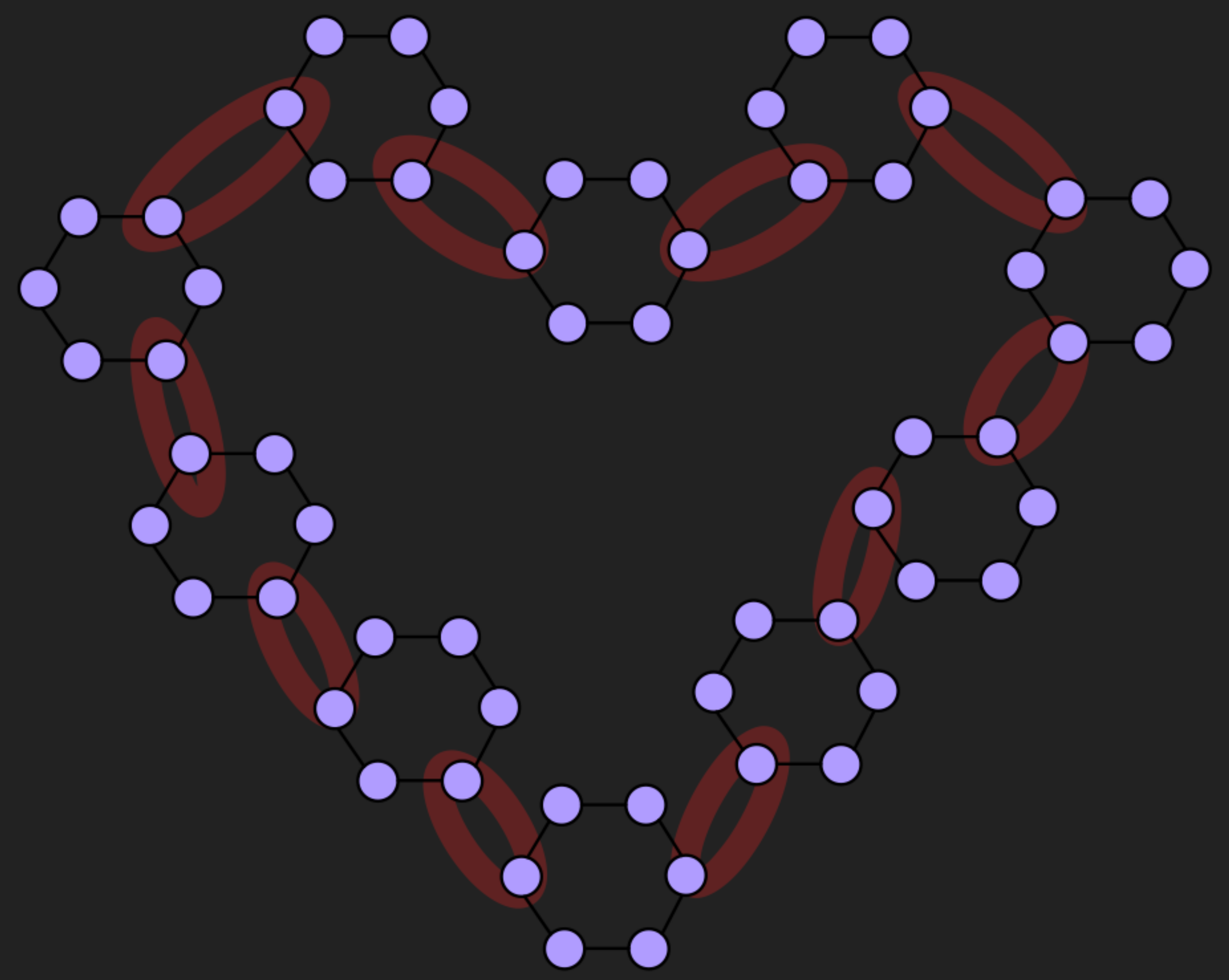


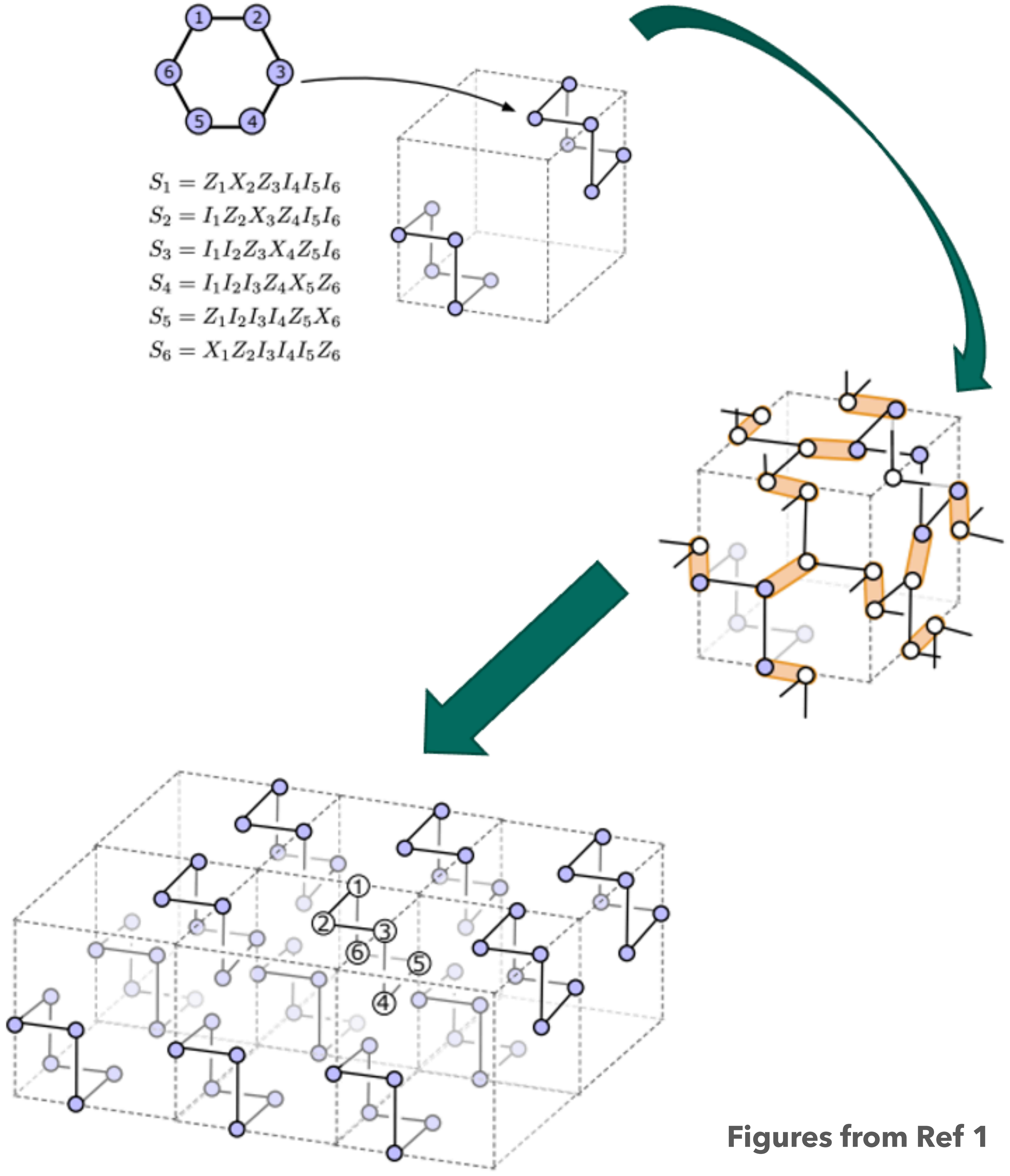
A BRIEF INTRODUCTION

FUSION-BASED QUANTUM COMPUTING

IEVA ČEPAITĖ
14/02/2022

OVERVIEW OF FBQC

- Fusion-based quantum computing (FBQC) is a recently developed QC scheme
- Particularly well-suited to photonics/linear optics QC
- Information propagated through fusion measurements of resource states and processing described via fusion networks
- FBQC is fault-tolerant: can tolerate/mitigate errors



Figures from Ref 1

REFERENCES

1. Fusion-based Quantum Computation, S. Bartolucci et al., 2021
2. Naomi Nickerson's wonderful talk on the topic: Architectures for fault tolerant quantum computing | QIP2021 Tutorial | Naomi Nickerson – YouTube
3. Interleaving: Modular architectures for fault-tolerant photonic quantum computing, H. Bombin et al., 2021
4. Resource-efficient linear optical quantum computation, D. E. Browne and T. Rudolph, 2005
5. Creation of Entangled Photonic States Using Linear Optics, S. Bartolucci et al., 2021
6. Logical blocks for fault-tolerant topological quantum computation, S. Roberts et al., 2021
7. Multi-party entanglement in graph states, M. Hein et al., 2005
8. Surface codes: Towards practical large-scale quantum computation, A. G. Fowler et al., 2012

FBQC PRIMITIVES (BUILDING BLOCKS)

LINEAR OPTICAL QUANTUM COMPUTING (LOQC)

- FBQC is well-suited to the LOQC (photonics) framework
- Photons are dynamical systems and measurements are destructive and non-deterministic
- While photons are short-lived and don't have many Pauli errors, they suffer from loss/erasure
- Qubits can be encoded in different ways – focus on single-photon (dual-rail) encoding

FUSION MEASUREMENTS

- Fusions act as entangling gates in photonics set-ups
- Inherently probabilistic and destructive
- There are two types:

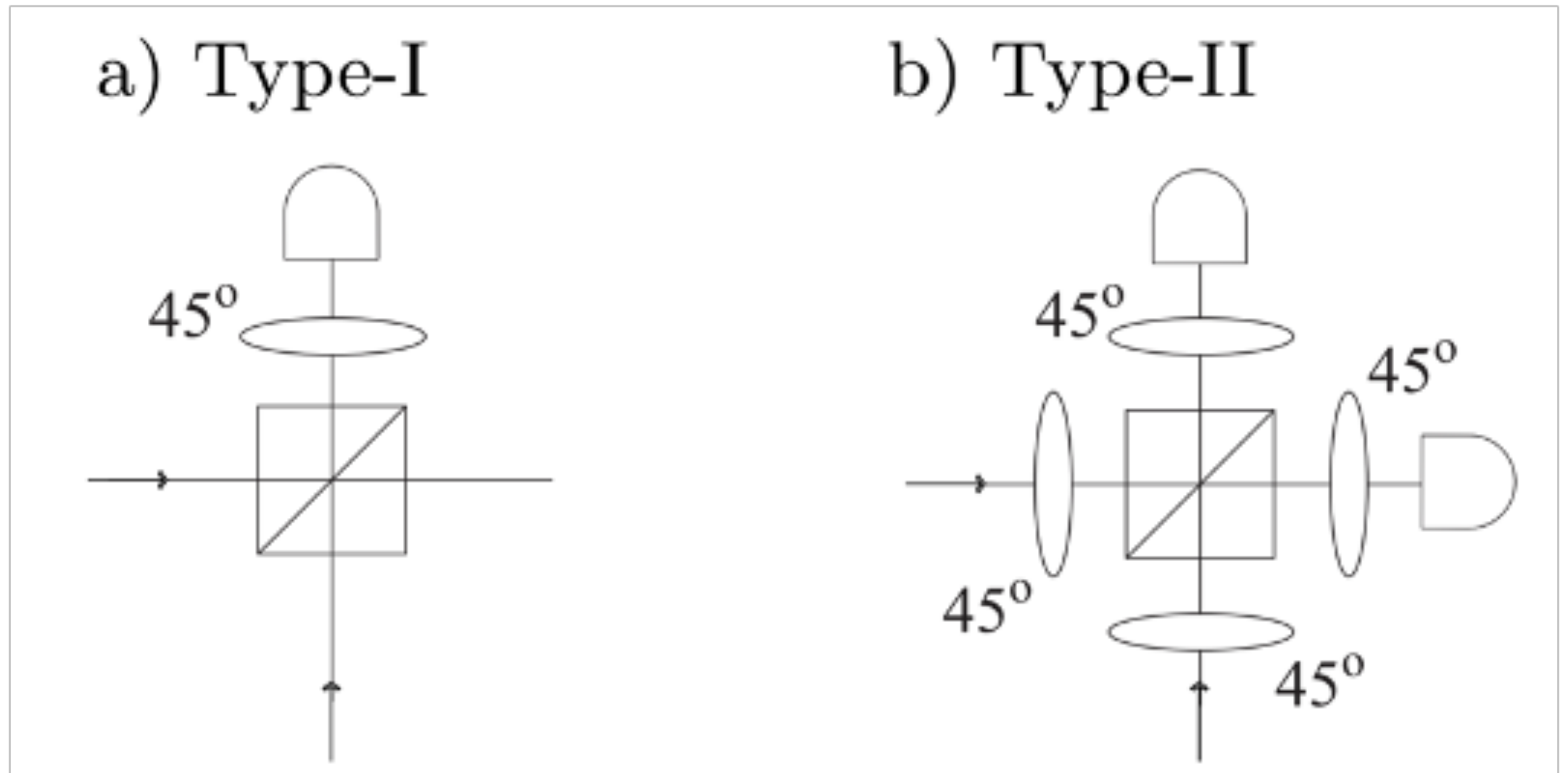


Figure from Ref 4

FUSION MEASUREMENTS

- Fusions act as entangling gates in photonics set-ups
- Inherently probabilistic and destructive
- There are two types:
  - Type I fusion takes 2 photons as input, but only one photon is measured, and hence one photon 'survives'

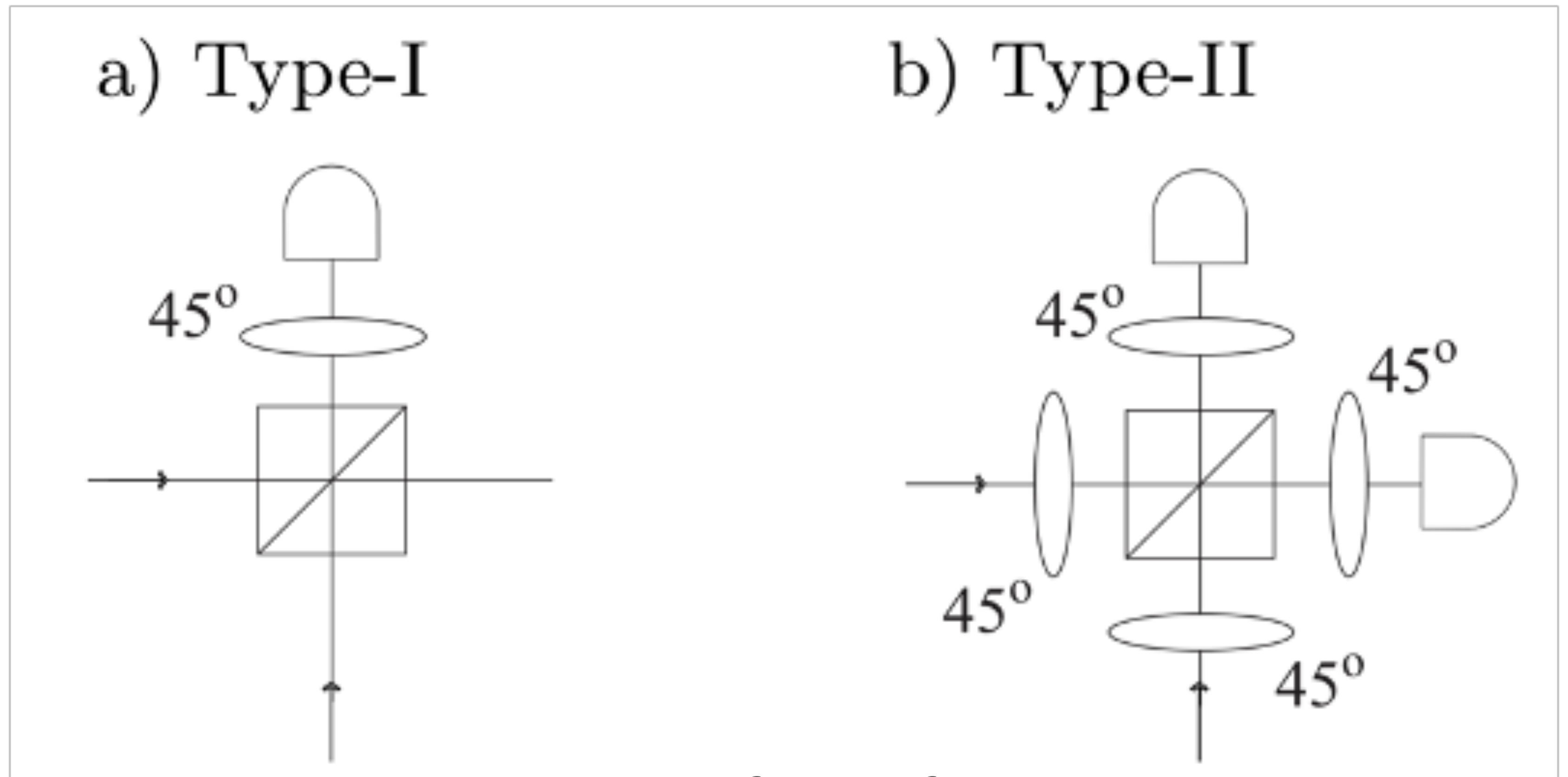


Figure from Ref 4

FUSION MEASUREMENTS

- Fusions act as entangling gates in photonics set-ups
- Inherently probabilistic and destructive
- There are two types:
  - Type I fusion takes 2 photons as input, but only one photon is measured, and hence one photon 'survives'
  - Type II fusion measures - and destroys - both photons. It is the basis of FBQC.

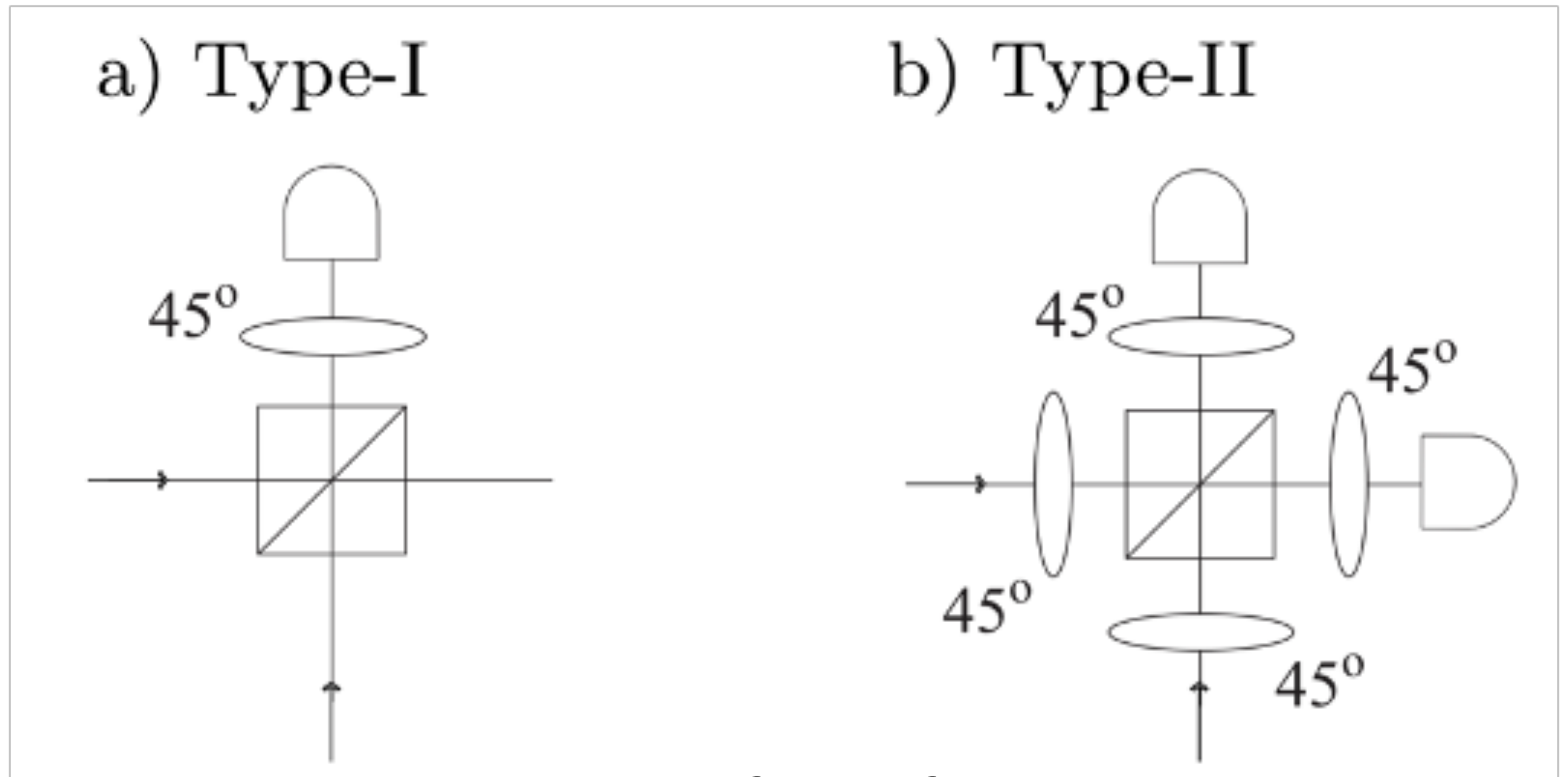
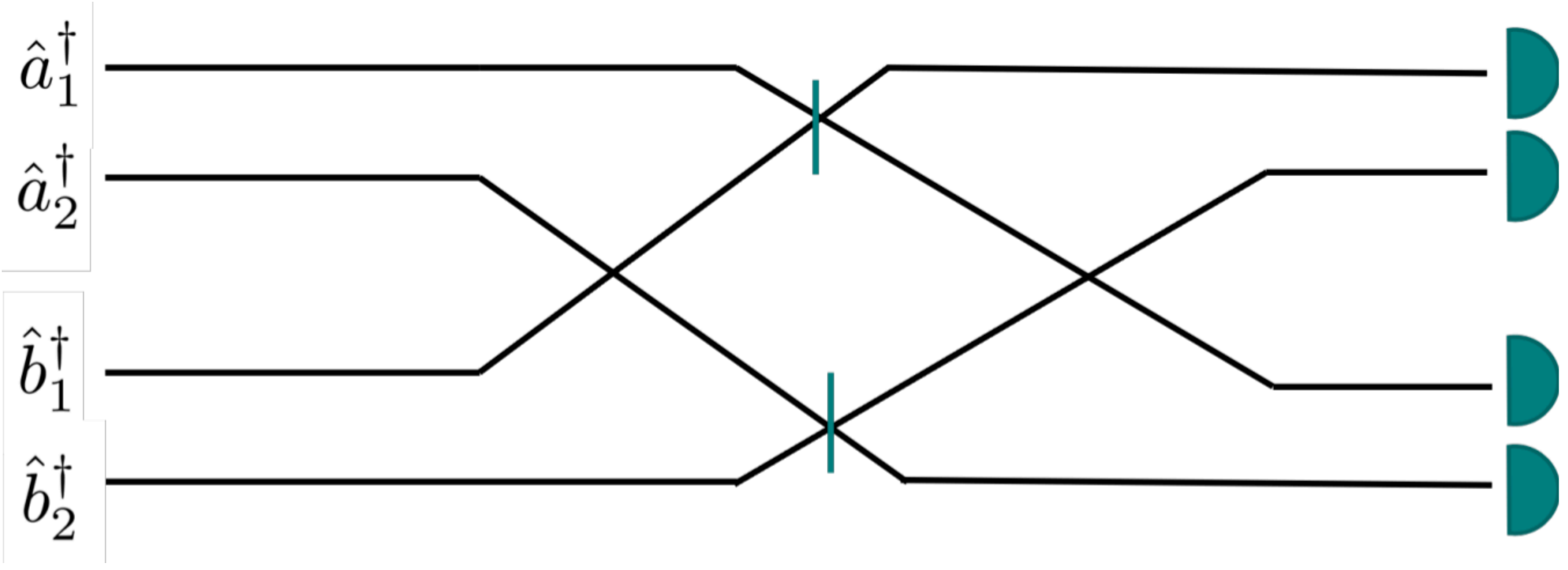


Figure from Ref 4



Type-II fusion in the dual-rail picture

Nice description of the mechanism in this talk:
[Quantum Computing at the Speed of Light - Terry Rudolph - 11/4/21]

TYPE-II FUSION MEASUREMENTS

- Yield 2 measurement outcomes, eg <XX, ZZ> or <XZ, ZX>
- Success heralded by a ‘click’ in both detectors

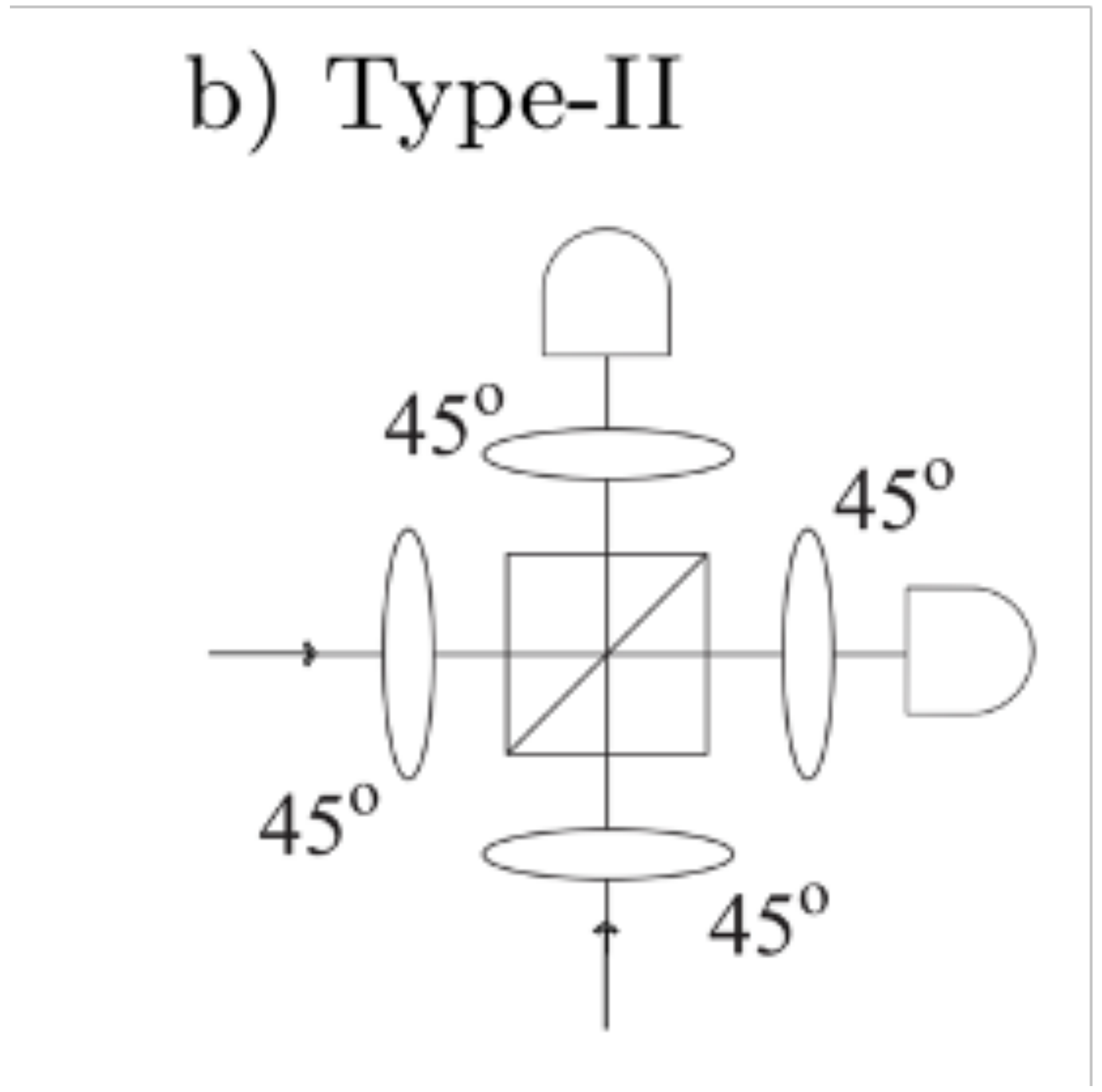


Figure from Ref 4

TYPE-II FUSION MEASUREMENTS

- Yield 2 measurement outcomes, eg <XX, ZZ> or <XZ, ZX>
- Success heralded by a ‘click’ in both detectors
- Extra motivation: failure → measurement in the X basis, *not* Z
  - Does not destroy the larger entangled state that the photon may be part of.

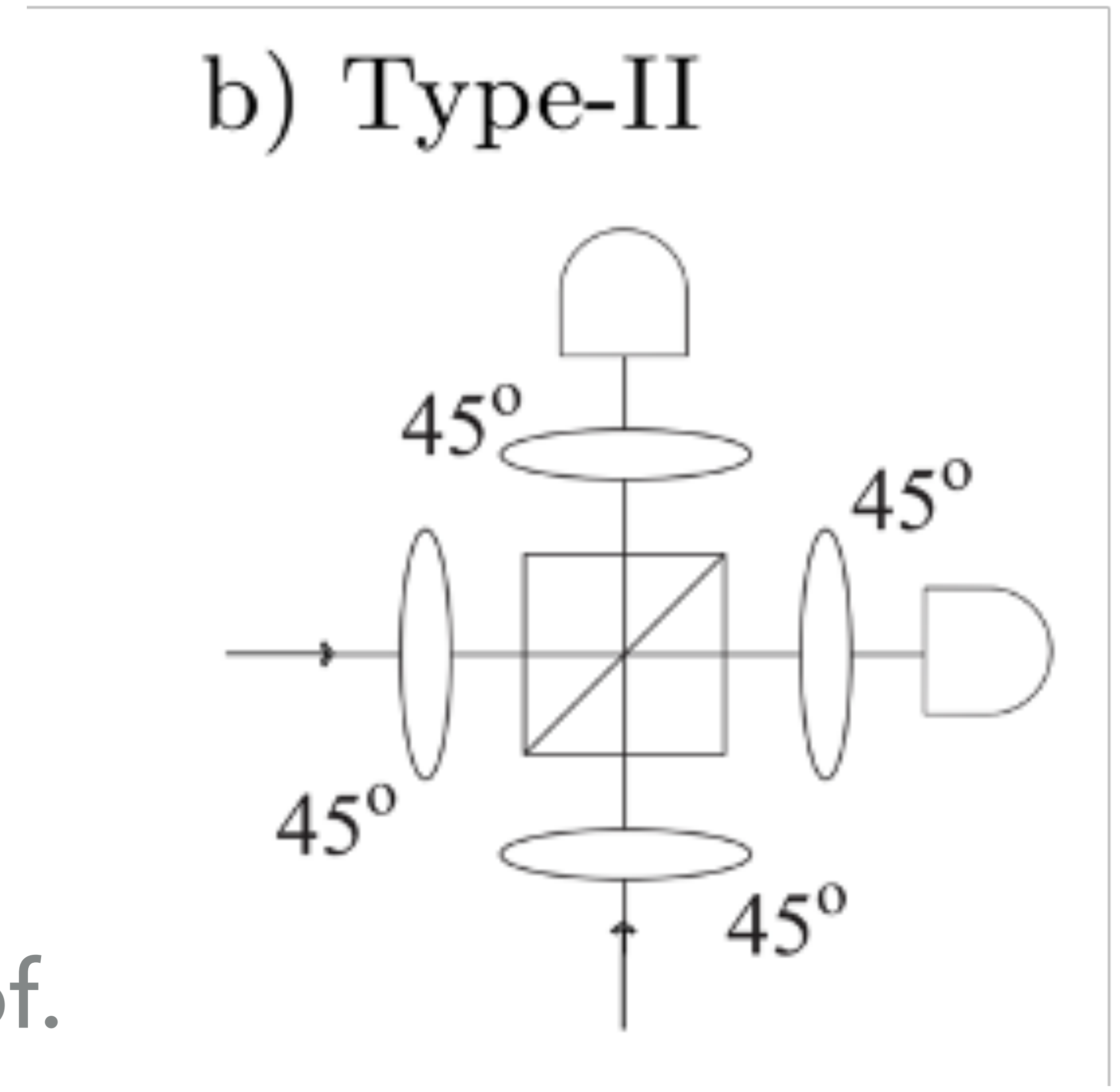


Figure from Ref 4

TYPE-II FUSION MEASUREMENTS

- Yield 2 measurement outcomes, eg <XX, ZZ> or <XZ, ZX>
- Success heralded by a 'click' in both detectors
- Extra motivation: failure → measurement in the X basis, *not* Z
    - Does not destroy the larger entangled state that the photon may be part of
- Propagate entanglement/perform computation in FBQC

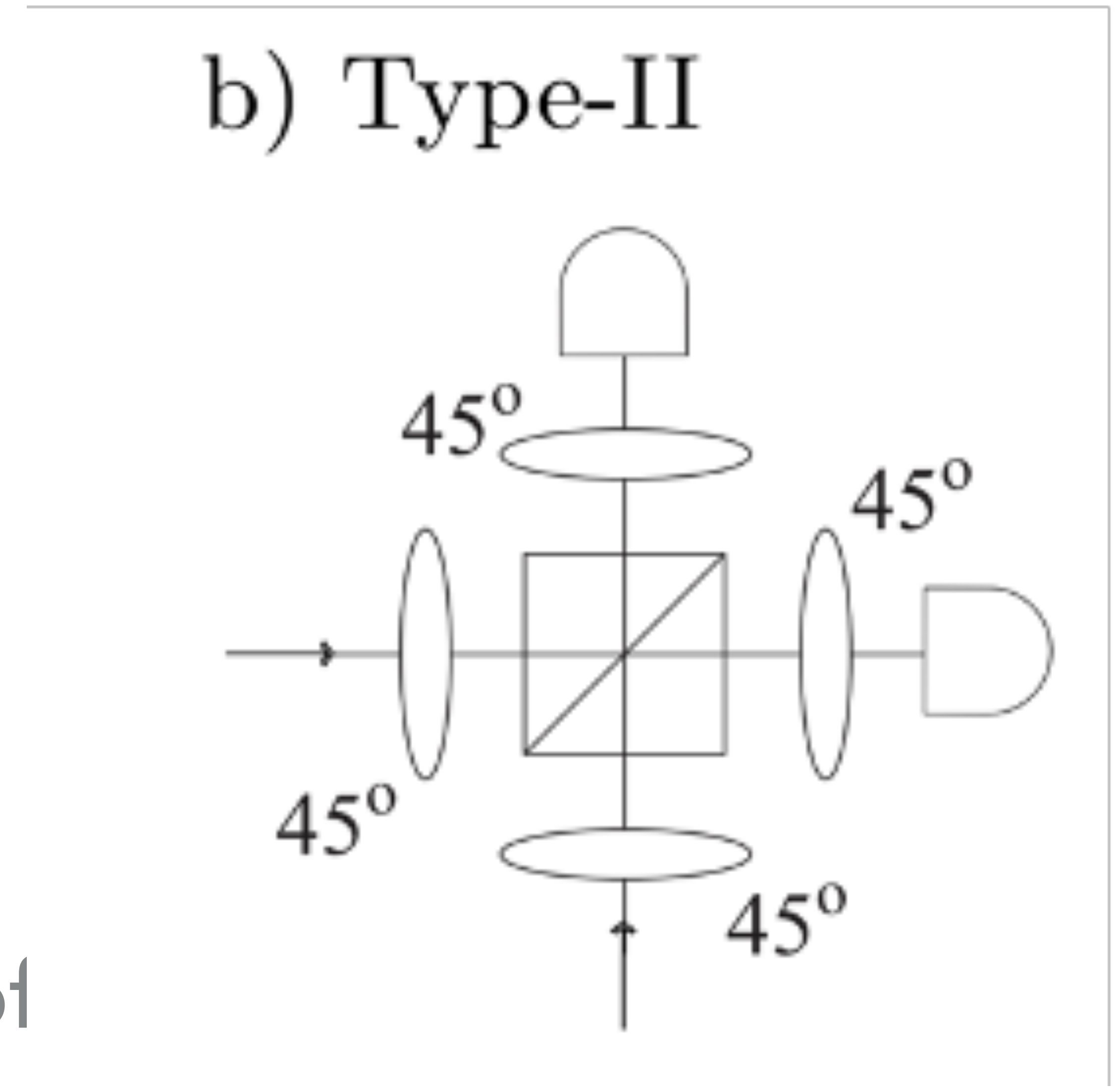


Figure from Ref 4

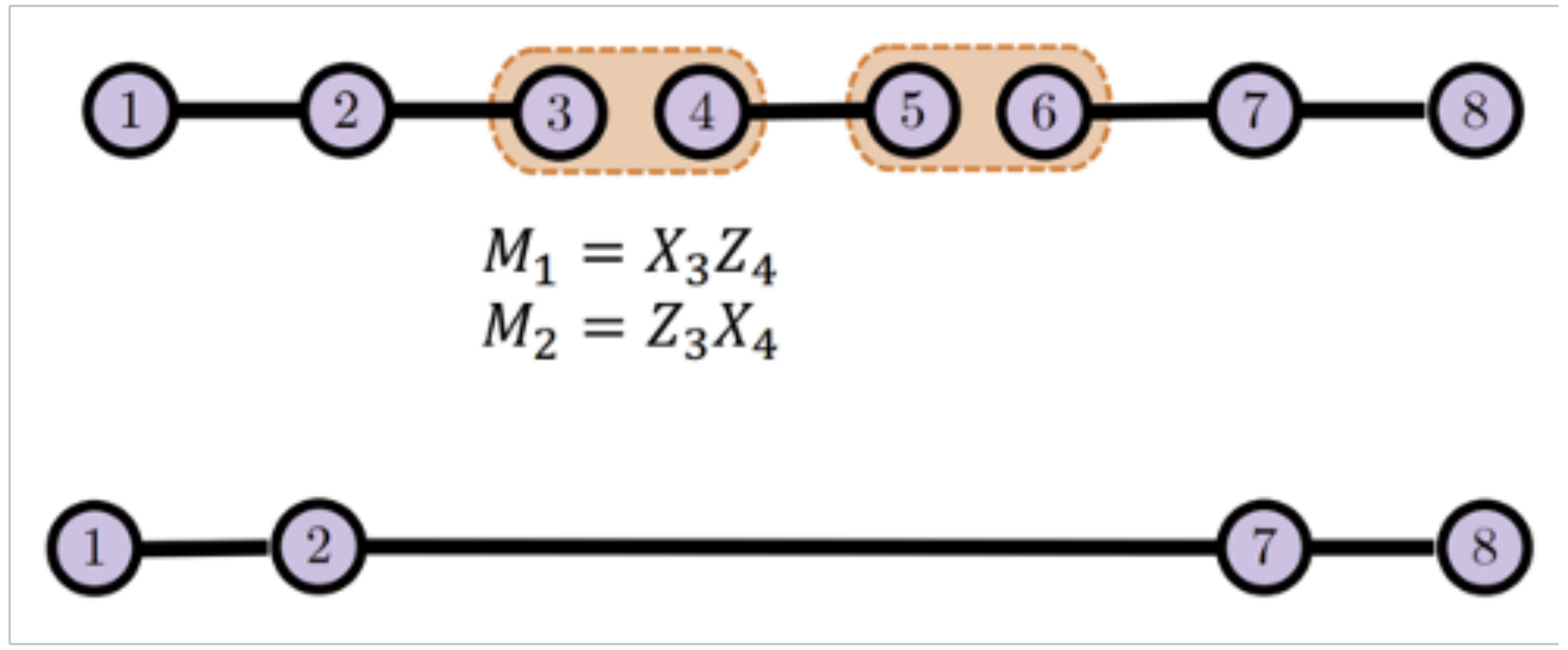
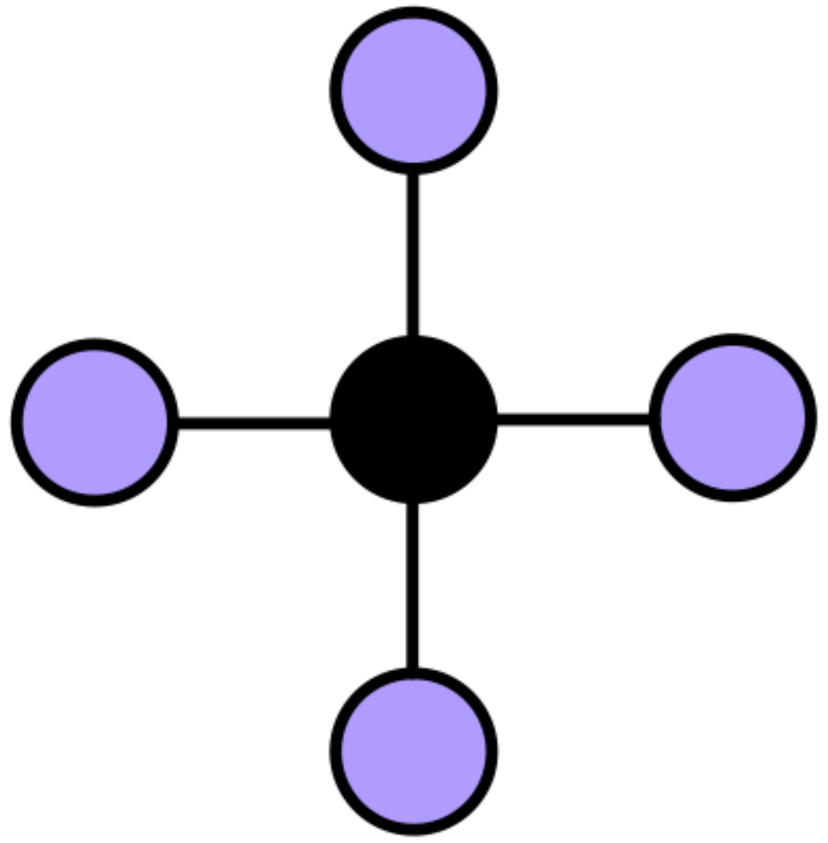
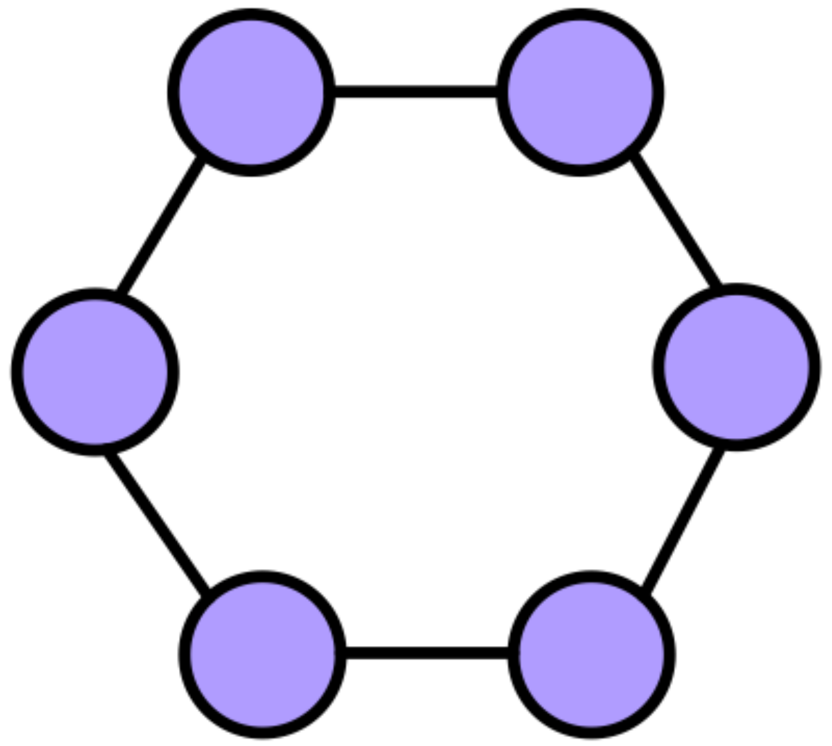


Figure from Ref 1

RESOURCE STATES

- In FBQC, information propagates via fusion between resource states
- Small entangled states usually O(10) qubits
  - (size independent of computation)
- As in MBQC, described via graph state representation





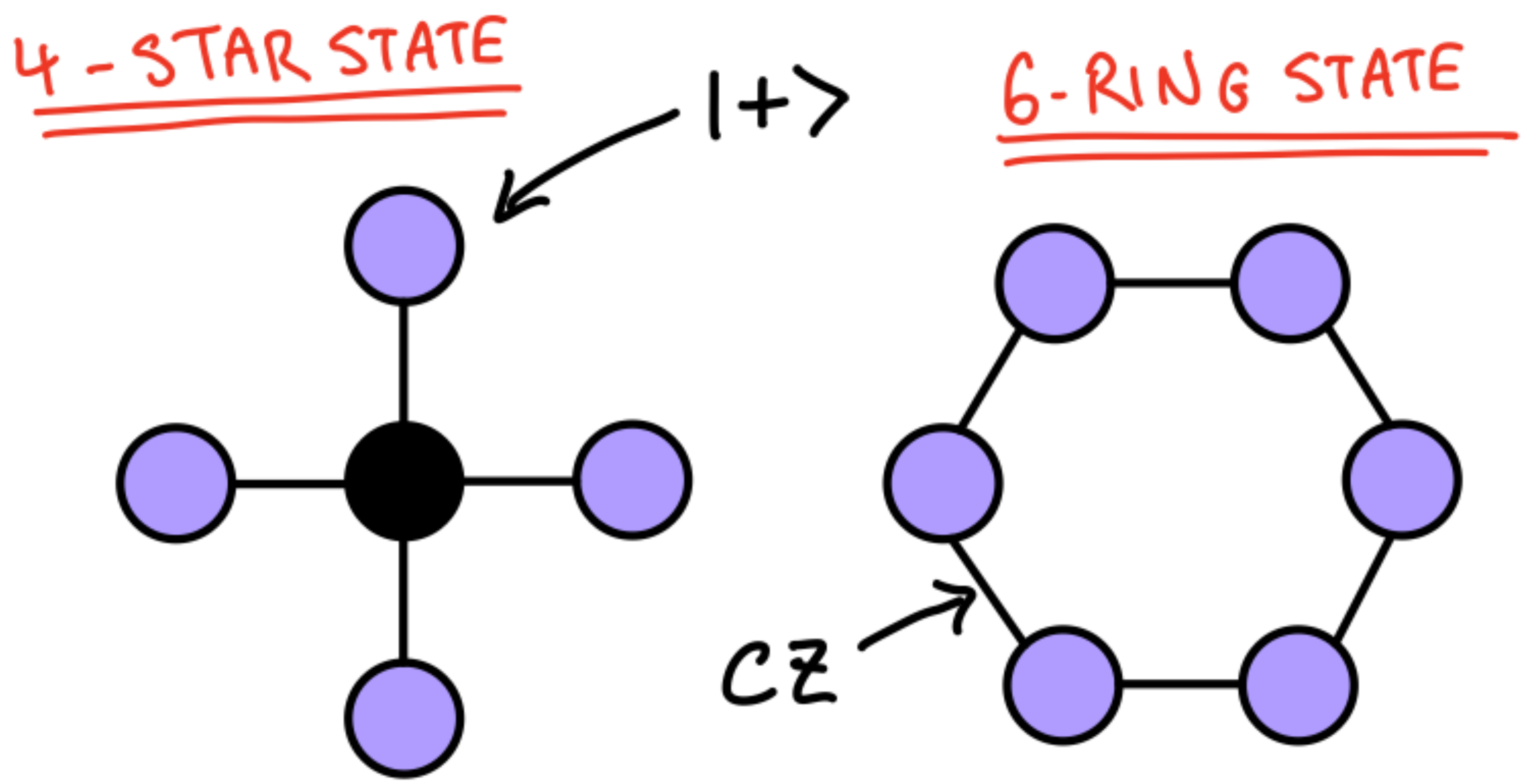
GRAPH ~~RESOURCE~~ STATES

- Graph states are a way to describe a quantum state and are useful in e.g. MBQC or QEC

- Given graph G = (V,E) we have:

$$|G\rangle = \prod_{(a,b)\in E} CZ^{\{a,b\}} |+\rangle^{\otimes V}$$

- Most useful when working in the stabilizer formalism



GRAPH

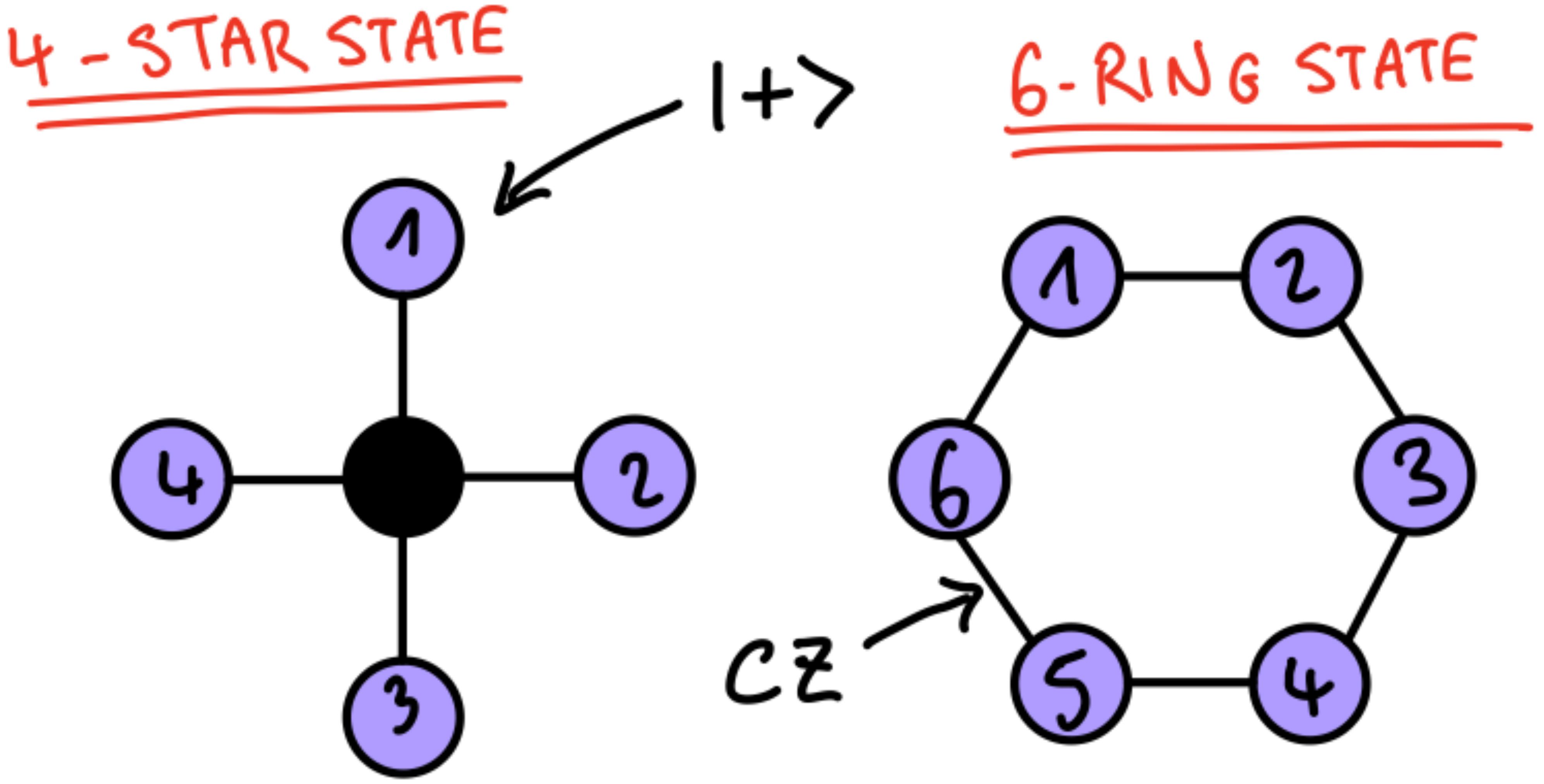
~~RESOURCE~~ STATES

- Stabilizer formalism: describe your state by which operations leave it invariant:
  - A stabilizer state on *n* qubits is a +1 eigenstate of a size *2n* subgroup of Pauli operators
- For graph states:

$$X_i \prod_{j \in \mathcal{N}(i)} Z_j$$

Neighbours of i

"stabilizers"



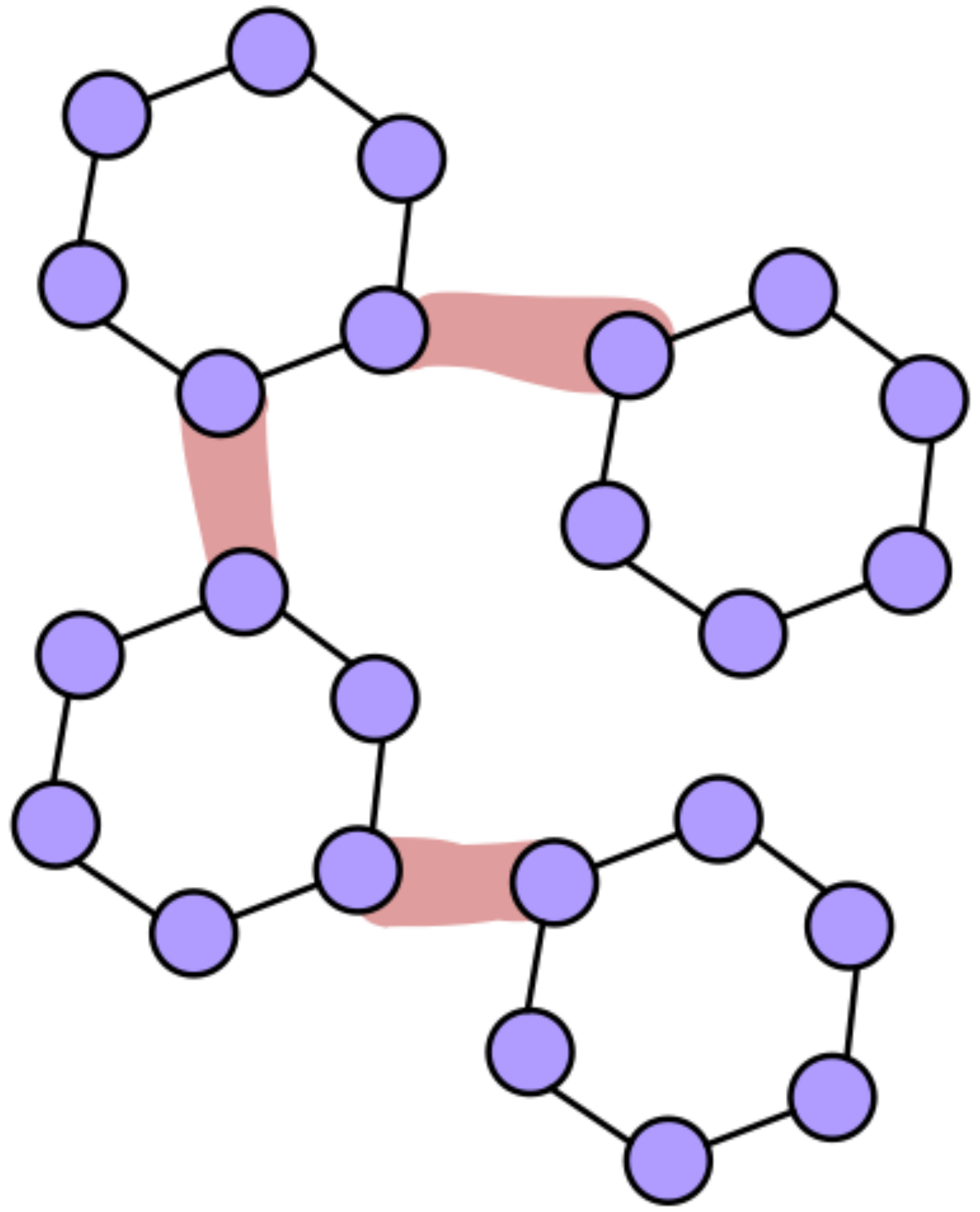
$$S_1 = Z_1 Z_2 Z_3 Z_4$$
$$S_2 = X_1 X_2 \mathbb{1}_3 \mathbb{1}_4$$
$$S_3 = \mathbb{1}_1 X_2 X_3 \mathbb{1}_4$$
$$S_4 = \mathbb{1}_1 \mathbb{1}_2 X_3 X_4$$

$$S_1 = Z_1 X_2 Z_3 \mathbb{1}_4 \mathbb{1}_5 \mathbb{1}_6$$
$$S_2 = \mathbb{1}_1 Z_2 X_3 Z_4 \mathbb{1}_5 \mathbb{1}_6$$
$$S_3 = \mathbb{1}_1 \mathbb{1}_2 Z_3 X_4 Z_5 \mathbb{1}_6$$
$$S_4 = \mathbb{1}_1 \mathbb{1}_2 \mathbb{1}_3 Z_4 X_5 Z_6$$
$$S_5 = Z_1 \mathbb{1}_2 \mathbb{1}_3 \mathbb{1}_4 Z_5 X_6$$
$$S_6 = X_1 Z_2 \mathbb{1}_3 \mathbb{1}_4 \mathbb{1}_5 Z_6$$

FUSION NETWORKS

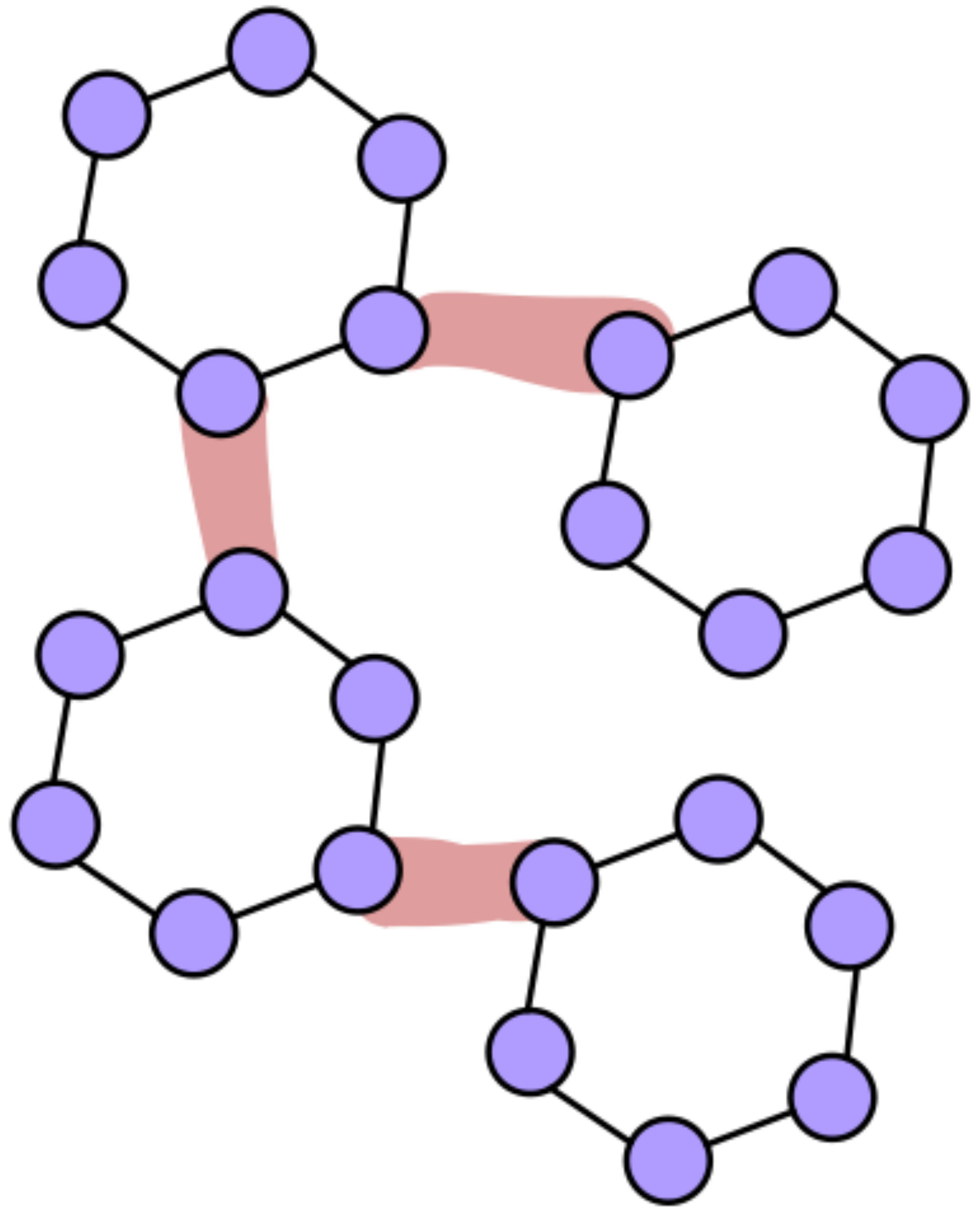
FUSION NETWORKS

- In a large-scale FBQC computation, FNs specify an arrangement of:
  - resource states and
  - a set of fusion measurements to be made on them



FUSION NETWORKS

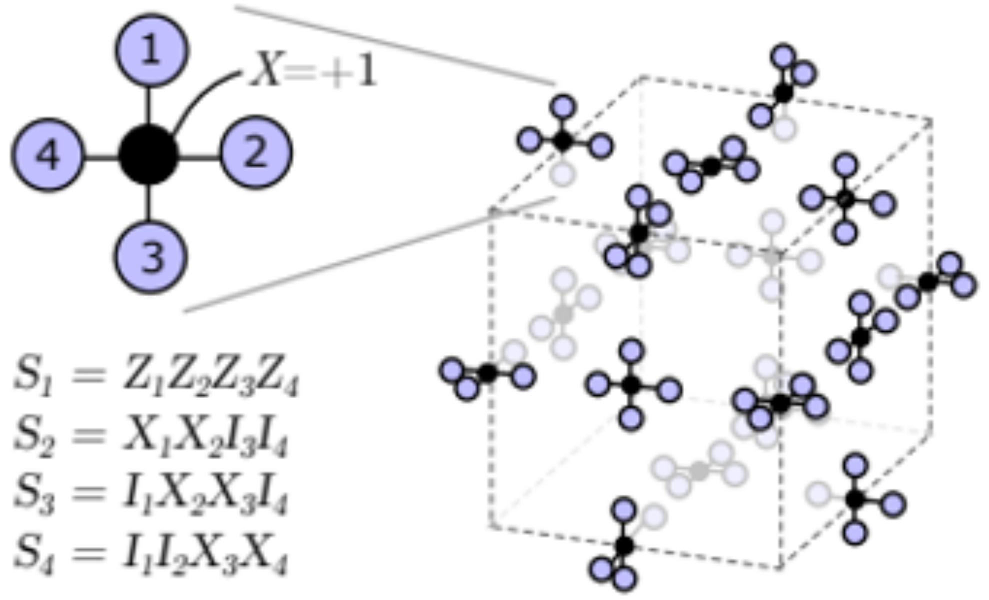
- In a large-scale FBQC computation, FNs specify an arrangement of:
  - resource states and
  - a set of fusion measurements to be made on them



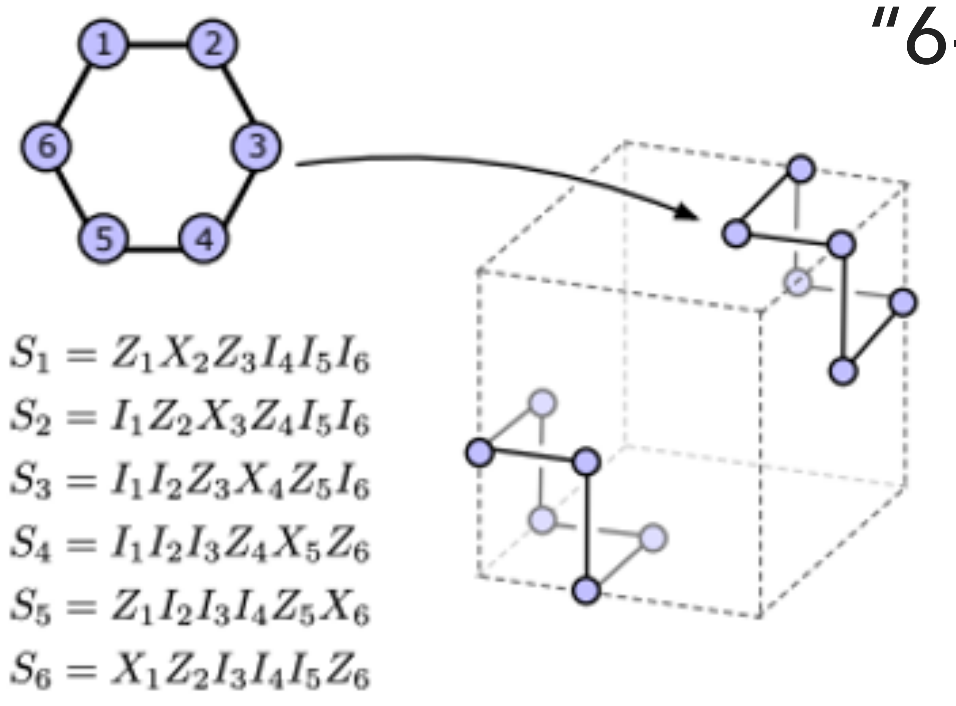
- This arrangement is informed by:
  - What quantum correlations remain in unmeasured, “output” qubits?
  - What classical information has been obtained during the measurements?

FUSION NETWORKS

"4-star" cell

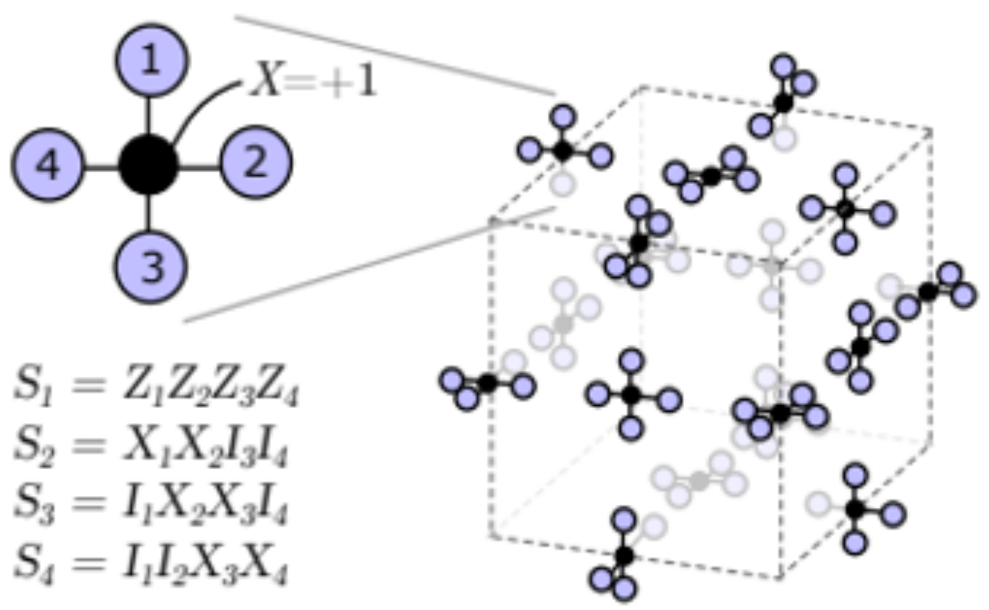


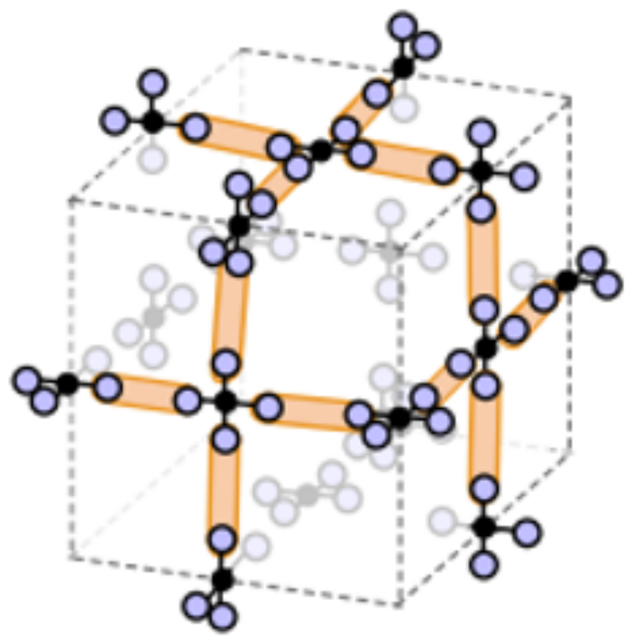
"6-ring" cell



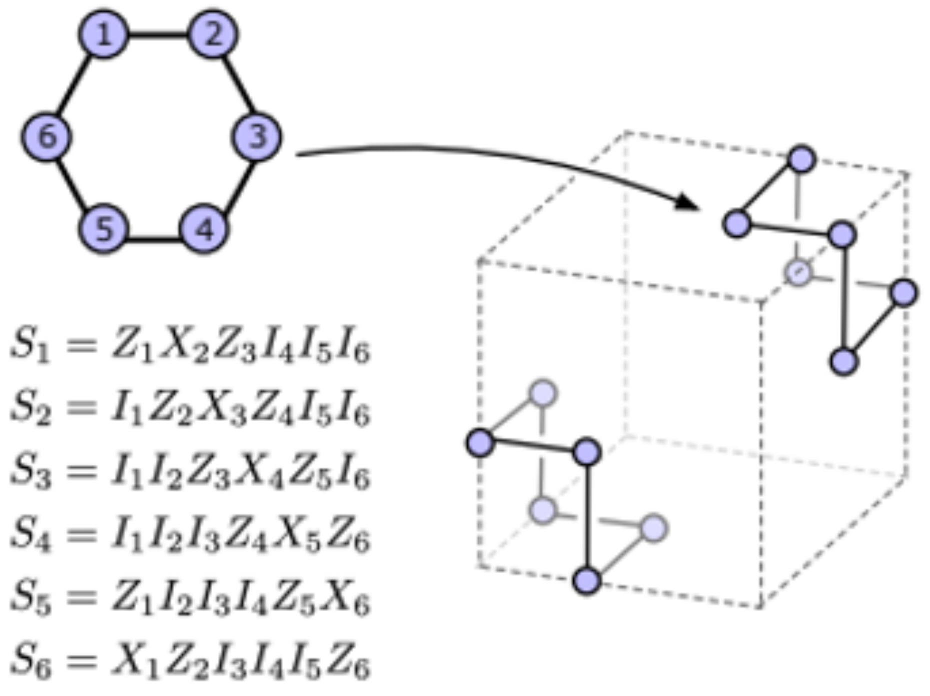
Figures from Ref 1

FUSION NETWORKS

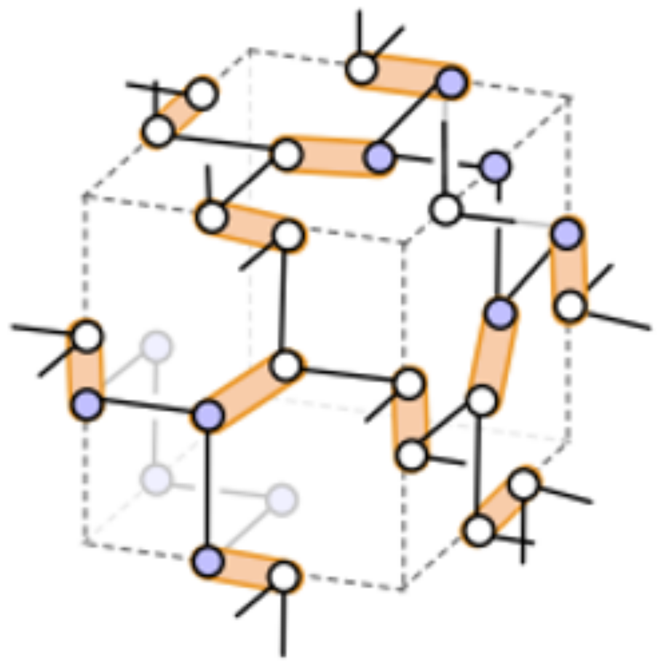




“4-star” cell FN

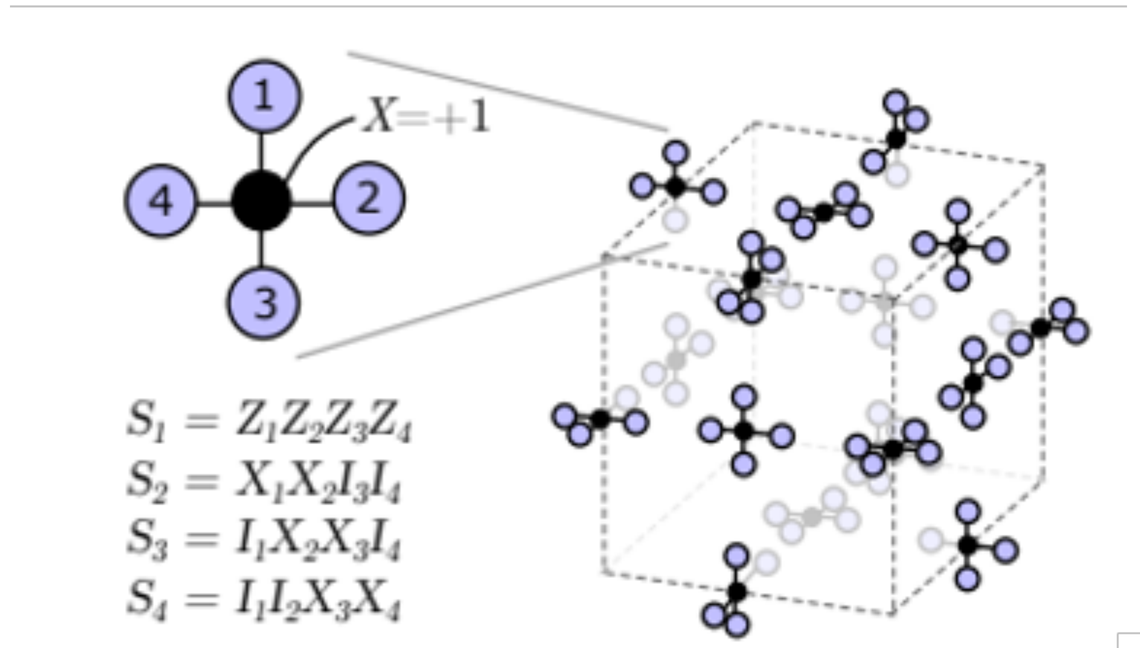


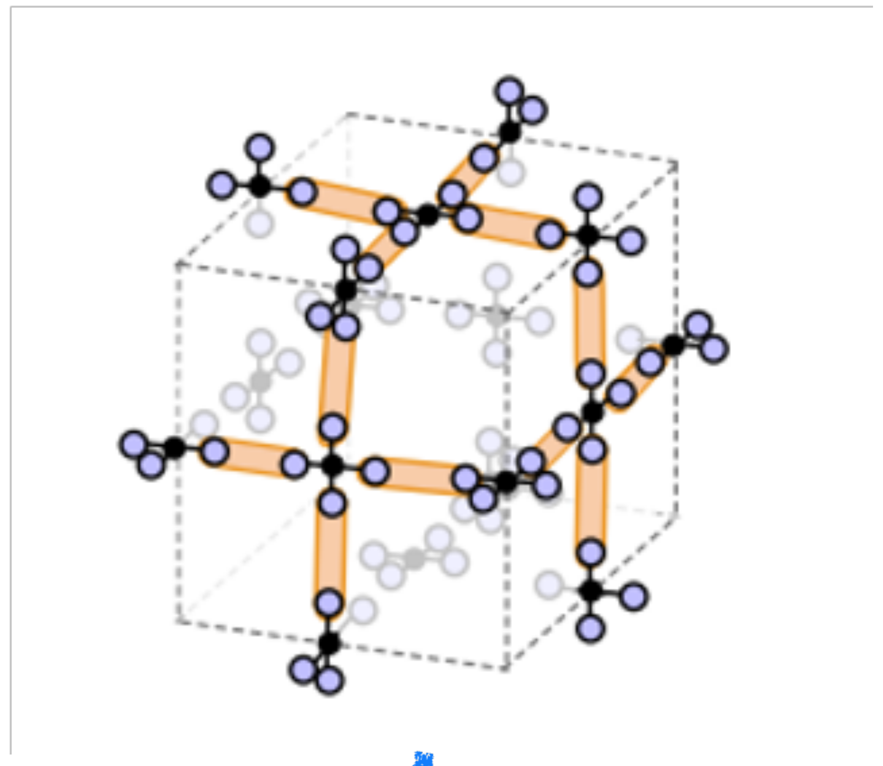
“6-ring” cell FN

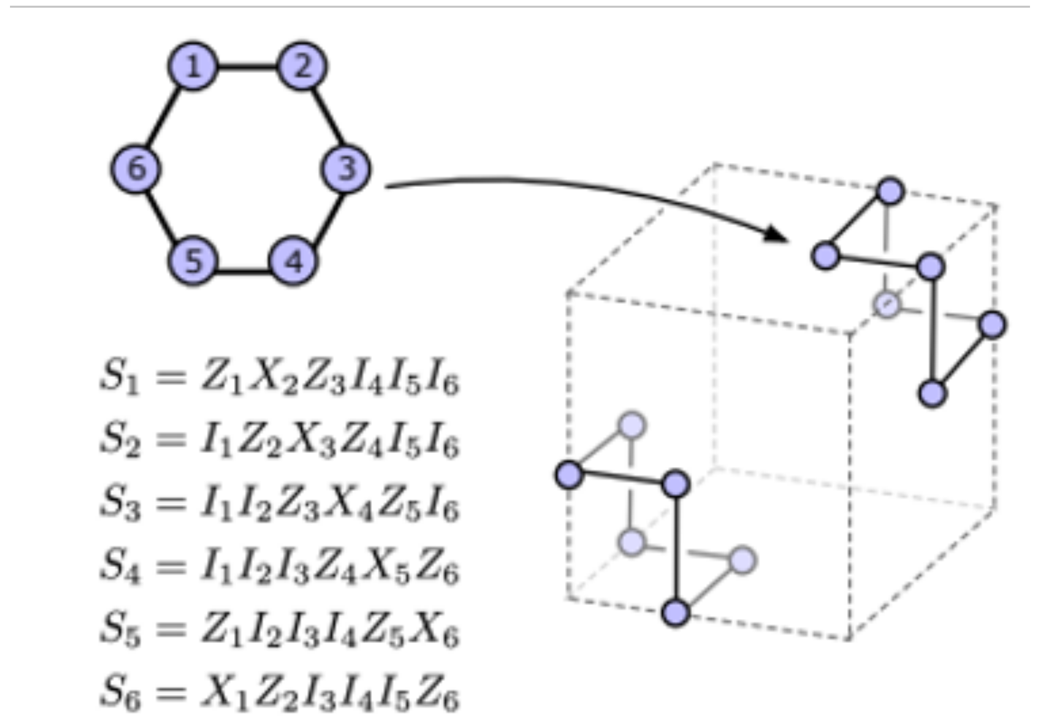


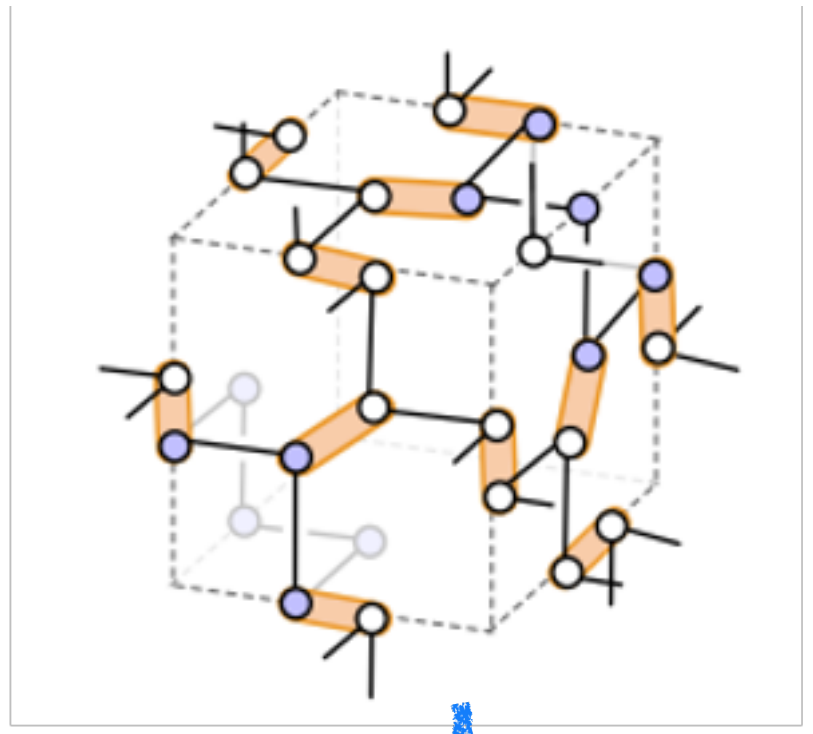
Figures from Ref 1

FUSION NETWORKS

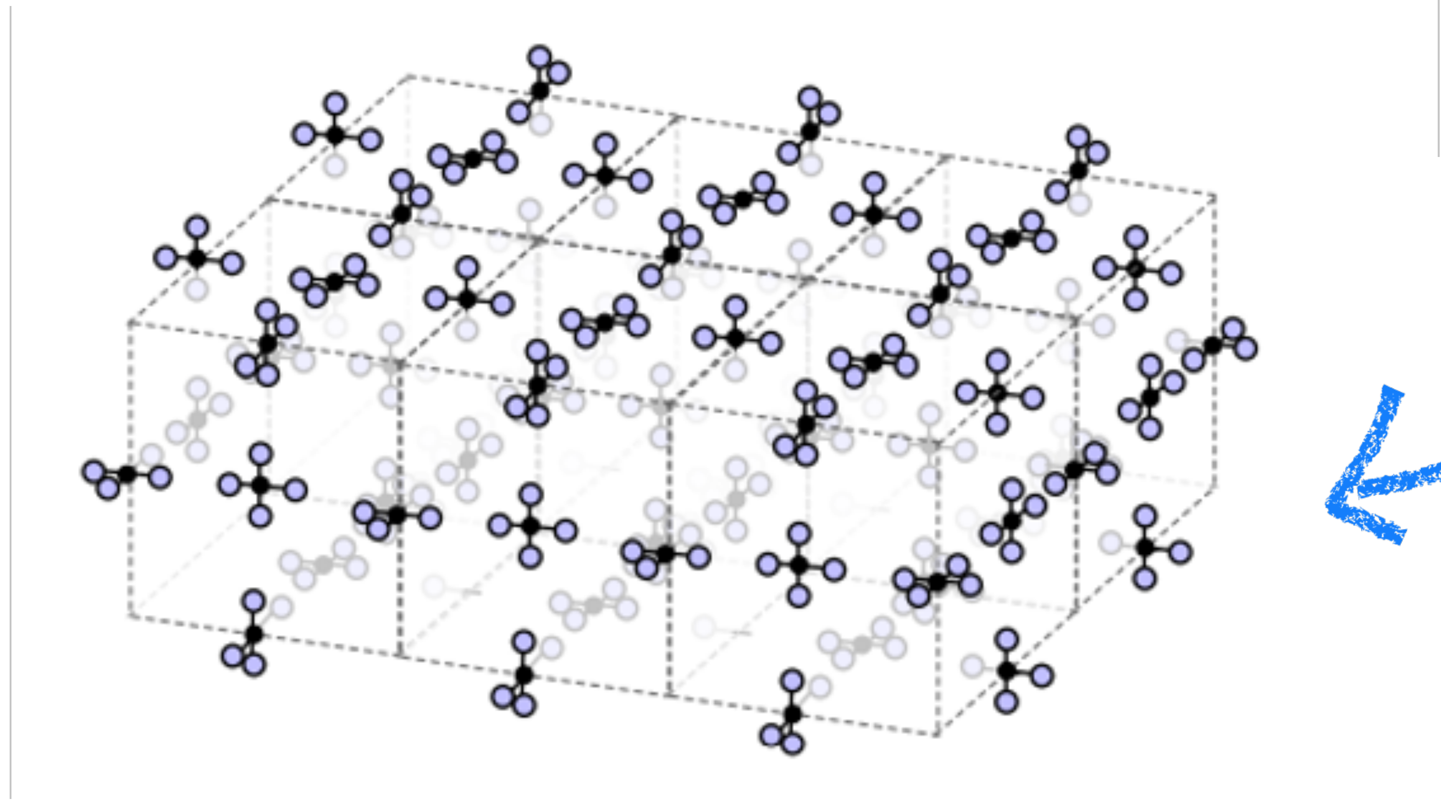




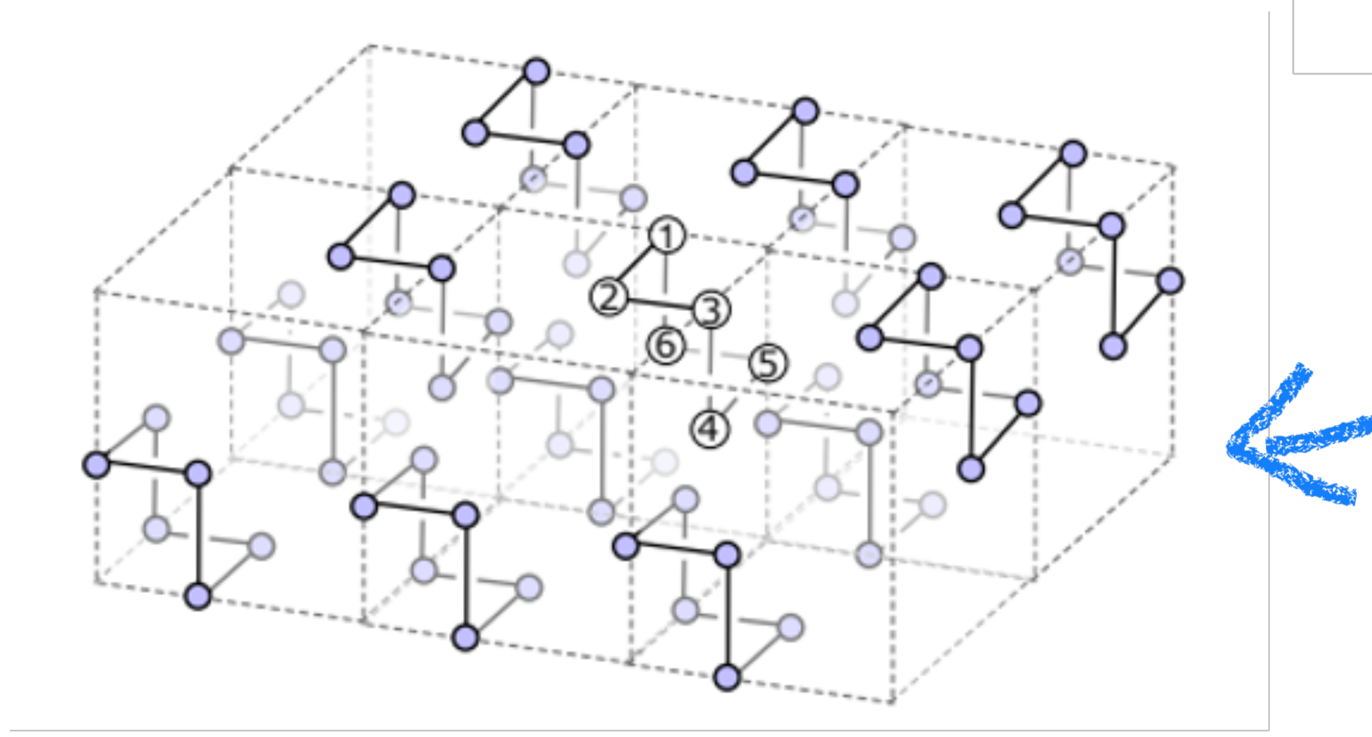




"4-star" cubic lattice FN



"6-ring" cubic lattice FN



Figures from Ref 1

FNS AND COMPUTATION

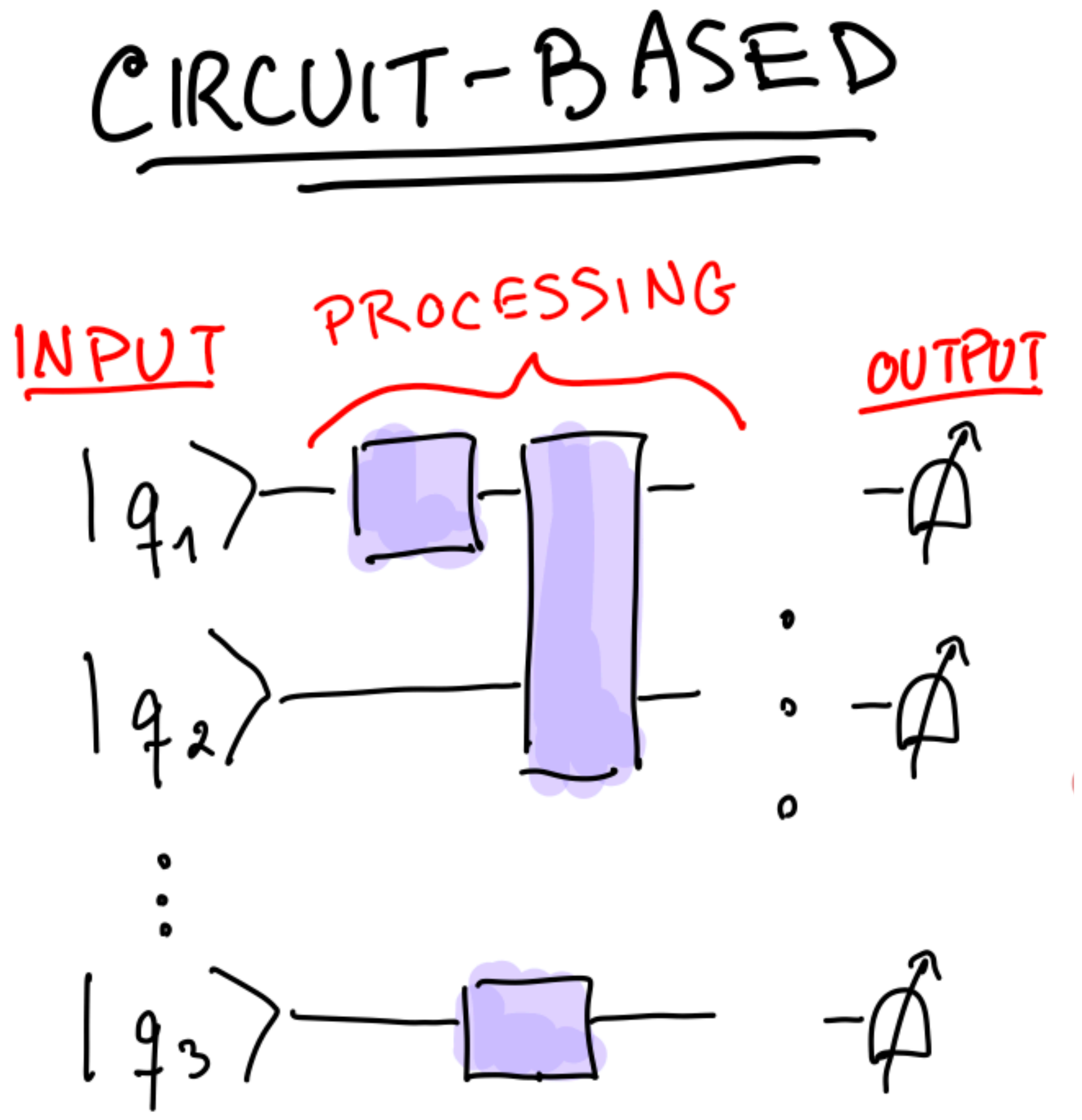
- How do you describe a computation in this model?

FNS AND COMPUTATION

- How do you describe a computation in this model?

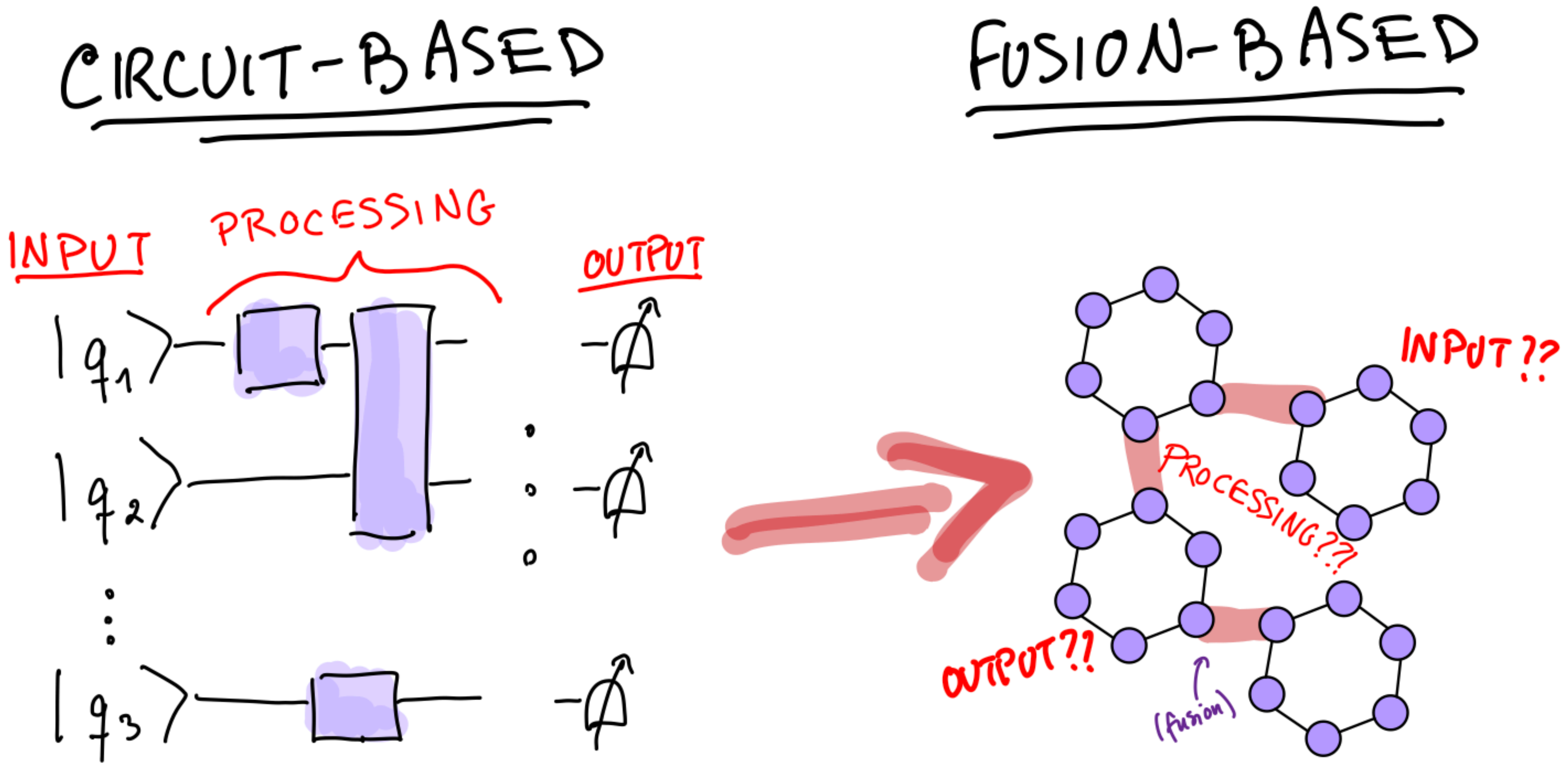
FNS AND COMPUTATION

- How do you describe a computation in this model?



FNS AND COMPUTATION

- How do you describe a computation in this model?



FNS AND COMPUTATION

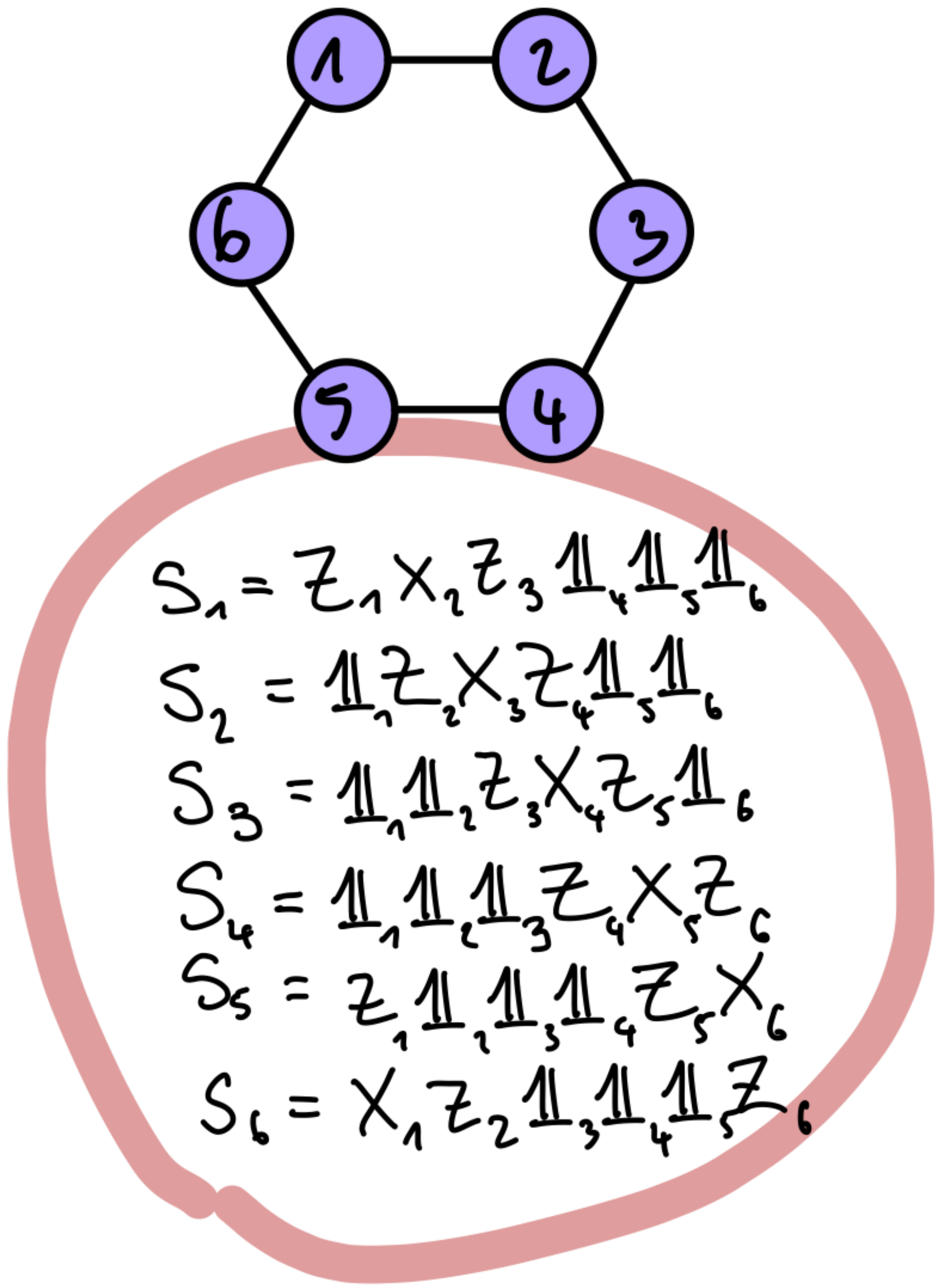
- Use the stabilizer formalism!

- FNs primarily described by 3 Pauli subgroups:

FNS AND COMPUTATION

- Use the stabilizer formalism!

- FNs primarily described by 3 Pauli subgroups:
  - INPUT: Stabilizer group **R** which describes ideal resource states



FNS AND COMPUTATION

- Use the stabilizer formalism!

- FNs primarily described by 3 Pauli subgroups:
    - INPUT: Stabilizer group **R** which describes ideal resource states
    - PROCESSING: Fusion group **F** which defines the fusion measurements and includes -1

e.g. $F = \langle X_1 X_2, Z_1 Z_2, -\mathbb{1} \rangle$

FNS AND COMPUTATION

- Use the stabilizer formalism!

- FNs primarily described by 3 Pauli subgroups:
    - INPUT: Stabilizer group **R** which describes ideal resource states
    - PROCESSING: Fusion group **F** which defines the fusion measurements and includes -1
    - OUTPUT: Surviving stabilizer group **S**, the set of elements of R that commute with all elements of F

FNS AND COMPUTATION

- Use the stabilizer formalism!

- FNs primarily described by 3 Pauli subgroups:
  - INPUT: Stabilizer group **R** which describes ideal resource states
  - PROCESSING: Fusion group **F** which defines the fusion measurements and includes -1
  - OUTPUT: Surviving stabilizer group **S**, the set of elements of R that commute with all elements of F
    - For destructive measurements, we need the output stabilizer group $\mathbf{S_{out}}$ which only contains qubits that are unmeasured + measurement outcomes

EXAMPLE

- Take two small graph states as input:

$$R = \langle (X_1 Z_2, Z_1 X_2), (X_3 Z_4, Z_3 X_4) \rangle$$

EXAMPLE

- Take two small graph states as input:

$$R = \langle (X_1 Z_2 , Z_1 X_2), (X_3 Z_4 , Z_3 X_4) \rangle$$

- Fusion between qubits 2 and 3:

$$F = \langle X_2 X_3 , Z_2 Z_3 , -1 \rangle$$

EXAMPLE

- Take two small graph states as input:

$$R = \langle (X_1 Z_2, Z_1 X_2), (X_3 Z_4, Z_3 X_4) \rangle$$

- Fusion between qubits 2 and 3:

$$F = \langle X_1 X_2, Z_1 Z_2, -1 \rangle$$

- Output:

$$S = \langle X_1 Z_2 Z_3 X_4, Z_1 X_2 X_3 Z_4 \rangle$$

EXAMPLE

- Take two small graph states as input:

$$R = \langle (X_1 Z_2, Z_1 X_2), (X_3 Z_4, Z_3 X_4) \rangle$$

- Fusion between qubits 2 and 3:

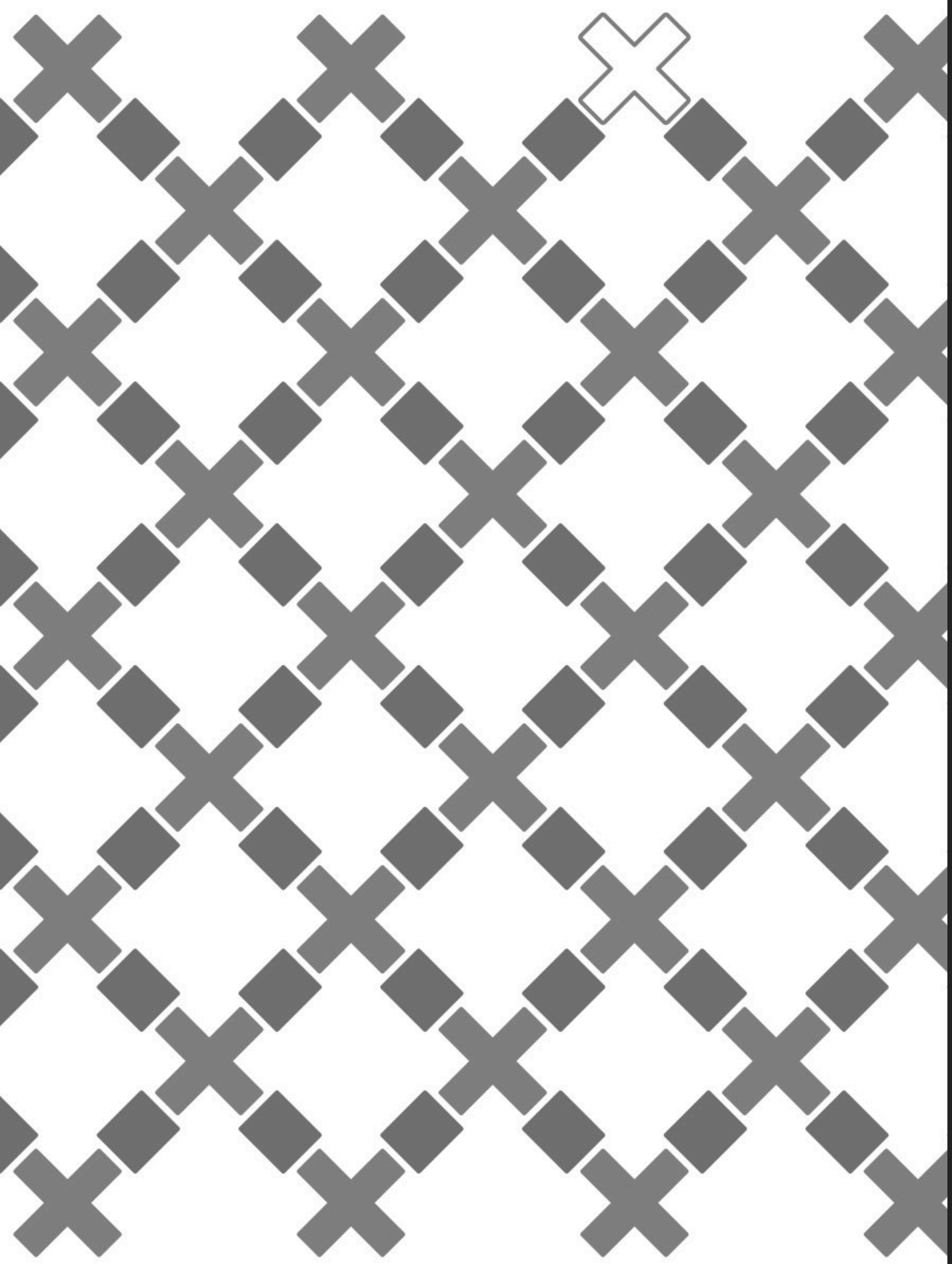
$$F = \langle X_1 X_2 , Z_1 Z_2 , -1 \rangle$$

- Output:

$$S_{out} = \langle X_1 \, m^{ZZ} \, X_4 , Z_1 \, m^{XX} \, Z_4 \rangle$$

EXAMPLE

$$R = \langle (X_1 Z_2,\ Z_1 X_2),\ (X_3 Z_4,\ Z_3 X_4) \rangle$$



FAULT-TOLERANT FUSION NETWORKS

QUICK NOTE ON FAULT-TOLERANCE

- Need to add redundancy to quantum state to be able to catch and correct errors
  - Logical qubit: encode state of single qubit into many qubits
  - Surface codes!

$$|0\rangle \longrightarrow |000\rangle = |0\rangle_L$$

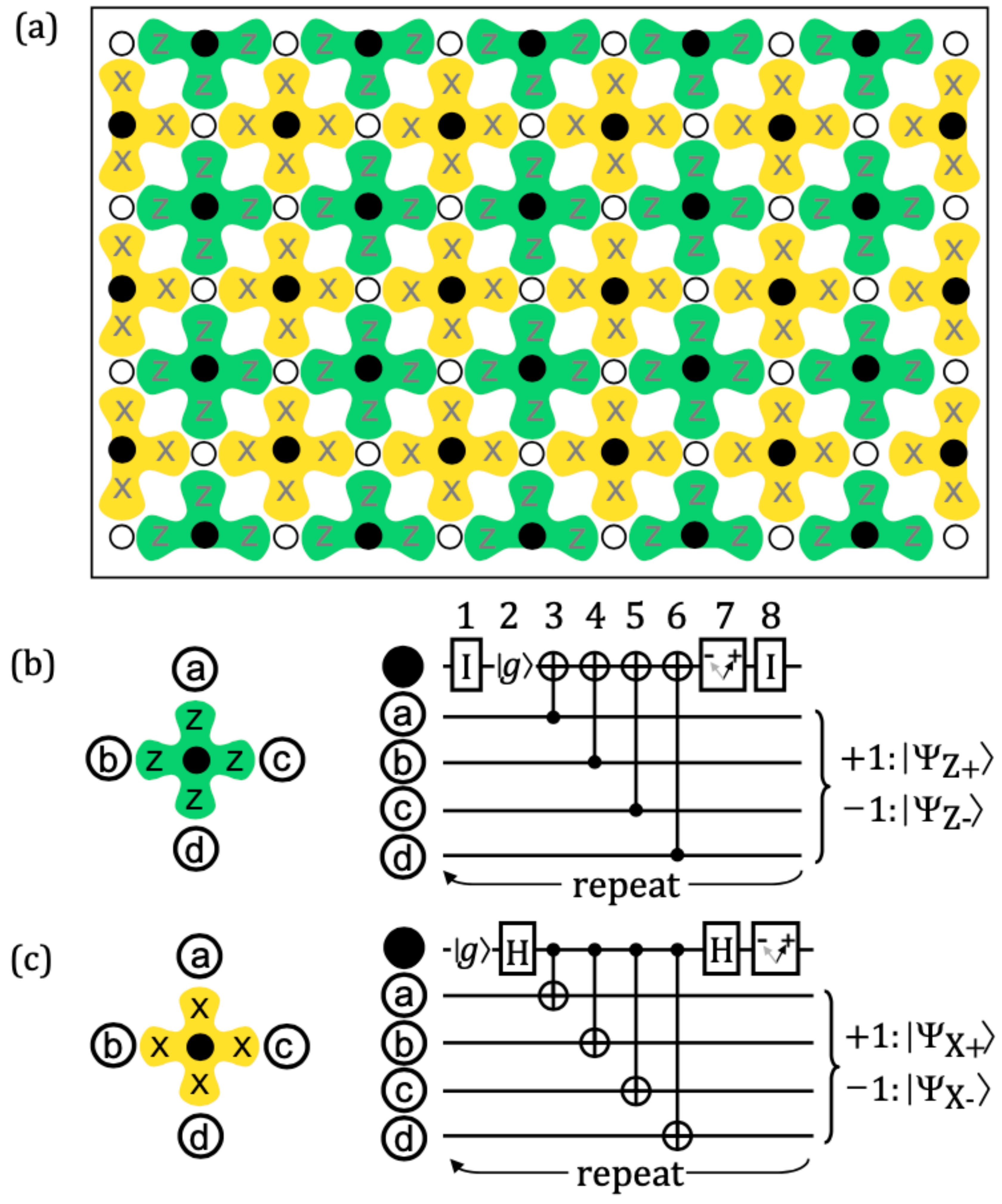


Figure from Ref 8

QUICK NOTE ON FAULT-TOLERANCE

- Need to add redundancy to quantum state to be able to catch and correct errors
  - Logical qubit: encode state of single qubit into many qubits
  - Surface codes!

$$|0\rangle \longrightarrow |000\rangle = |0\rangle_L$$

- Set of surface codes not complete:
  - Leave degrees of freedom like boundaries/(topological) defects to perform logical operations on logical qubits

$$X_L|0\rangle_L = |1\rangle_L = |111\rangle$$

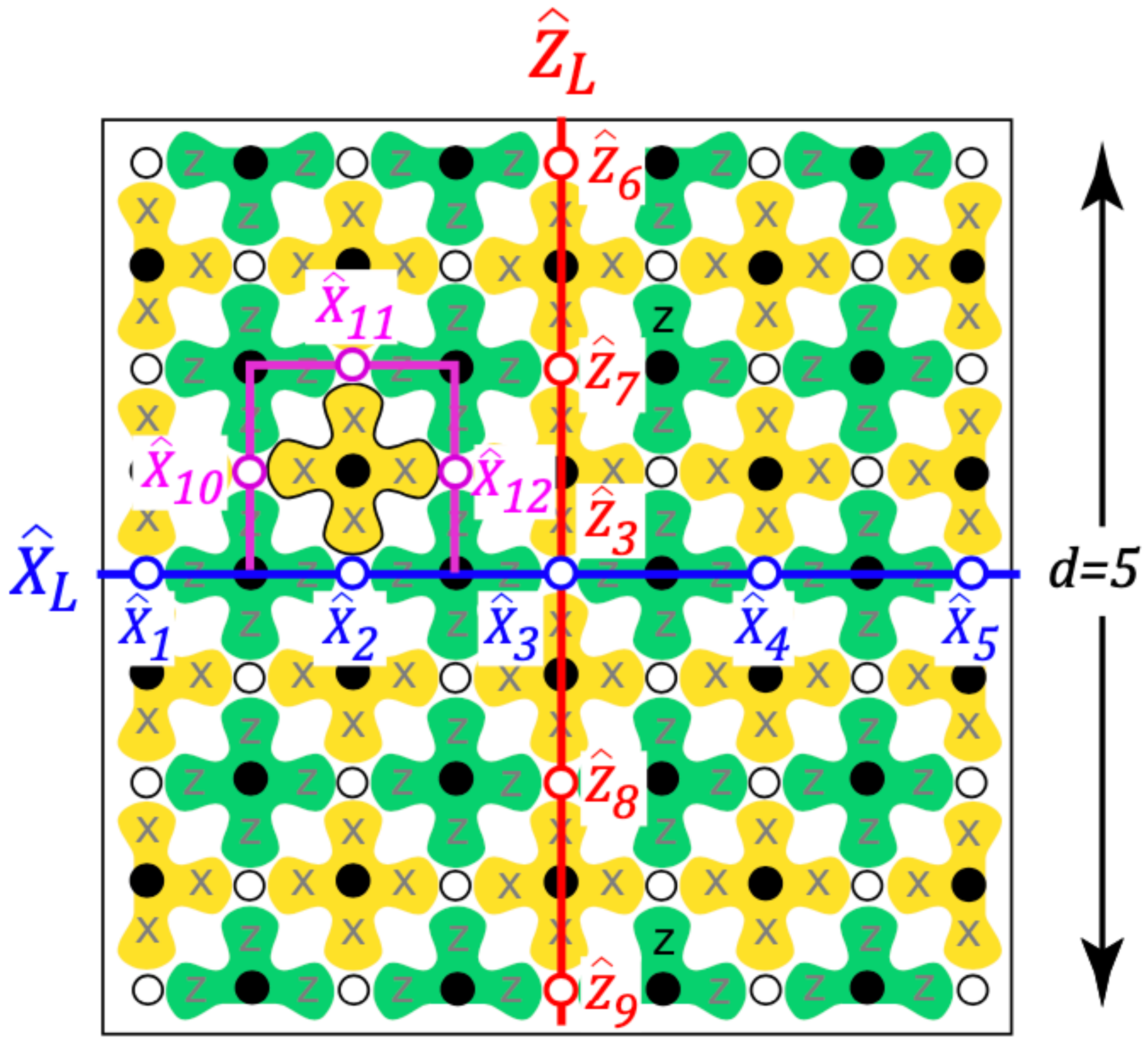


Figure from Ref 8

CHECK OPERATORS

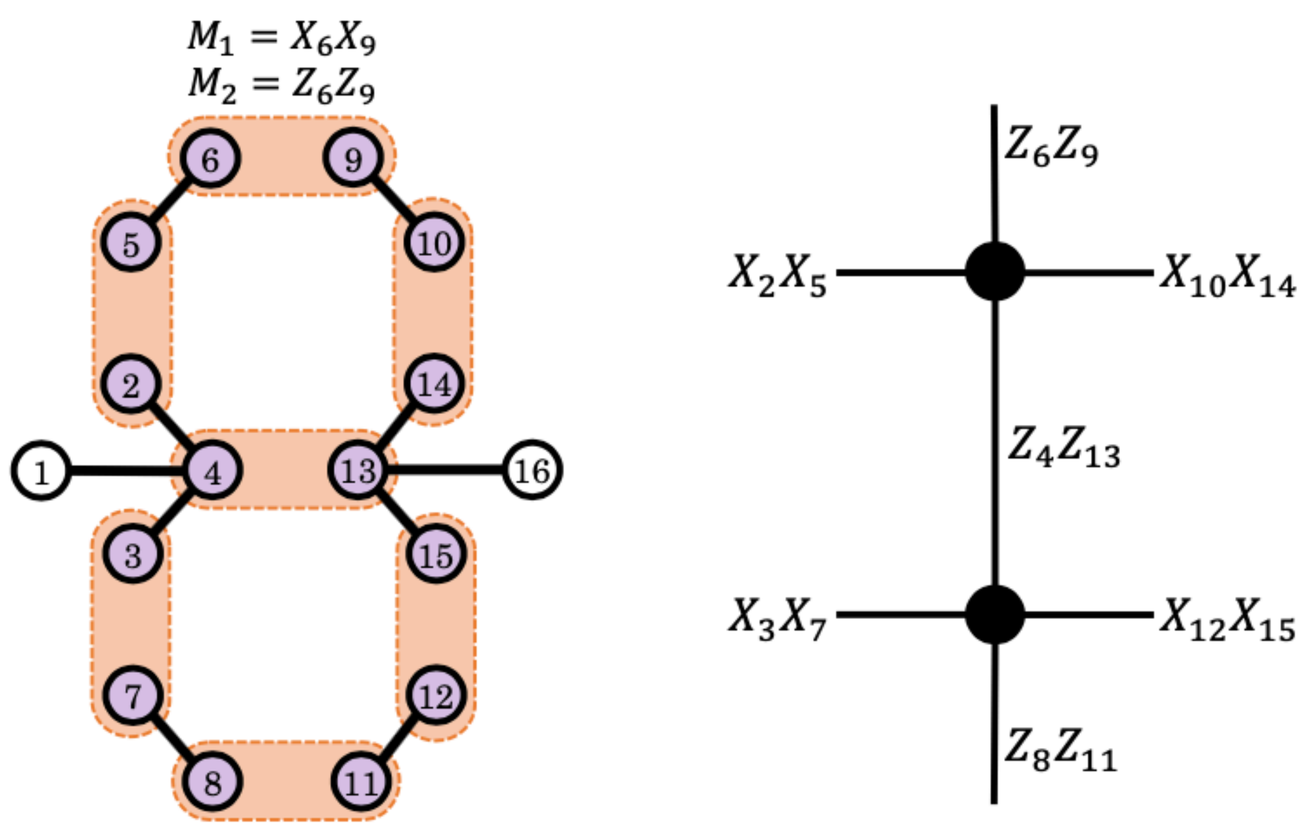
- FTFNs use check operators **C:**

$$C = R \cap F$$

- These identify and keep track of/correct errors in the computation.
- No errors = measurement outcomes → generators of C have only positive values

SYNDROME GRAPHS

- The value of all the check operators is referred to as a syndrome
- A syndrome graph has vertices corresponding to generators in C and edges corresponding to generators of F

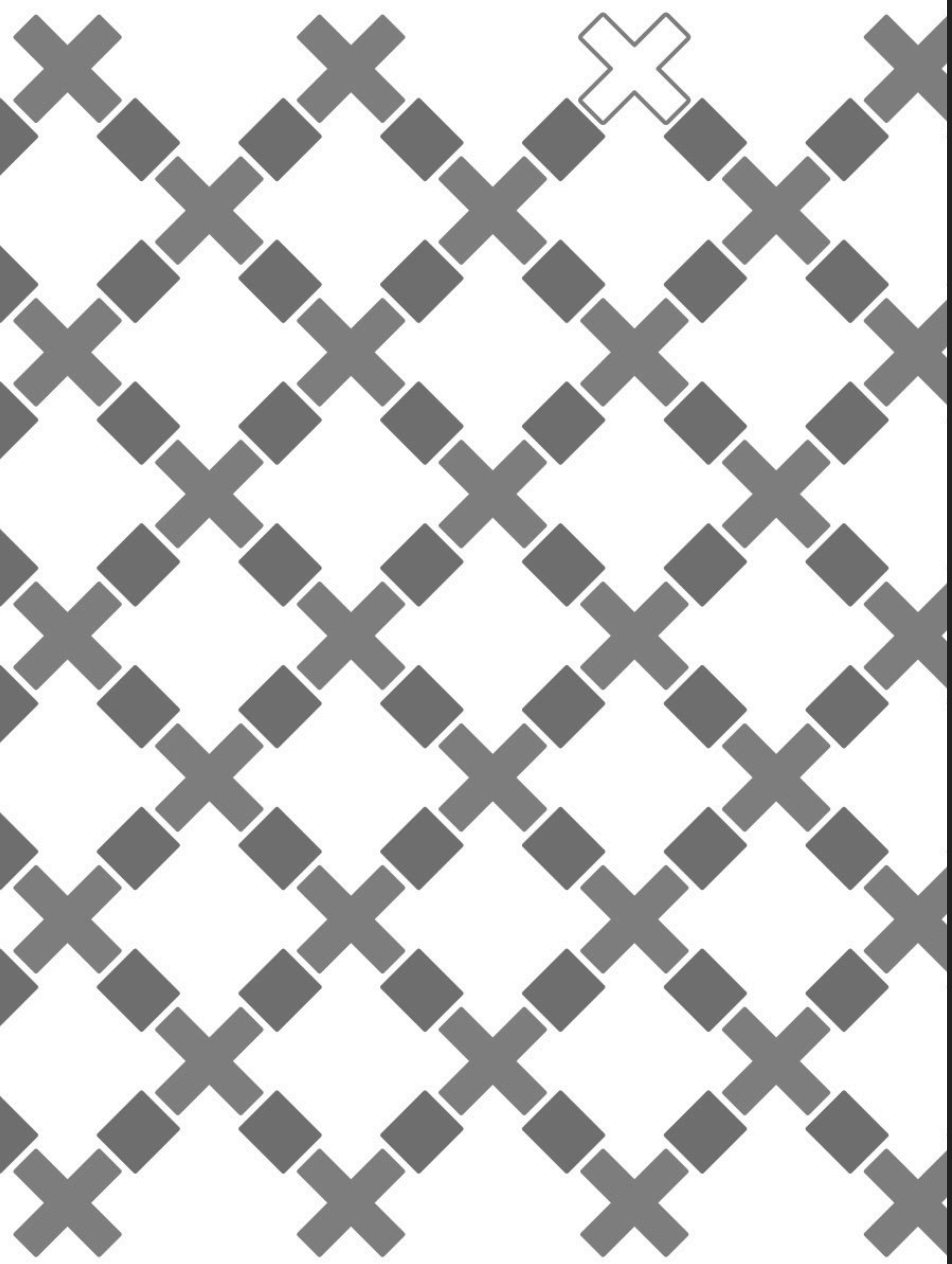


(a) Fusion network

(b) Syndrome graph

Resource States, R	$\left\langle \begin{pmatrix} X_1Z_4, \\ X_2Z_4, \\ X_3Z_4, \\ X_4Z_1Z_2Z_3, \end{pmatrix}, \begin{pmatrix} X_{14}Z_{13}, \\ X_{15}Z_{13}, \\ X_{16}Z_{13}, \\ X_{13}Z_{14}Z_{15}Z_{16} \end{pmatrix}, \begin{pmatrix} X_5Z_6, \\ X_6Z_5 \end{pmatrix}, \begin{pmatrix} X_7Z_8, \\ X_8Z_7 \end{pmatrix}, \begin{pmatrix} X_9Z_{10}, \\ X_{10}Z_9 \end{pmatrix}, \begin{pmatrix} X_{11}Z_{12}, \\ X_{12}Z_{11} \end{pmatrix} \right\rangle$
Fusion, F	$\left\langle \begin{pmatrix} X_2X_5, \\ Z_2Z_5 \end{pmatrix}, \begin{pmatrix} X_3X_7, \\ Z_3Z_7 \end{pmatrix}, \begin{pmatrix} X_6X_9, \\ Z_6Z_9 \end{pmatrix}, \begin{pmatrix} X_8X_{11}, \\ Z_8Z_{11} \end{pmatrix}, \begin{pmatrix} X_4X_{13}, \\ Z_4Z_{13} \end{pmatrix}, \begin{pmatrix} X_{10}X_{14}, \\ Z_{10}Z_{14}, \end{pmatrix}, \begin{pmatrix} X_{12}X_{15}, \\ Z_{12}Z_{15}, \end{pmatrix}, -\mathbf{1} \right\rangle$
Checks, C	$\langle X_2X_5Z_6Z_9X_{10}X_{14}Z_{13}Z_4, \quad X_3X_7Z_8Z_{11}X_{12}X_{15}Z_{13}Z_4 \rangle$
Surviving Stabilizer, S	$\langle \mathrm{C}, X_1(Z_4Z_{13})X_{16}, Z_1(Z_2Z_5)(X_6X_9)(Z_{10}Z_{14})(Z_{12}Z_{15})(X_8X_{11})(Z_3Z_7)(Z_4Z_{13})Z_{16} \rangle$

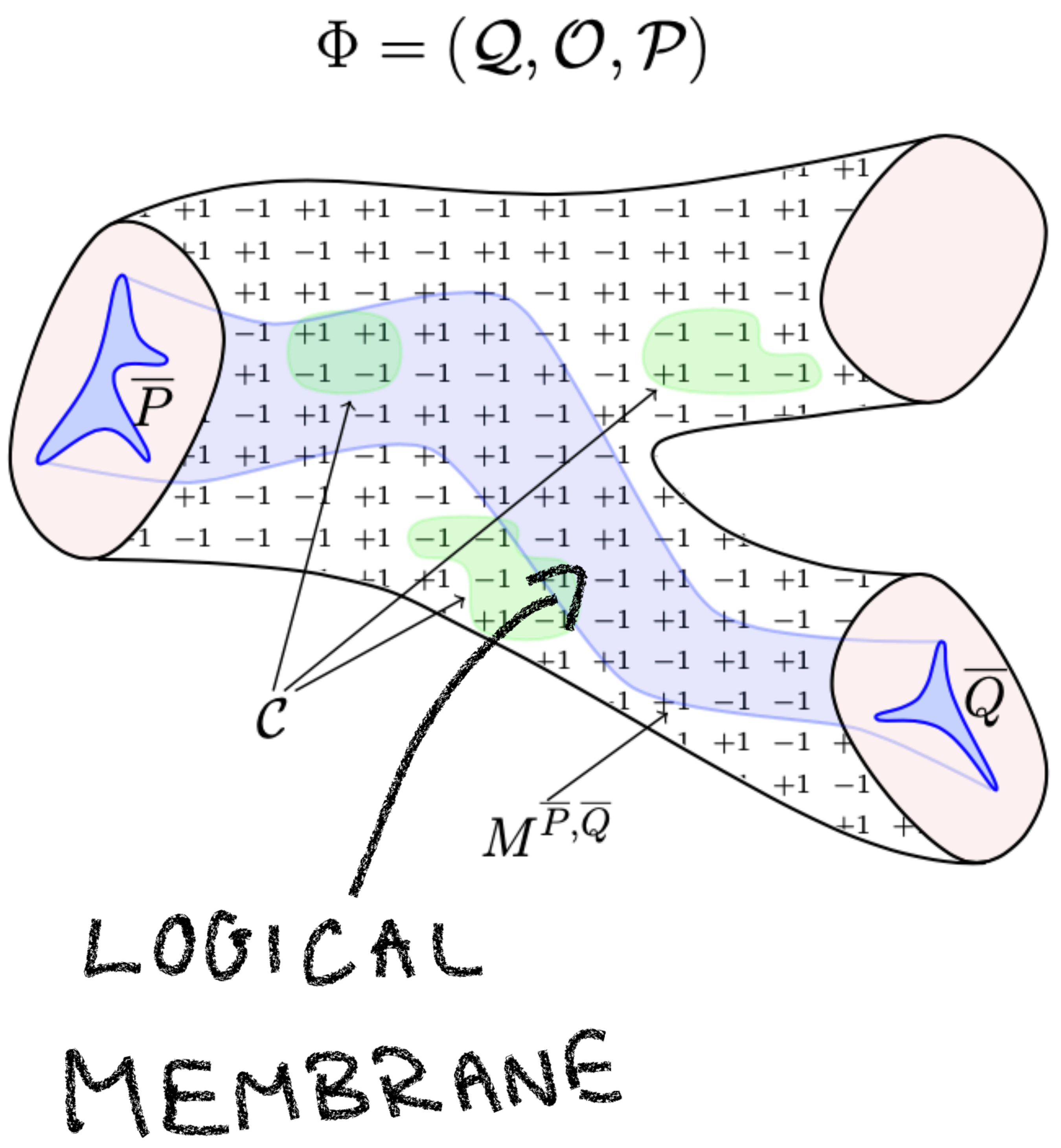
(c) Stabilizer group in the fusion network



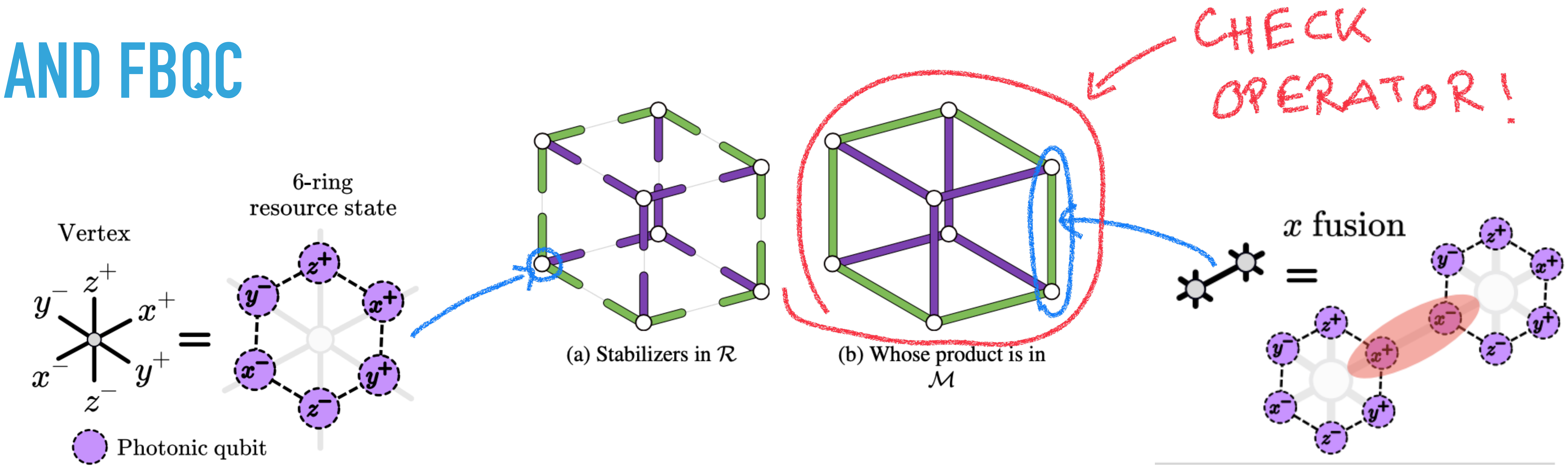
LARGE-SCALE COMPUTATION (FAULT-TOLERANT CHANNELS)

FAULT-TOLERANT LOGICAL CHANNELS

- FTLCs are a beautiful tool to describe fault-tolerant computations:
    1. Take some number of encoded states as inputs
    2. Perform some encoded operations
    3. Output some number of encoded states
- Quantum computing model-independent!
    - MBQC, FBQC, surface codes…



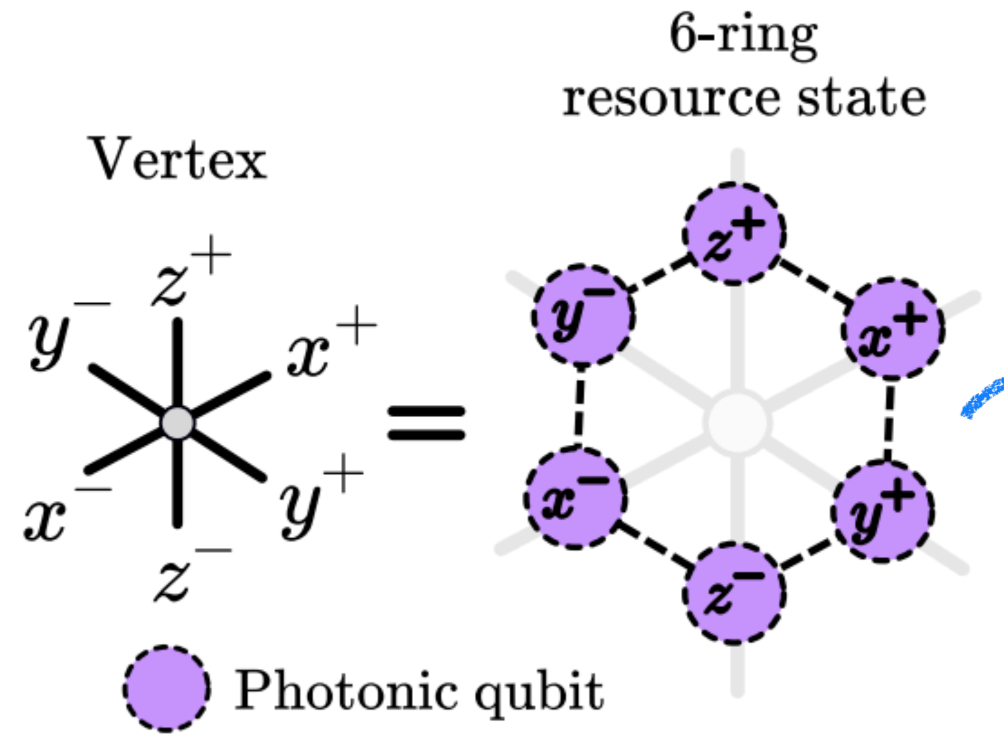
FTLCS AND FBQC

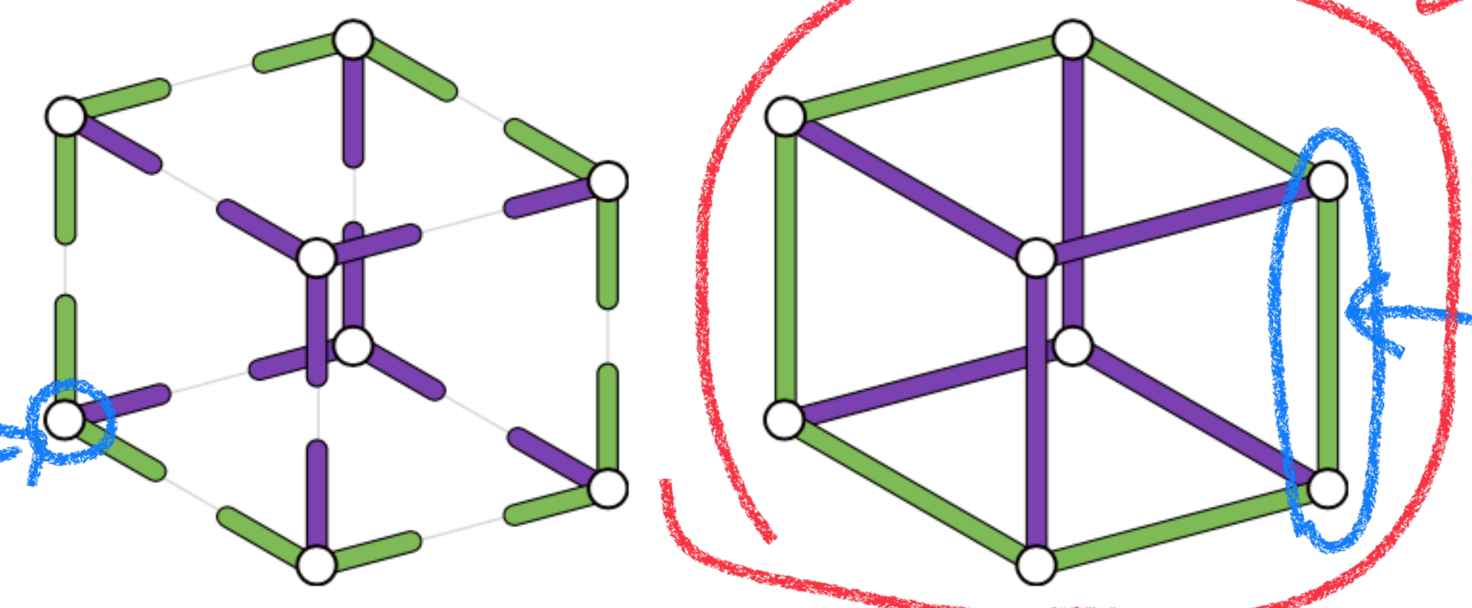


Figures from Refs 3,6

FTLCS AND FBQC

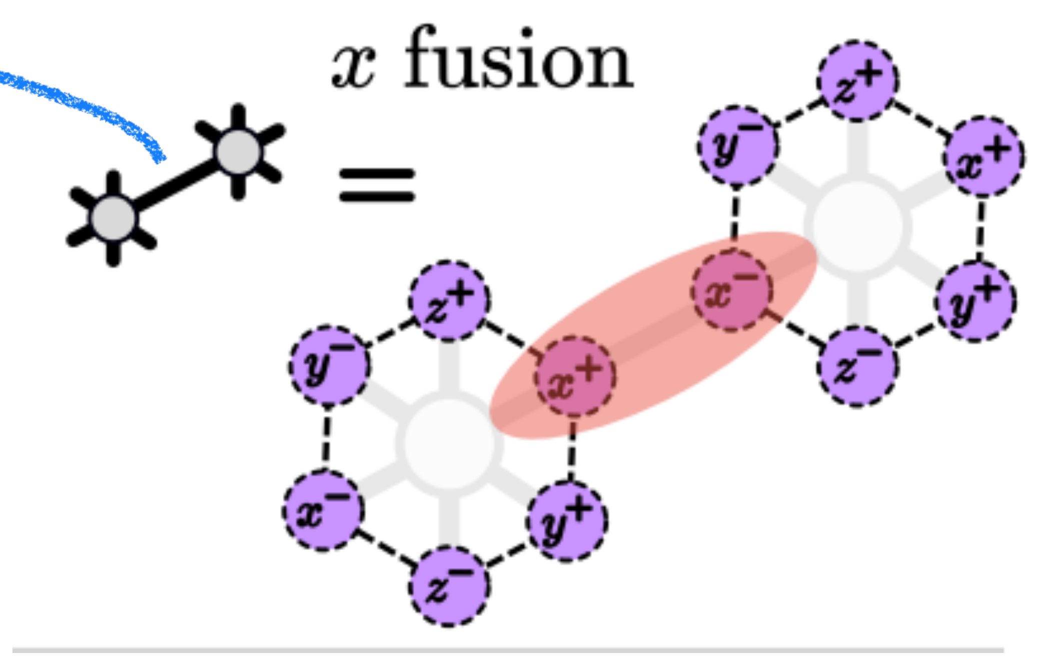
CHECK OPERATOR!

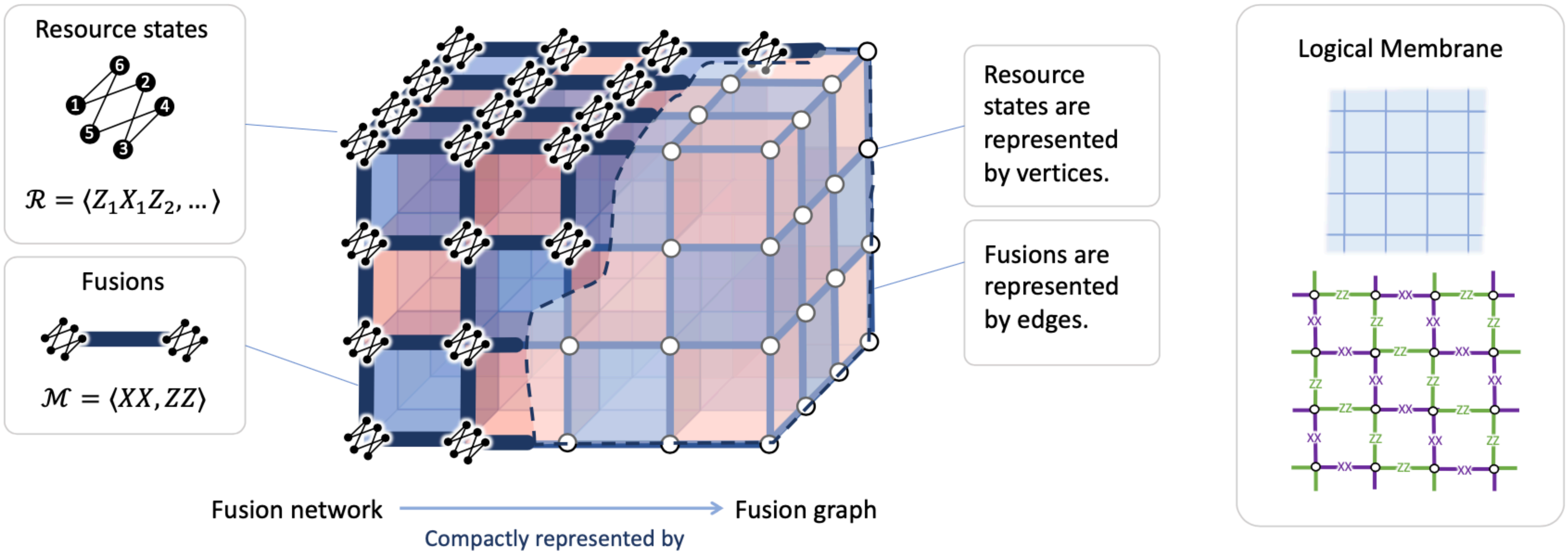




(a) Stabilizers in $\mathcal{R}$

(b) Whose product is in $\mathcal{M}$

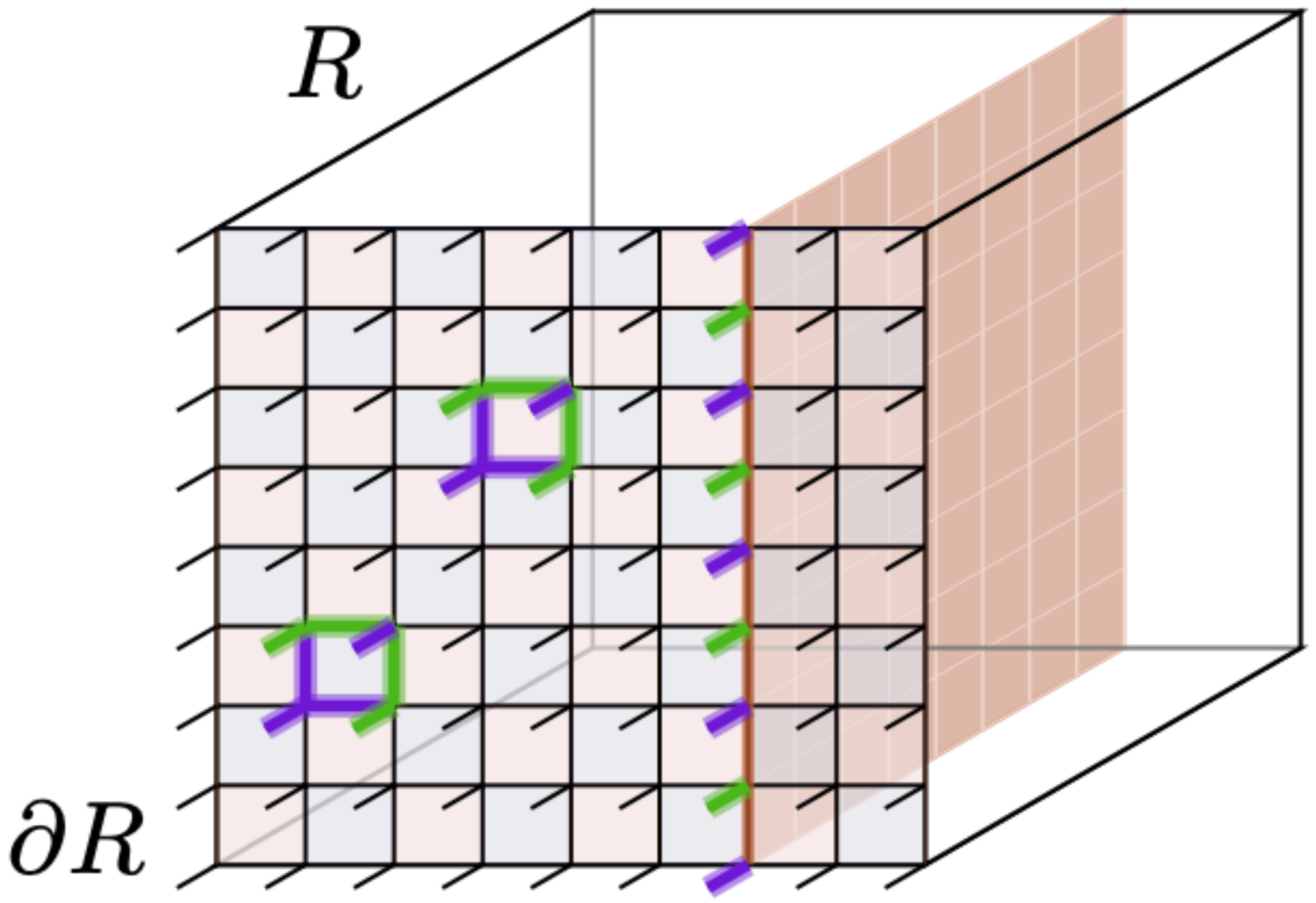


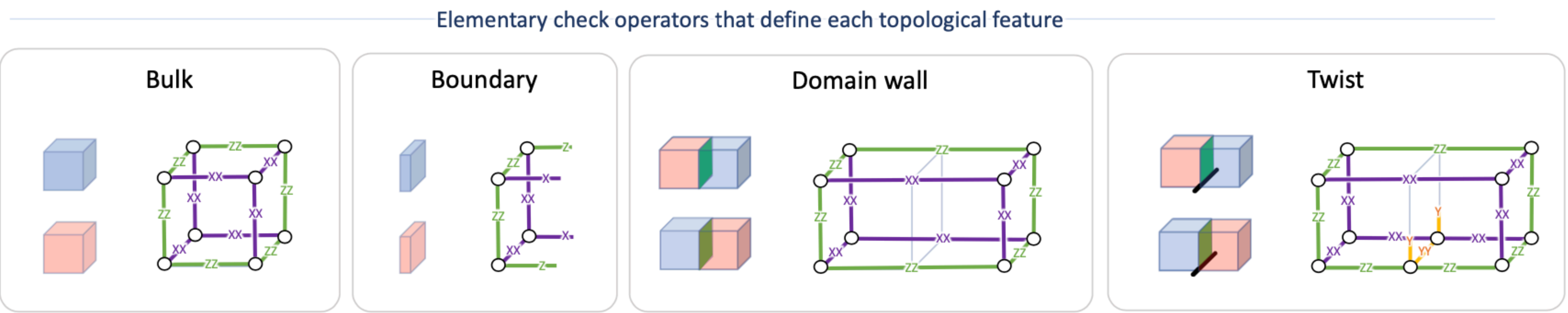


Figures from Refs 3,6

LOGICAL OPERATORS

- Logical membranes can be seen as world-sheets of logical operators
- Implement logic via topological features of the bulk:
  - Boundaries, twists, domain walls…

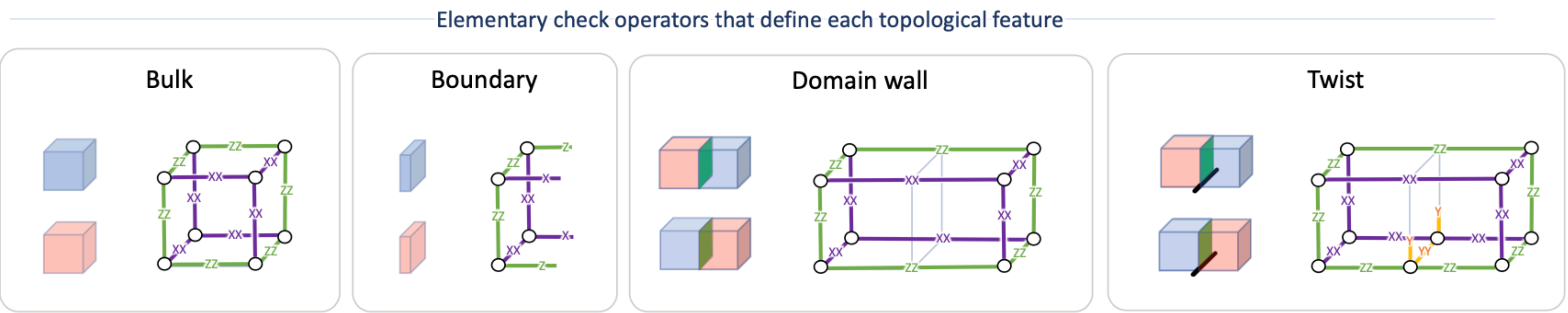


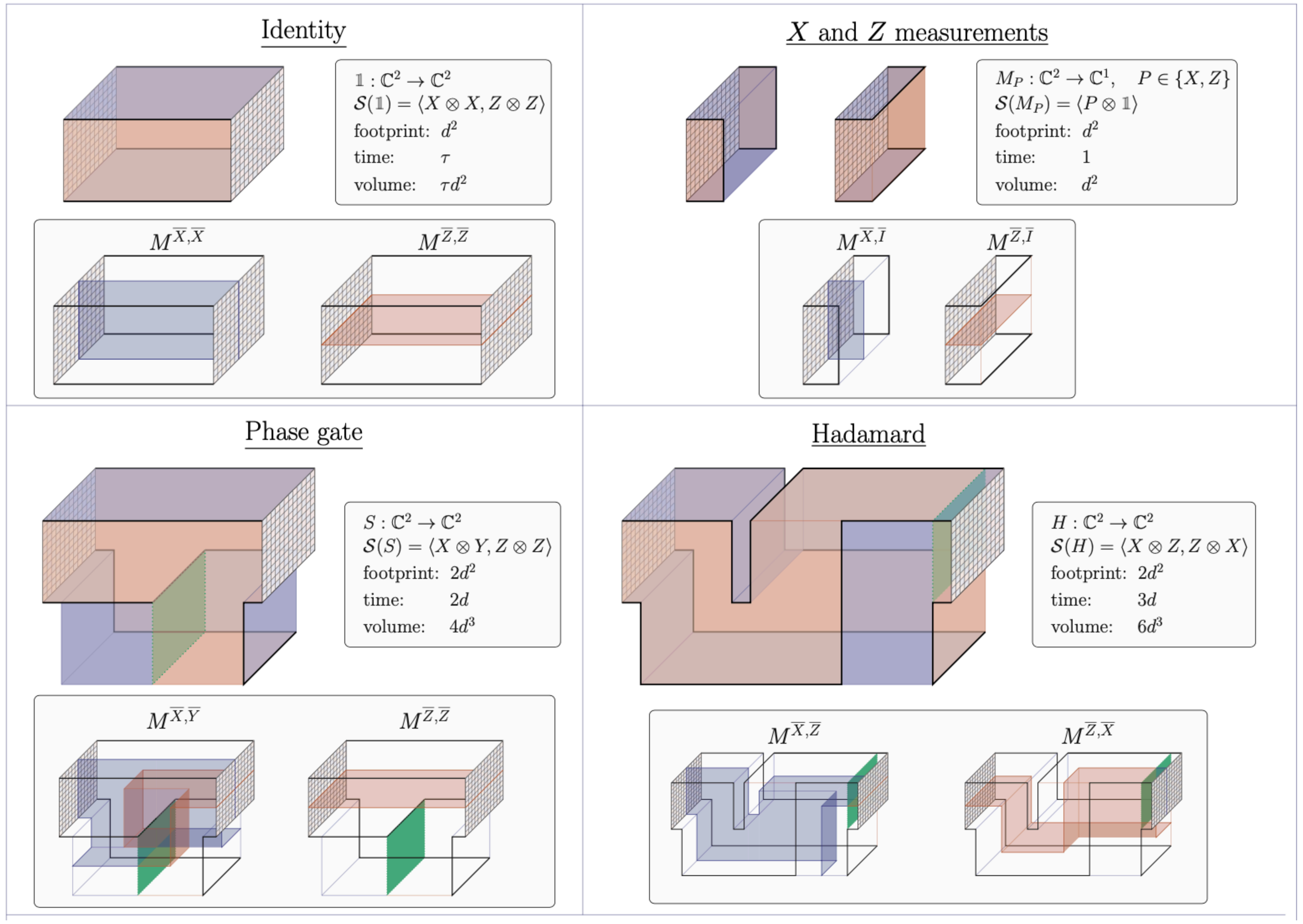


Figures from Ref 6

LOGICAL OPERATORS

- Logical membranes can be seen as world-sheets of logical operators
- Implement logic via topological features of the bulk:
  - Boundaries, twists, domain walls…

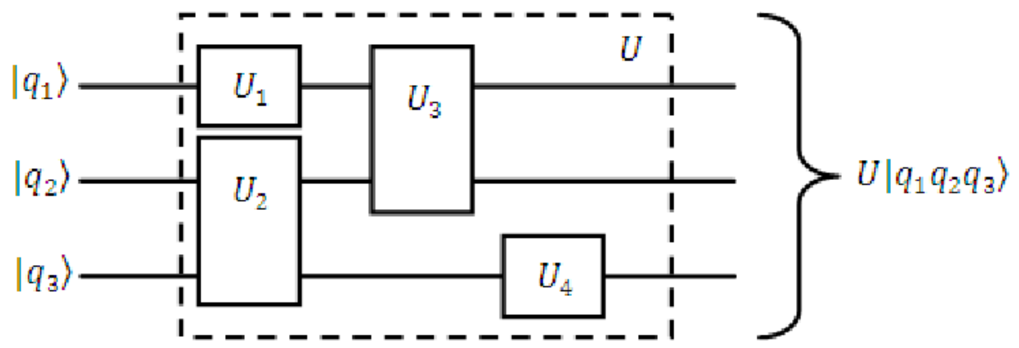


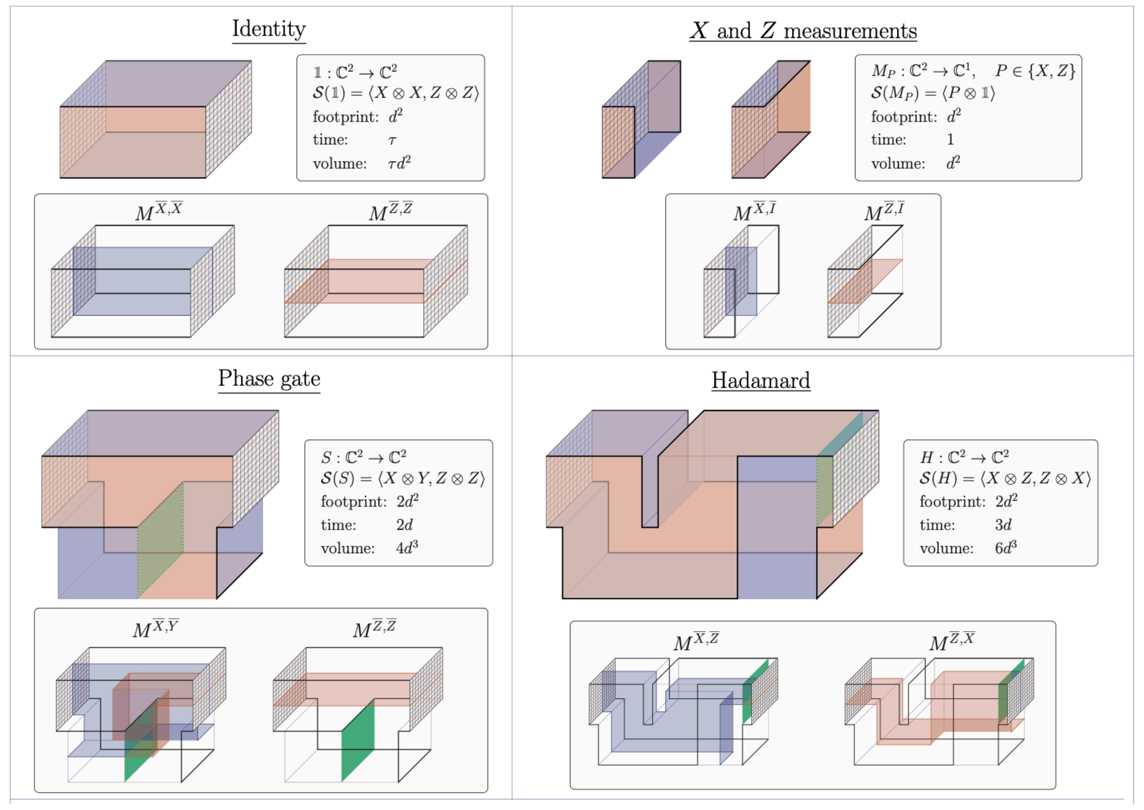


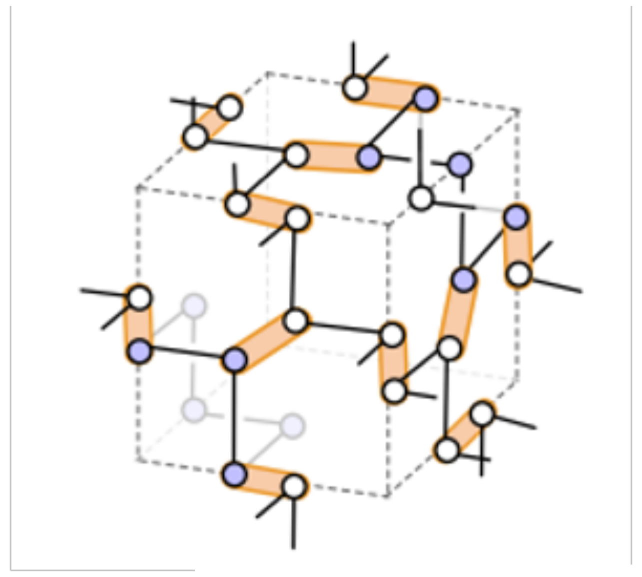
Figures from Ref 6

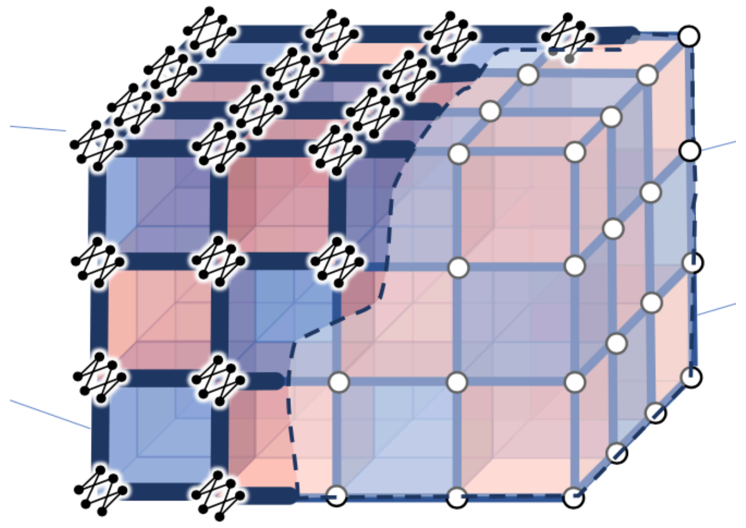
SUMMARY OF A COMPUTATION:

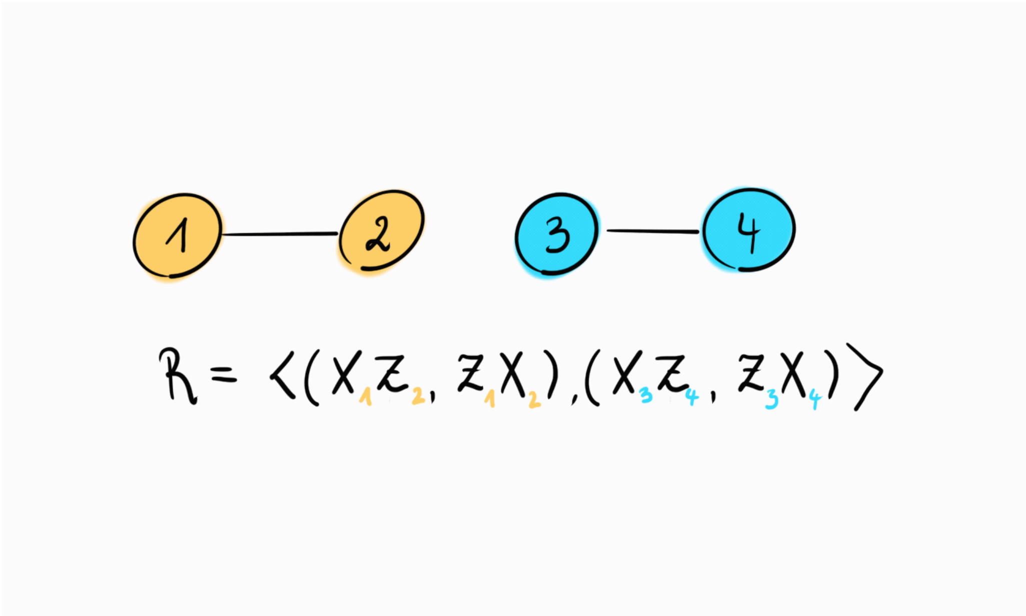
QUANTUM ALGORITHM (EG AS CIRCUIT) → LOGICAL BLOCK PICTURE → (FAULT-TOLERANT) FUSION NETWORK → MEASUREMENTS/ CORRECTIONS/OUTPUT







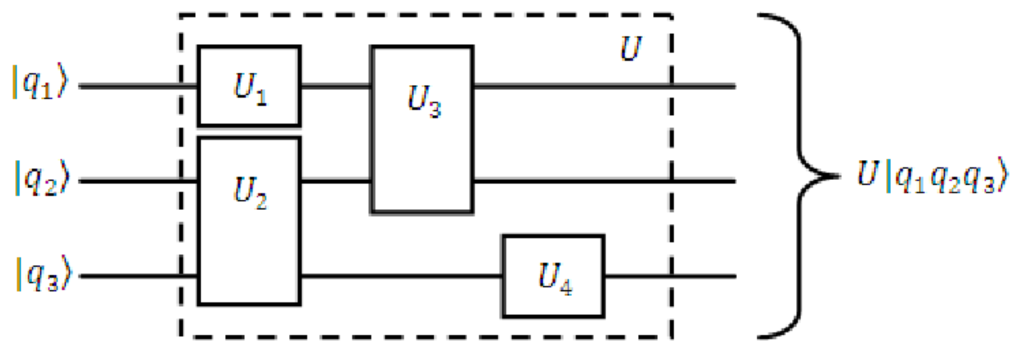


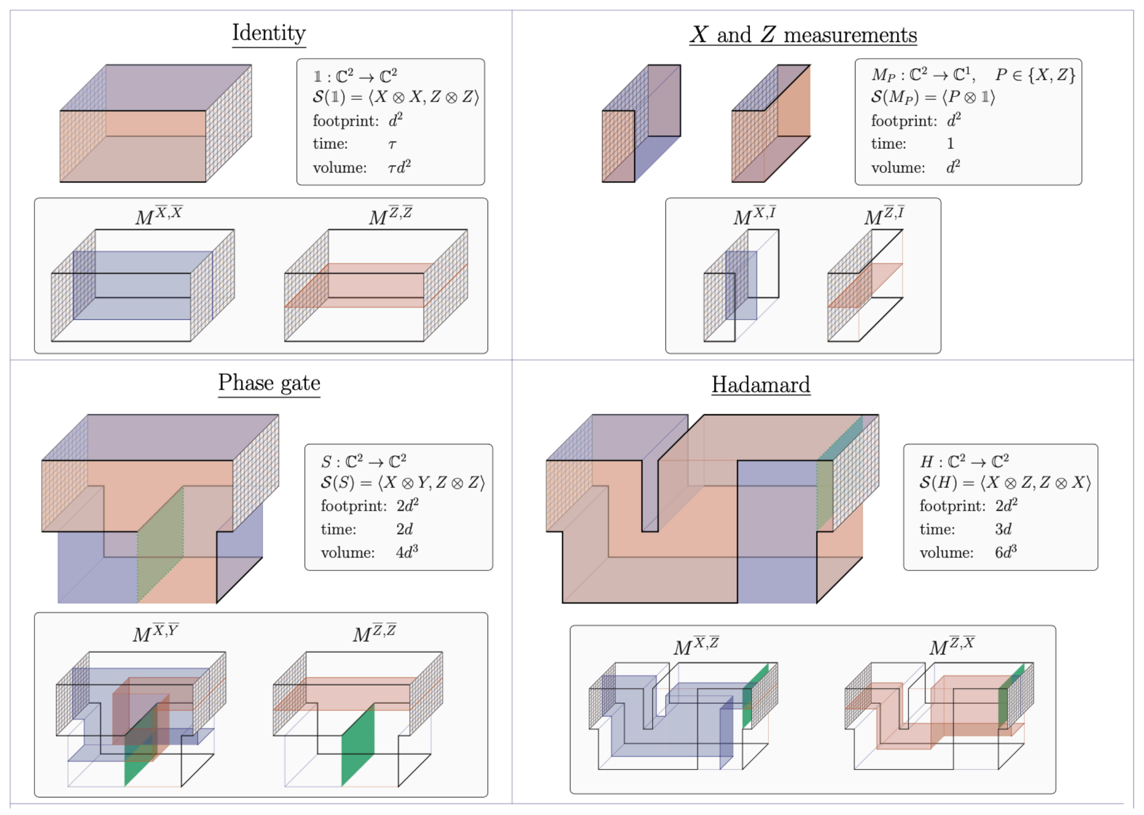


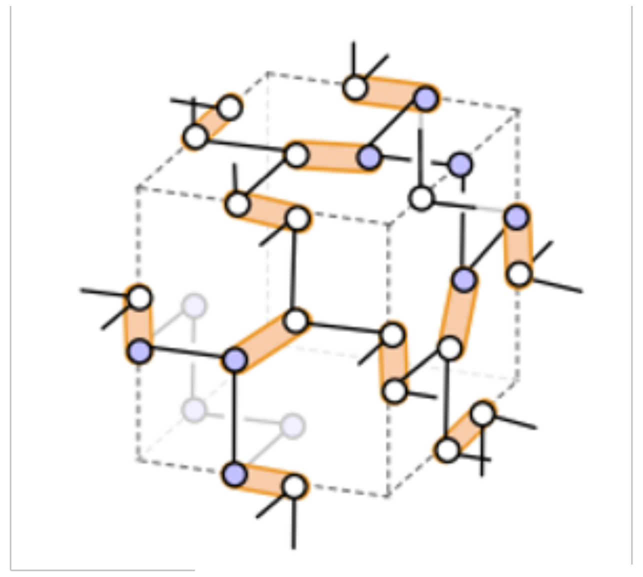
Figures from Refs 6, 8

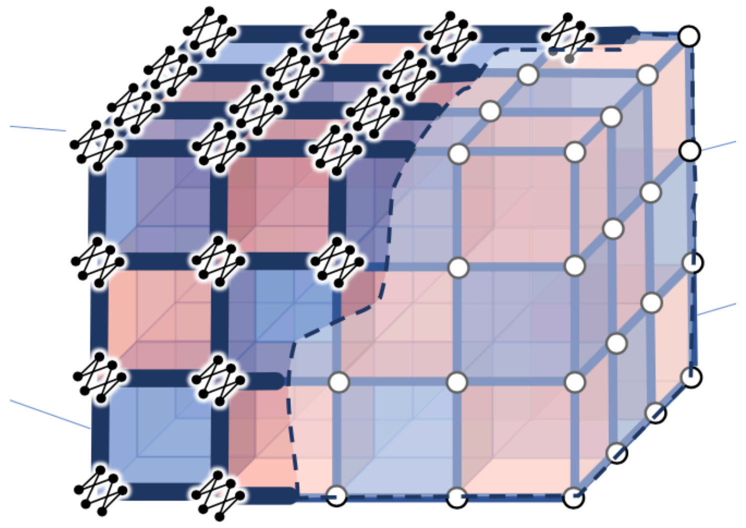
SUMMARY OF A COMPUTATION:

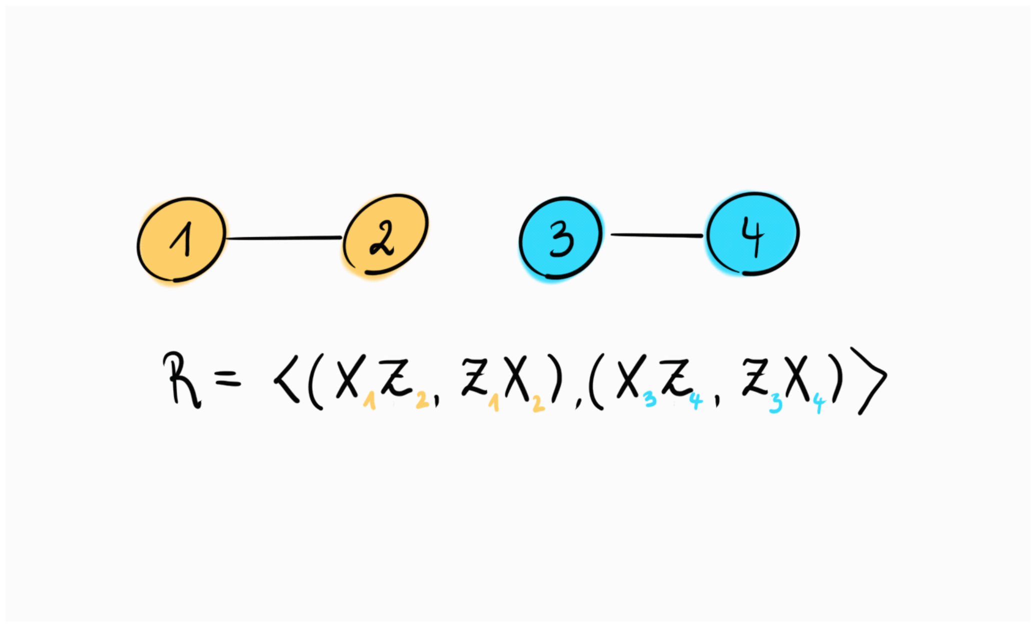
QUANTUM ALGORITHM (EG AS CIRCUIT) → LOGICAL BLOCK PICTURE → (FAULT-TOLERANT) FUSION NETWORK → MEASUREMENTS/ CORRECTIONS/OUTPUT











Also check out the blog post!
https://ievacepaite.com/2022/02/08/fusion-based-quantum-computing-1-building-blocks/

Figures from Refs 6, 8