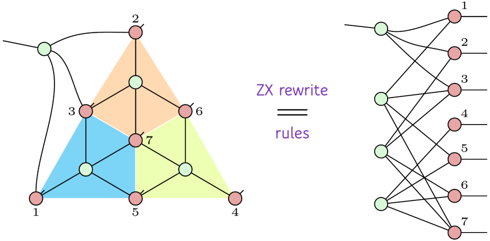
Encoder Circuit

YES

Can we unify the representations of a CSS code in one graphical language?



The ZX-calculus is an intuitive graphical language for quantum computation[1].

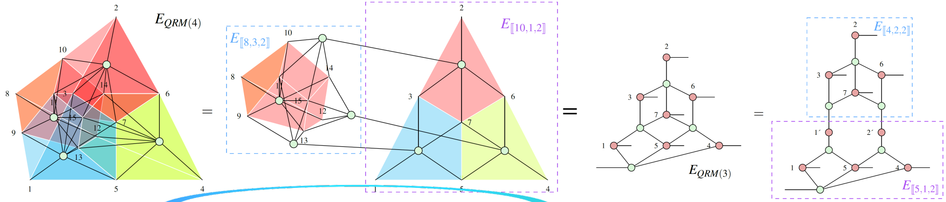


Any CSS encoder has a phase-free ZX normal form which corresponds to both the Tanner graph and geometry[2].

[1] Coecke, B., & Kissinger, A. (2018). Picturing quantum processes: A first course on quantum theory and diagrammatic reasoning. In *Diagrammatic Representation and Inference: 10th International Conference, Diagrams 2018, Edinburgh, UK, June 18-22, 2018, Proceedings 10* (pp. 28-31). Springer International Publishing.
[2] Kissinger, A. (2022). Phase-free ZX diagrams are CSS codes (... or how to graphically grok the surface code). In 19th International Conference on Quantum Physics and Logic.

Graphical transformation between CSS codes

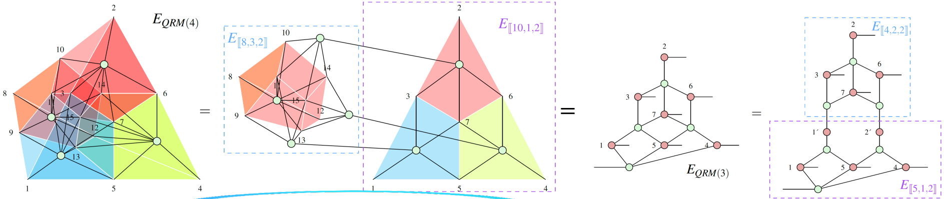
Code Morphing



Graphical transformation between CSS codes

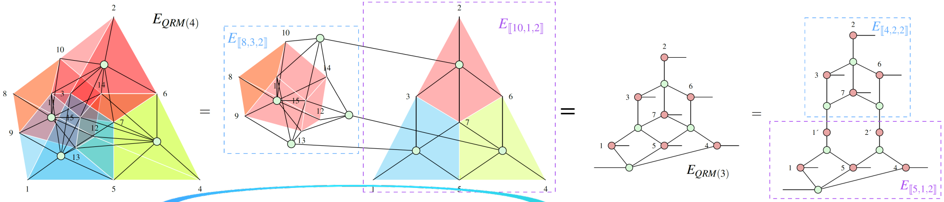
Code Morphing

[3] Vasmer, M., & Kubica, A. (2022). Morphing quantum codes. PRX Quantum, 3(3), 030319.

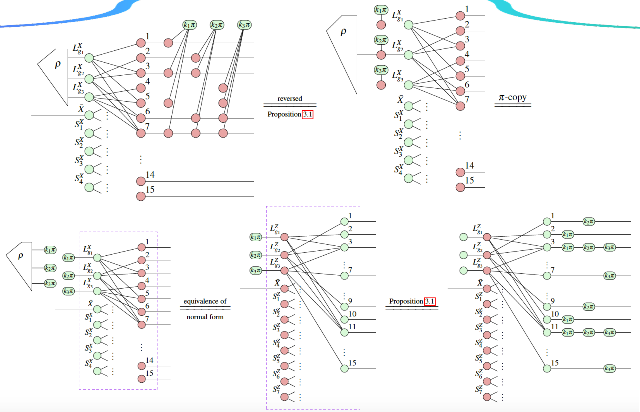


Graphical transformation between CSS codes

Code Morphing



Code Switching

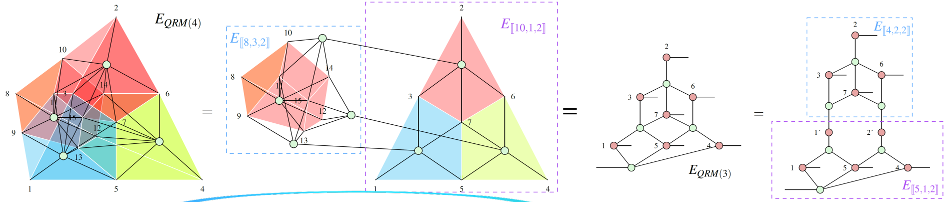


Subsystem Code Gauge Fixing

Graphical transformation between CSS codes

Code Morphing

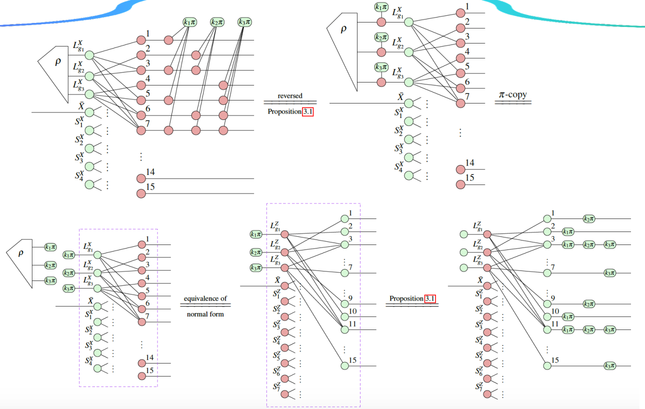
[3] Vasmer, M., & Kubica, A. (2022). Morphing quantum codes. PRX Quantum, 3(3), 030319.



Code Switching

[4] Anderson, J. T., Duclos-Cianci, G., & Poulin, D. (2014). Fault-tolerant conversion between the steane and reed-muller quantum codes. Physical review letters, 113(8), 080501.

[5] Quan, D. X., Zhu, L. L., Pei, C. X., & Sanders, B. C. (2018). Fault-tolerant conversion between adjacent Reed–Muller quantum codes based on gauge fixing. Journal of Physics A: Mathematical and Theoretical, 51(11), 115305.



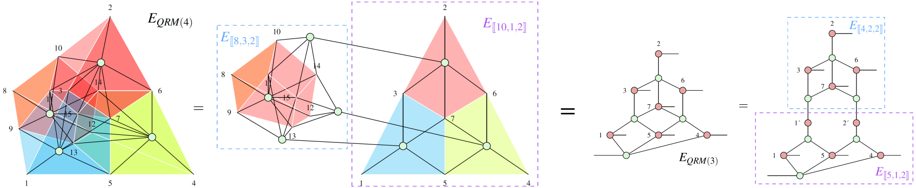
Subsystem Code Gauge Fixing

[6] Paetznick, A., & Reichardt, B. W. (2013). Universal fault-tolerant quantum computation with only transversal gates and error correction. Physical review letters, 111(9), 090505.

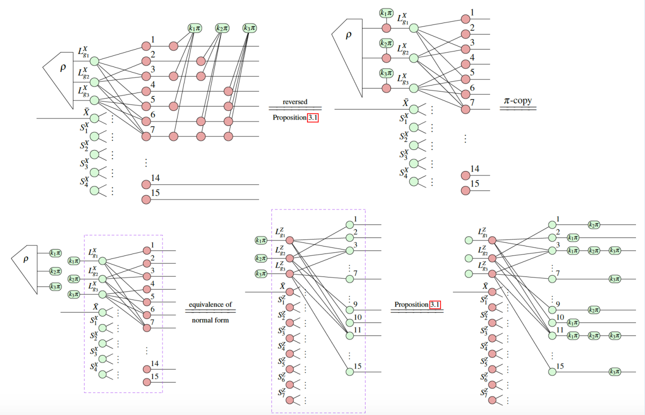
[7] Vuillot, C., Lao, L., Criger, B., Almud´ever, C. G., Bertels, K., & Terhal, B. M. (2019). Code deformation and lattice surgery are gauge fixing. New Journal of Physics, 21(3), 033028.

Graphical transformation between CSS codes

Code Morphing



Code Switching

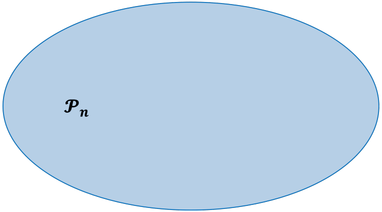


Subsystem Code Gauge Fixing

Stabilizer Code

Consider three groups of Pauli operators.

1. Pauli group on n qubits: $\mathcal{P}_n = \{i^c \left(\bigotimes_{i=1}^{n} P_i\right);\ P_i \in \{X, Y, Z, I\},\ 0 \leq c \leq 3\}$.

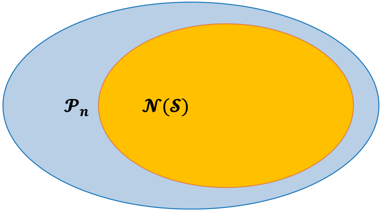


[1] Gottesman, D. (1997). Stabilizer codes and quantum error correction. California Institute of Technology.

Stabilizer Code

Consider three groups of Pauli operators.

1. Pauli group on n qubits: $\mathcal{P}_n = \{i^c \left(\bigotimes_{i=1}^n P_i\right);\ P_i \in \{X, Y, Z, I\},\ 0 \le c \le 3\}$.
2. Stabilizer group: $\mathcal{S} = \langle M_1, M_2, \ldots, M_{n-k}\rangle$, $-I \notin \mathcal{S}$. $\mathcal{S} \subset \mathcal{P}_n$. $\mathcal{S}$ Abelian.

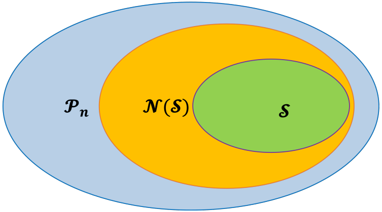


[1] Gottesman, D. (1997). Stabilizer codes and quantum error correction. California Institute of Technology.

Stabilizer Code

Consider three groups of Pauli operators.

1. Pauli group on n qubits: $\mathcal{P}_n = \{i^c \left(\bigotimes_{i=1}^{n} P_i\right);\ P_i \in \{X, Y, Z, I\},\ 0 \leq c \leq 3\}$.
2. Stabilizer group: $\mathcal{S} = \langle M_1, M_2, \ldots, M_{n-k} \rangle$, $-I \notin \mathcal{S}$. $\mathcal{S} \subset \mathcal{P}_n$. $\mathcal{S}$ Abelian.
3. Centralizer of $\mathcal{S}$: $\mathcal{N}(\mathcal{S}) = \{U \in \mathcal{P}_n;\ [U, M] = 0,\ \forall M \in \mathcal{S}\}$.

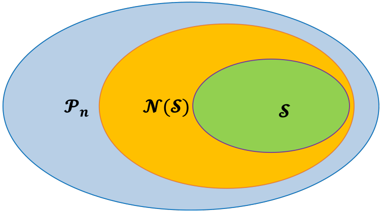


[1] Gottesman, D. (1997). Stabilizer codes and quantum error correction. California Institute of Technology.

Stabilizer Code

Consider three groups of Pauli operators.

1. Pauli group on n qubits: $\mathcal{P}_n = \{i^c \left(\bigotimes_{i=1}^n P_i\right);\ P_i \in \{X, Y, Z, I\},\ 0 \leq c \leq 3\}$.
2. Stabilizer group: $\mathcal{S} = \langle M_1, M_2, \ldots, M_{n-k} \rangle$, $-I \notin \mathcal{S}$. $\mathcal{S} \subset \mathcal{P}_n$. $\mathcal{S}$ Abelian.
3. Centralizer of $\mathcal{S}$: $\mathcal{N}(\mathcal{S}) = \{U \in \mathcal{P}_n;\ [U, M] = 0,\ \forall M \in \mathcal{S}\}$.



Definition

Stabilizer codes are a class of quantum error-correcting codes. Its code space $\mathcal{C}$ is the joint $+1$ eigenspace of $\mathcal{S}$.

[1] Gottesman, D. (1997). Stabilizer codes and quantum error correction. California Institute of Technology.

Code Space

$\lvert\overline{\psi}\rangle$ is called a *codeword* in $\mathcal{C}$, where

$$\mathcal{C} := \{n\text{-qubit state}\,\lvert\overline{\psi}\rangle\,;\, M\lvert\overline{\psi}\rangle = \lvert\overline{\psi}\rangle\,, \forall M \in \mathcal{S}\}$$

There are three important parameters for a stabilizer code: $[[n, k, d]]$.

- n is the number of physical qubits.
- k is the number of logical (or encoded) qubits.
- d is the code distance.

Code Space

$|\overline{\psi}\rangle$ is called a *codeword* in $\mathcal{C}$, where

$$\mathcal{C} := \{n\text{-qubit state } |\overline{\psi}\rangle\,;\, M|\overline{\psi}\rangle = |\overline{\psi}\rangle\,, \forall M \in \mathcal{S}\}$$

There are three important parameters for a stabilizer code: $[[n, k, d]]$.

- n is the number of physical qubits.
- k is the number of logical (or encoded) qubits.
- d is the code distance.

Example

Consider $\mathcal{S} = \langle XX, ZZ\rangle$ on two qubits. Then $\mathcal{C} = \left\{\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)\right\}$.

Logical Operators

Consider the centralizer of $\mathcal{S}$,

$$\mathcal{N}(\mathcal{S}) := \{U \in \mathcal{P}_n ; [U, M] = 0, \forall M \in \mathcal{S}\}.$$

Logical Operators

Consider the centralizer of $\mathcal{S}$,

$$\mathcal{N}(\mathcal{S}) := \{U \in \mathcal{P}_n; [U, M] = 0, \forall M \in \mathcal{S}\}.$$

- Since $\mathcal{S}$ is Abelian, $\mathcal{S} \subset \mathcal{N}(\mathcal{S})$. They act trivially on $|\overline{\psi}\rangle$.

Logical Operators

Consider the centralizer of $\mathcal{S}$,

$$\mathcal{N}(\mathcal{S}) := \{U \in \mathcal{P}_n; [U, M] = 0, \forall M \in \mathcal{S}\}.$$

- Since $\mathcal{S}$ is Abelian, $\mathcal{S} \subset \mathcal{N}(\mathcal{S})$. They act trivially on $|\overline{\psi}\rangle$.
- $\overline{X_1}, \overline{Z_1}, \ldots, \overline{X_k}, \overline{Z_k} \in \mathcal{N}(\mathcal{S})/\mathcal{S}$, up to the generators of $\mathcal{S}$.

  They are anticommuting Pauli pairs acting non-trivially on $|\overline{\psi}\rangle$.

Logical Operators

Consider the centralizer of $\mathcal{S}$,

$$\mathcal{N}(\mathcal{S}) := \{U \in \mathcal{P}_n ; [U, M] = 0, \forall M \in \mathcal{S}\}.$$

- Since $\mathcal{S}$ is Abelian, $\mathcal{S} \subset \mathcal{N}(\mathcal{S})$. They act trivially on $|\overline{\psi}\rangle$.
- $\overline{X_1}, \overline{Z_1}, \ldots, \overline{X_k}, \overline{Z_k} \in \mathcal{N}(\mathcal{S})/\mathcal{S}$, up to the generators of $\mathcal{S}$.

  They are anticommuting Pauli pairs acting non-trivially on $|\overline{\psi}\rangle$.
- All other operators in $\mathcal{P}_n$ anti-commute with at least one element in $\mathcal{S}$ and map a codeword $|\overline{\psi}\rangle$ onto a state outside the code space $\mathcal{C}$.

Fundamental Theorem of Stabilizer Theory

Theorem

If $\mathcal{S} \subset \mathcal{P}_n$ has m generators, then $\mathcal{C}$ is a 2^k dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$, $k = n - m$.

Fundamental Theorem of Stabilizer Theory

Theorem

If $\mathcal{S} \subset \mathcal{P}_n$ has m generators, then $\mathcal{C}$ is a 2^k dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$, $k = n - m$.

- $\mathcal{S}$ is *maximal* when $m = n$. $\mathcal{S}$ fixes a $2^0 = 1$ dimensional subspace, i.e. a quantum state, up to scalar factor.

Fundamental Theorem of Stabilizer Theory

Theorem

If $\mathcal{S} \subset \mathcal{P}_n$ has m generators, then $\mathcal{C}$ is a 2^k dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$, $k = n - m$.

- $\mathcal{S}$ is *maximal* when $m = n$. $\mathcal{S}$ fixes a $2^0 = 1$ dimensional subspace, i.e. a quantum state, up to scalar factor.
- More generally, we think of non-maximal stabiliser groups as a description for the embedding of $k = n - m$ "logical" qubits into a space of n "physical" qubits.

Fundamental Theorem of Stabilizer Theory

Theorem

If $\mathcal{S} \subset \mathcal{P}_n$ has m generators, then $\mathcal{C}$ is a 2^k dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$, $k = n - m$.

- $\mathcal{S}$ is *maximal* when $m = n$. $\mathcal{S}$ fixes a $2^0 = 1$ dimensional subspace, i.e. a quantum state, up to scalar factor.
- More generally, we think of non-maximal stabiliser groups as a description for the embedding of $k = n - m$ "logical" qubits into a space of n "physical" qubits.

Example: Four-qubit code [[4, 2, 2]]

$$\mathcal{S} = \langle XXXX, ZZZZ \rangle$$

- What is the dimension of the code space?

Code Distance

Definition

Let d be the distance of a stabilizer code $\mathcal{C}(\mathcal{S})$, $|P|$ denotes the weight of $P \in \mathcal{P}_n$, the number of physical qubits on which P acts nontrivially. Then

$$d := \min_{P \in \mathcal{N}(\mathcal{S})/\mathcal{S}} |P|.$$

The code distance is the **minimum weight** of any logical operator.

Code Distance

Definition

Let d be the distance of a stabilizer code $\mathcal{C}(\mathcal{S})$, $|P|$ denotes the weight of $P \in \mathcal{P}_n$, the number of physical qubits on which P acts nontrivially. Then

$$d := \min_{P \in \mathcal{N}(\mathcal{S})/\mathcal{S}} |P|.$$

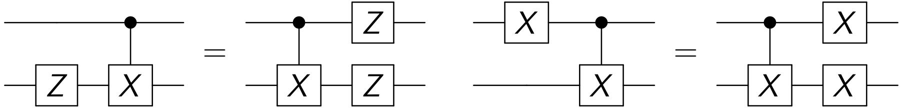
The code distance is the **minimum weight** of any logical operator.

Example: Four-qubit code $[[4, 2, 2]]$

$$\mathcal{S} = \langle XXXX, ZZZZ \rangle$$

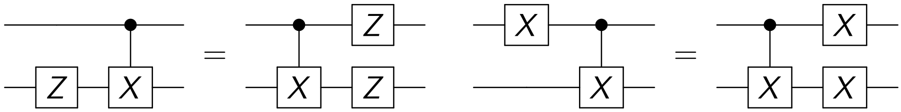
- Find pairs of mutually anti-commuting Paulis which commute with $XXXX$, $ZZZZ$.
- What is the code distance?

Fault-tolerant Technique: Transversality



[2]Gottesman, D. (2000). Fault-tolerant quantum computation with local gates. Journal of Modern Optics, 47(2-3), 333-345.

Fault-tolerant Technique: Transversality

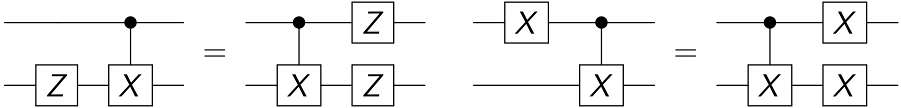


Definition

A transversal logical operator is **NOT** implemented by any multi-qubit physical operation acting on the same code block.

[2] Gottesman, D. (2000). Fault-tolerant quantum computation with local gates. Journal of Modern Optics, 47(2-3), 333-345.

Fault-tolerant Technique: Transversality



Definition

A transversal logical operator is **NOT** implemented by any multi-qubit physical operation acting on the same code block.

- Transversality prevents any errors from spreading within a block, so a single physical error cannot cause a whole block of codes to go bad.

[2]Gottesman, D. (2000). Fault-tolerant quantum computation with local gates. Journal of Modern Optics, 47(2-3), 333-345.

Calderbank-Shor-Steane (CSS) Codes

Definition

CSS codes are stabilizer codes whose stabilizer generators are defined by two orthogonal binary matrices G, H, $GH^T = 0$. Moreover,

$$\mathcal{S} = \langle G^X, H^Z \rangle.$$

Calderbank-Shor-Steane (CSS) Codes

Definition

CSS codes are stabilizer codes whose stabilizer generators are defined by two orthogonal binary matrices G, H, $GH^T = 0$. Moreover,

$$\mathcal{S} = \langle G^X, H^Z \rangle.$$

- The stabilizer generators can be divided into two types: X type and Z type.

Calderbank-Shor-Steane (CSS) Codes

Definition

CSS codes are stabilizer codes whose stabilizer generators are defined by two orthogonal binary matrices G, H, $GH^T = 0$. Moreover,

$$\mathcal{S} = \langle G^X, H^Z \rangle.$$

- The stabilizer generators can be divided into two types: X type and Z type.
- $GH^T = 0$ implies that each X generator overlaps with a Z generator in an even number of places.

Calderbank-Shor-Steane (CSS) Codes

Definition

CSS codes are stabilizer codes whose stabilizer generators are defined by two orthogonal binary matrices G, H, $GH^T = 0$. Moreover,

$$\mathcal{S} = \langle G^X, H^Z \rangle.$$

- The stabilizer generators can be divided into two types: X type and Z type.
- $GH^T = 0$ implies that each X generator overlaps with a Z generator in an even number of places.
- Example: The $[\![7, 1, 3]\!]$ Steane code $\mathcal{S} = \langle S_1^X, S_2^X, S_3^X, S_1^Z, S_2^Z, S_3^Z \rangle$.

$$\begin{aligned} S_1^X &= X \otimes I \otimes X \otimes I \otimes X \otimes I \otimes X \\ &= X_1 X_3 X_5 X_7 \\ S_2^X &= X_2 X_3 X_6 X_7, \;\; S_3^X = X_4 X_5 X_6 X_7 \end{aligned}$$

$$\begin{aligned} S_1^Z &= Z \otimes I \otimes Z \otimes I \otimes Z \otimes I \otimes Z \\ &= Z_1 Z_3 Z_5 Z_7 \\ S_2^Z &= Z_2 Z_3 Z_6 Z_7, \;\; S_3^Z = Z_4 Z_5 Z_6 Z_7 \end{aligned}$$

Record Stabilizers for CSS Codes

Definition

Let M be a $k \times n$ binary matrix and $T \in \mathcal{X}, \mathcal{Z}$, then

$$M^T := \left\{ \bigotimes_{j=1}^{n} T^{[M]_{ij}} : 1 \leq i \leq k \right\} \subset \mathcal{P}^{\otimes n}.$$

Record Stabilizers for CSS Codes

Definition

Let M be a $k \times n$ binary matrix and $T \in \mathcal{X}, \mathcal{Z}$, then

$$M^T := \left\{ \bigotimes_{j=1}^{n} T^{[M]_{ij}} : 1 \leq i \leq k \right\} \subset \mathcal{P}^{\otimes n}.$$

- Example: For the Steane code,

$$M = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}_{3\times 7}$$

Record Stabilizers for CSS Codes

Definition

Let M be a $k \times n$ binary matrix and $T \in \mathcal{X}, \mathcal{Z}$, then

$$M^T := \left\{ \bigotimes_{j=1}^{n} T^{[M]_{ij}} : 1 \leq i \leq k \right\} \subset \mathcal{P}^{\otimes n}.$$

- Example: For the Steane code,

$$M = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}_{3\times 7}$$

- Then $\mathcal{S} = \langle M^X, M^Z \rangle$.

The ZX Calculus

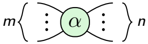
- An intuitive graphical language for quantum computation.

The ZX Calculus

- An intuitive graphical language for quantum computation.
- Every ZX diagram is composed of two types of generators:

The ZX Calculus

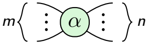
- An intuitive graphical language for quantum computation.
- Every ZX diagram is composed of two types of generators:
  - Z spiders, which sum over the eigenbasis of the Z operator:



$$:= |0\rangle^{\otimes n} \langle 0|^{\otimes m} + e^{i\alpha} |1\rangle^{\otimes n} \langle 1|^{\otimes m},$$

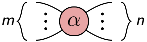
The ZX Calculus

- An intuitive graphical language for quantum computation.
- Every ZX diagram is composed of two types of generators:
  - Z spiders, which sum over the eigenbasis of the Z operator:



$$:= \; |0\rangle^{\otimes n} \langle 0|^{\otimes m} \; + \; e^{i\alpha} \, |1\rangle^{\otimes n} \langle 1|^{\otimes m} \, ,$$

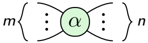
  - X spiders, which sum over the eigenbasis of the X operator:



$$:= \; |+\rangle^{\otimes n} \langle +|^{\otimes m} \; + \; e^{i\alpha} \, |-\rangle^{\otimes n} \langle -|^{\otimes m} \, .$$

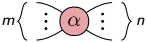
The ZX Calculus

- An intuitive graphical language for quantum computation.
- Every ZX diagram is composed of two types of generators:
  - Z spiders, which sum over the eigenbasis of the Z operator:



$$:= |0\rangle^{\otimes n} \langle 0|^{\otimes m} + e^{i\alpha} |1\rangle^{\otimes n} \langle 1|^{\otimes m},$$

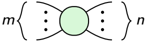
  - X spiders, which sum over the eigenbasis of the X operator:



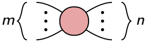
$$:= |+\rangle^{\otimes n} \langle +|^{\otimes m} + e^{i\alpha} |-\rangle^{\otimes n} \langle -|^{\otimes m}.$$

Definition

A ZX diagram is *phase-free* if its spiders have no phases.



$$:= |0\rangle^{\otimes n} \langle 0|^{\otimes m} + |1\rangle^{\otimes n} \langle 1|^{\otimes m}$$

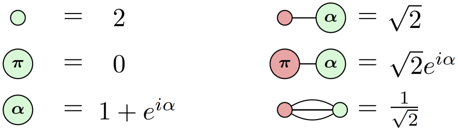


$$:= |+\rangle^{\otimes n} \langle +|^{\otimes m} + |-\rangle^{\otimes n} \langle -|^{\otimes m}$$

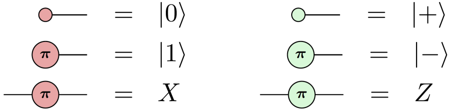
The ZX Calculus is Universal

Any linear map from *m* to *n* qubits corresponds exactly to a ZX diagram.

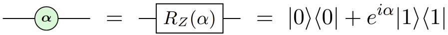
- A ZX diagram with 0 input and output represents a scalar.



- A ZX diagram with 0 input and 1 output represents a state.

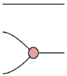


- A ZX diagram with the same number of inputs and outputs represents a unitary.

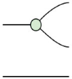


Represent the CNOT Gate in ZX

Proof. Horizontally composing the two diagrams below,

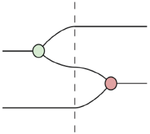


$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

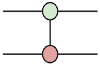


$$= \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

we get



$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Therefore, $CNOT = \sqrt{2}$ 

The ZX Calculus is Complete

If two ZX diagrams represent the same linear map, then there should be a sequence of rewrites that transforms one diagram into the other.

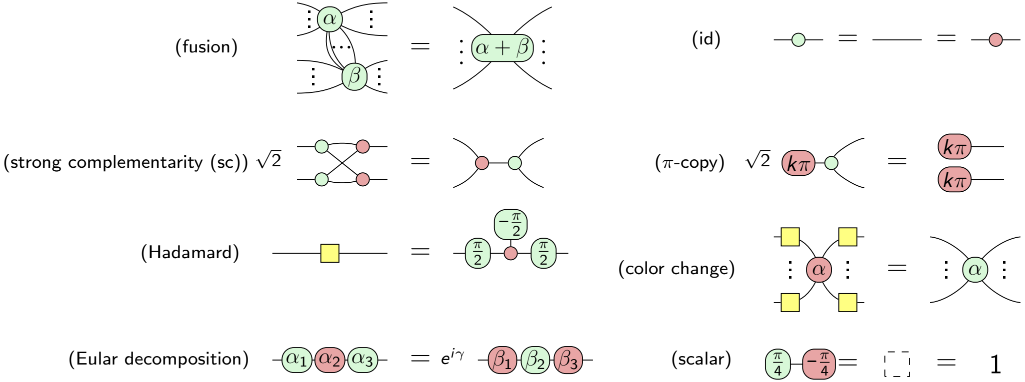
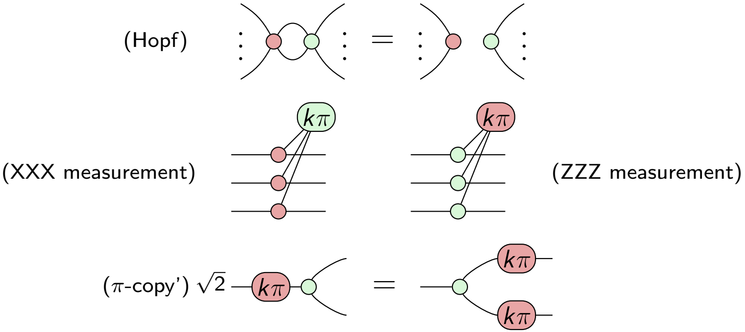


Figure: The minimal complete rule set for ZX calculus.

[3] Vilmart, R. (2019, June). A near-minimal axiomatisation of zx-calculus for pure qubit quantum mechanics. In 2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS) (pp. 1-10). IEEE.

Additional ZX Rules

These rules are derivable from the minimal rule set. Used extensively in this work.



Phase-free ZX Diagrams are CSS Codes

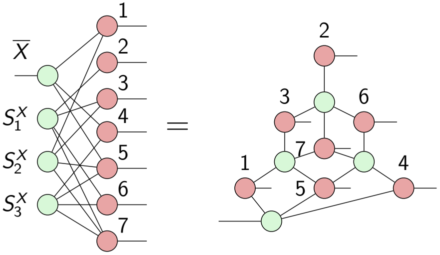
Consider an $[\![n, k, d]\!]$ CSS code with X-type stabilizers $\{S_1^X, S_2^X, \dots, S_{m'}^X\}$ and logical operators $\{\overline{X}_1, \overline{X}_2, \dots, \overline{X}_k\}$, it has a unique ZX normal form.

Example: The Steane code

- $n = 7,\ k = 1,\ d = 3.$
- 3 X-type stabilizers:

$$S_1^X = X_2X_3X_6X_7$$
$$S_2^X = X_1X_3X_5X_7$$
$$S_3^X = X_4X_5X_6X_7$$

- 1 logical X operator: $\overline{X} = X_1X_4X_5$



The Steane code encoder in ZX normal form.

[4]Kissinger, A. (2022). Phase-free ZX diagrams are CSS codes (... or how to graphically grok the surface code). arXiv preprint arXiv:2204.14038.

Subsystem Codes

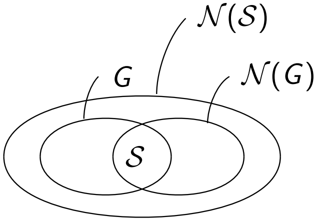
Subsystem codes are stabilizer codes where some of the logical qubits are **NOT** used for information storage and processing. These logical qubits are called *gauge qubits*.

[5] David Kribs, Raymond Laflamme & David Poulin (2005): Unified and Generalized Approach to Quantum Error Correction. Physical Review Letters, 94.

Subsystem Codes

Subsystem codes are stabilizer codes where some of the logical qubits are **NOT** used for information storage and processing. These logical qubits are called *gauge qubits*.

- G is an arbitrary subgroup of the Pauli group $\mathcal{P}$.
- $\mathcal{S} = \mathcal{N}(G) \cap G$, where $\mathcal{N}(G) = \{P \in \mathcal{P} : PM = MP, \forall M \in G\}$.
- $L_g = G \setminus \mathcal{S}$.

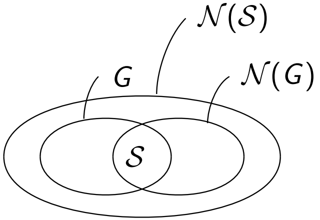


[5] David Kribs, Raymond Laflamme & David Poulin (2005): Unified and Generalized Approach to Quantum Error Correction. Physical Review Letters, 94.

Subsystem Codes

Subsystem codes are stabilizer codes where some of the logical qubits are **NOT** used for information storage and processing. These logical qubits are called *gauge qubits*.

- G is an arbitrary subgroup of the Pauli group $\mathcal{P}$.
- $\mathcal{S} = \mathcal{N}(G) \cap G$, where $\mathcal{N}(G) = \{P \in \mathcal{P} : PM = MP, \forall M \in G\}$.
- $L_g = G \setminus \mathcal{S}$.



Definition

A subsystem code defined by G has a group $\mathcal{S}$ of stabilizers and a set L_g of gauge operators.

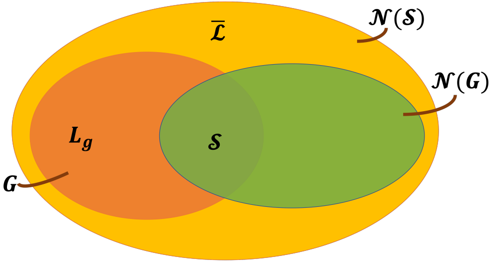
[5] David Kribs, Raymond Laflamme & David Poulin (2005): Unified and Generalized Approach to Quantum Error Correction. Physical Review Letters, 94.

CSS Subsystem Codes

Definition

CSS subsystem codes are subsystem codes whose stabilizers and gauge operators are either X-type or Z-type.

- When $L_g \neq \emptyset$, it contains pairs of anticommuting Pauli operators.

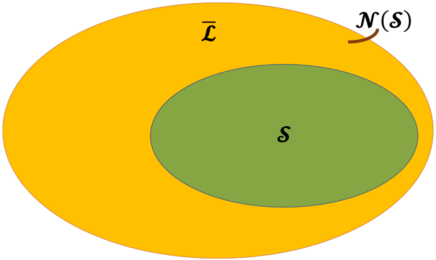


CSS Subsystem Codes

Definition

CSS subsystem codes are subsystem codes whose stabilizers and gauge operators are either X-type or Z-type.

- When $L_g \neq \emptyset$, it contains pairs of anticommuting Pauli operators.
- When G is Abelian, $G = \mathcal{S}$ and $L_g = \emptyset$.



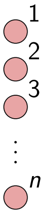
Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

$L_{g_i}^X$ is a gauge operator acting non-trivially on the logical gauge qubit i. It acts trivially on other logical gauge qubits.

Step 1: For each physical qubit, introduce an X spider.

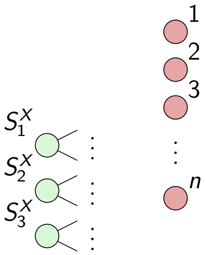


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 2.1: For each X-type stabilizer S_i^X, introduce an Z spider. Connect it to all X spiders where S_i^X has support.

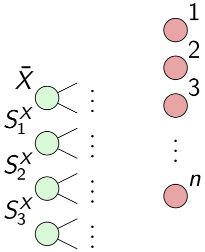


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 2.2: For each X-type logical operator $\overline{X_j}$, introduce an Z spider. Connect it to all X spiders where $\overline{X_j}$ has support.

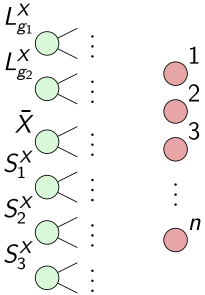


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 2.3: For each X-type gauge operator $L_{g_t}^X$, introduce an Z spider. Connect it to all X spiders where $L_{g_t}^X$ has support.

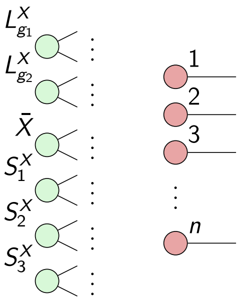


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 3: Give each X spider an output wire.

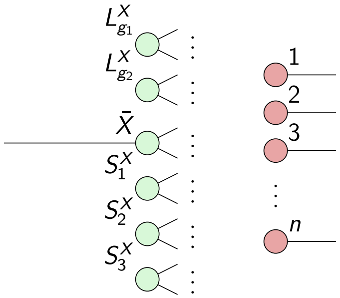


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 4: For each Z spider representing $\overline{X_j}$, give it an output wire.

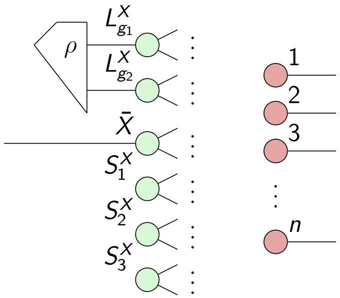


Normal Form for CSS Subsystem Codes

Example: For a CSS subsystem code $[[n, 1, d]]$, let $\overline{X}$ be the logical operator,

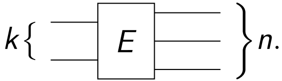
$$\mathcal{S}^X = \langle S_1^X, S_2^X, S_3^X \rangle, \quad L_g^X = \{L_{g_1}^X, L_{g_2}^X\}.$$

Step 5: For all Z spiders representing $L_{g_t}^X$, attach them to a joint arbitrary input state (i.e., a density operator ρ).



Pushing through the Encoder

For any $[\![n, k, d]\!]$ CSS code, its encoder map E is of the form:

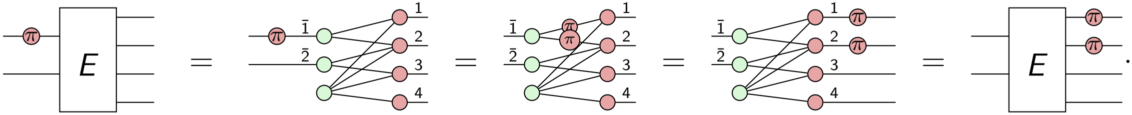


E is an isometry. $E^\dagger E = I$.

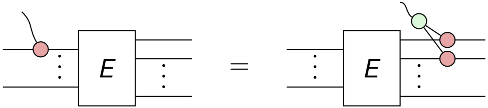
Lemma

In any CSS code, all $\overline{X_i}$ and $\overline{Z_i}$ must be multi-qubit Pauli operators.

Example: For the $[\![4, 2, 2]\!]$ code, $\overline{X_1} = X_1 X_2$:



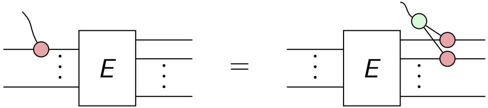
Physically Implement a Logical Operator



Proposition

For any ZX diagram L on the left-hand side of E, one can write down a corresponding ZX diagram P on the right-hand side of E, such that $EL = PE$.

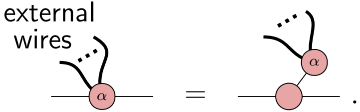
Physically Implement a Logical Operator



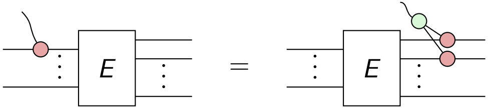
Proposition

For any ZX diagram L on the left-hand side of E, one can write down a corresponding ZX diagram P on the right-hand side of E, such that $EL = PE$.

- Unfuse all spiders on logical qubit wires of L, whenever they are not phase-free or have more than one external wire.



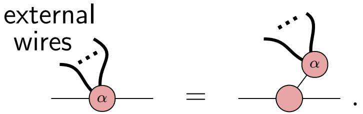
Physically Implement a Logical Operator



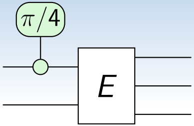
Proposition

For any ZX diagram L on the left-hand side of E, one can write down a corresponding ZX diagram P on the right-hand side of E, such that $EL = PE$.

- Unfuse all spiders on logical qubit wires of L, whenever they are not phase-free or have more than one external wire.

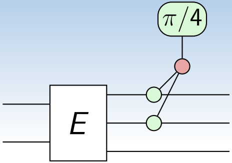


- For each X spider on logical qubit wires, rewriting E to be in ZX normal form and then applying the strong complementarity (sc) rule.



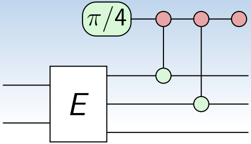
π/4
E

=

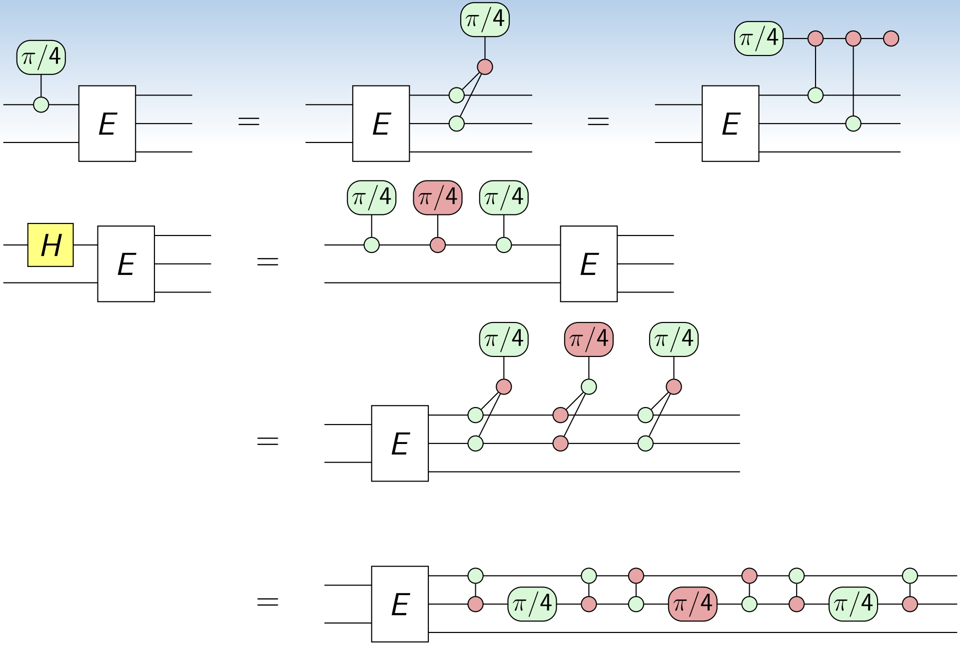


π/4
E

=

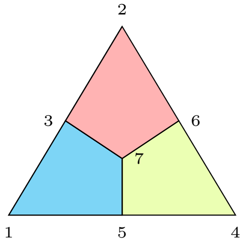


π/4
E

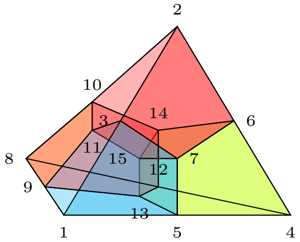

π/4
E
=
H

Switch between Two CSS Codes

	Steane Code: QRM(3)	Quantum Reed-Muller Code: QRM(4)
Code parameters	$[\![7, 1, 3]\!]$	$[\![15, 1, 3]\!]$
Logical operators	$\overline{X} = X_1 X_4 X_5$, $\overline{Z} = Z_1 Z_4 Z_5$	$\overline{X} = X_1 X_4 X_5$, $\overline{Z} = Z_1 Z_4 Z_5$
Transversal gates	$CX, S,$ **H**	$CX, S,$ **T**
Towards universality	Need transversal logical T	Need transversal logical H



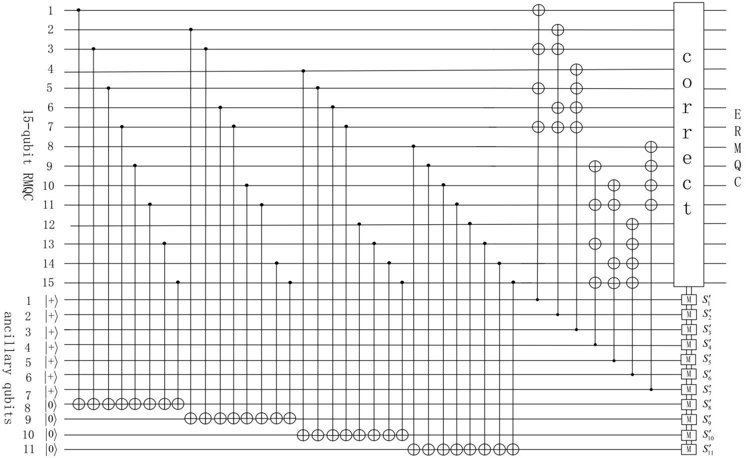
(a) QRM(3) as a 2D color code.



(b) QRM(4) as a 3D color code.

Code Switching[6,7]

- Codes with complementary fault-tolerant gate sets are switched between each other to realize a universal set of logical operations.
- Fault-tolerantly switch between QRM(3) and QRM(4)



[6] Anderson, J. T., Duclos-Cianci, G., & Poulin, D. (2014). Fault-tolerant conversion between the steane and reed-muller quantum codes. Physical review letters, 113(8), 080501.

[7] Quan, D. X., Zhu, L. L., Pei, C. X., & Sanders, B. C. (2018). Fault-tolerant conversion between adjacent Reed–Muller quantum codes based on gauge fixing. Journal of Physics A: Mathematical and Theoretical, 51(11), 115305.

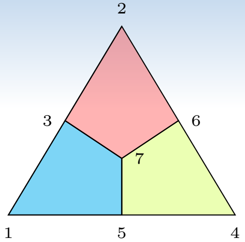
Steane Code & Quantum Reed-Muller Code

QRM(3) & QRM(4) are stabilizer codes defined by the stabilizers $\mathcal{S}_3$ & $\mathcal{S}_4$ respectively.

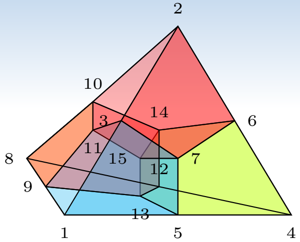
$$\mathcal{S}_3 = \langle M^X, M^Z \rangle, \qquad \mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle, \qquad \text{where}$$

$$M = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}_{3\times 7}, \quad N = \begin{bmatrix} M & 0 & M \\ \mathbf{0} & 1 & \mathbf{1} \end{bmatrix}_{4\times 15}, \quad H = \begin{bmatrix} M & \mathbf{0} \end{bmatrix}_{3\times 15}$$

$$T = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{3\times 15}$$



(a) QRM(3) as a 2D color code.



(b) QRM(4) as a 3D color code.

- $\mathcal{S}_3 = \langle M^X, M^Z \rangle$.
- Coloured face $\mapsto X/Z$ stabilizer generator.
- QRM(3) is self-dual.
- $\overline{X} = X_1X_4X_5$, $\overline{Z} = Z_1Z_4Z_5$.
- $d = 3$.

- $\mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle$.
- Coloured face $\mapsto Z$ stabilizer generator.
- Coloured cell $\mapsto X/Z$ stabilizer generator.
- $\overline{X} = X_1X_4X_5$, $\overline{Z} = Z_1Z_4Z_5$.
- $d = 3$.

Subsystem Quantum Reed-Muller Code: SQRM

Definition

SQRM is defined by the gauge group:

$$G = \langle N^X, N^Z, H^Z, H^X, T^Z \rangle.$$

- The associated stabilizer group, gauge operators and logical operators are:

$$\mathcal{S}_S = \langle N^X, N^Z, H^Z \rangle_{11}, \qquad L_g = \langle H^X, T^Z \rangle_{6}, \qquad \overline{L} = \langle \overline{X}, \overline{Z} \rangle_{2}.$$

- For brevity, we will use $L_g^X = H^X$ and $L_g^Z = T^Z$.

$\Rightarrow$ SQRM has 1 logical qubit and 3 gauge qubits.

$\Rightarrow$ Alternatively, $\mathcal{S}_S$ stabilizes the $[\![15, 4, 3]\!]$ CSS code, with logical operators $\{L_g, \overline{L}\}$.

Extended Quantum Reed-Muller Code: EQRM

EQRM is defined by the stabilizer group $\mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X \rangle$, where

$$M = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}_{3\times 7}, \qquad N = \begin{bmatrix} M & 0 & M \\ \mathbf{0} & 1 & \mathbf{1} \end{bmatrix}_{4\times 15}, \qquad H = \begin{bmatrix} M & \mathbf{0} \end{bmatrix}_{3\times 15}.$$

- Let $|\bar{0}\rangle$ and $|\bar{1}\rangle$ be the logical 0 and 1 encoded in QRM(3).
- Let $|\bar{\psi}\rangle = \alpha |\bar{0}\rangle + \beta |\bar{1}\rangle$ be the single-qubit logical information encoded in QRM(3).

Lemma

An EQRM codeword $|\bar{\Phi}\rangle$ can be decomposed as

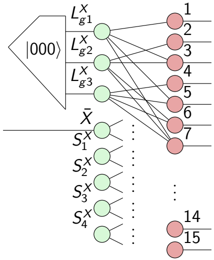
$$|\bar{\Phi}\rangle = |\bar{\psi}\rangle \otimes |\phi\rangle, \text{ where}$$

$|\phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle |\bar{0}\rangle + |1\rangle |\bar{1}\rangle)$.

Gauge Fixing SQRM

Lemma

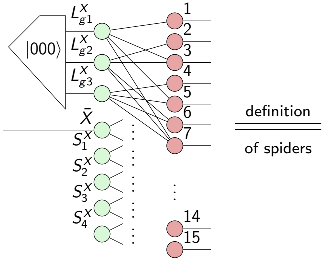
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

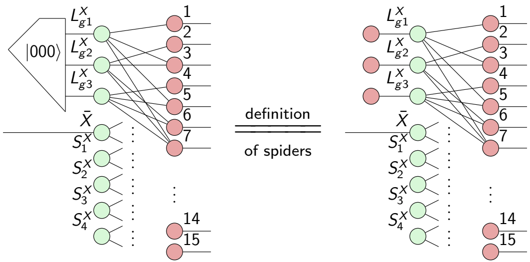
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

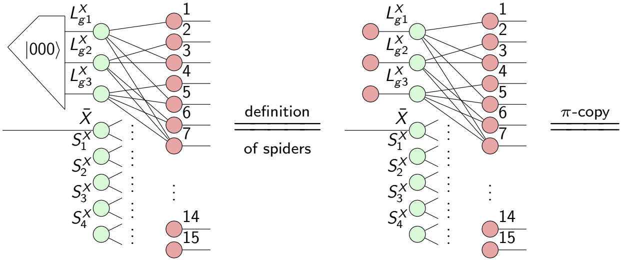
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

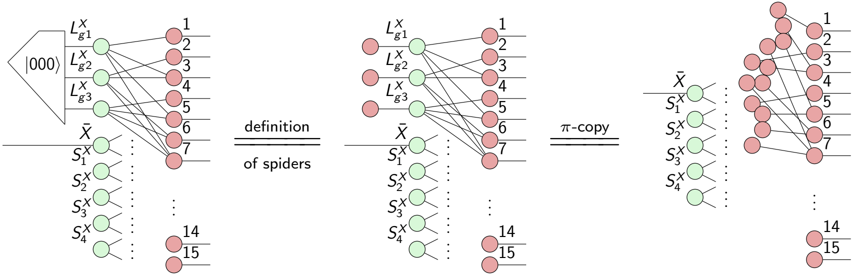
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

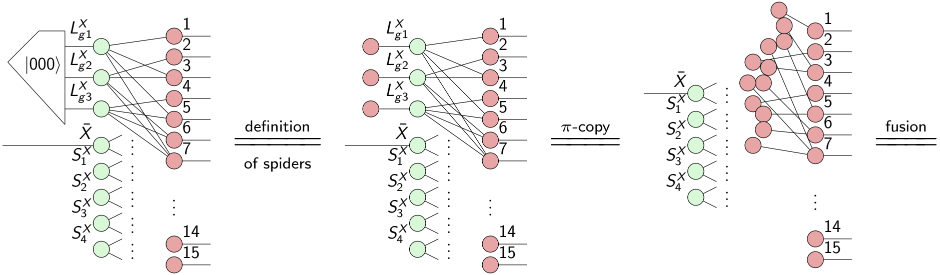
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

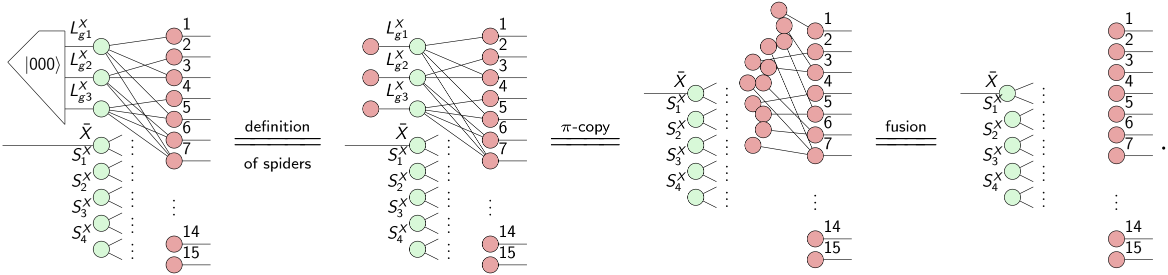
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

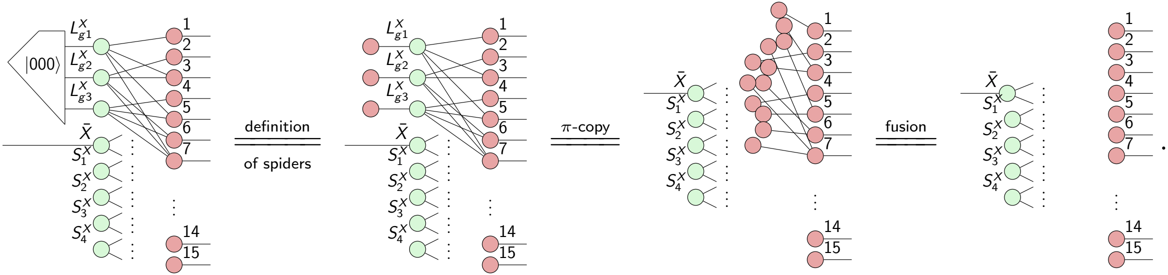
When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



Gauge Fixing SQRM

Lemma

When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).

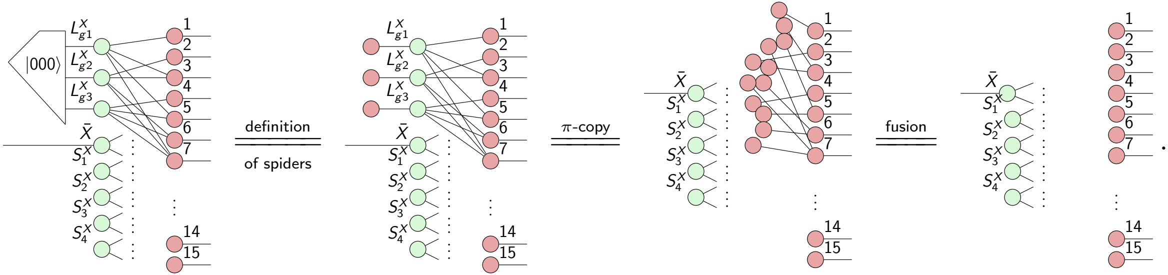


- When the three gauge qubits are in the $|\overline{+++}\rangle$ state, SQRM is fixed to EQRM.

Gauge Fixing SQRM

Lemma

When the three gauge qubits are in the $|\overline{000}\rangle$ state, SQRM is fixed to QRM(4).



- When the three gauge qubits are in the $|\overline{+++}\rangle$ state, SQRM is fixed to EQRM.

$\Rightarrow$ Start with the XZ normal form of the SQRM encoder.

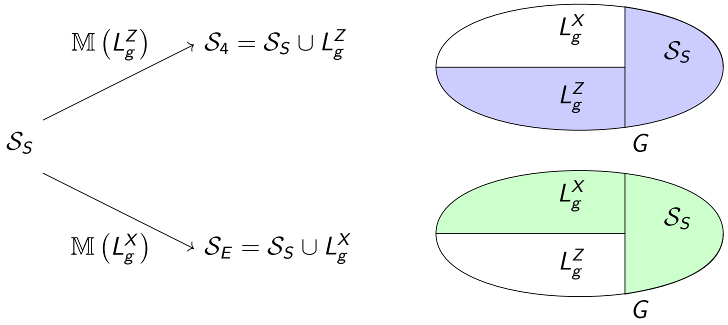
Forward & Backward Switching

- Forward: EQRM $\Rightarrow$ QRM(4) $\mathcal{S}_4$

[8] Paetznick, A., & Reichardt, B. W. (2013). Universal fault-tolerant quantum computation with only transversal gates and error correction. Physical review letters, 111(9), 090505.

Forward & Backward Switching

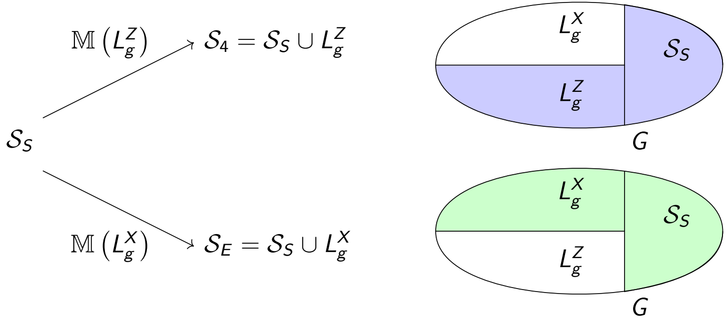
- Forward: EQRM $\Rightarrow$ QRM(4) $\mathcal{S}_{\mathbf{4}}$
- Backward: QRM(4) $\Rightarrow$ EQRM $\mathcal{S}_{\mathbf{E}}$



[8] Paetznick, A., & Reichardt, B. W. (2013). Universal fault-tolerant quantum computation with only transversal gates and error correction. Physical review letters, 111(9), 090505.

Forward & Backward Switching

- Forward: EQRM $\Rightarrow$ QRM(4) $\mathcal{S}_4$
- Backward: QRM(4) $\Rightarrow$ EQRM $\mathcal{S}_E$



[8] Paetznick, A., & Reichardt, B. W. (2013). Universal fault-tolerant quantum computation with only transversal gates and error correction. Physical review letters, 111(9), 090505.

Gauge Fixing SQRM Realizes EQRM

Step 1: : Measure a commuting subset of gauge operators. E.g., measure three X-type gauge operators $L^X_{g_i}$ and obtain the corresponding outcomes $k_1, k_2, k_3 \in \{0, 1\}$:

$$L^X_{g_i} |\overline{\psi}\rangle = (-1)^{k_i} |\overline{\psi}\rangle$$

Gauge Fixing SQRM Realizes EQRM

Step 1: : Measure a commuting subset of gauge operators. E.g., measure three X-type gauge operators $L^X_{g_i}$ and obtain the corresponding outcomes $k_1, k_2, k_3 \in \{0, 1\}$:

$$L^X_{g_i} |\overline{\psi}\rangle = (-1)^{k_i} |\overline{\psi}\rangle$$

Step 2: When $k_i = 1$, the state of gauge qubit i is projected to the wrong state $|\overline{-}\rangle$. Applying the recovery operator $L^Z_{g_i}$ corrects the collapsed state:

$$|\overline{-}\rangle \Rightarrow L^Z_{g_i} |\overline{-}\rangle = |\overline{+}\rangle .$$

Gauge Fixing SQRM Realizes EQRM

Step 1: : Measure a commuting subset of gauge operators. E.g., measure three X-type gauge operators $L^X_{g_i}$ and obtain the corresponding outcomes $k_1, k_2, k_3 \in \{0, 1\}$:

$$L^X_{g_i} |\overline{\psi}\rangle = (-1)^{k_i} |\overline{\psi}\rangle$$

Step 2: When $k_i = 1$, the state of gauge qubit i is projected to the wrong state $|\overline{-}\rangle$. Applying the recovery operator $L^Z_{g_i}$ corrects the collapsed state:

$$|\overline{-}\rangle \Rightarrow L^Z_{g_i} |\overline{-}\rangle = |\overline{+}\rangle .$$

Example 1: Measure an X-type stabilizer

$$X|+\rangle = |+\rangle, \; X|-\rangle = -|-\rangle$$

$$|+\rangle \overset{Z}{\leftrightarrow} |-\rangle$$

Gauge Fixing SQRM Realizes EQRM

Step 1: : Measure a commuting subset of gauge operators. E.g., measure three X-type gauge operators $L^X_{g_i}$ and obtain the corresponding outcomes $k_1, k_2, k_3 \in \{0, 1\}$:

$$L^X_{g_i} \left|\overline{\psi}\right\rangle = (-1)^{k_i} \left|\overline{\psi}\right\rangle$$

Step 2: When $k_i = 1$, the state of gauge qubit i is projected to the wrong state $\left|\overline{-}\right\rangle$. Applying the recovery operator $L^Z_{g_i}$ corrects the collapsed state:

$$\left|\overline{-}\right\rangle \Rightarrow L^Z_{g_i} \left|\overline{-}\right\rangle = \left|\overline{+}\right\rangle .$$

Example 1: Measure an X-type stabilizer

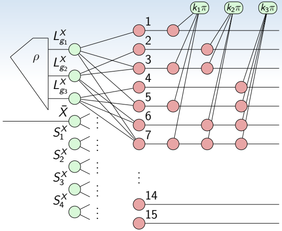
$$X \left|+\right\rangle = \left|+\right\rangle , \; X \left|-\right\rangle = -\left|-\right\rangle$$

$$\left|+\right\rangle \overset{Z}{\longleftrightarrow} \left|-\right\rangle$$

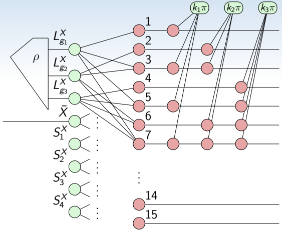
Example 2: Measure a Z-type stabilizer

$$Z \left|0\right\rangle = \left|0\right\rangle , \; Z \left|1\right\rangle = -\left|1\right\rangle$$

$$\left|0\right\rangle \overset{X}{\longleftrightarrow} \left|1\right\rangle$$

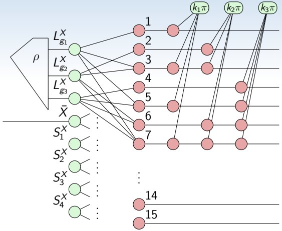


$k_1\pi$
$k_2\pi$
$k_3\pi$
ρ
$L^X_{g_1}$
$L^X_{g_2}$
$L^X_{g_3}$
$\bar{X}$
S^X_1
S^X_2
S^X_3
S^X_4
1
2
3
4
5
6
7
14
15

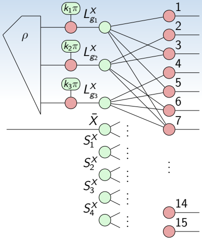


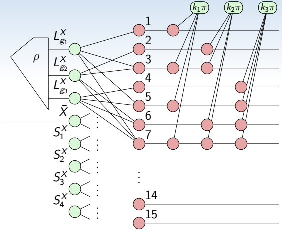
reversed

PTE

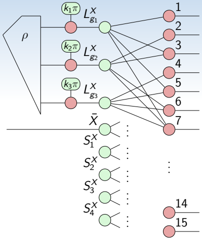


$$\overset{\text{reversed}}{\underset{\text{PTE}}{=\!=}}$$

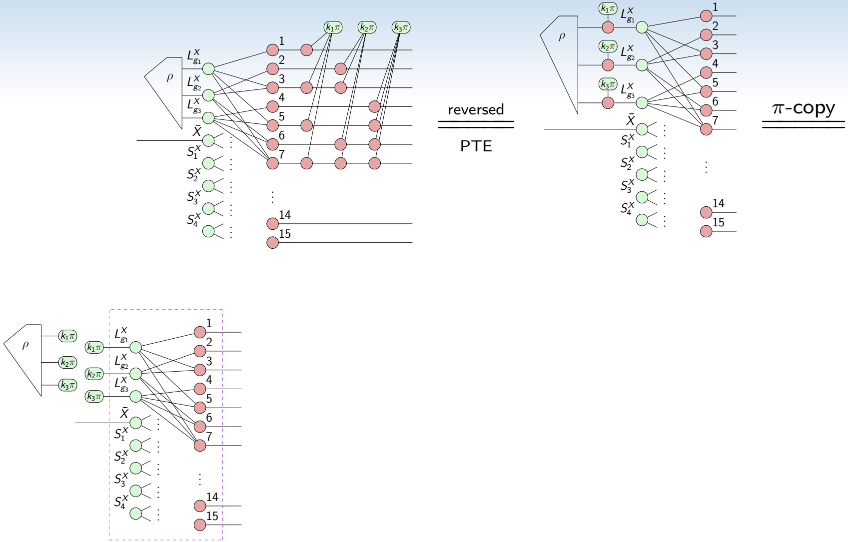




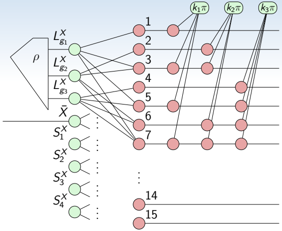
reversed
PTE



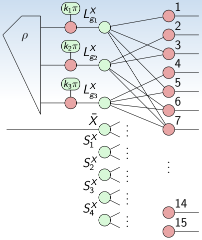
π-copy



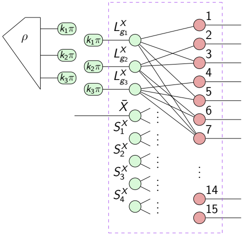
reversed
PTE
π-copy



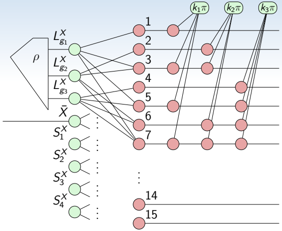
reversed
PTE



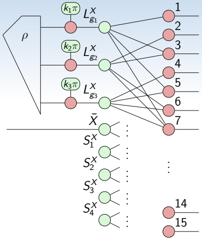
π-copy



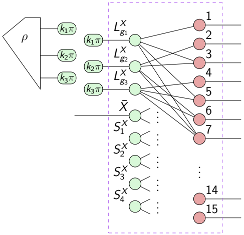
equivalence of
normal forms



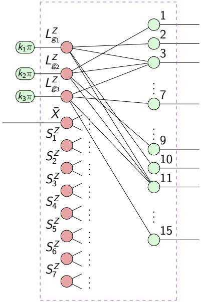
reversed
PTE

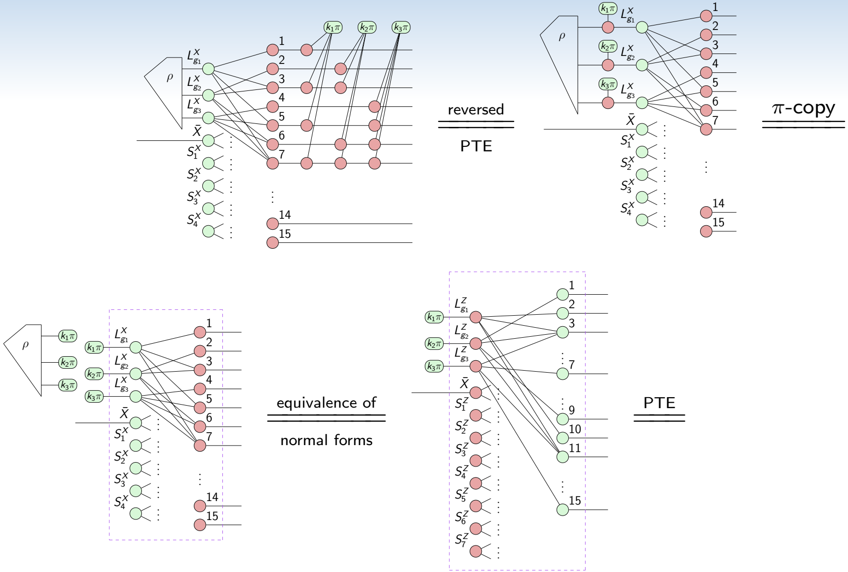


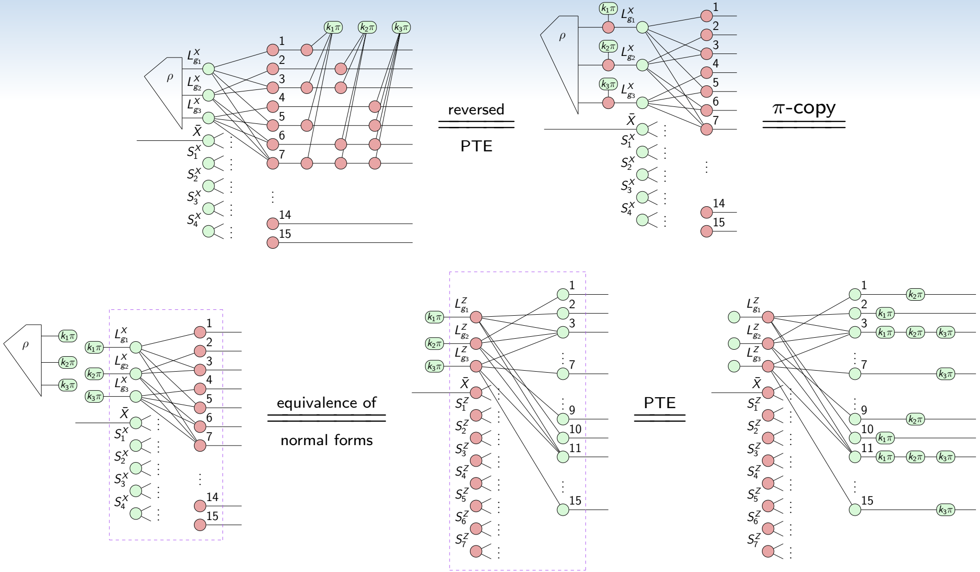
π-copy



equivalence of
normal forms




reversed PTE
π-copy
equivalence of normal forms
PTE

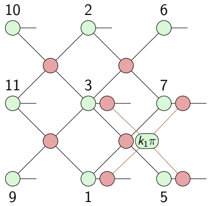

ρ
reversed
PTE
π-copy
equivalence of
normal forms
PTE

Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.

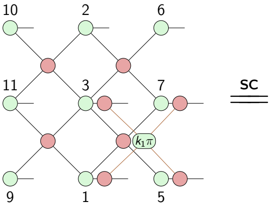
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



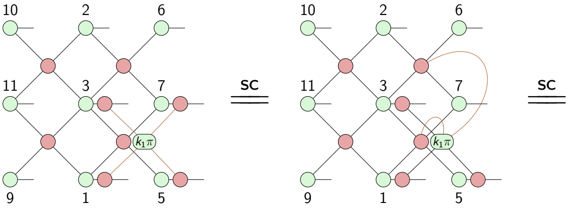
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



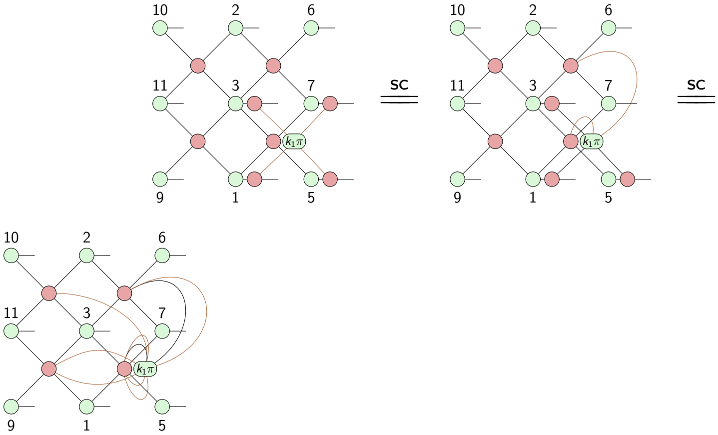
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



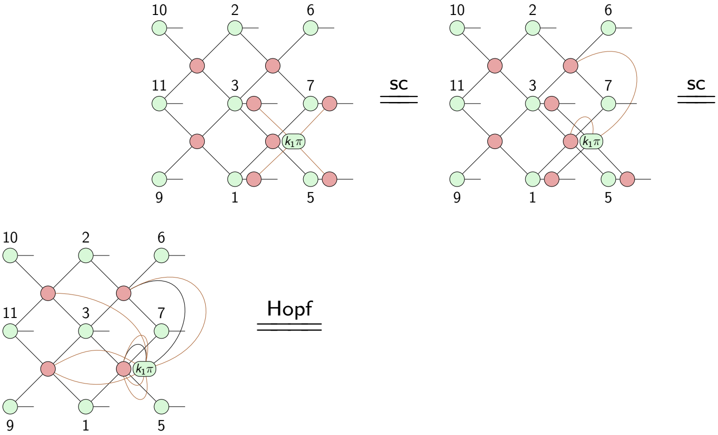
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



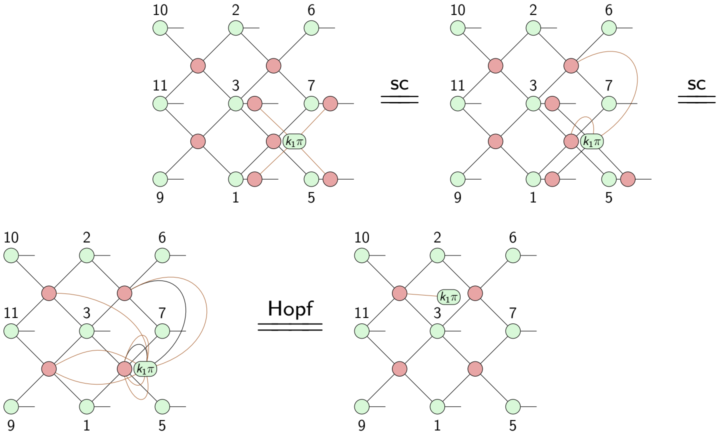
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



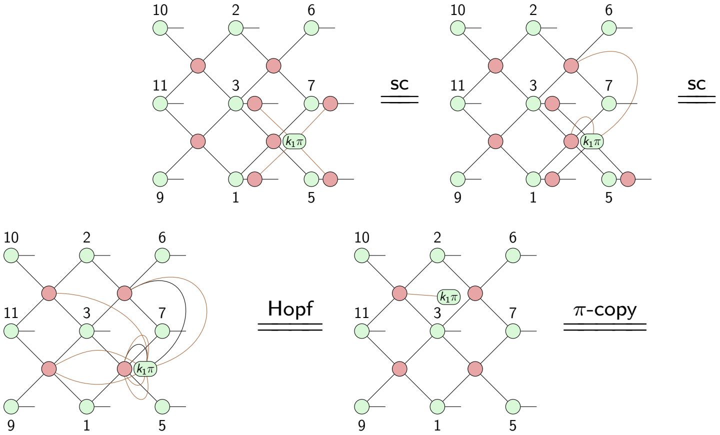
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



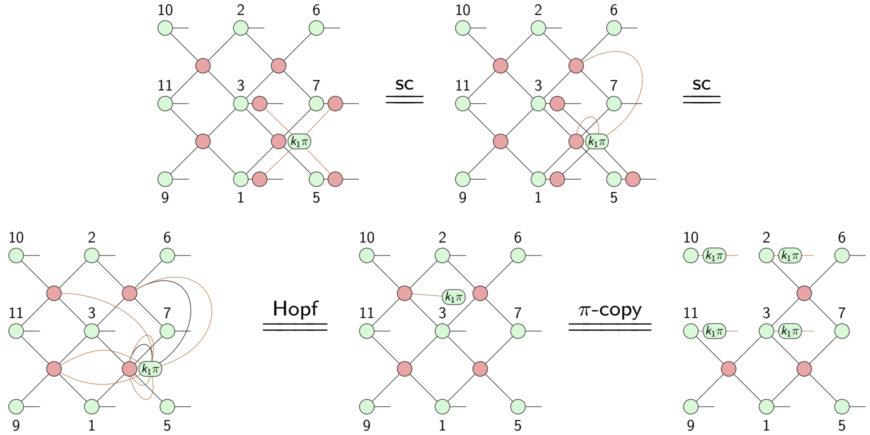
Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



Syndrome-determined Recovery Operation

- Measuring L_g^X adds these operators into the stabilizer group and removes stabilizers L_g^Z. Moreover, the fixing operations be readily read-off from the graphical derivation.



Construct SQRM from QRM(4) & EQRM

- QRM(4): $\mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle$
- EQRM: $\mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X \rangle$
- SQRM: $(G,\ \mathcal{S}_S,\ L_g,\ \overline{L})$, where

$$
\begin{aligned}
G &= \mathcal{S}_4 \cup \mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X, T^Z \rangle \\
\mathcal{S}_S &= \mathcal{S}_4 \cap \mathcal{S}_E = \langle N^X, N^Z, H^Z \rangle \\
L_g &= G \setminus \mathcal{S}_S = \langle H^X, T^Z \rangle \\
\overline{L} &= \langle \overline{X}, \overline{Z} \rangle
\end{aligned}
$$

Construct SQRM from QRM(4) & EQRM

- QRM(4): $\mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle$
- EQRM: $\mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X \rangle$
- SQRM: $(G,\ \mathcal{S}_S,\ L_g,\ \overline{L})$, where

$$\begin{aligned} G &= \mathcal{S}_4 \cup \mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X, T^Z \rangle \\ \mathcal{S}_S &= \mathcal{S}_4 \cap \mathcal{S}_E = \langle N^X, N^Z, H^Z \rangle \\ L_g &= G \setminus \mathcal{S}_S = \langle H^X, T^Z \rangle \\ \overline{L} &= \langle \overline{X}, \overline{Z} \rangle \end{aligned}$$

Construct SQRM from QRM(4) & EQRM

- QRM(4): $\mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle$
- EQRM: $\mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X \rangle$
- SQRM: $(G,\ \mathcal{S}_S,\ L_g,\ \overline{L})$, where

$$
\begin{aligned}
G &= \mathcal{S}_4 \cup \mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X, T^Z \rangle \\
\mathcal{S}_S &= \mathcal{S}_4 \cap \mathcal{S}_E = \langle N^X, N^Z, H^Z \rangle \\
L_g &= G \setminus \mathcal{S}_S = \langle T^Z, H^X \rangle \\
\overline{L} &= \langle \overline{X}, \overline{Z} \rangle
\end{aligned}
$$

Construct SQRM from QRM(4) & EQRM

- QRM(4): $\mathcal{S}_4 = \langle N^X, N^Z, H^Z, T^Z \rangle$
- EQRM: $\mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X \rangle$
- SQRM: $(G,\ \mathcal{S}_S,\ L_g,\ \overline{L})$, where

$$\begin{aligned} G &= \mathcal{S}_4 \cup \mathcal{S}_E = \langle N^X, N^Z, H^Z, H^X, T^Z \rangle \\ \mathcal{S}_S &= \mathcal{S}_4 \cap \mathcal{S}_E = \langle N^X, N^Z, H^Z \rangle \\ L_g &= G \setminus \mathcal{S}_S = \langle T^Z, H^X \rangle \\ \overline{L} &= \langle \overline{\mathbf{X}}, \overline{\mathbf{Z}} \rangle \end{aligned}$$

Open Problems

Through the lens of ZX calculus, we will

- **Present** CSS code deformations.

[9] Vuillot, C., Lao, L., Criger, B., Almudéver, C. G., Bertels, K., & Terhal, B. M. (2019). Code deformation and lattice surgery are gauge fixing. New Journal of Physics, 21(3), 033028.

[10] Knill, E., & Laflamme, R. (1996). Concatenated quantum codes. arXiv preprint quant-ph/9608012.

[11] Vasmer, M., & Kubica, A. (2022). Morphing quantum codes. PRX Quantum, 3(3), 030319.

[12] Bombin, H., Litinski, D., Nickerson, N., Pastawski, F., & Roberts, S. (2023). Unifying flavors of fault tolerance with the ZX calculus. arXiv preprint arXiv:2303.08829.

[13] Khesin, A. B., Lu, J. Z., & Shor, P. W. (2023). Graphical quantum Clifford-encoder compilers from the ZX calculus. arXiv preprint arXiv:2301.02356.

Open Problems

Through the lens of ZX calculus, we will

- **Present** CSS code deformations.
- **Understand** code concatenation.

---

[9] Vuillot, C., Lao, L., Criger, B., Almudéver, C. G., Bertels, K., & Terhal, B. M. (2019). Code deformation and lattice surgery are gauge fixing. New Journal of Physics, 21(3), 033028.

[10] Knill, E., & Laflamme, R. (1996). Concatenated quantum codes. arXiv preprint quant-ph/9608012.

[11] Vasmer, M., & Kubica, A. (2022). Morphing quantum codes. PRX Quantum, 3(3), 030319.

[12] Bombin, H., Litinski, D., Nickerson, N., Pastawski, F., & Roberts, S. (2023). Unifying flavors of fault tolerance with the ZX calculus. arXiv preprint arXiv:2303.08829.

[13] Khesin, A. B., Lu, J. Z., & Shor, P. W. (2023). Graphical quantum Clifford-encoder compilers from the ZX calculus. arXiv preprint arXiv:2301.02356.

Open Problems

Through the lens of ZX calculus, we will

- **Present** CSS code deformations.
- **Understand** code concatenation.
- **Derive** new good QECCs from the existing QECCs.
- **Unify** fault-tolerant protocols for stabilizer codes.

---

[9] Vuillot, C., Lao, L., Criger, B., Almudéver, C. G., Bertels, K., & Terhal, B. M. (2019). Code deformation and lattice surgery are gauge fixing. New Journal of Physics, 21(3), 033028.

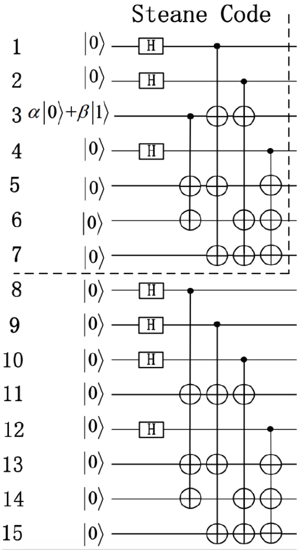
[10] Knill, E., & Laflamme, R. (1996). Concatenated quantum codes. arXiv preprint quant-ph/9608012.

[11] Vasmer, M., & Kubica, A. (2022). Morphing quantum codes. PRX Quantum, 3(3), 030319.

[12] Bombin, H., Litinski, D., Nickerson, N., Pastawski, F., & Roberts, S. (2023). Unifying flavors of fault tolerance with the ZX calculus. arXiv preprint arXiv:2303.08829.

[13] Khesin, A. B., Lu, J. Z., & Shor, P. W. (2023). Graphical quantum Clifford-encoder compilers from the ZX calculus. arXiv preprint arXiv:2301.02356.

Thank you!



$|\bar{\psi}\rangle = \alpha|\bar{0}\rangle_3 + \beta|\bar{1}\rangle_3$

$\mathbb{S}_3$ stabilizes $|\bar{\psi}\rangle$

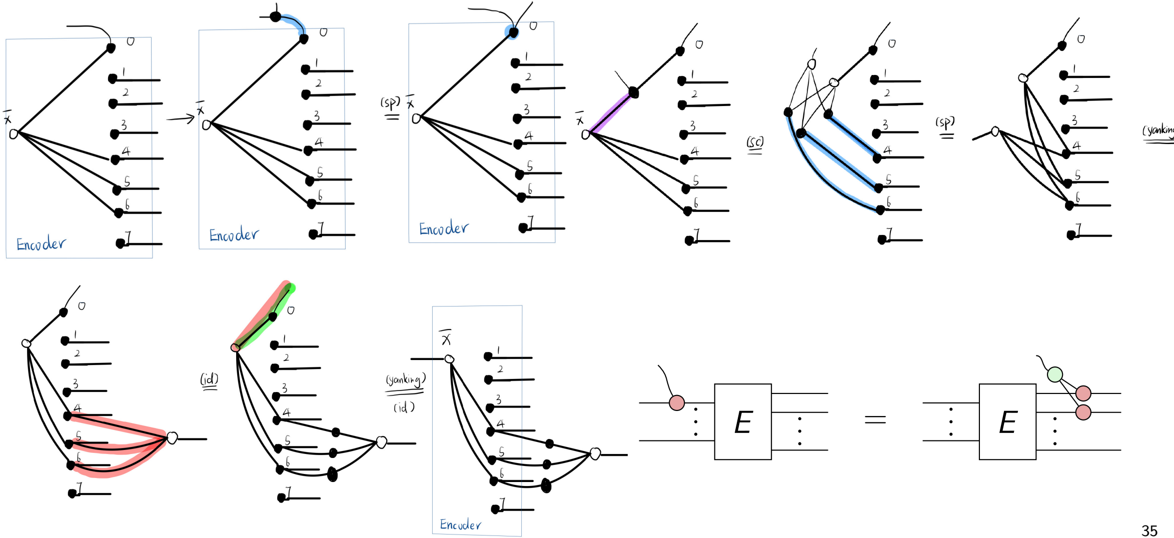
$|\phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle|\bar{0}\rangle_3 + |1\rangle|\bar{1}\rangle_3)$

No information is stored on the last eight qubits.

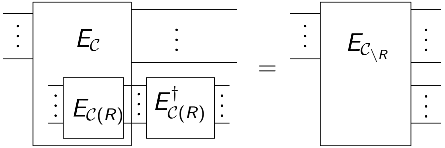
$|\bar{\Phi}\rangle = |\bar{\psi}\rangle|\phi\rangle$

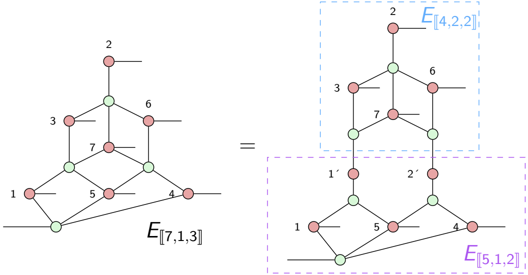
$\mathbb{S}_E$ stabilizes $|\bar{\Phi}\rangle$

Example: $\overline{X} = X_4X_5X_6$



Code Morphing[10]





[10] Michael Vasmer & Aleksander Kubica (2022): Morphing Quantum Codes. PRX Quantum 3(3), doi:10.1103/prxquantum.3.030319