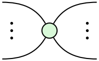
Quantum software:
Phase-free ZX diagrams and CSS codes

Aleks Kissinger

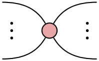
February 16, 2024

Phase-free ZX-diagrams

...are made of spiders with $\alpha = 0$:



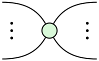
$$:= \quad |0...0\rangle\langle 0...0| + |1...1\rangle\langle 1...1|$$



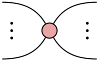
$$:= \quad |+...+\rangle\langle +...+| + |-...-\rangle\langle -...-|$$

Phase-free ZX-diagrams

...are made of spiders with $\alpha = 0$:



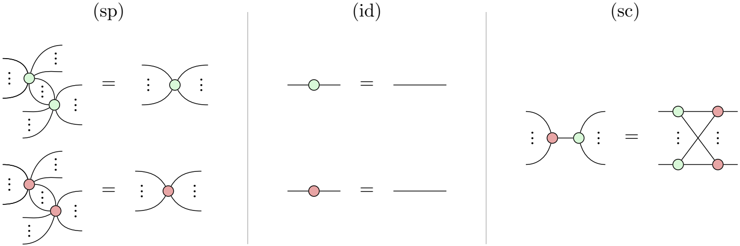
$$:= \quad |0...0\rangle\langle 0...0| + |1...1\rangle\langle 1...1|$$



$$:= \quad |+...+\rangle\langle +...+| + |-...-\rangle\langle -...-|$$

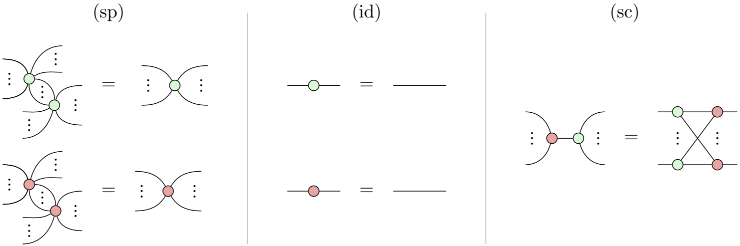
$$= \quad N \sum_{\oplus_i b_i = 0} |b_1 ... b_n\rangle\langle b_{n+1} ... b_{n+m}|$$

Phase-free ZX-calculus



Simplification

1. Apply (sp) and (id) as much as possible.
2. Apply (sc) where
   - ○ is **not** an input and
   - ● is **not** an output.
3. Repeat as long as step 2 applies.



Simplification

1. Apply (sp) and (id) as much as possible.
2. Apply (sc) where
   - ○ is **not** an input and
   - ● is **not** an output.
3. Repeat as long as step 2 applies.

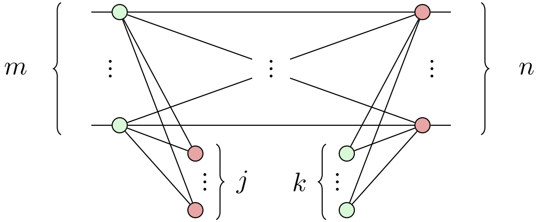
Each iteration **strictly** decreases:

$$(\#\ \text{non-input } \circ\text{'s}) + (\#\ \text{non-output } \bullet\text{'s})$$

Simplification

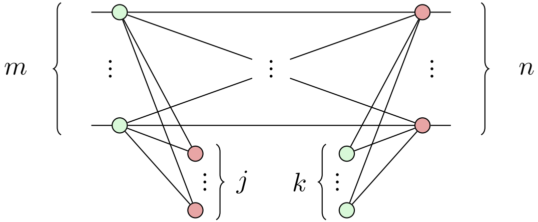
1. Apply (sp) and (id) as much as possible.
2. Apply (sc) where
   - ○ is **not** an input and
   - ● is **not** an output.
3. Repeat as long as step 2 applies.

Terminates with:



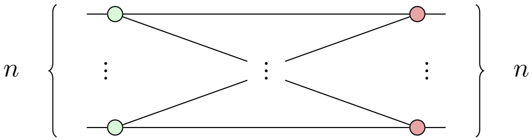
Unitaries

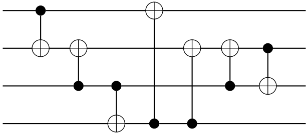
$$\text{Unitary} \quad \Longrightarrow \quad m = n, \quad j = k = 0$$

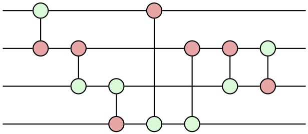


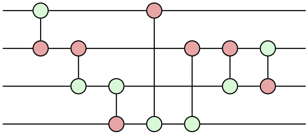
Unitaries

$$\text{Unitary} \quad \Longrightarrow \quad m = n, \quad j = k = 0$$

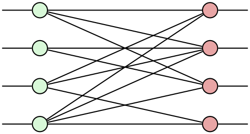


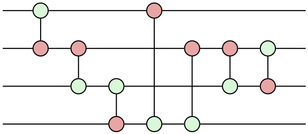




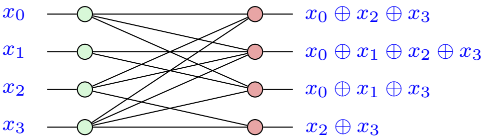


$\downarrow^*$



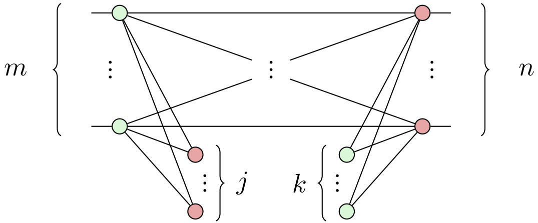


$\downarrow^*$


x_0
x_1
x_2
x_3
$x_0 \oplus x_2 \oplus x_3$
$x_0 \oplus x_1 \oplus x_2 \oplus x_3$
$x_0 \oplus x_1 \oplus x_3$
$x_2 \oplus x_3$

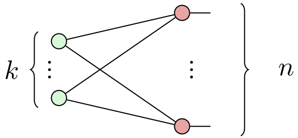
States

$$\text{State} \quad \Longrightarrow \quad m = 0, \quad j = 0$$



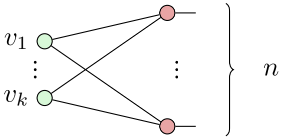
States

$$\text{State} \quad \Longrightarrow \quad m = 0, \quad j = 0$$



States

$$\text{State} \quad \Longrightarrow \quad m = 0, \quad j = 0$$

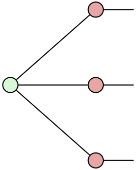


$$|\psi\rangle = \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \mathrm{span}\{v_1, \ldots, v_k\} \subseteq \mathbb{F}_2^n$$

$$|\mathrm{GHZ}\rangle = |000\rangle + |111\rangle$$

$$
\begin{aligned}
|\mathrm{GHZ}\rangle &= |000\rangle + |111\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\}
\end{aligned}
$$

$$
\begin{aligned}
|\mathrm{GHZ}\rangle &= |000\rangle + |111\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\}
\end{aligned}
$$

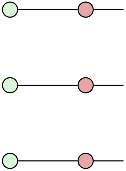
$=$ 

$$|{+}{+}{+}\rangle \quad = \quad \sum_{v \in \mathbb{F}_2^3} |v\rangle$$

$$
\begin{aligned}
|{+}{+}{+}\rangle &= \sum_{v \in \mathbb{F}_2^3} |v\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \operatorname{span}\left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\end{pmatrix} \right\}
\end{aligned}
$$

$$|{+}{+}{+}\rangle = \sum_{v \in \mathbb{F}_2^3} |v\rangle$$

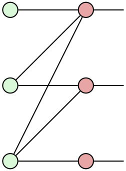
$$= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \operatorname{span}\left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\end{pmatrix} \right\}$$

$=$ 

$$|{+}{+}{+}\rangle = \sum_{v \in \mathbb{F}_2^3} |v\rangle$$

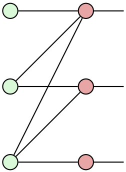
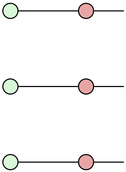
$$
\begin{aligned}
|{+}{+}{+}\rangle &= \sum_{v \in \mathbb{F}_2^3} |v\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \operatorname{span}\left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\end{pmatrix}, \begin{pmatrix}1\\1\\1\end{pmatrix} \right\}
\end{aligned}
$$

$$
\begin{aligned}
|{+}{+}{+}\rangle &= \sum_{v \in \mathbb{F}_2^3} |v\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \mathrm{span}\left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\end{pmatrix}, \begin{pmatrix}1\\1\\1\end{pmatrix} \right\} \\
&=
\end{aligned}
$$



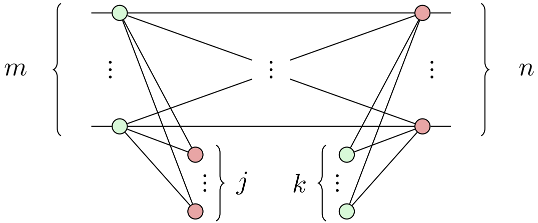
$$|{+}{+}{+}\rangle = \sum_{v \in \mathbb{F}_2^3} |v\rangle$$

$$= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \text{span}\left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\end{pmatrix}, \begin{pmatrix}1\\1\\1\end{pmatrix} \right\}$$

=  = 

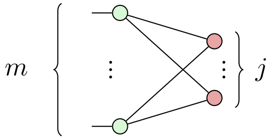
Effects

$$\text{Effect} \implies n = 0, \quad k = 0$$



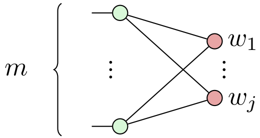
Effects

$$\text{Effect} \quad \Longrightarrow \quad n = 0, \quad k = 0$$



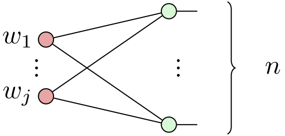
Effects

$$\text{Effect} \implies n = 0, \quad k = 0$$



$$\langle \phi | = \sum_{v \in S} \langle v | \quad \text{where} \quad S^{\perp} = \mathrm{span}\{w_1, \ldots, w_j\} \subseteq \mathbb{F}_2^m$$

Or a second way to write states...



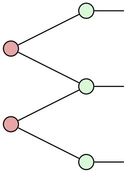
$$|\psi\rangle = \sum_{v \in S} |v\rangle \quad \text{where} \quad S^{\perp} = \mathrm{span}\{w_1, \ldots, w_j\} \subseteq \mathbb{F}_2^n$$

$$|\mathrm{GHZ}\rangle = |000\rangle + |111\rangle$$

$$
\begin{aligned}
|\mathrm{GHZ}\rangle &= |000\rangle + |111\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \{v \mid v_1 \oplus v_2 = 0, v_2 \oplus v_3 = 0\}
\end{aligned}
$$

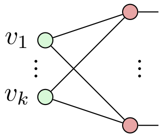
$$
\begin{aligned}
|\mathrm{GHZ}\rangle &= |000\rangle + |111\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \{v \mid v_1 \oplus v_2 = 0, v_2 \oplus v_3 = 0\} \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S^{\perp} = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\}
\end{aligned}
$$

$$
\begin{aligned}
|\mathrm{GHZ}\rangle &= |000\rangle + |111\rangle \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \{v \mid v_1 \oplus v_2 = 0, v_2 \oplus v_3 = 0\} \\
&= \sum_{v \in S} |v\rangle \quad \text{where} \quad S^{\perp} = \mathrm{span}\left\{ \begin{pmatrix}1\\1\\0\end{pmatrix}, \begin{pmatrix}0\\1\\1\end{pmatrix} \right\} \\
&=
\end{aligned}
$$

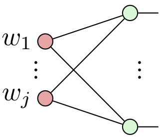


Theorem

A state represented by a phase-free ZX-diagram is uniquely fixed by a subspace $S \subseteq \mathbb{F}_2^n$ *(or equivalently* $S^\perp \subseteq F_2^n$*).*



$$= \sum_{v \in S} |v\rangle \quad \text{where} \quad S = \mathrm{span}\{v_1, \ldots, v_k\}$$



$$= \sum_{v \in S} |v\rangle \quad \text{where} \quad S^\perp = \mathrm{span}\{w_1, \ldots, w_j\}$$

Stabiliser Theory

Theorem (FTST)

If $\mathcal{S}$ has k generators, then $\mathrm{Stab}(\mathcal{S})$ is a 2^{n-k} dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$.

Stabiliser Theory

Theorem (FTST)

If $\mathcal{S}$ has k generators, then $\mathrm{Stab}(\mathcal{S})$ is a 2^{n-k} dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$.

$$k = n \qquad \Longrightarrow \qquad \mathrm{Stab}(\mathcal{S}) = \{\lambda|\psi\rangle \mid \lambda \in \mathbb{C}\}$$

maximal *1D subspace*

CSS codes

Definition

For $S \subseteq \mathbb{F}_2^n$, $T \subseteq S^\perp$, a **CSS code** is a stabiliser group with generators:

$$\vec{X}_i := \bigotimes_{q=1}^{\dim S} X^{(v_i)_q} \qquad \vec{Z}_j := \bigotimes_{q=1}^{\dim T} Z^{(w_j)_q}$$

where $S = \mathrm{span}\{v_i\}$ and $T = \mathrm{span}\{w_i\}$.

A CSS code is maximal iff $T = S^{\perp}$, i.e. it has generators:

$$\vec{X}_i := X^{(v_i)_1} \otimes \ldots \otimes X^{(v_i)_n} \qquad \vec{Z}_j := Z^{(w_j)_1} \otimes \ldots \otimes Z^{(w_j)_n}$$

where $S = \mathrm{span}\{v_i\}$ and $S^{\perp} = \mathrm{span}\{w_j\}$.

Example

The stabiliser group of $|\mathrm{GHZ}\rangle$ is generated by:

$$X \otimes X \otimes X \qquad Z \otimes Z \otimes I \qquad I \otimes Z \otimes Z$$

Example

The stabiliser group of $|\mathrm{GHZ}\rangle$ is generated by:

$$X \otimes X \otimes X \qquad Z \otimes Z \otimes I \qquad I \otimes Z \otimes Z$$

This is a maximal CSS code, where:

$$S = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\} \qquad S^{\perp} = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\}$$

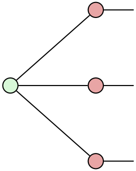
Example

The stabiliser group of $|\mathrm{GHZ}\rangle$ is generated by:

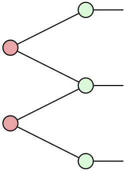
$$X \otimes X \otimes X \qquad Z \otimes Z \otimes I \qquad I \otimes Z \otimes Z$$

This is a maximal CSS code, where:

$$S = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\} \qquad S^{\perp} = \mathrm{span}\left\{ \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\}$$



$=$

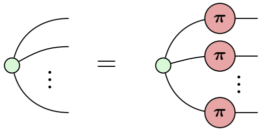


Theorem

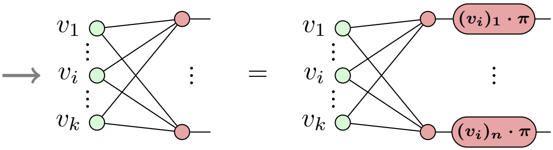
The ZX-diagram associated with $S \subseteq \mathbb{F}_2^n$ is the unique stabiliser state of the maximal CSS code defined by $(S, S^\perp)$.

Proof

Using:

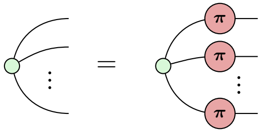


compute the X-stabilisers by “firing” each basis vector of S:

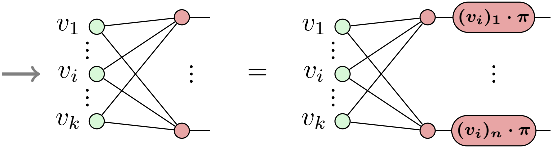


Proof

Using:



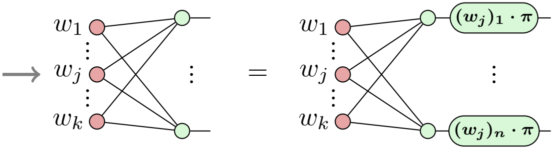
compute the X-stabilisers by "firing" each basis vector of S:



$$|\psi\rangle = (X^{(v_i)_1} \otimes \ldots \otimes X^{(v_i)_n})|\psi\rangle$$

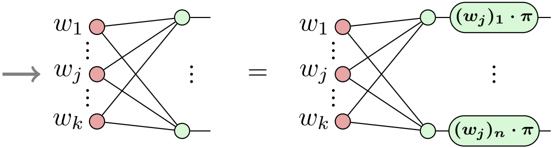
Proof (cont'd)

Similarly, compute the Z-stabilisers from $S^{\perp}$:



Proof (cont'd)

Similarly, compute the Z-stabilisers from $S^\perp$:



$$|\psi\rangle = (Z^{(w_j)_1} \otimes \ldots \otimes Z^{(w_j)_n})|\psi\rangle$$

This gives $\dim S + \dim S^\perp = n$ generators for n qubits, so $|\psi\rangle$ uniquely fixed by FTST. □

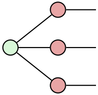
Corollary

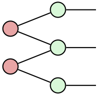
We can translate a maximal CSS code directly into a ZX-diagram in 2 ways.

Corollary

We can translate a maximal CSS code directly into a ZX-diagram in 2 ways.

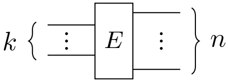
For example, $\{X \otimes X \otimes X, Z \otimes Z \otimes I, I \otimes Z \otimes Z\}$ gives:

X-representation: $\{X \otimes X \otimes X\}$ $\rightsquigarrow$ 

Z-representation: $\{Z \otimes Z \otimes I, I \otimes Z \otimes Z\}$ $\rightsquigarrow$ 

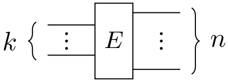
Quantum error correction

...is done by encoding some **logical** qubits into a bigger space of **physical** qubits:

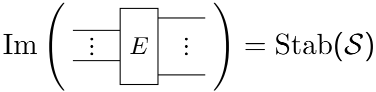


Quantum error correction

...is done by encoding some **logical** qubits into a bigger space of **physical** qubits:



E defines a **stabiliser code** when:



where $\mathcal{S}$ is a stabiliser group with $n - k$ generators.

Quantum error correction

We can detect errors without destroying the state by measuring stabilisers in $\mathcal{S}$.

For CSS codes, 2 kinds of stabiliser measurements are relevant:

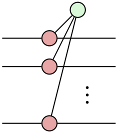
$$\mathcal{M}_{X\dots X} := \{\Pi^{(0)}_{X\dots X}, \Pi^{(1)}_{X\dots X}\}$$

$$\mathcal{M}_{Z\dots Z} := \{\Pi^{(0)}_{Z\dots Z}, \Pi^{(1)}_{Z\dots Z}\}$$

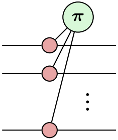
X measurements

$$\mathcal{M}_{X\ldots X} = \left\{ \Pi^{(k)}_{X\ldots X} := \frac{1}{2}(I + (-1)^k X \otimes \ldots \otimes X) \right\}$$

$$\Pi^{(0)}_{X\ldots X} =$$

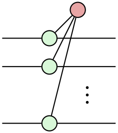


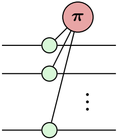
$$\Pi^{(1)}_{X\ldots X} =$$



Z measurements

$$\mathcal{M}_{Z\ldots Z} = \left\{ \Pi^{(k)}_{Z\ldots Z} := \frac{1}{2}(I + (-1)^k Z \otimes \ldots \otimes Z) \right\}$$

$\Pi^{(0)}_{Z\ldots Z} =$ 

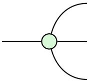
$\Pi^{(1)}_{Z\ldots Z} =$ 

Example

The GHZ code:

$$\mathcal{S} := \{\cancel{X \otimes X \otimes X}, \quad Z \otimes Z \otimes I, \quad I \otimes Z \otimes Z\}$$

Then:

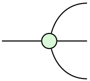
$$\mathrm{Im}\left(\right.$$

$$\left. \right) = \mathrm{span}\{|000\rangle, |111\rangle\} = \mathrm{Stab}(\mathcal{S})$$

Example

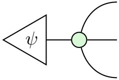
The GHZ code:

$$\mathcal{S} := \{\cancel{X \otimes X \otimes X}, \quad Z \otimes Z \otimes I, \quad I \otimes Z \otimes Z\}$$

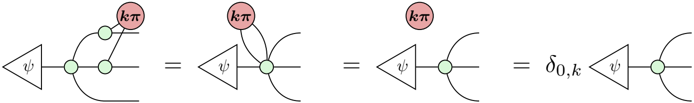
Then:

$\mathrm{Im}\Big($  $\Big) \;=\; \mathrm{span}\{|000\rangle, |111\rangle\} \;=\; \mathrm{Stab}(\mathcal{S})$

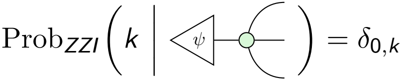
So, we can encode states like this:



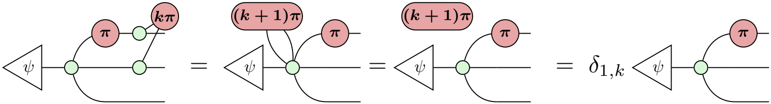
Applying $\Pi^{\pm}_{ZZI}$ to an encoded state:



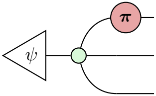
Hence:

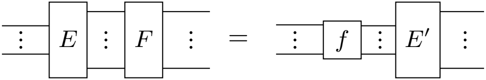


Applying $\Pi^{\pm}_{ZZI}$ to an encoded state with an error:



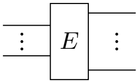
Hence:

$$\mathrm{Prob}_{ZZI}\left(k \;\middle|\; \right.$$  $$\left. \right) = \delta_{1,k}$$


E
F
=
f
E′

Logical operators

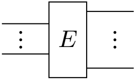
Note:

$\mathrm{Im}\Bigg($  $\Bigg) = \mathrm{Stab}(\mathcal{S})$

only fixes the **image** of E, not E itself.

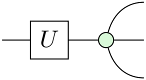
Logical operators

Note:

$\mathrm{Im}\Big($  $\Big) = \mathrm{Stab}(\mathcal{S})$

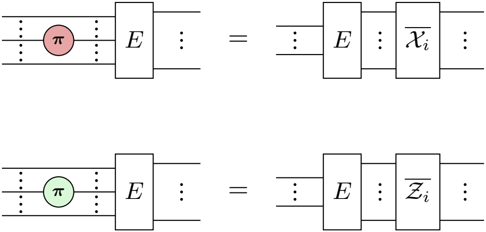
only fixes the **image** of E, not E itself.

For example, the following is also a GHZ encoder:



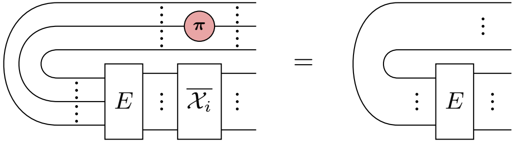
Logical operators

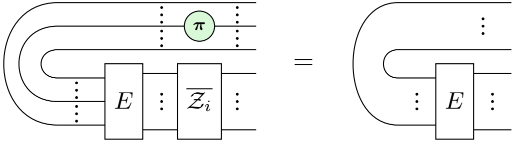
To fix E, we should fix $2k$ more **logical operators** by “pushing” Pauli X and Z ops through the encoder:



Logical operators

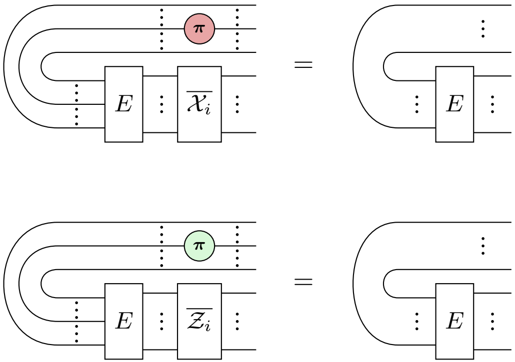
Equivalently, we fix $2k$ more stabilisers for the $n + k$ qubit state $|E\rangle := (I \otimes E)|\cup\rangle$:





Logical operators

Equivalently, we fix $2k$ more stabilisers for the $n+k$ qubit state $|E\rangle := (I \otimes E)|\cup\rangle$:



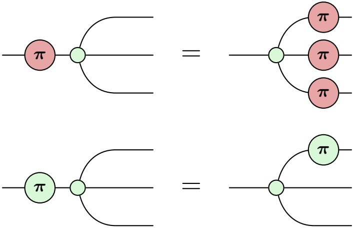
$$(n-k) + 2k = n + k \text{ stabilisers for } |E\rangle$$

Example

The GHZ code has stabiliers and logical operators:

$$\vec{Z}_1 = Z_1 Z_2 \qquad \vec{Z}_2 = Z_2 Z_3$$

$$\vec{\mathcal{X}} = X_1 X_2 X_3 \qquad \vec{\mathcal{Z}} = Z_1$$

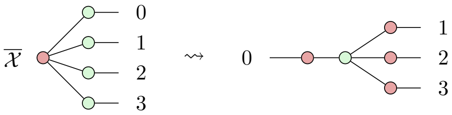


Stabiliers for $|E\rangle$:

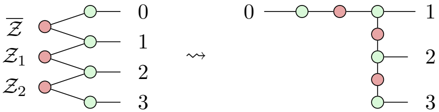
$$\vec{X}_1' = X_1 X_2 X_3 \qquad \vec{Z}_1' = Z_1 Z_2 \qquad \vec{Z}_2' = Z_2 Z_3$$

$$\vec{\mathcal{X}}' = X_0 X_1 X_2 X_3 \qquad \vec{\mathcal{Z}}' = Z_0 Z_1$$

X-representation:



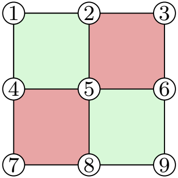
Z-representation:



The surface code

The surface code

...is a 2D lattice of $d \times e$ qubits:



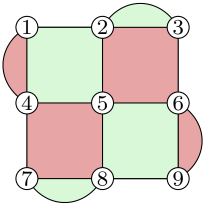
$\rightsquigarrow$

$$\vec{X}_1 := X_2X_3X_5X_6 \quad \vec{X}_2 := X_4X_5X_7X_8$$
$$\vec{Z}_1 := Z_1Z_2Z_4Z_5 \quad \vec{Z}_2 := Z_5Z_6Z_8Z_9$$

$(d-1)(e-1)$ stabilisers

The surface code

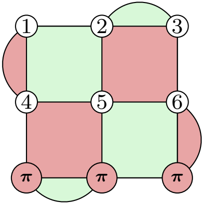
...is a 2D lattice of $d \times e$ qubits:



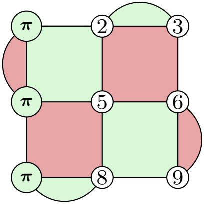
$\rightsquigarrow$

$$\vec{X}_1 := X_2X_3X_5X_6 \quad \vec{X}_2 := X_4X_5X_7X_8$$
$$\vec{X}_3 := X_1X_4 \quad \vec{X}_4 := X_6X_9$$
$$\vec{Z}_1 := Z_1Z_2Z_4Z_5 \quad \vec{Z}_2 := Z_5Z_6Z_8Z_9$$
$$\vec{Z}_3 := Z_2Z_3 \quad \vec{Z}_4 := Z_7Z_8$$

$(d-1)(e-1) + d - 1 + e - 1 = de - 1$ stabilisers

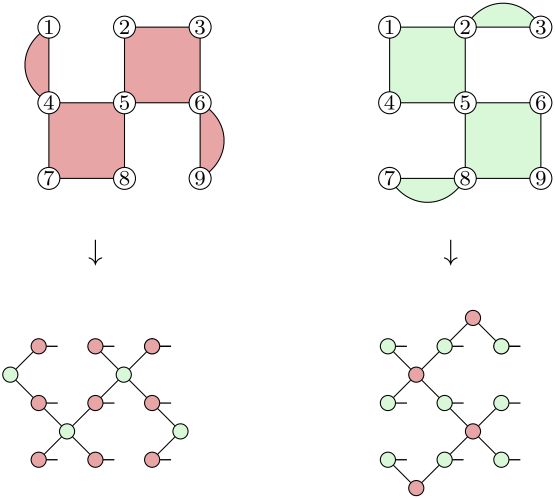


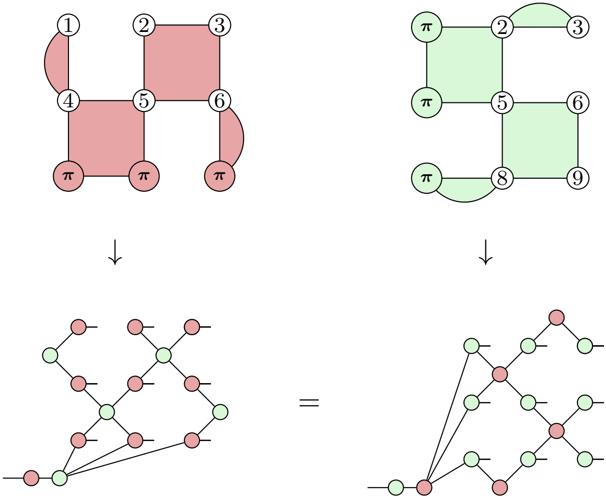
$\rightsquigarrow \quad \vec{\mathcal{X}} := X_7 X_8 X_9$



$\rightsquigarrow \quad \vec{\mathcal{Z}} := Z_1 Z_4 Z_7$

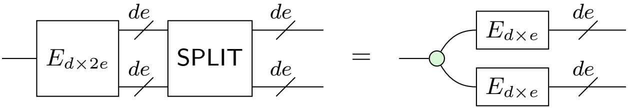
2 logical operators

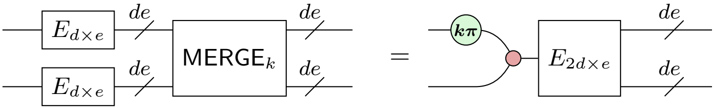

1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
7
8
9


1 2 3
4 5 6
π π π
π 2 3
π 5 6
π 8 9
=

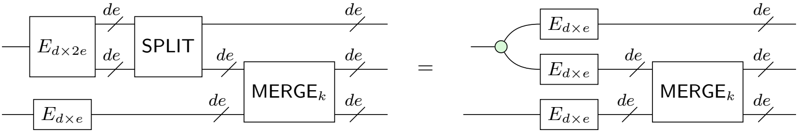
Lattice surgery

In the surface code, we can implement physical operations that behave like **SPLIT** and **MERGE** on logical qubits:

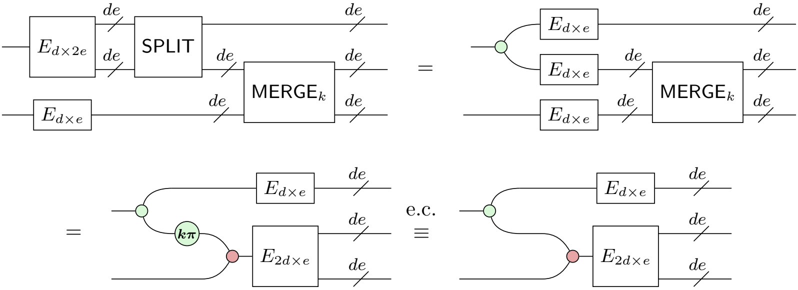




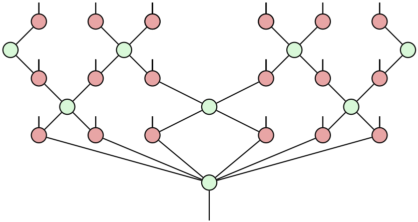
This lets us do entangling operations, e.g. CNOT:



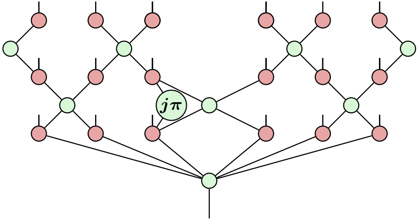
This lets us do entangling operations, e.g. CNOT:



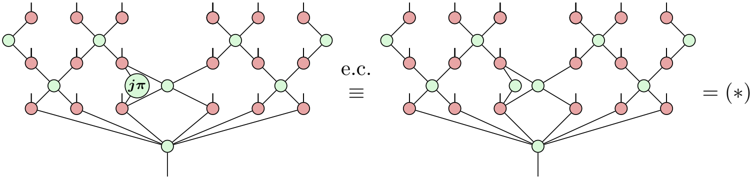
Split



Split

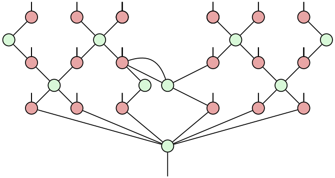


Split

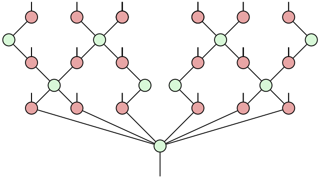


Split

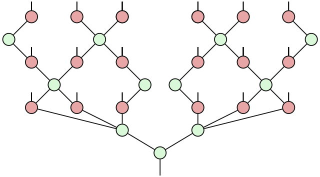
$(*) =$



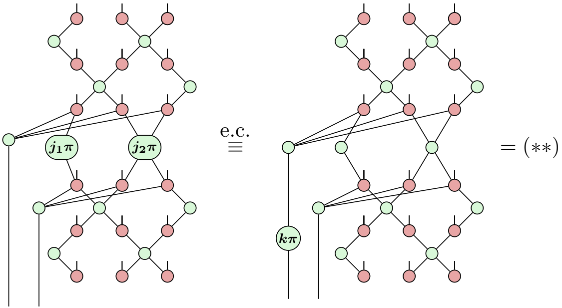
$=$



$=$

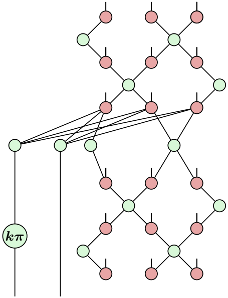
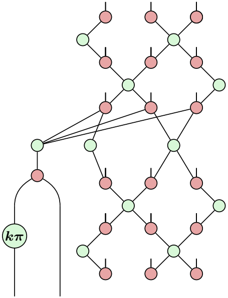


Merge



$$k := j_1 \oplus j_2$$

Merge

$(**) =$  $=$ 

Final notes

Final notes

This all used the X-representation. Flip to the Z-representation to get the colour-reversed split and merge.

Final notes

This all used the X-representation. Flip to the Z-representation to get the colour-reversed split and merge.

Similar tricks implement:

- Entangled measurements
- Magic state injection
- $\implies$ universal FTQC

Final notes

This all used the X-representation. Flip to the Z-representation to get the colour-reversed split and merge.

Similar tricks implement:

- Entangled measurements
- Magic state injection
- $\implies$ universal FTQC

Other CSS codes like colour codes translate to ZX very similarly. L.S. should pretty much work the same way.