An Introduction to Quantum Error Correction

with a particular appetite for surface code

Yiming Zhang

2024-11-14

USTC

Outline

1. Background 2
2. Surface Code 8
3. Experiment Progress 24

1. Background

1.1 Stabilizers

A compact <u>representation of quantum states</u> with (signed) Pauli operators:

$$|\Psi\rangle = \frac{1}{\sqrt{2}} |000\rangle + |111\rangle \Rightarrow \langle +XXX, +ZZI, +IZZ \rangle$$

The state $|\Psi\rangle$ is *stabilized* by the 3 independent stabilizers.

For complex states, it's tedious or even impossible to write down the full state vector. The stabilizer representation comes to rescue.

1.2 Stabilizer Sign

The sign of a stabilizer is useful for detecting errors. For a *Pauli error* E that anticommutes with a stabilizer S:

$$|\Psi\rangle = S|\Psi\rangle \xrightarrow{E} E|\Psi\rangle = ES|\Psi\rangle = -SE|\Psi\rangle$$

1.3 Stabilizer Projector

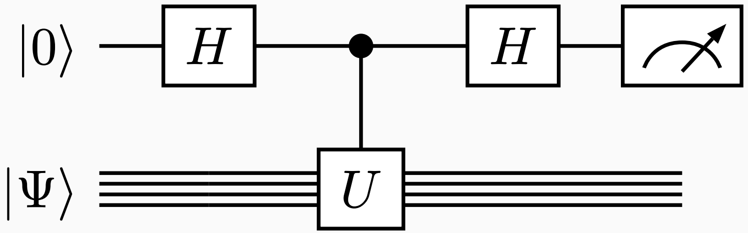


Figure 1: Operator Projector

Two perspectives of the above circuit:

1. If $|\Psi\rangle = \pm U \ |\Psi\rangle$, then it's a simple phase kickback to measure the eigenvalue, i.e. the sign of the stabilizer.
2. Otherwise, it plays the role of a projector operator $\frac{I \pm U}{2} \ |\Psi\rangle$ with the measurement backaction. Continuous errors will be digitized as Pauli errors.

1.4 Identify Errors

For the state $|\Psi\rangle = \frac{1}{\sqrt{2}}\,|000\rangle + |111\rangle \Rightarrow \langle +XXX, +ZZI, +IZZ\rangle$:

- $X_0|\Psi\rangle = \frac{1}{\sqrt{2}}\,|100\rangle + |011\rangle \Rightarrow \langle +XXX, -ZZI, +IZZ\rangle$:
- $X_1|\Psi\rangle = \frac{1}{\sqrt{2}}\,|010\rangle + |101\rangle \Rightarrow \langle +XXX, -ZZI, -IZZ\rangle$:
- $X_2|\Psi\rangle = \frac{1}{\sqrt{2}}\,|001\rangle + |110\rangle \Rightarrow \langle +XXX, +ZZI, +IZZ\rangle$:

By measuring the stabilizers, we get the so-called *syndromes*.

1.5 Logical Qubit

What is a logical qubit?

- *Error Detection*: A set of independent stabilizer generators to detect errors on the state.
- *Degree of Freedom*: #Stabilizers $\leq$ #Data Qubits

We can define a three-qubit state under the stabilizer constraints $\langle ZZI, IZZ \rangle$, then

$$|\Psi\rangle_L = \alpha\ |0\rangle_L + \beta\ |1\rangle_L$$

where

$$|0\rangle_L = \frac{1}{\sqrt{2}}\ |000\rangle + |111\rangle, |1\rangle_L = \frac{1}{\sqrt{2}}\ |000\rangle - |111\rangle,$$

And we can define the logical operator pairs $X_L = XXX, Z_L = ZII$

2. Surface Code

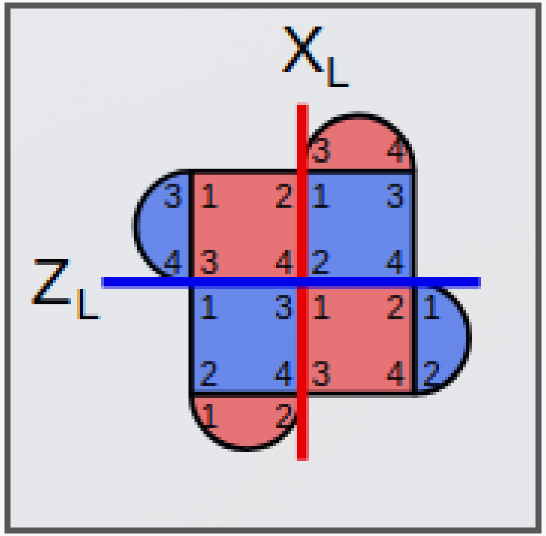


Figure 2: Surface Code

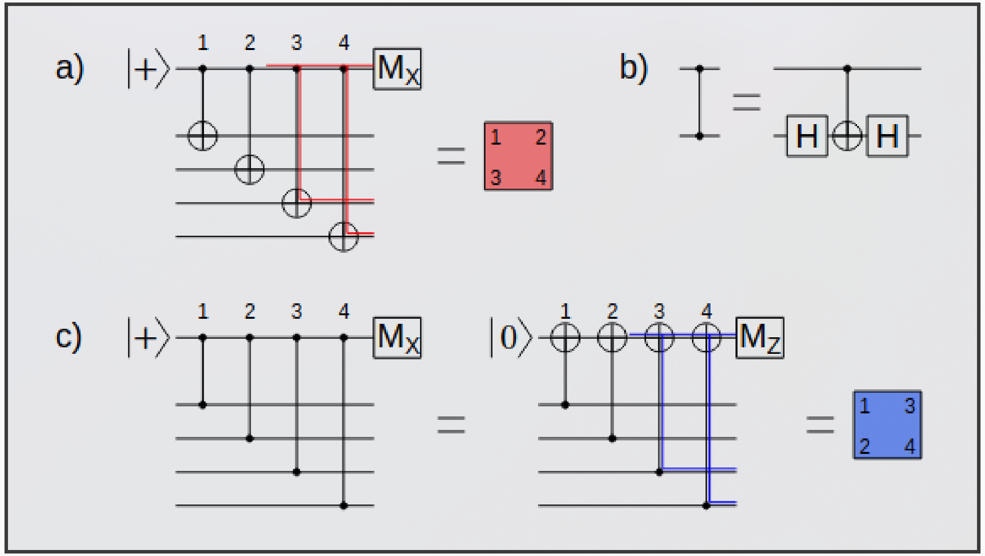


Figure 3: Stabilizer Measurements

2.2 Logical Operations

It's all about how to map the logical operators!

- I_L :

$$Z_L \xrightarrow{I_L} Z_L, X_L \xrightarrow{I_L} X_L$$

- X_L :

$$Z_L \xrightarrow{X_L} -Z_L, X_L \xrightarrow{X_L} X_L$$

- H_L :

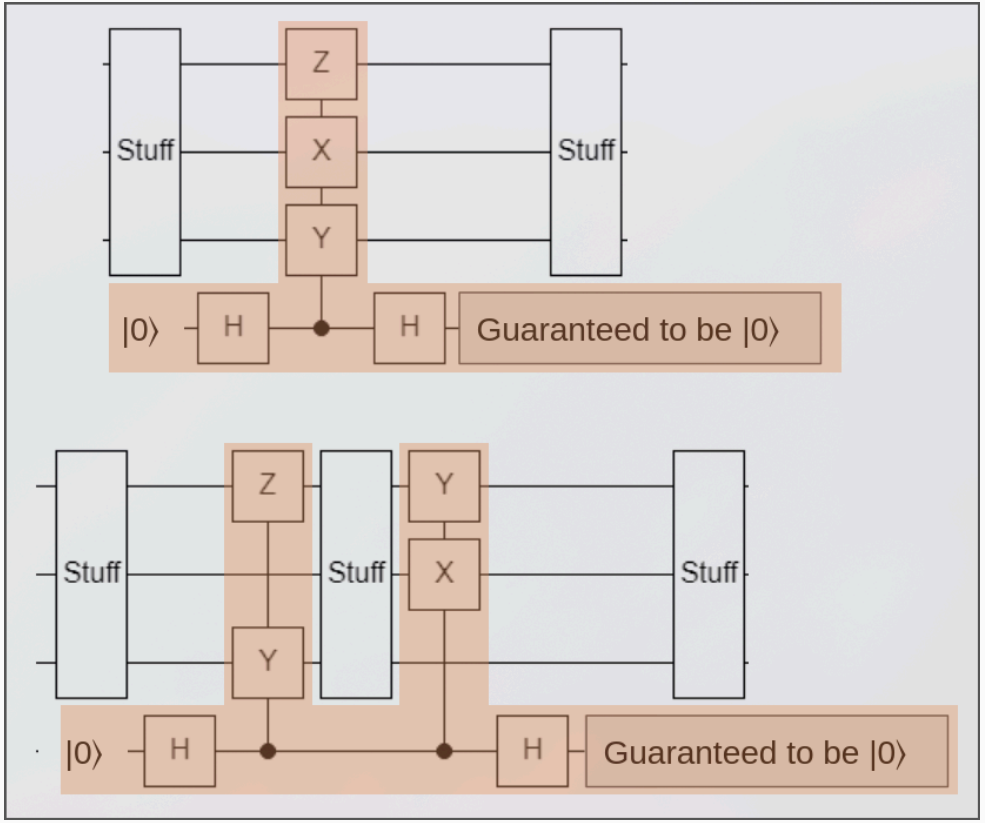
$$Z_L \xrightarrow{H_L} X_L, X_L \xrightarrow{H_L} Z_L$$

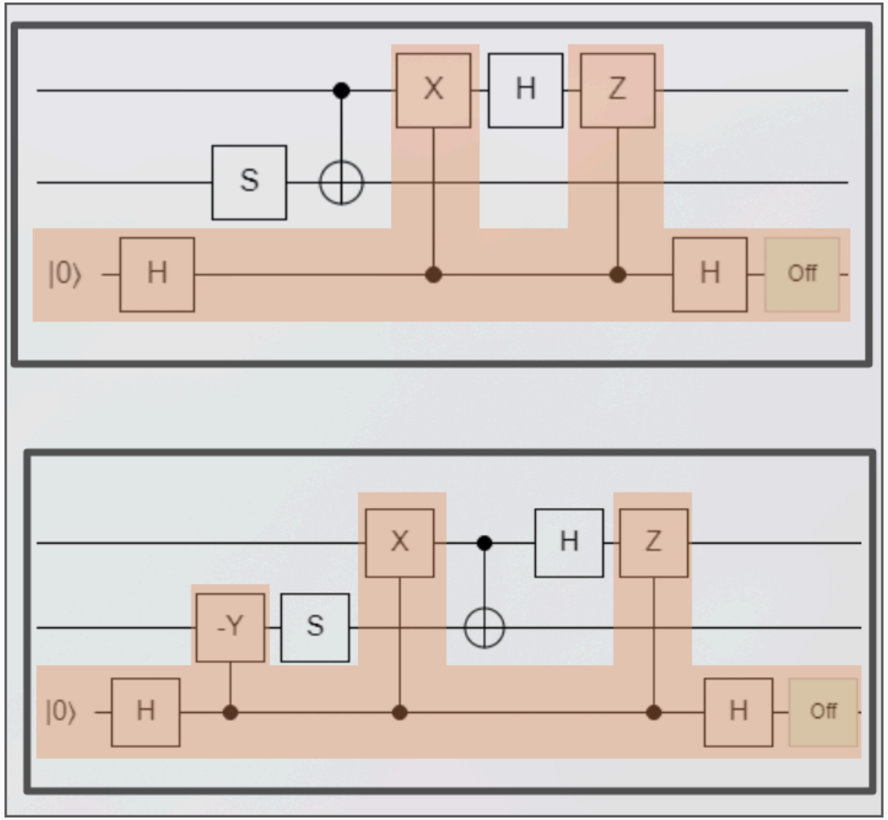
2.2 Logical Operations

- $CNOT_L$:

$$
\begin{aligned}
Z_L^C I_L^T &\overset{CNOT_L}{\longrightarrow} Z_L^C I_L^T, \\
I_L^C Z_L^T &\overset{CNOT_L}{\longrightarrow} Z_L^C Z_L^T, \\
X_L^C I_L^T &\overset{CNOT_L}{\longrightarrow} X_L^C X_L^T, \\
I_L^C X_L^T &\overset{CNOT_L}{\longrightarrow} I_L^C X_L^T
\end{aligned}
$$

Spacetime Stabilizer: another way to think about “gate as map”.





Look at the code at circuit level under a 3D picture. The logical operators can be moved by multiplying the spacetime stabilizers.

Stabilizer measurement circuits revisited:

- <u>Logical operator preserved in time</u>
- <u>Logical operator moved in space</u>: be careful about the measurements.

What is the famous *Lattice Surgery*?

<u>Nothing more than logical operator movements</u>

The simplest logical operation is "doing nothing".

Or… Everything?

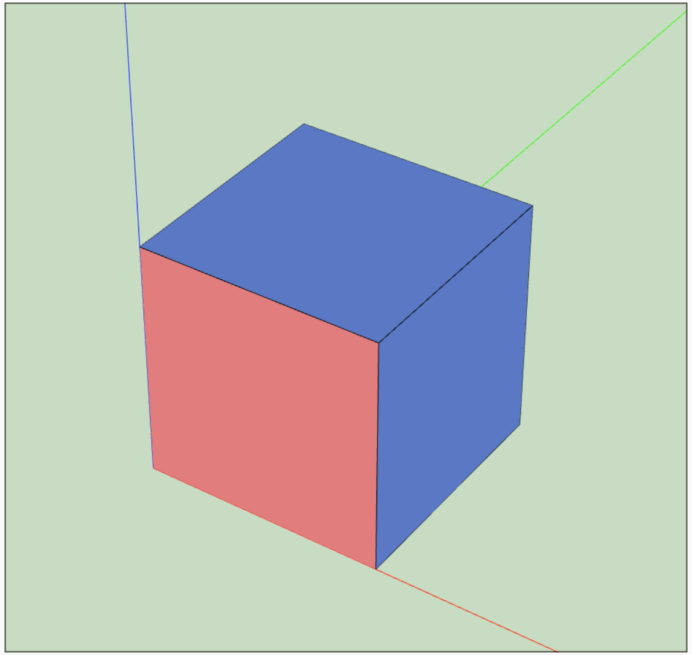


Figure 6: Logical Memory Spacetime Diagram

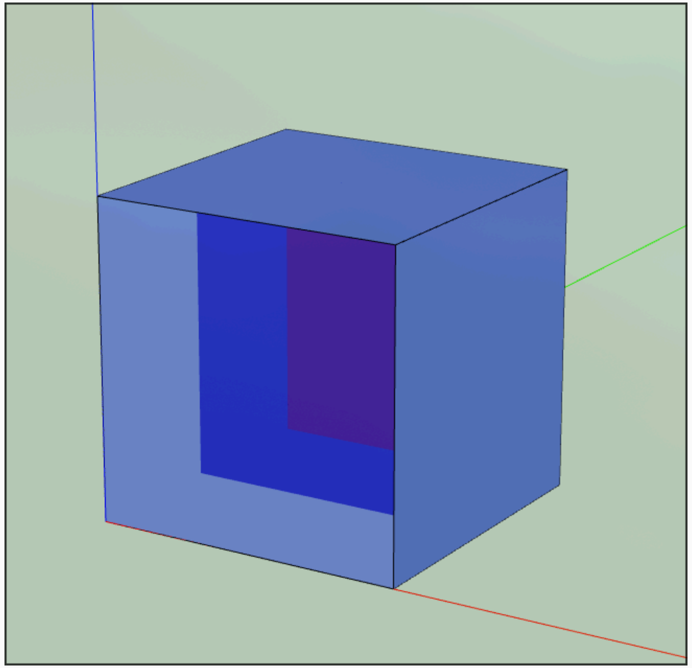


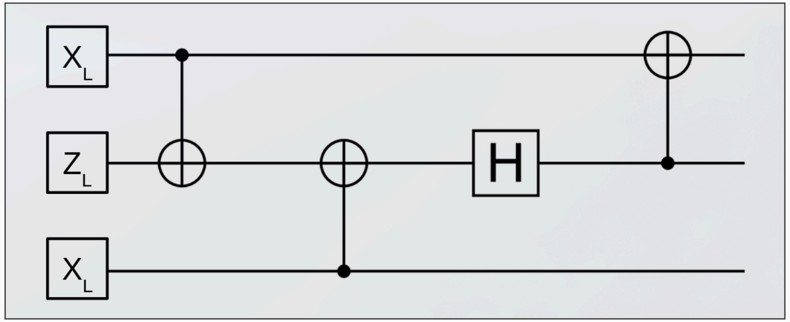
Figure 7: Logical Memory Correlation Surface

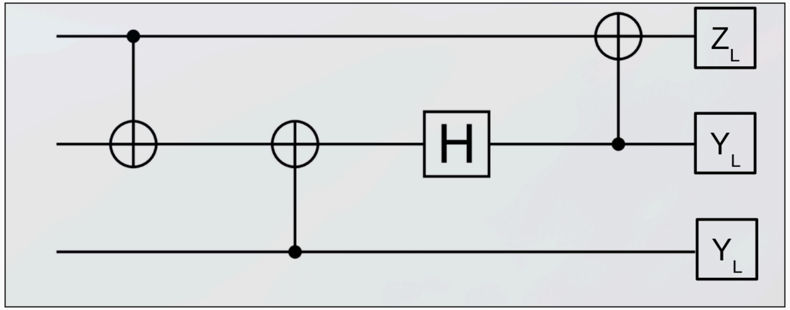
2.5 Logical Pauli Gates

$$X_L, Z_L \; \text{---} \boxed{Z_L^a} \text{---} \boxed{X_L^b} \text{---} \; (-1)^a X_L, (-1)^b Z_L$$

Track the byproduct operators in software(Logical Pauli Frame) to avoid applying logical Pauli gates on hardware.

Byproduct will change over the circuit:





2.6 Logical Hadamard

A layer of transversal H gates on data qubits will do the hadamard transition, but the boundary will be changed:

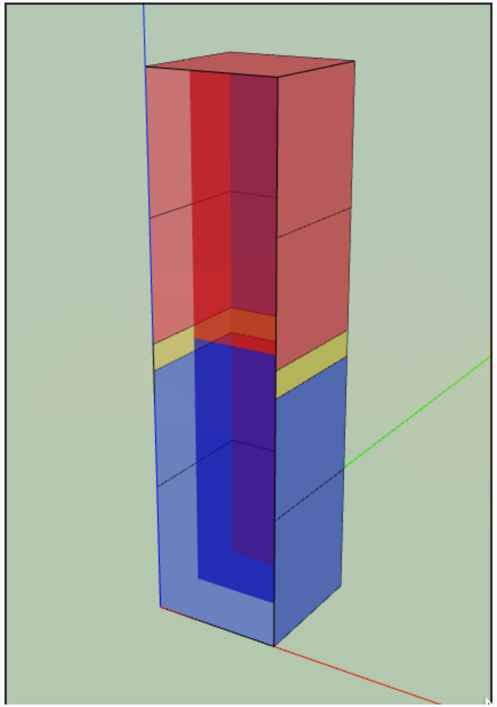


Figure 11: Temporal Hadamard

Think about that: it's all about mapping the logical operators. We can actually rotate the spacetime! (Ummm... the circuit details will change though...)

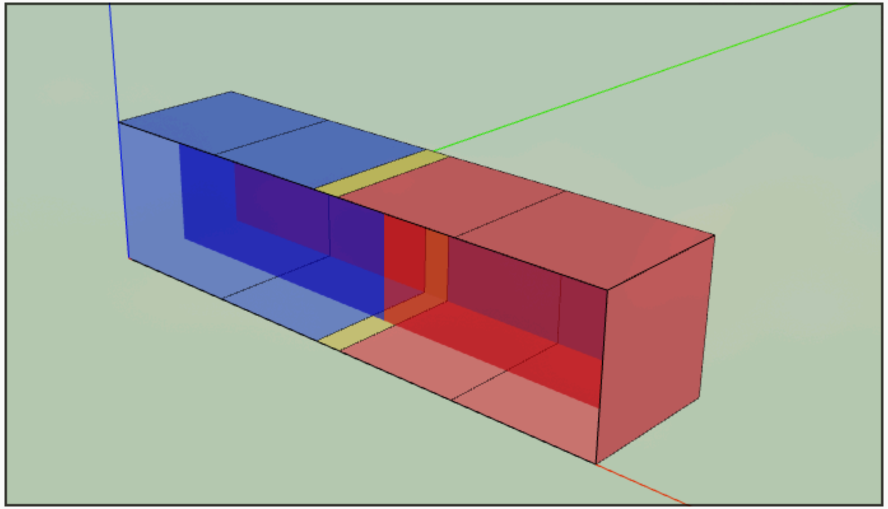


Figure 12: Spatial Hadamard

Forget about the "Lattice Surgery" for now, just build a topological structure that can move and map the logical operators correctly.

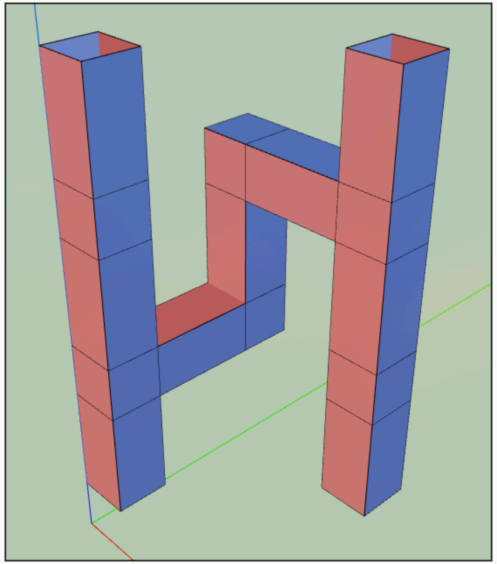


Figure 13: Logical CNOT

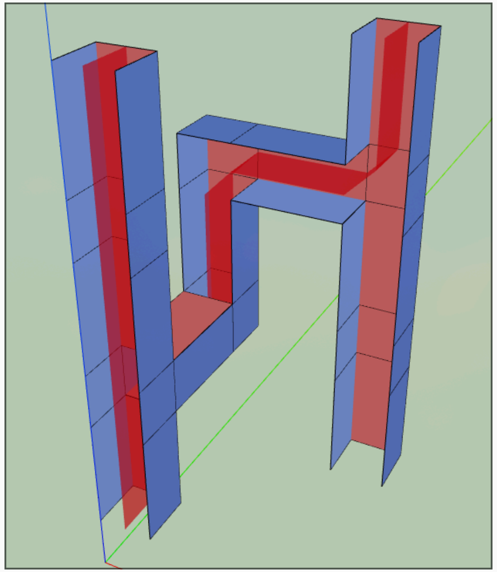


Figure 14: $XI \to XX$

Gate teleportation with inplace Y basis measurement.

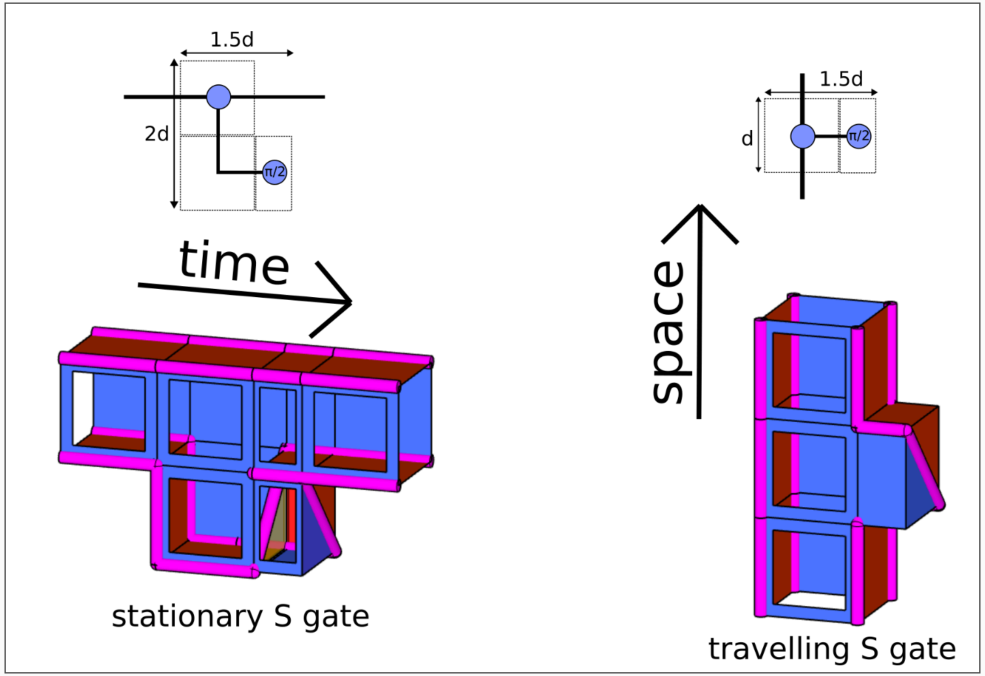


Figure 15: Logical S via gate teleportation

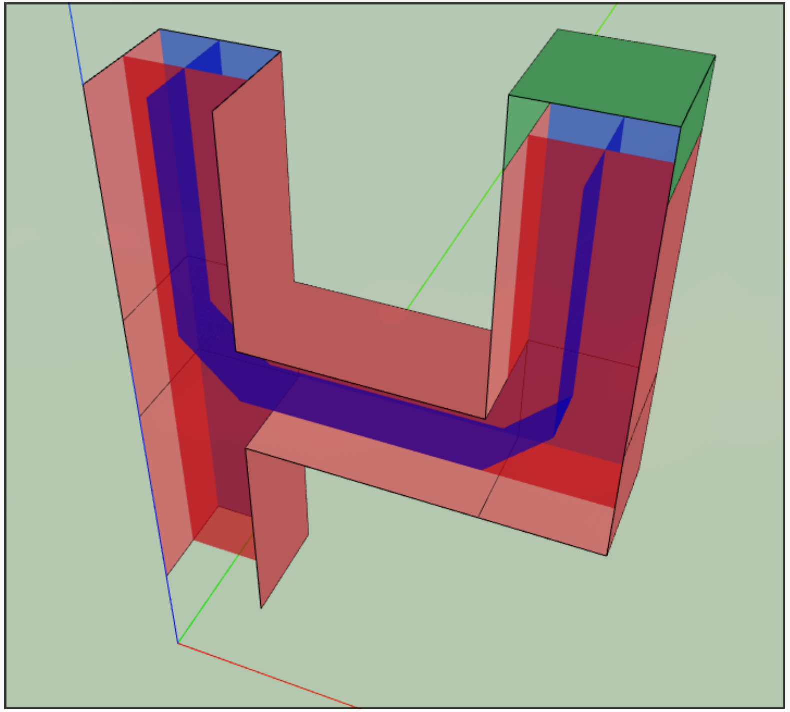


Figure 16: $X \to Y$

Non-clifford Gate: beyond the Gottesman–Knill theorem, necessary for practical universal quantum computation.

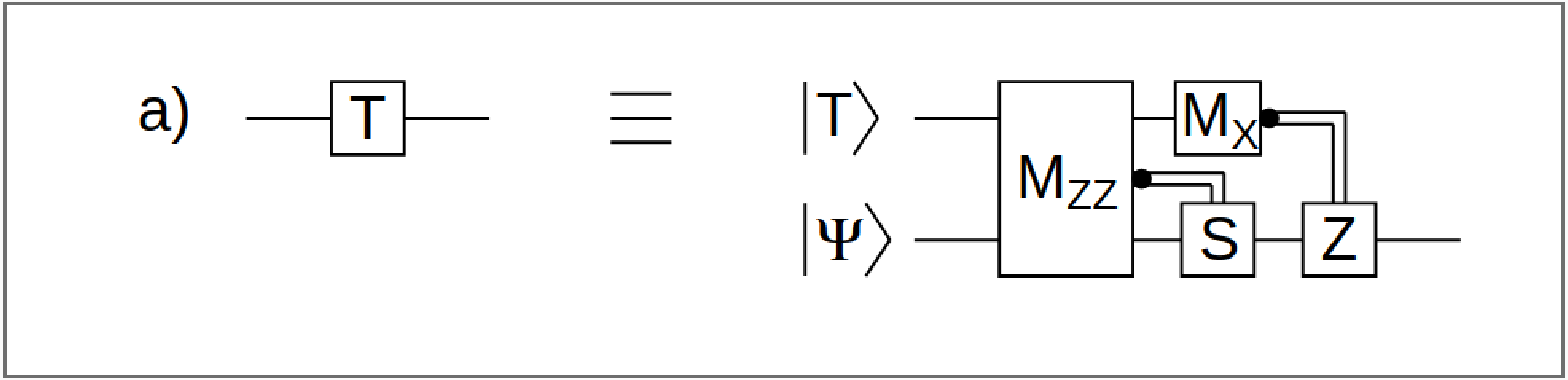


Figure 17: Logical T gate

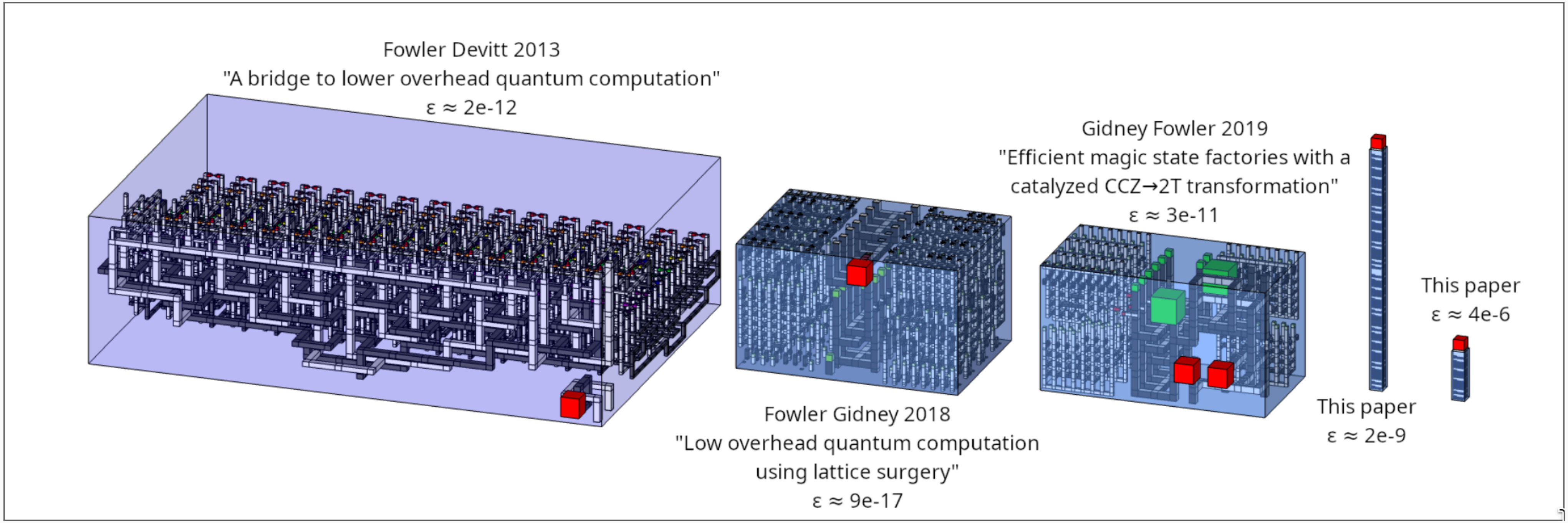


Figure 18: Magic state preparation improvements over the years

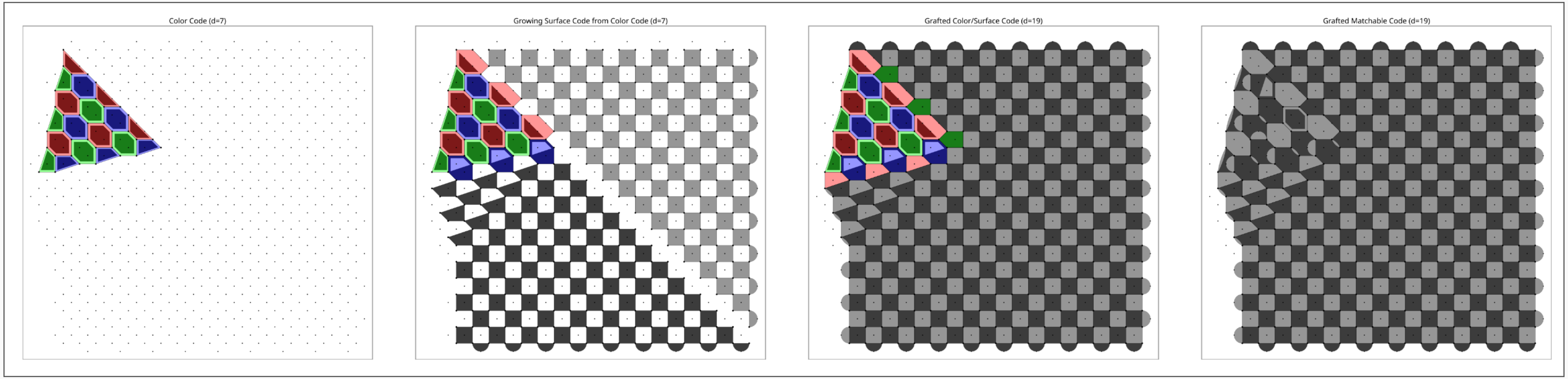


Figure 19: Magica State Cultivation

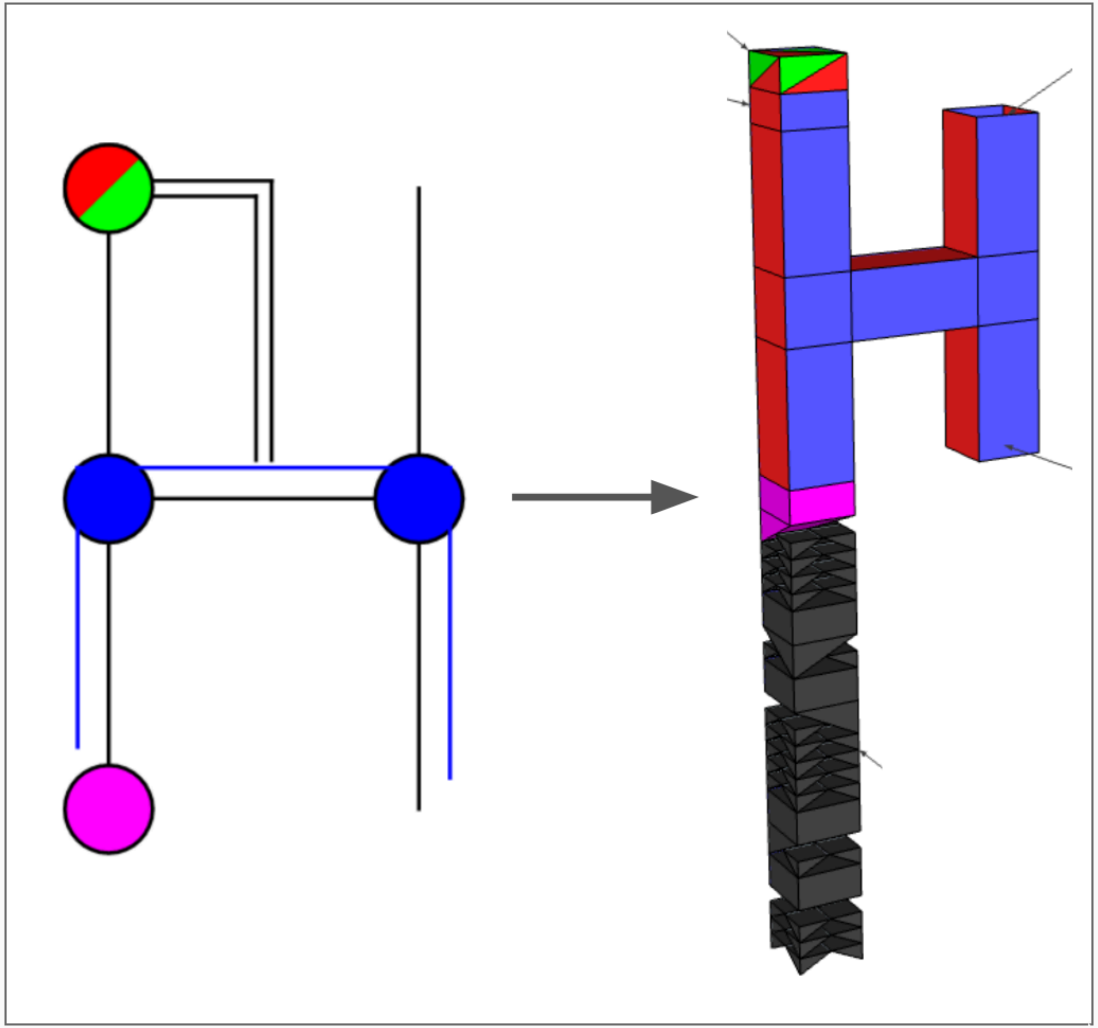


Figure 20: Logical T diagram

2.10 TQEC

- Building software tools to manage the complexity.
- It's not for good, but necessary…
- Take a look at https://github.com/tqec/tqec, it's still in early stage.

3. Experiment Progress

Neutral Atoms

Bluvstein, D., Evered, S.J., Geim, A.A. et al. Logical quantum processor based on reconfigurable atom arrays. Nature 626, 58–65 (2024).

- Connectivity: arbitray in theory, tradeoffs in practice.
- Scalability: not clear for millions of qubits.
- Gates: qubit loss problem, not suitable for repeated measurements yet.
- Clock time: $\sim ms$, extremely slow compared to superconducting qubits. This is important for practical large-scale FT quantum algorithm.

Trapped Ions

High-fidelity teleportation of a logical qubit using transversal gates and lattice surgery

Superconducting Qubits

Quantum error correction below the surface code threshold

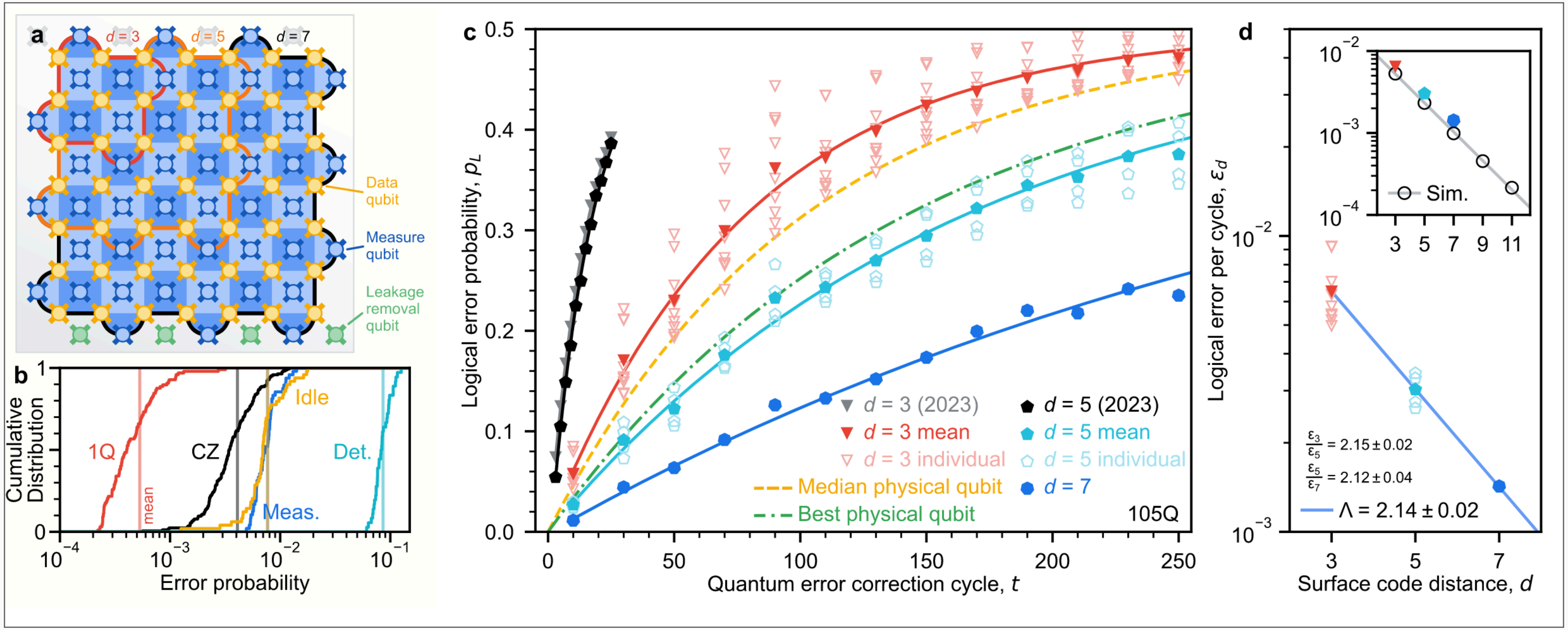


Figure 21: Google d=3,5,7 surface code