

Lecture 2: Spacetime cost in early FTQC

Nobuyuki Yoshioka
UTokyo



Agenda

Lecture 1 : Spacetime cost in NISQ

Introduction	15min
Introduction to QEM	30min
Fundamental limit of QEM	40min

Lecture 2 : Spacetime cost in early FTQC

Overview of QEC	20min
Decoders for surface code	20min
Decoder confidence for postselection	20min
Syndrome-aware estimation	20min

Lecture 3 : Spacetime cost in FTQC

Overview of resource estimation	10min
Level 1 estimation, surface code	25min
Level 2 estimation, surface code	20min
Partial fault tolerance	30min

Overview of QEC

Classical linear codes

Let C be an (n, k, d) code, and let B_C be the set of all its codewords (the code space).

The kernel of parity check matrix $H \in \mathbb{F}_2^{(n-k) \times n}$ gives the code space:

$$Hx = 0, \forall x \in B_C.$$

e.g. Check matrix for 3-bit repetition code

$$H = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad H \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = (0,0)^T \quad H \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = (0,0)^T$$

check parity of
neighboring bits

Codewords

$$H \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = (0,1)^T \quad H \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = (1,0)^T$$

Not codewords

\leftrightarrow Simultaneous eigenspace of $\{Z_1Z_2, Z_2Z_3\}$

Stabilizer group

Definition (Commutative subgroup)

A subset $H \subseteq G$ of a group G is a commutative subgroup of G if :

- (1) H is itself a subgroup,
- (2) $h_1 h_2 = h_2 h_1$ for any $h_1, h_2 \in H$.

Definition (Stabilizer group)

A commutative subgroup of the n -qubit Pauli group P_n that does not contain $-I^{\otimes n}$ is called a stabilizer group, denoted by \mathcal{S} .

$\{+II, +ZZ\}$ **OK**

$\{+II, +ZZ, +XX, -YY\}$ **OK**

$\{+II, +ZZ, +XX, +YY\}$ **NG** (not a group)

$\{+II, +iZZ, -II, -iZZ\}$ **NG** ($\pm I^{\otimes 2}$)

Stabilize

Definition (Stabilize)

An operator S stabilizes $|\psi\rangle$ when $S|\psi\rangle = |\psi\rangle$.

Moreover, the set of quantum states stabilized by every operator in a stabilizer group \mathcal{S} is defined by

$$\mathcal{H}_{\mathcal{S}} := \left\{ |\psi\rangle \mid S|\psi\rangle = |\psi\rangle, \forall S \in \mathcal{S}_n \right\}$$

e.g. Bell state

$$\mathcal{S} = \{+II, +XX, +ZZ, -YY\} \leftrightarrow \mathcal{H}_{\mathcal{S}} = \left\{ \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \right\}$$

$$\mathcal{S} = \{+II, +XX, -ZZ, +YY\} \leftrightarrow \mathcal{H}_{\mathcal{S}} = \left\{ \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \right\}$$

e.g. Bitflip code space ($d = 3$)

$$\mathcal{S} = \{+II, +ZZI, +IZZ, +ZIZ\} = \langle ZZI, IZZ \rangle \leftrightarrow \mathcal{H}_{\mathcal{S}} = \{|000\rangle, |111\rangle\}$$

Stabilizer generator and stabilizer codes

Definition (Stabilizer generator):

When a stabilizer group \mathcal{S}_n has order $|\mathcal{S}_n| = 2^{n-k}$, there exists a set of $n - k$ Pauli operators $\{g_1, \dots, g_{n-k}\}$ that generate the group:

$$\mathcal{S}_n = \langle g_1, \dots, g_{n-k} \rangle = \left\{ \prod_i g_i^{b_i} \mid b_i \in \{0, 1\} \right\}$$

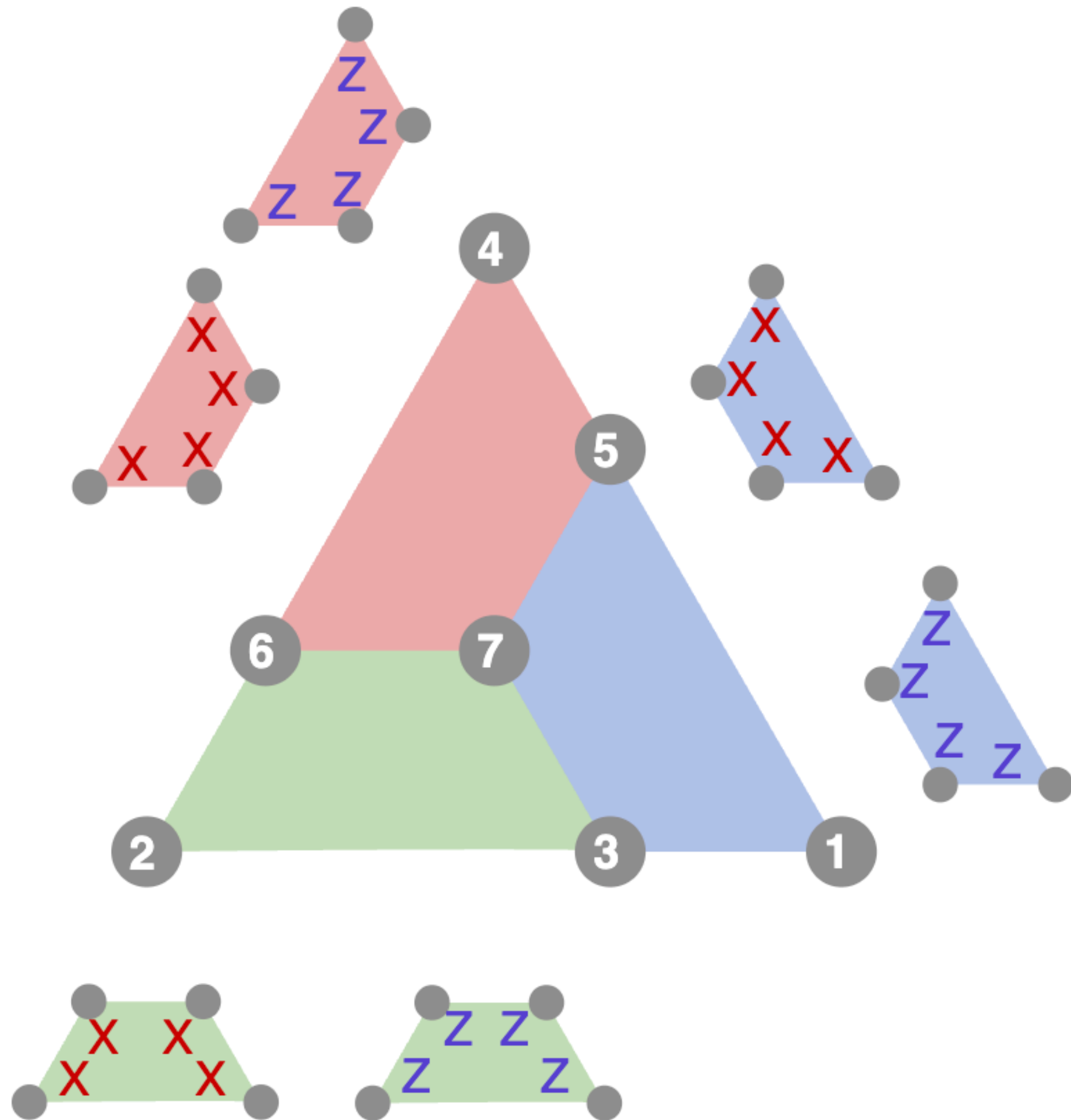
Definition (Stabilizer code)

A stabilizer code $C_{\mathcal{S}}$ is a code whose code space $\mathcal{H}_{\mathcal{S}}$ is spanned by states stabilized by a stabilizer group \mathcal{S} .

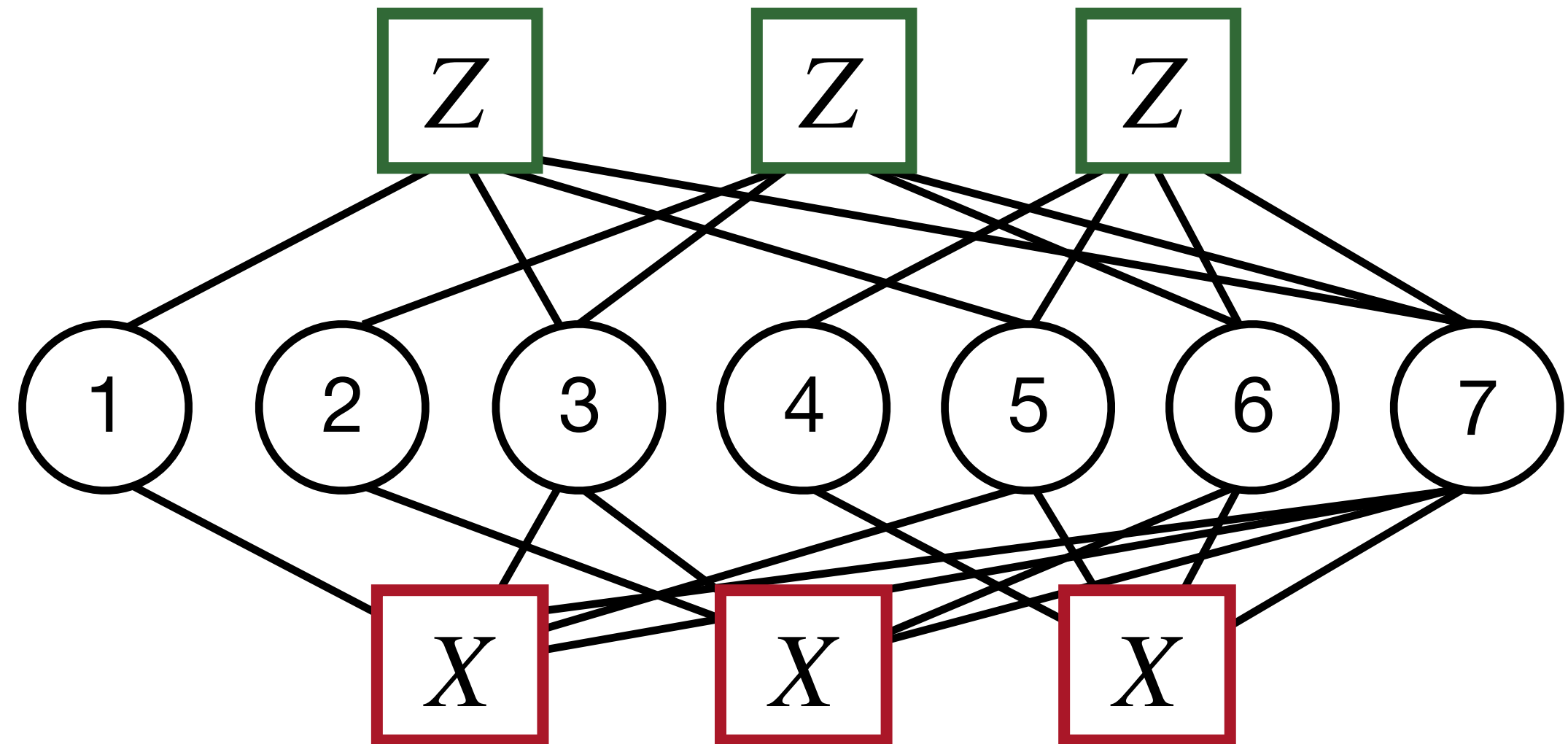
$$\mathcal{H}_{\mathcal{S}} = \left\{ |\psi\rangle \mid S|\psi\rangle = |\psi\rangle, \forall S \in \mathcal{S} \right\}$$

Stabilizers in Steane code

Graphical expression



Tanner graph



Check matrix

Bit representation of stabilizer support location

Indicates that the code is CSS

$$H_X = H_Z = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

What we want for QECC

- 1. High threshold**
- 2. Low-density parity check**
- 3. Fast decoder**
- 4. Fault-tolerant logical operation**
- 5. High encoding rate**
- 6. Hardware friendly**

What we want for QECC

1. High threshold

The whole point of QEC is to reduce error exponentially with distance.

However, since syndrome extraction is noisy, there is a break-even error rate p_{th} :

If $p > p_{\text{th}}$, logical error worsens,

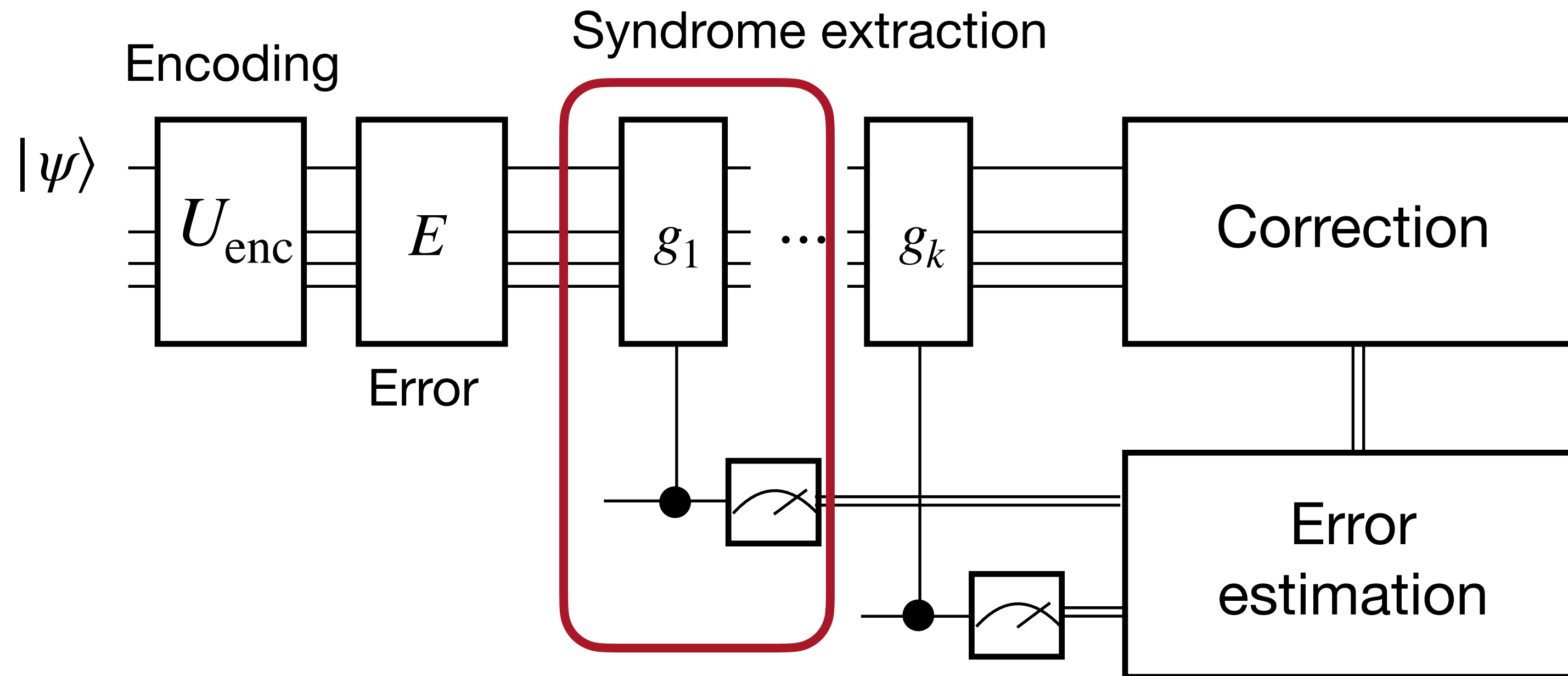
If $p < p_{\text{th}}$, logical error improves.

For instance, threshold of $p_{\text{th}} = 10^{-10}$ would be difficult to benefit from in reality.

What we want for QECC

2. Low-density parity check

Keep the syndrome extraction circuit small
(= construct fault-tolerant quantum circuit)



Gate depth typically depends on weight of stabilizer

What we want for QECC

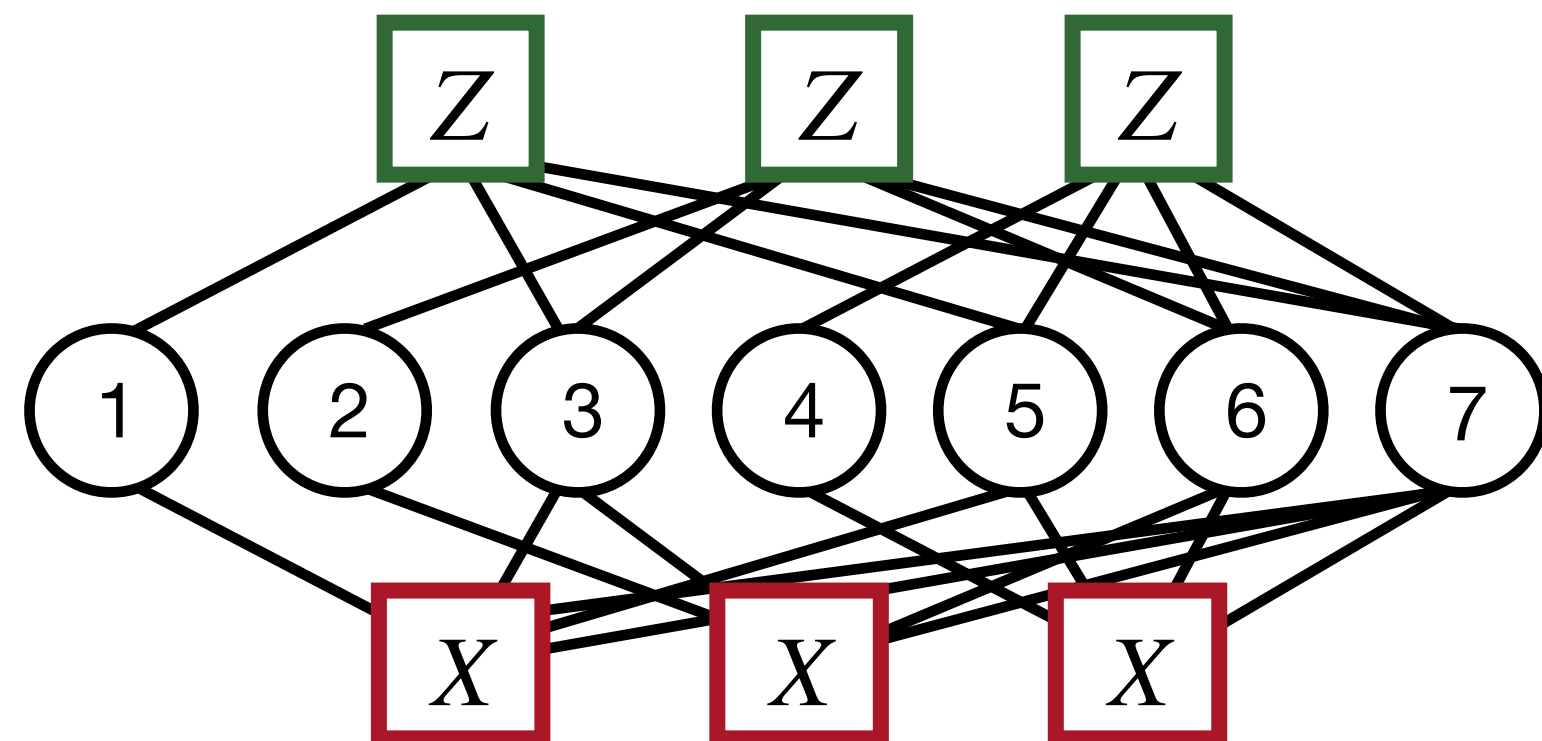
3. **Fast decoder** especially real-time decoder is required.

Otherwise we have backlog problem: the syndrome meas. data piles up exponentially with depth (Error needs to be corrected faster than the next error happens)

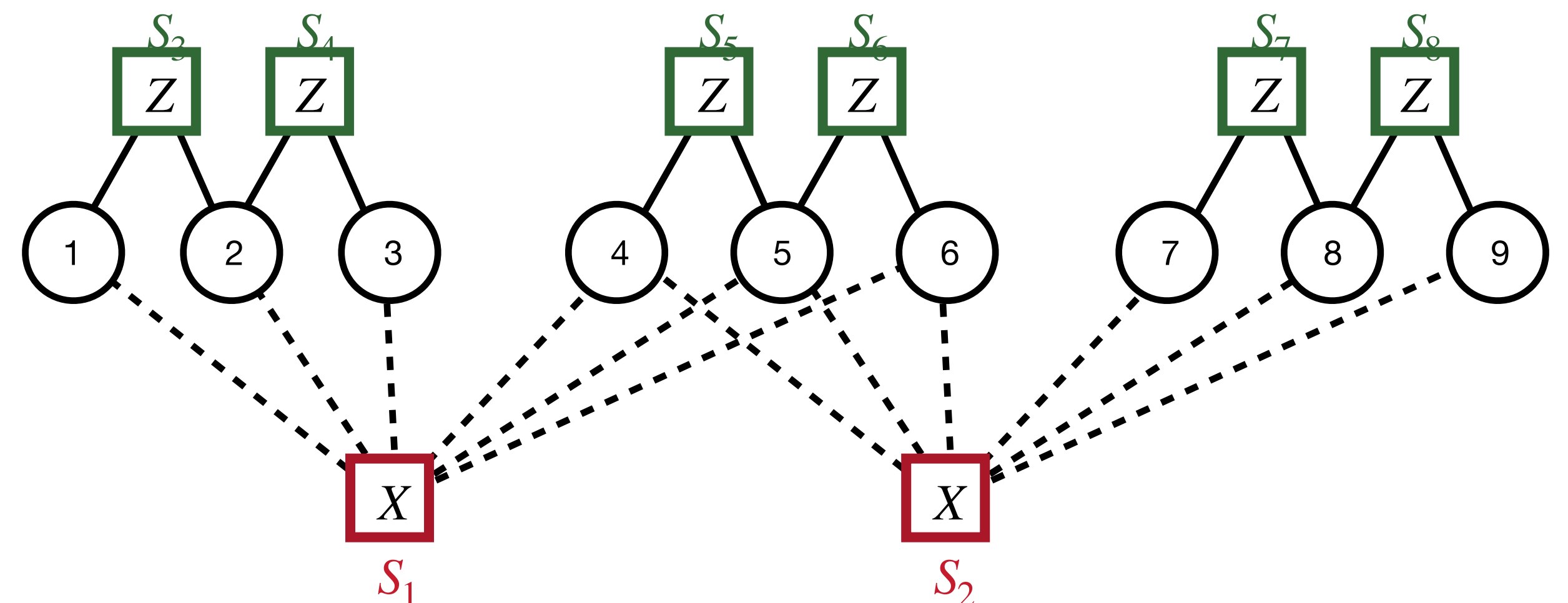
e.g. **Matchable codes**

if data \circ participates in at most two \square_Z and two \square_X ,
we can use the $O(n)$ -time decoders called matching decoders e.g. Sparse blossom

Not matchable



Matchable

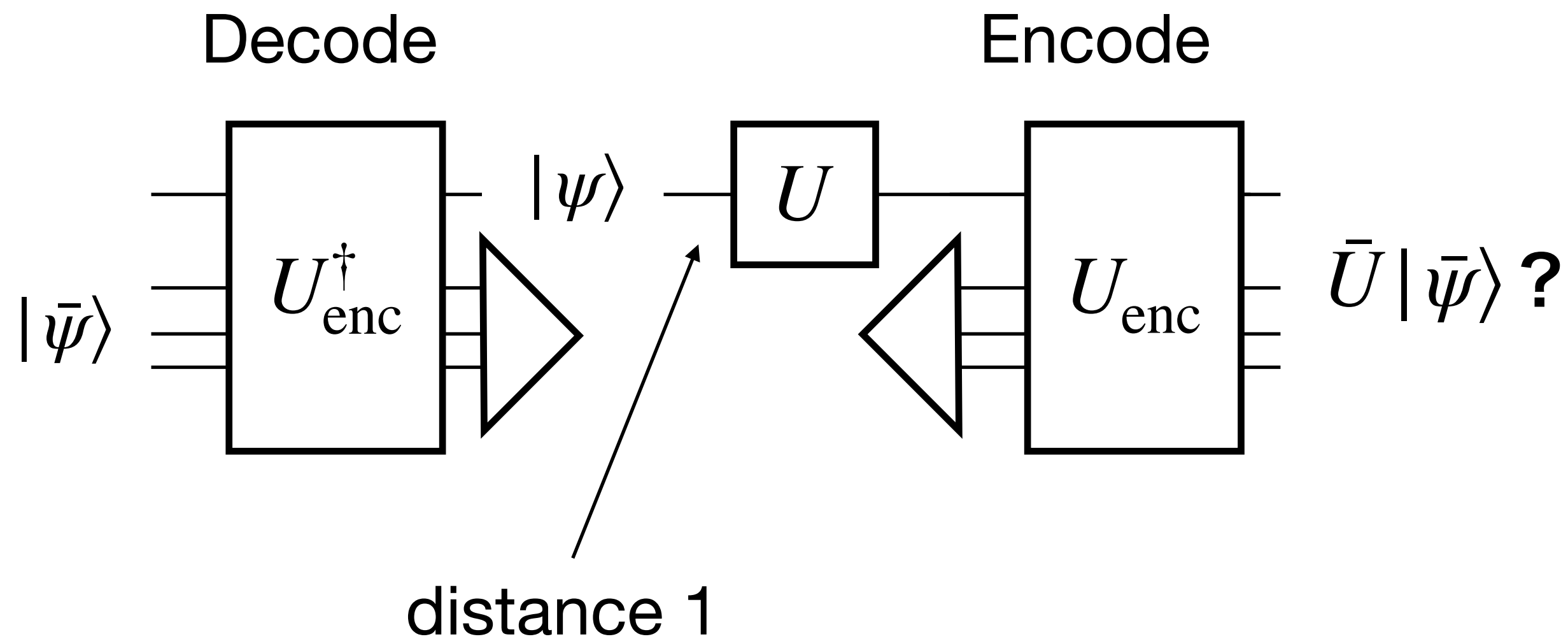


What we want for QECC

4. Fault-tolerant logical operation

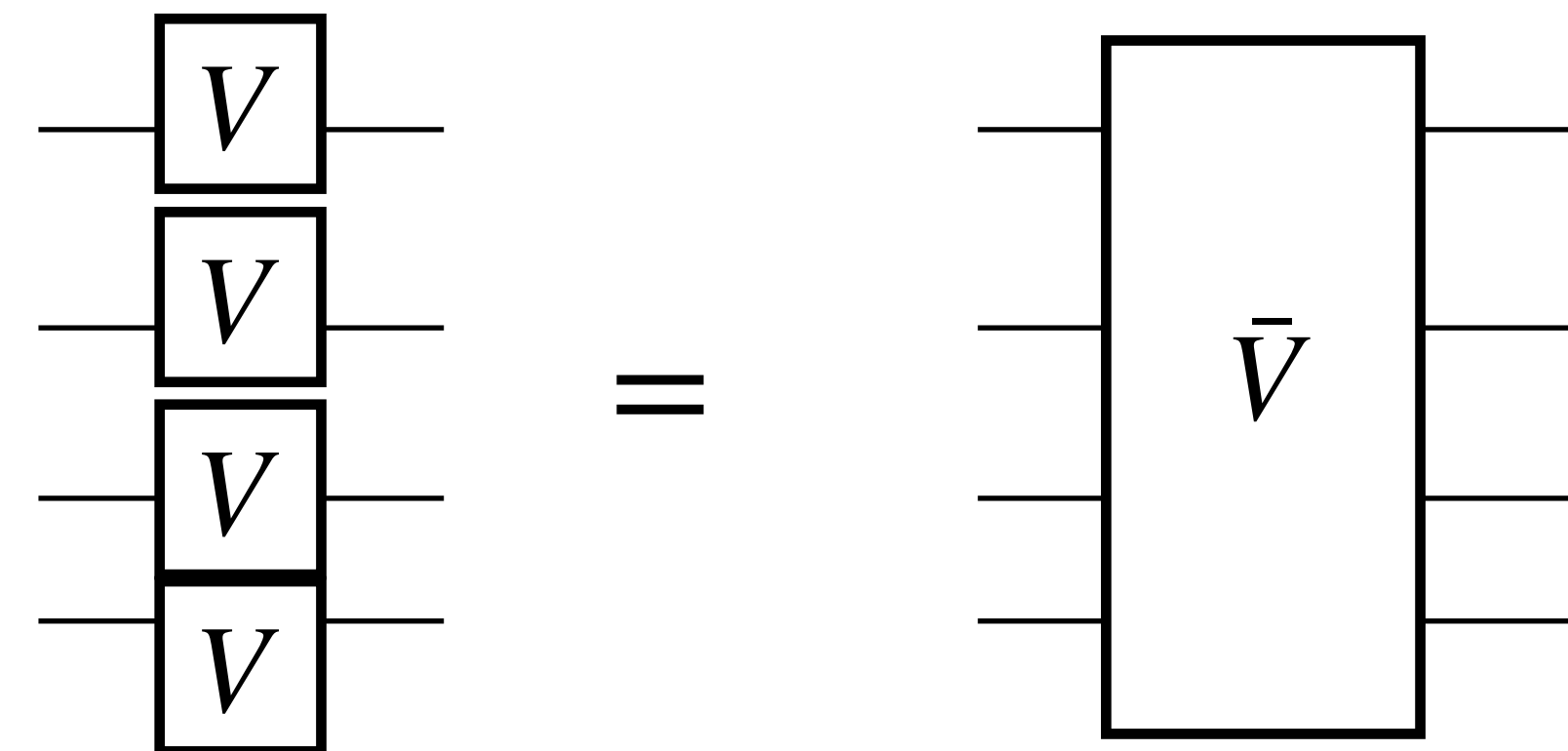
Scale the quantum circuit with larger code distance with universal gateset

Non-fault-tolerant

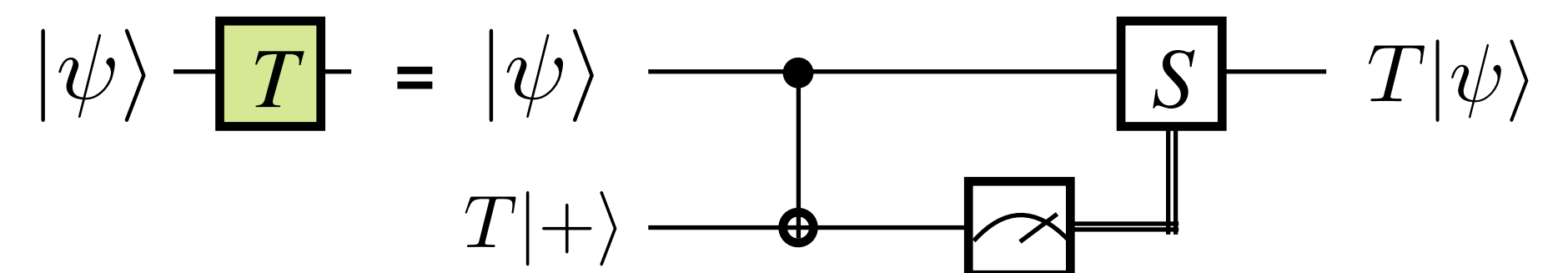


Fault-tolerant

Transversal gates



Gate teleportation



What we want for QECC

5. High encoding rate

Encodes many logical qubit using fewer physical qubits

Code parameters for QECC

$$[[n, k, d]]$$

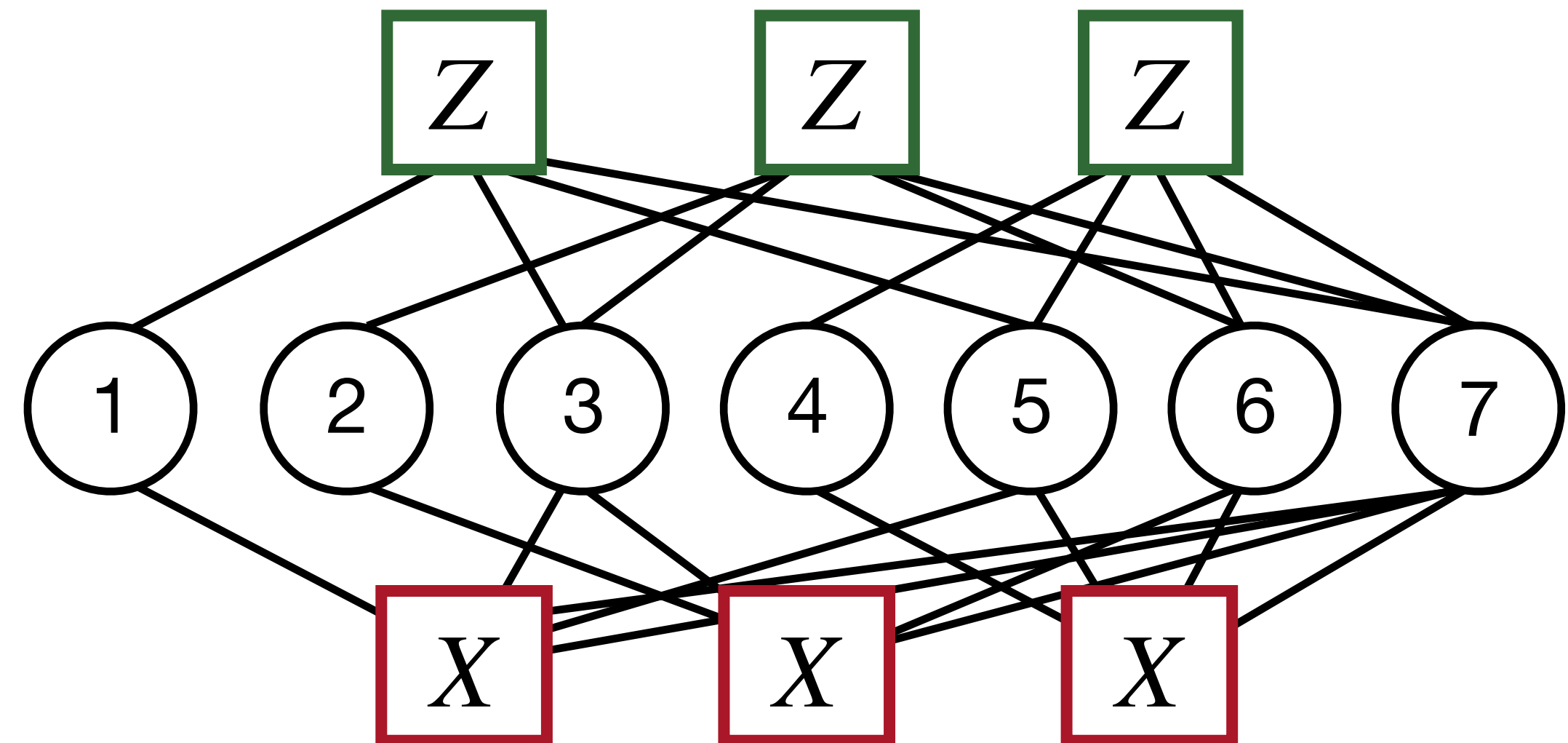
n : # physical qubit

k : # logical qubit

d : code distance

Encoding rate : $R = k/n$

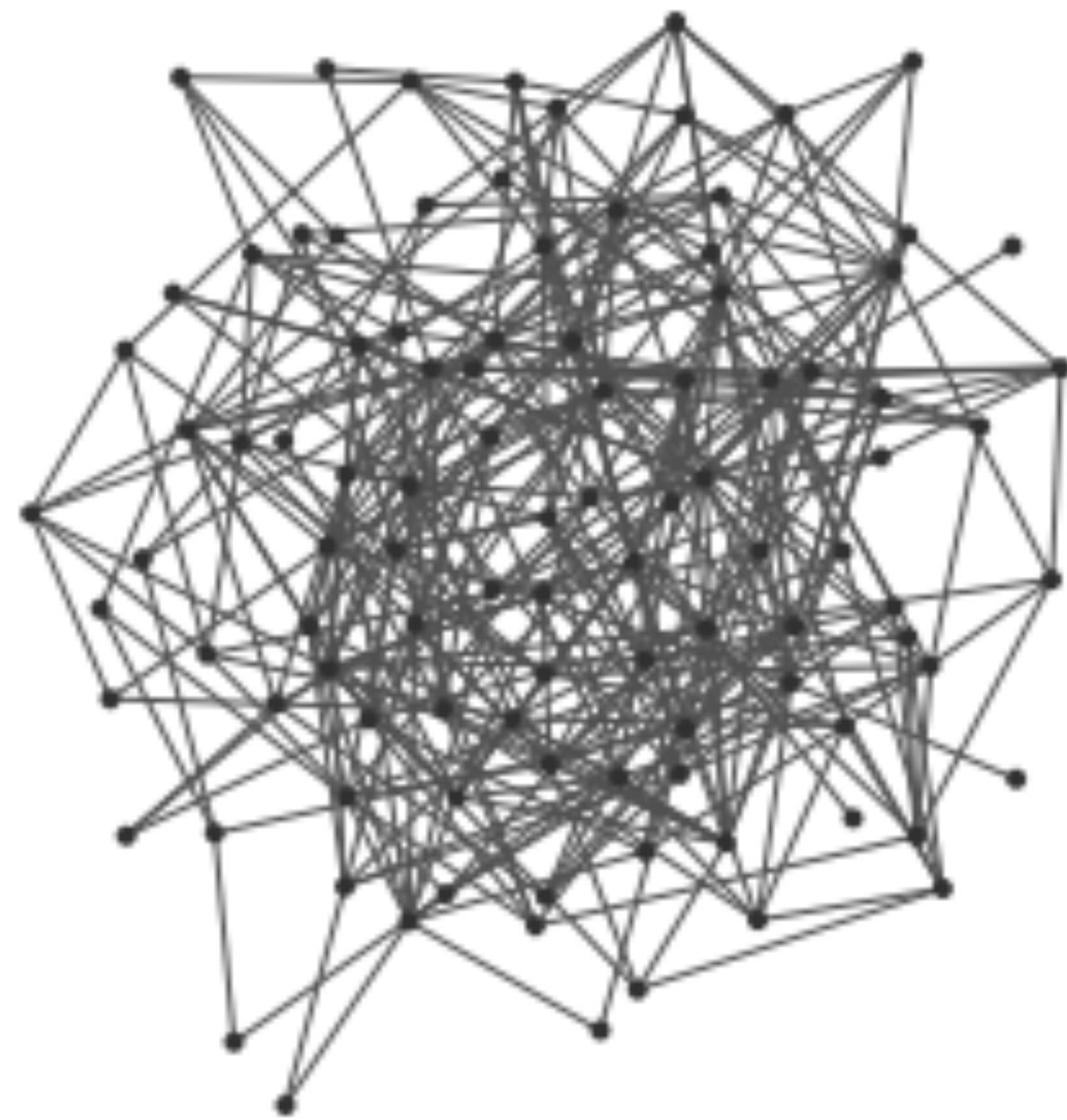
e.g. Steane code encodes 1 logical qubit per 7 physical qubit



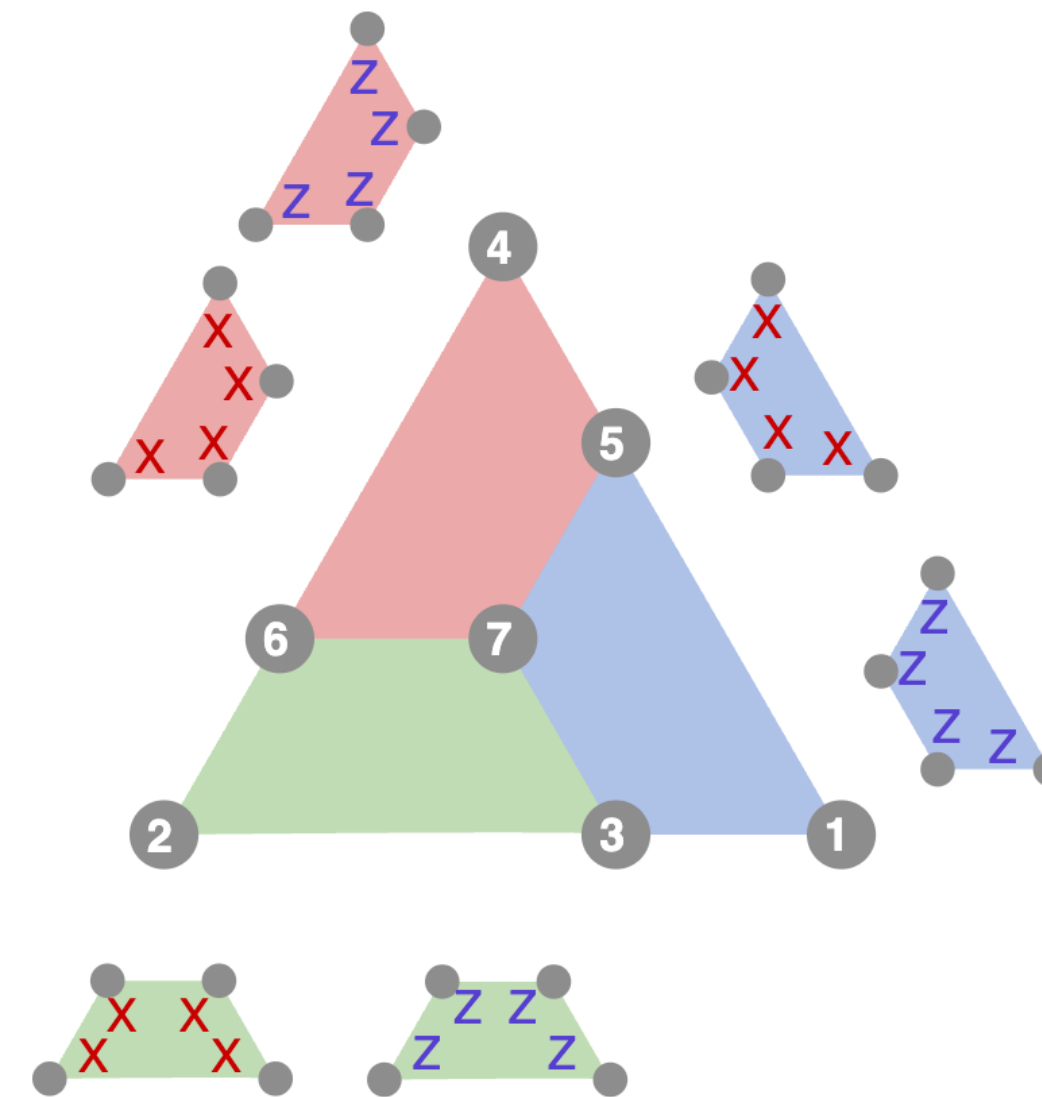
What we want for QECC

6. Hardware-friendly

Random codes have good properties but not option for hardware
(long-range, dense connectivity, high check weight)



unfriendly



friendly

What we want for QECC

1. High threshold

Encodes many logical qubit using fewer physical qubits

2. Low-density parity check

Keep the syndrome extraction circuit small for larger d

3. Fast decoder

Especially real-time decoder is required.

4. Fault-tolerant logical operation

Scale the quantum circuit with larger code distance with universal gateset

5. High encoding rate

Encodes many logical qubit using fewer physical qubits

6. Hardware friendly

We want to build quantum computers in reality

What we want for QECC

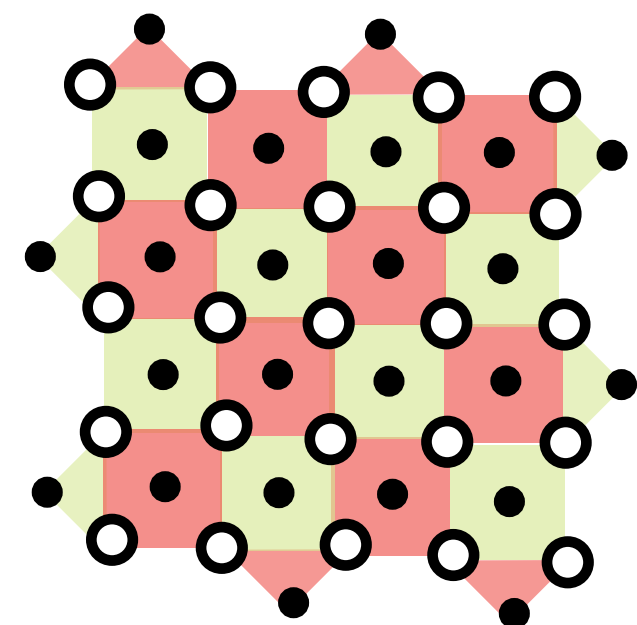
	Topological code	good qLDPC	Concatenated
High threshold	✓	✓	?
LDPC	✓	✓	✗
Fast decoder	✓	✓	✓
Logical ops.	✓	✓	?
High encoding rate	✗	✓	✓✓
Hardware friendly	✓	Depends	?

What we want for QECC

	Topological code	good qLDPC	Concatenated
High threshold	✓	✓	?
LDPC	✓	✓	✗
Fast decoder	✓	✓	✓
Logical ops.	✓	✓	?
High encoding rate	✗	✓	✓✓
Hardware friendly	✓	Depends	?

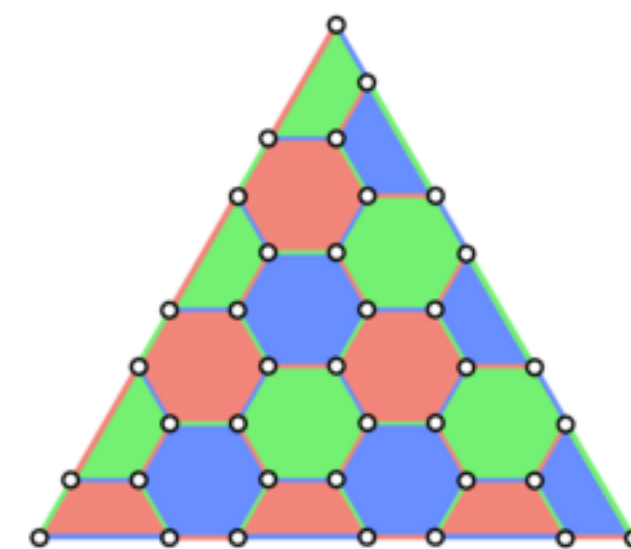
Surface code

Kitaev, arXiv:quant-ph/9707021
 Bravyi&Kitaev, arXiv:quant-ph/9811052



Color code

Bombin&Martin-Delgado, PRL ('06)



Bravyi-Poulin-Terhal bound BPT, PRL ('10)

For $[[n, k, d]]$ code in D -dim system:

$$kd^{2/(D-1)} = O(n)$$

In 2D, this implies $[[d^2, O(1), d]]$

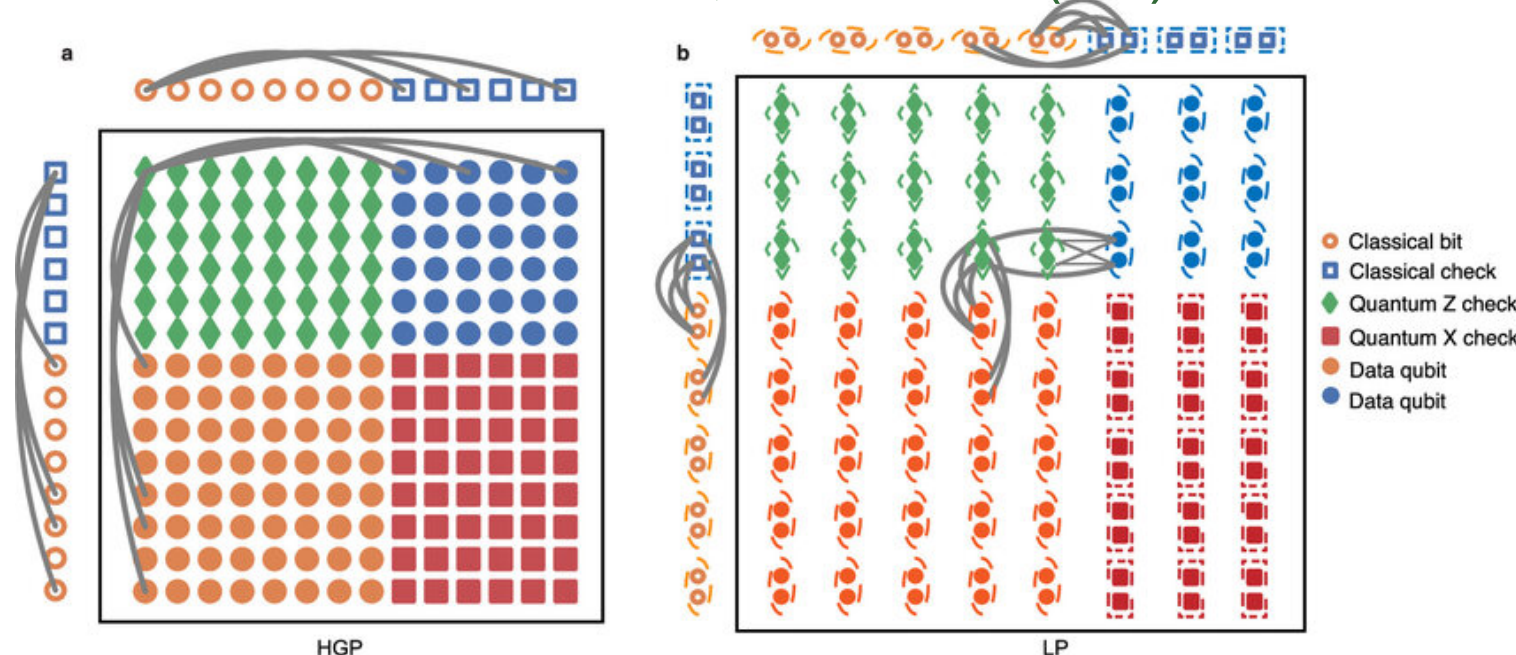
(surface code is almost best you can do)

What we want for QECC

	Topological code	good qLDPC	Concatenated
High threshold	✓	✓	?
LDPC	✓	✓	✗
Fast decoder	✓	✓	✓
Logical ops.	✓	✓	?
High encoding rate	✗	✓	✓✓
Hardware friendly	✓	Depends	?

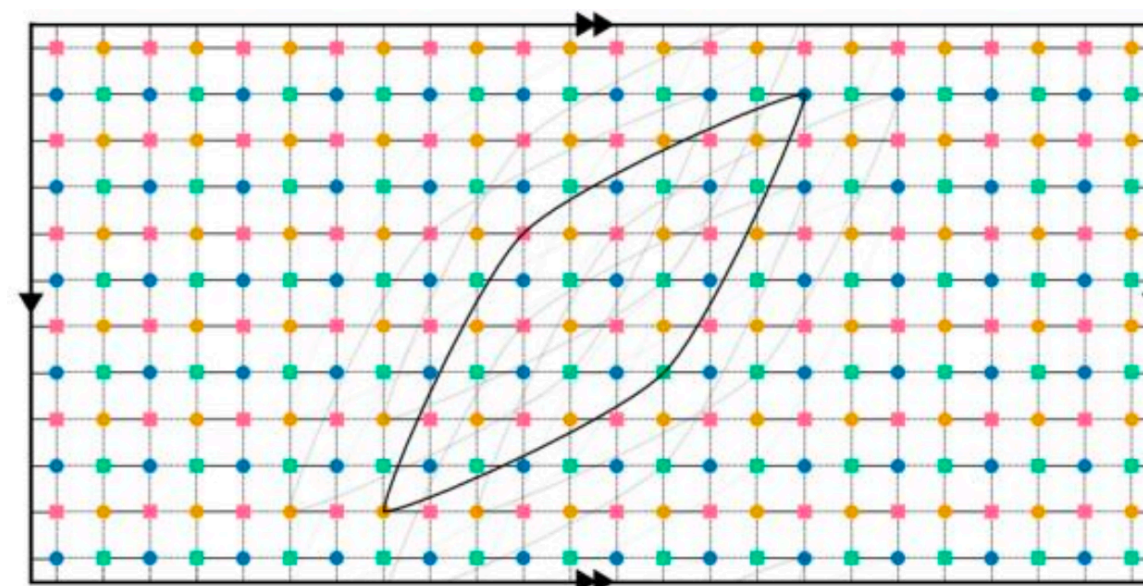
Hypergraph product codes

Tillich and Zemor, IEEE Trans. Inf. Theory ('14)
Panteleev&Kalachev, Quantum ('21)



Bivariate Bicycle codes

Bravyi et al., Nature ('24)



Universal logical operations

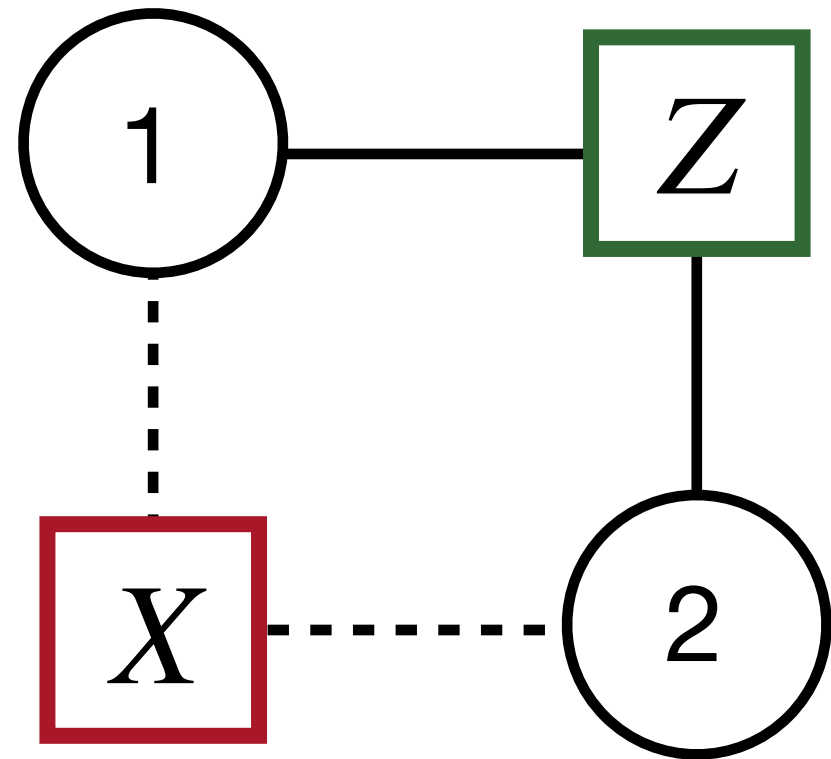
usually relies on teleportation from magic states prepared on surface code

Decoders for surface code

Surface code

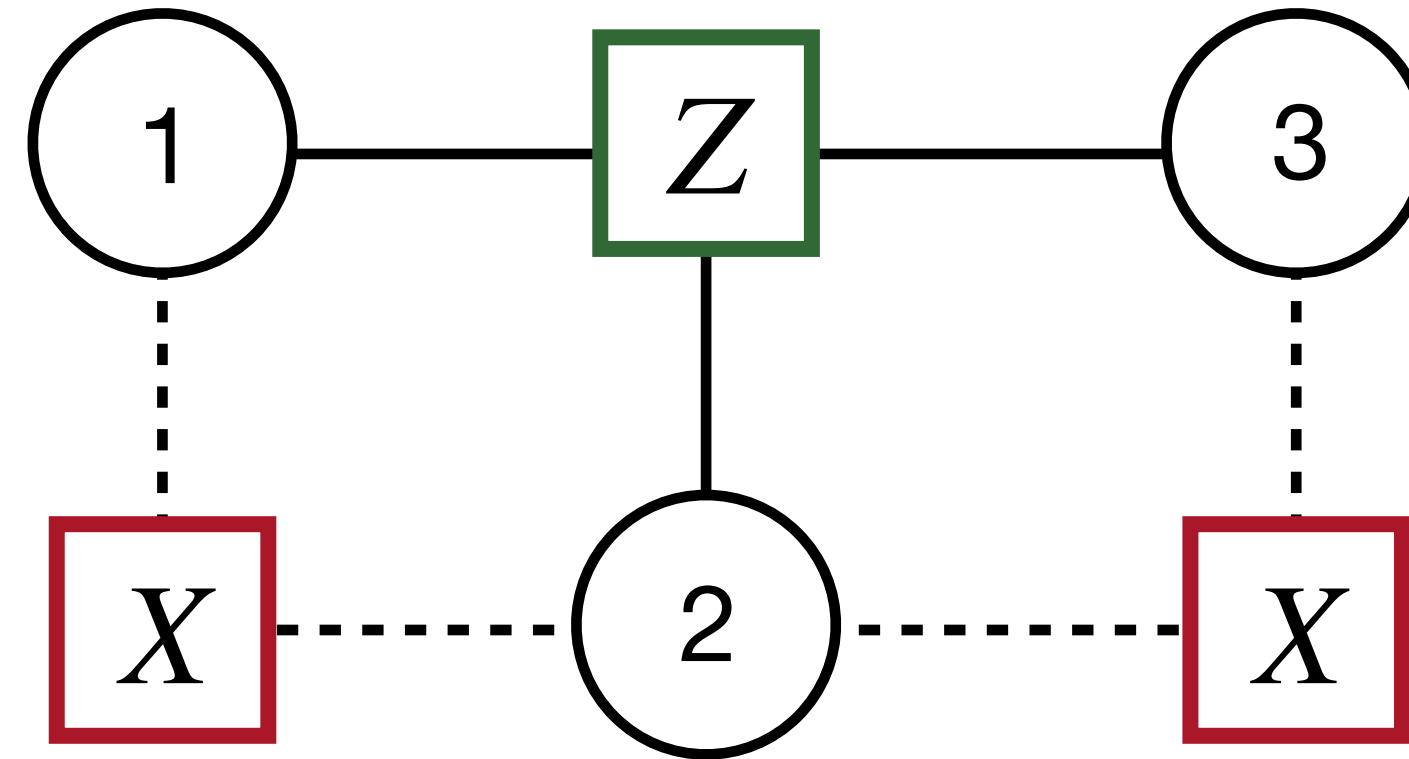
A cycle in Tanner graph = X and Z stabilizers commutes.

Connectivity-aware code can be constructed by regarding checks as ancilla.



$$S_1 = X_1 X_2$$

$$S_2 = Z_1 Z_2$$



$$S_1 = X_1 X_2$$

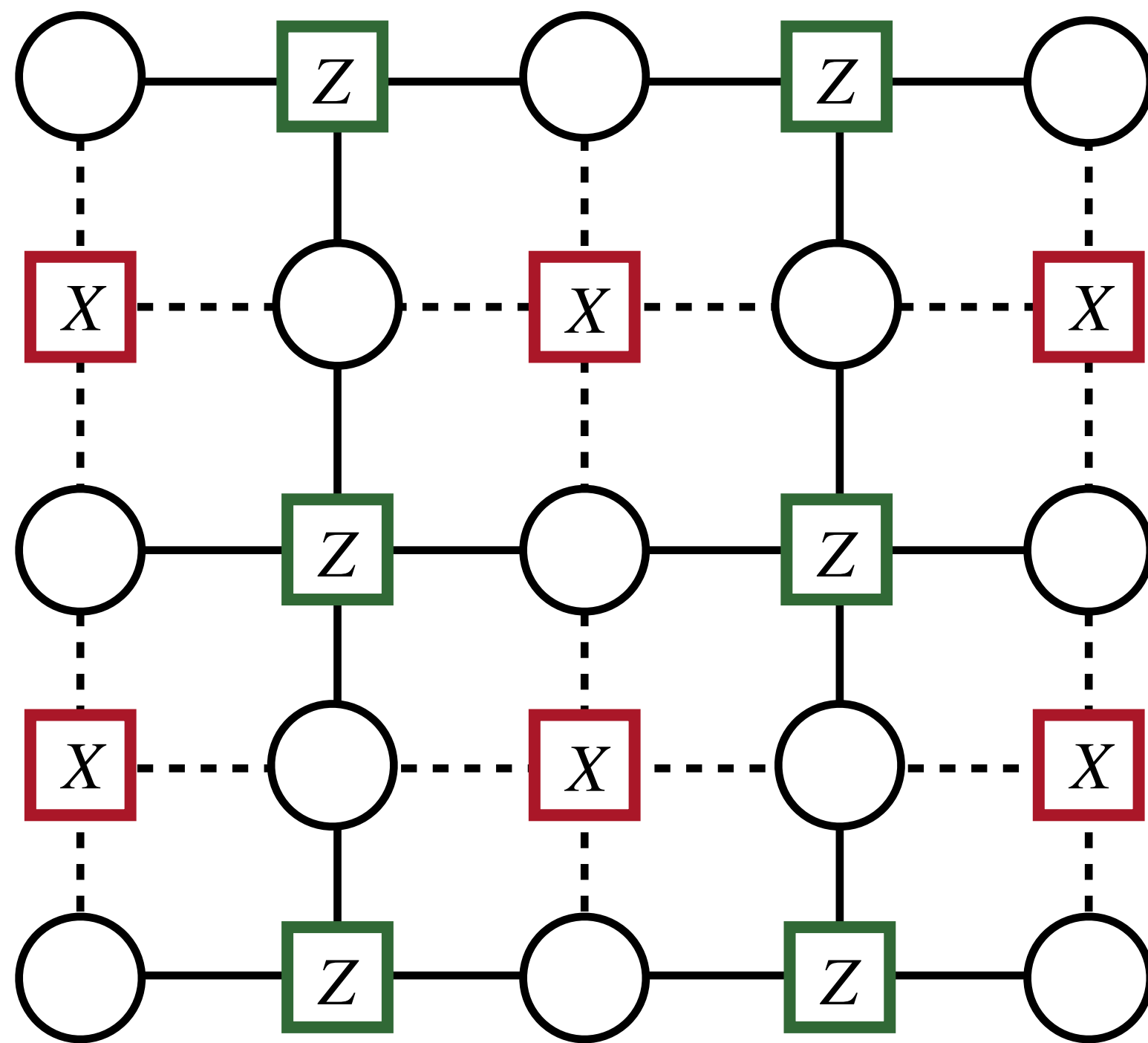
$$S_2 = Z_1 Z_2 Z_3$$

$$S_3 = X_2 X_3$$

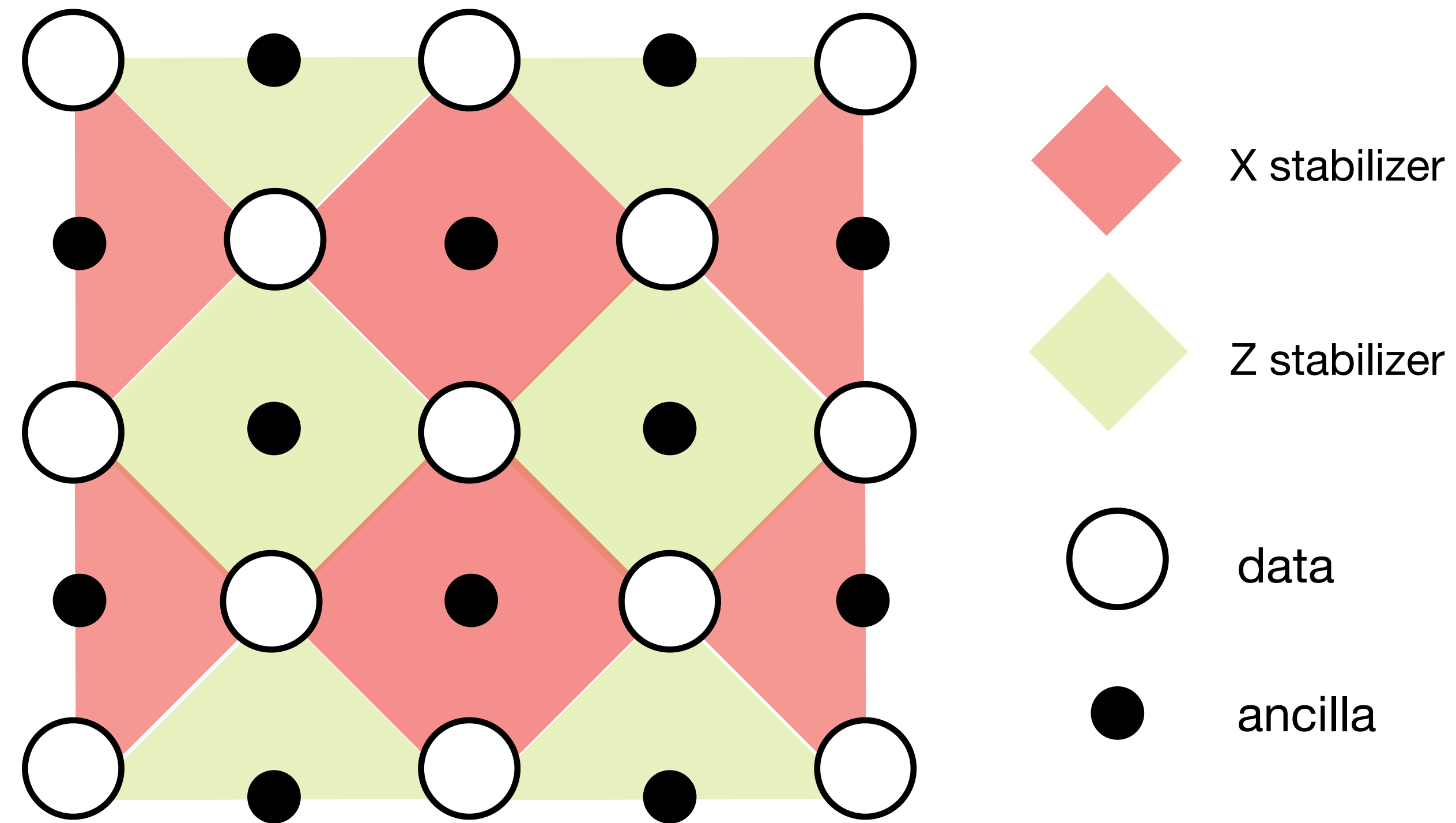
Surface code

Surface code is a CSS code that requires 2d nearest-neighbor connectivity

Tanner graph

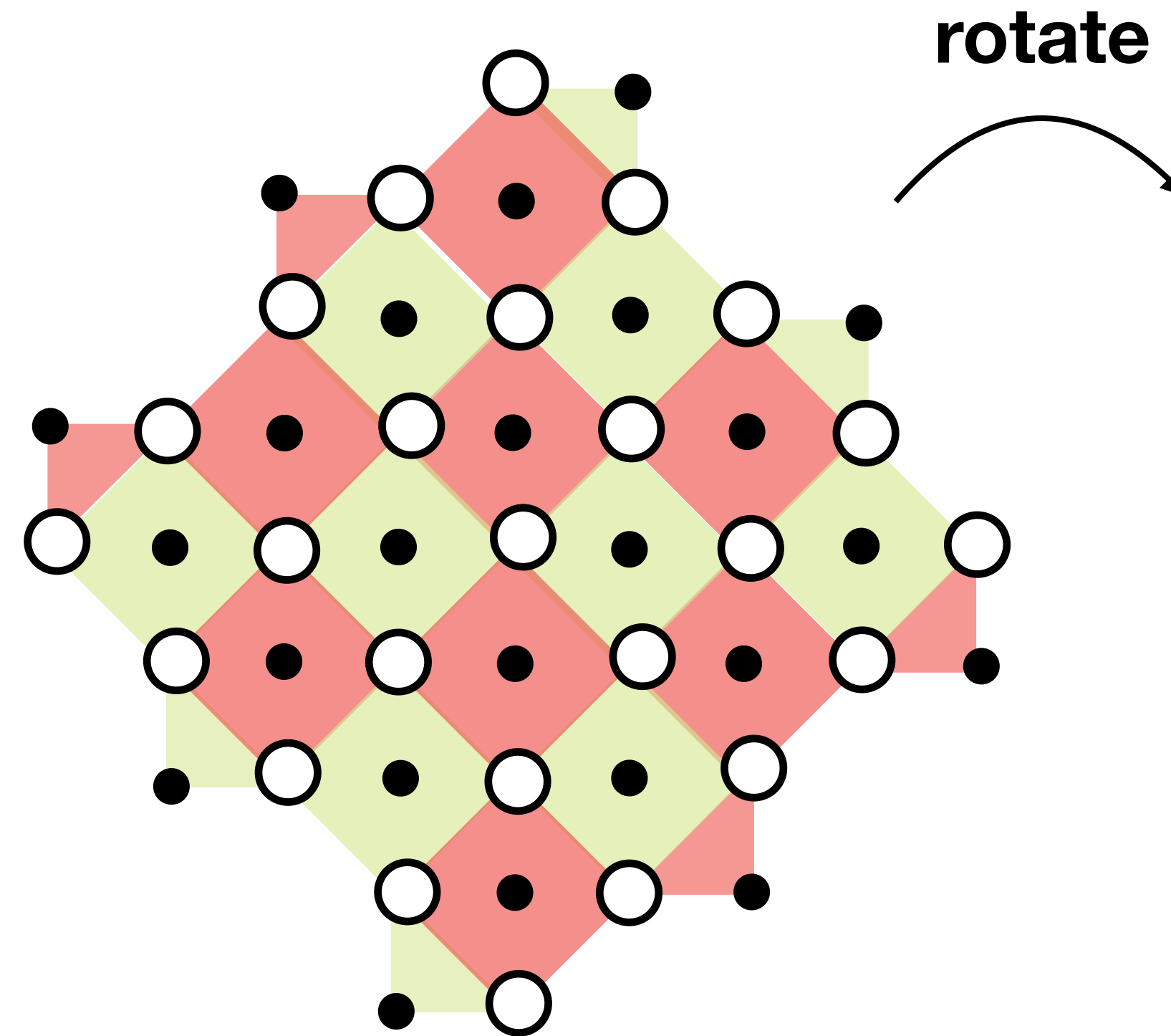
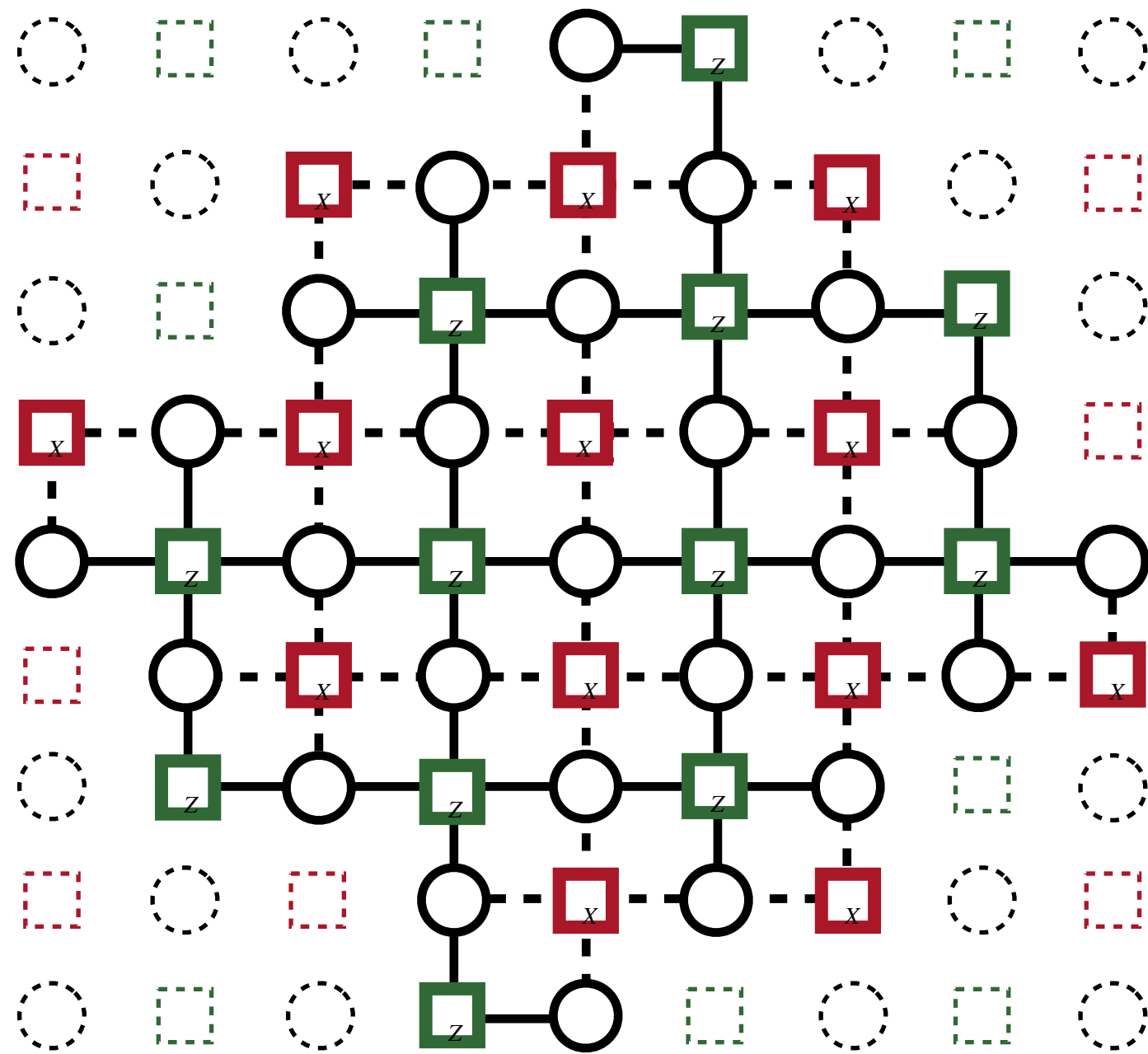


Graphical expression

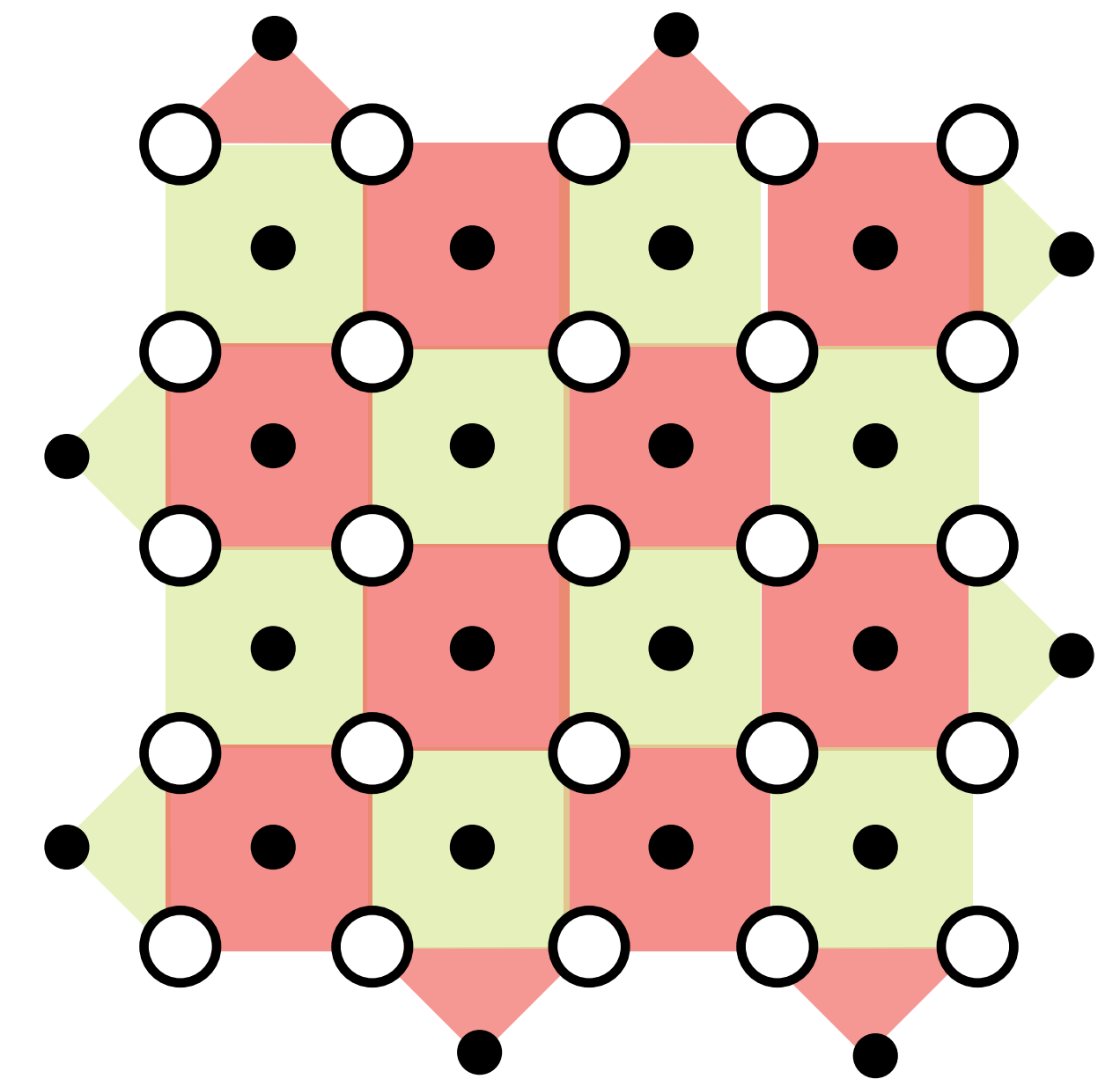


Rotated surface code

Chopped Tanner graph



Rotated surface code



Original : $[[d^2 + (d - 1)^2, 1, d]]$

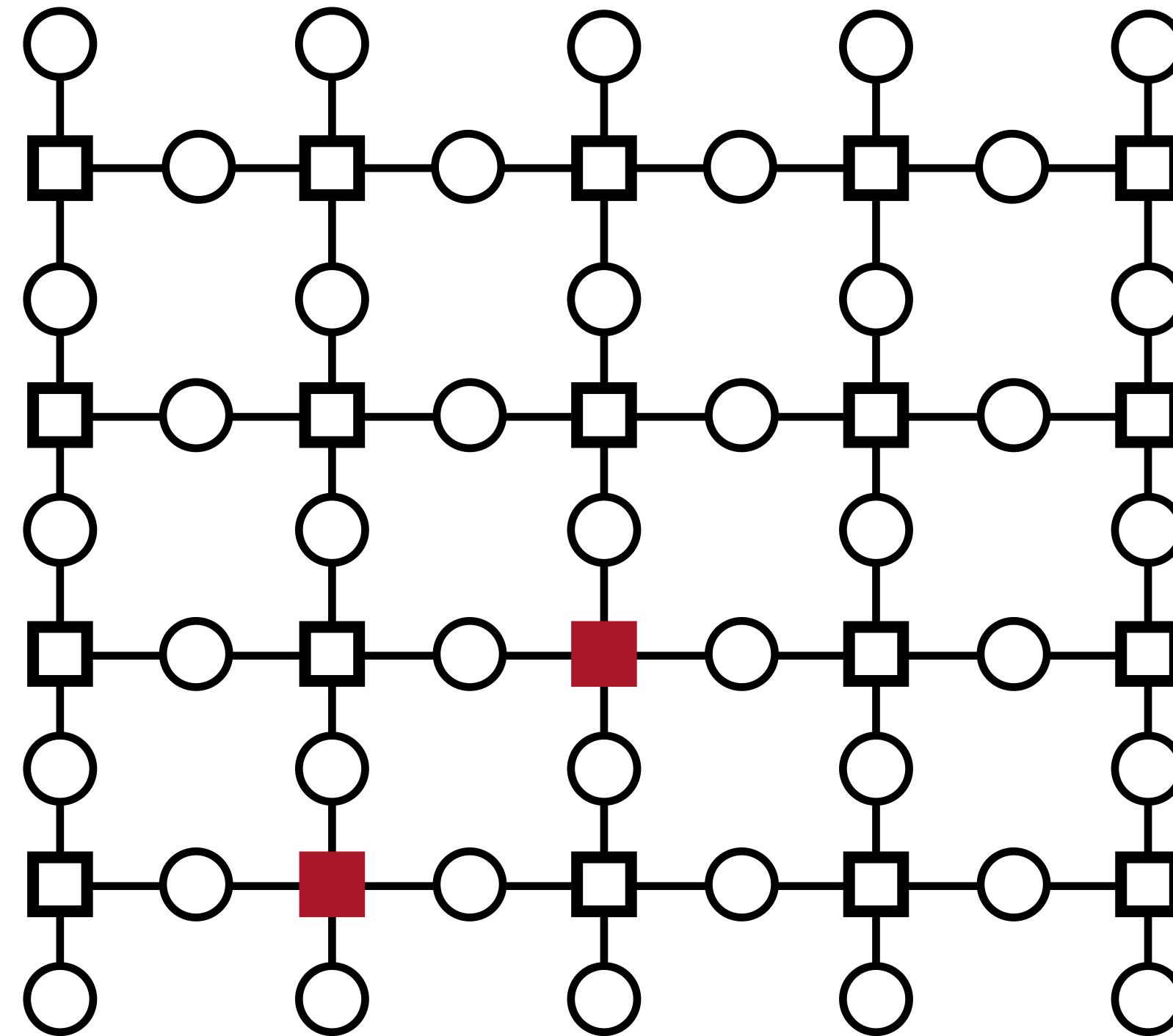
Rotated: $[[d^2, 1, d]]$

nearly halves data qubits!

Estimating error locations

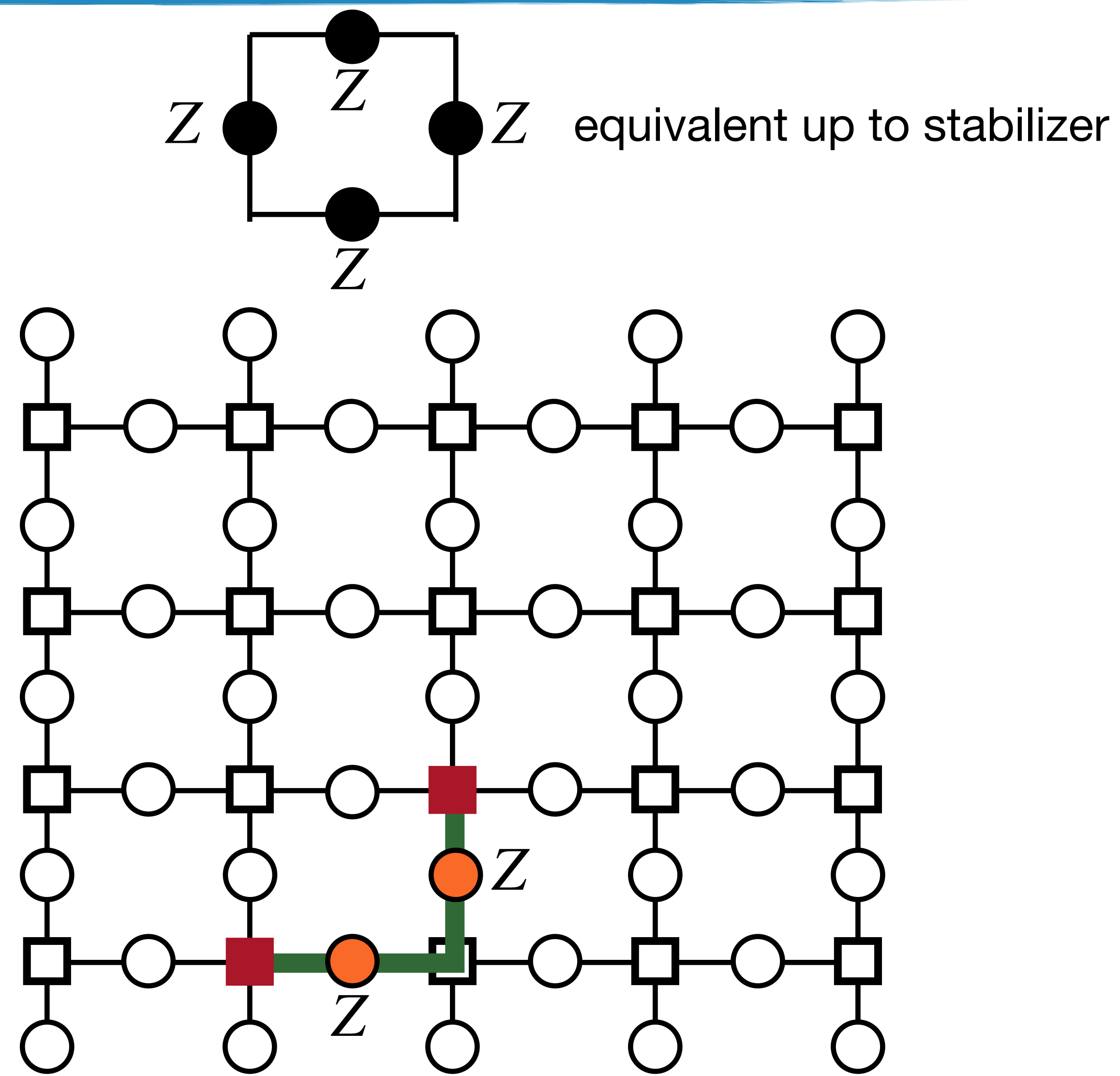
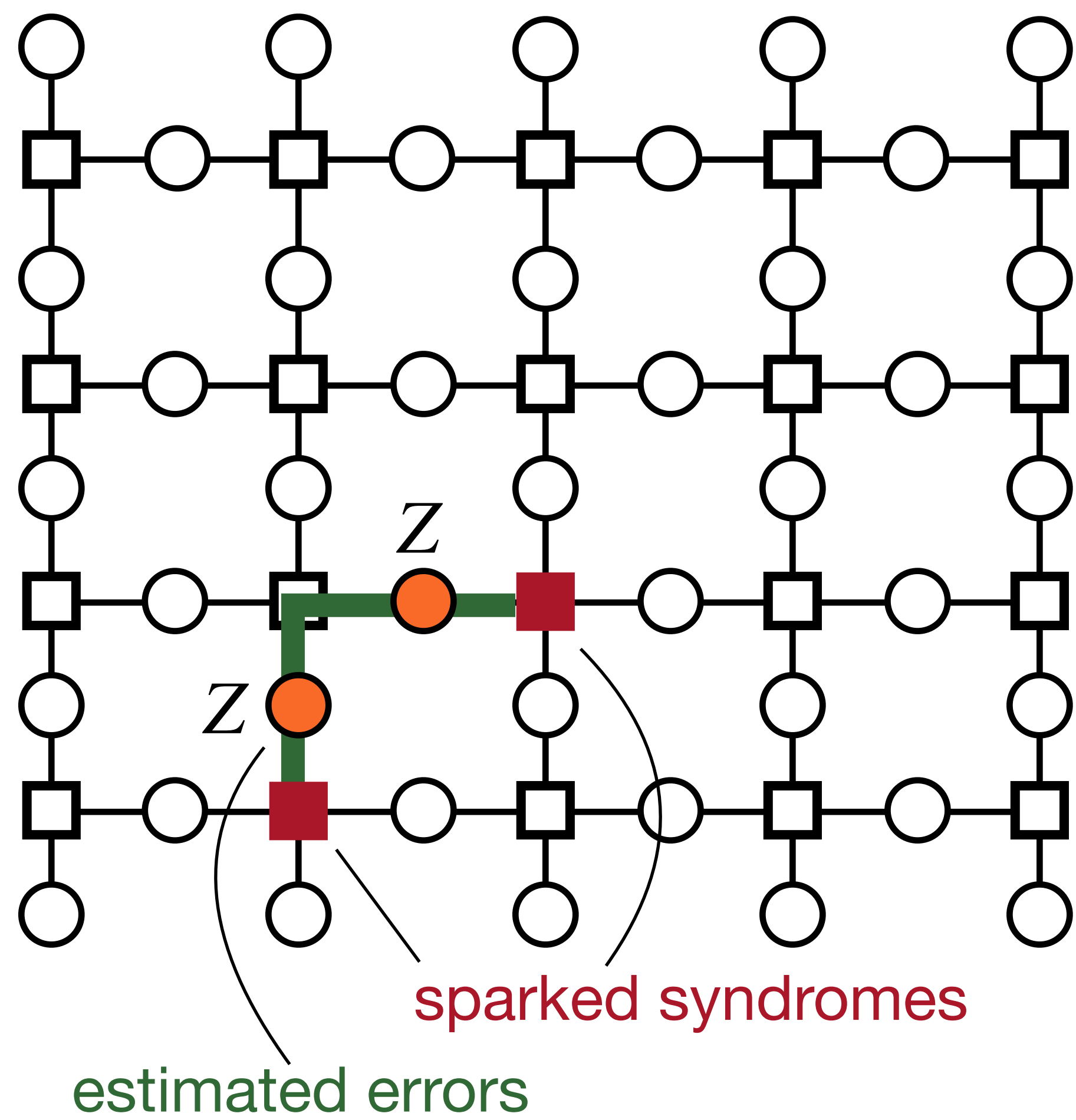
For simplicity, we assume that

- (1) Errors are Z-type,
- (2) Markovian,
- (3) Spacially independent

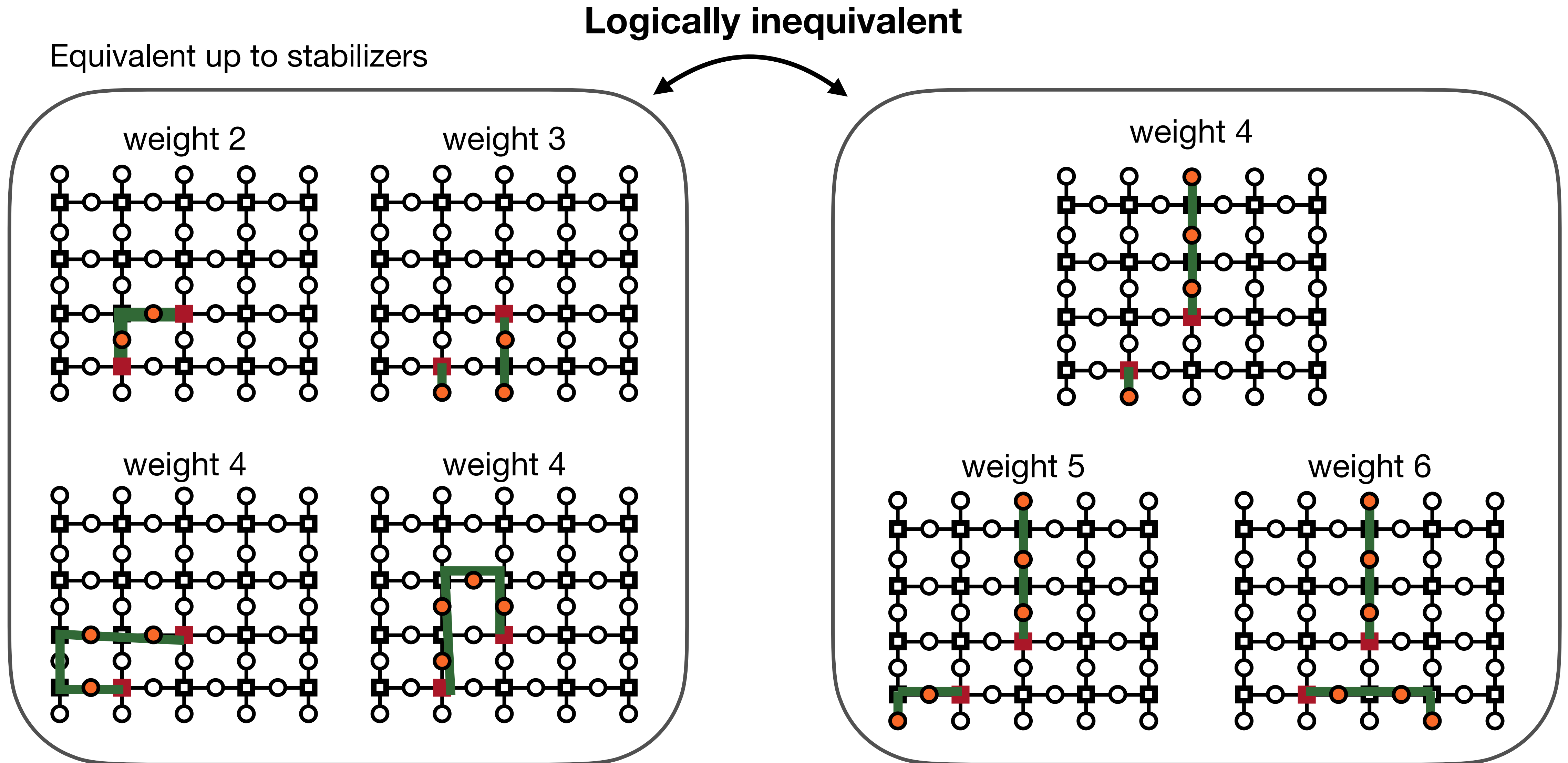


What is the most likely error?

Estimating error locations



Estimating error locations



Logical error occur when you misestimate the logical class.

Maximum Likelihood Decoding (MLD)

Assume that error is single-qubit channel with probability p as

$$\mathcal{E} = \bigcirc_i \mathcal{E}_i = \bigcirc_i ((1-p)\mathcal{I} + p\mathcal{Z}_i)$$

Then, the probability of physical error $E = Z^\eta$ with $\eta \in \{0,1\}^N$ is given by

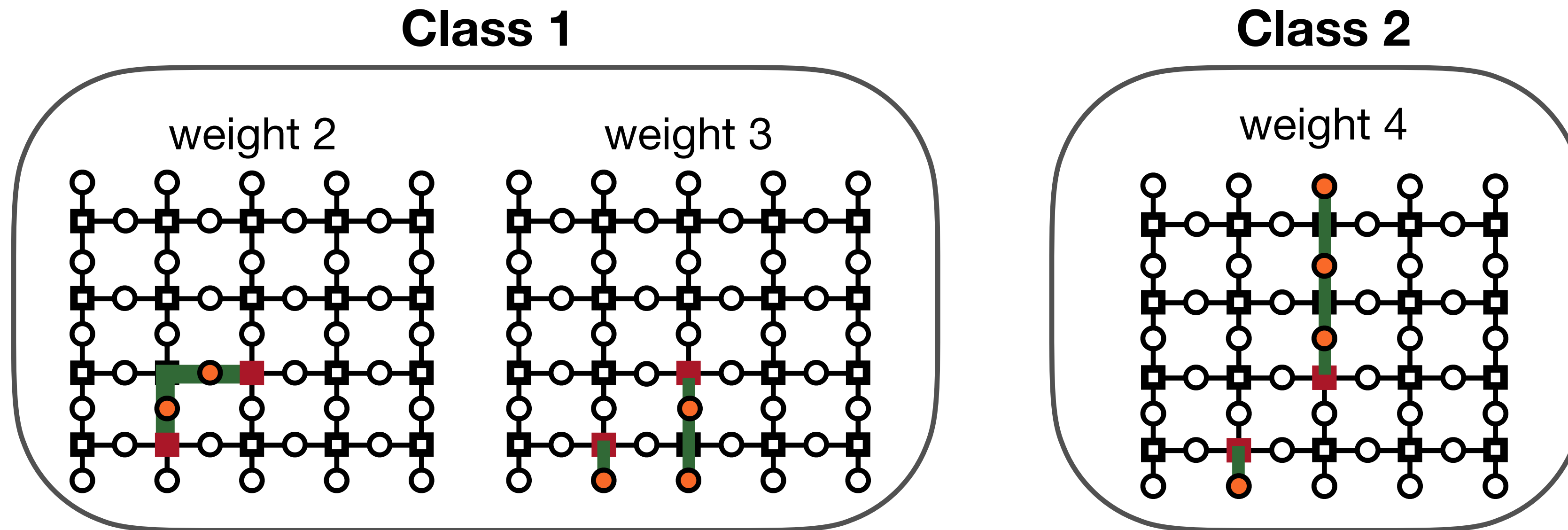
$$P(E) = \prod_{i=1}^N \left(\frac{p}{1-p} \right)^{\eta_i} \prod_{i=1}^N (1-p)$$

Correspondingly, the negative log likelihood is given as

$$L(E) = -\log P(E) = \sum_{i=1}^N \eta_i \log \left(\frac{1-p}{p} \right) + \text{const.}$$

The goal of decoding is to find $\hat{E} = \arg \min \{L(E) \mid s(E) = s\}$ where s is the syndrome.

Most likely errors



$\Pr(E)$

$O(p^2)$

$O(p^3)$

$O(p^4)$

$L(E)$

$2 \log(1/p)$

$3 \log(1/p)$

$4 \log(1/p)$

Most likely

Conditional error rate is $O(p^2)$ in this case

$$\sum \Pr(E_{\text{class 1}} | s) / \sum \Pr(E_{\text{class 2}} | s)$$

Minimum-Weight Perfect Matching

The goal of decoding is to find $\hat{E} = \arg \min \{L(E) \mid s(E) = s\}$ where s is the syndrome.

$$L(E) = -\log P(E) = \sum_{i=1}^N \eta_i \log \left(\frac{1-p}{p} \right) + \text{const.}$$

In surface code, this is **Minimum-Weight Perfect Matching**:

$$\hat{E} = \arg \min \{W(E) \mid s(E) = s\}$$

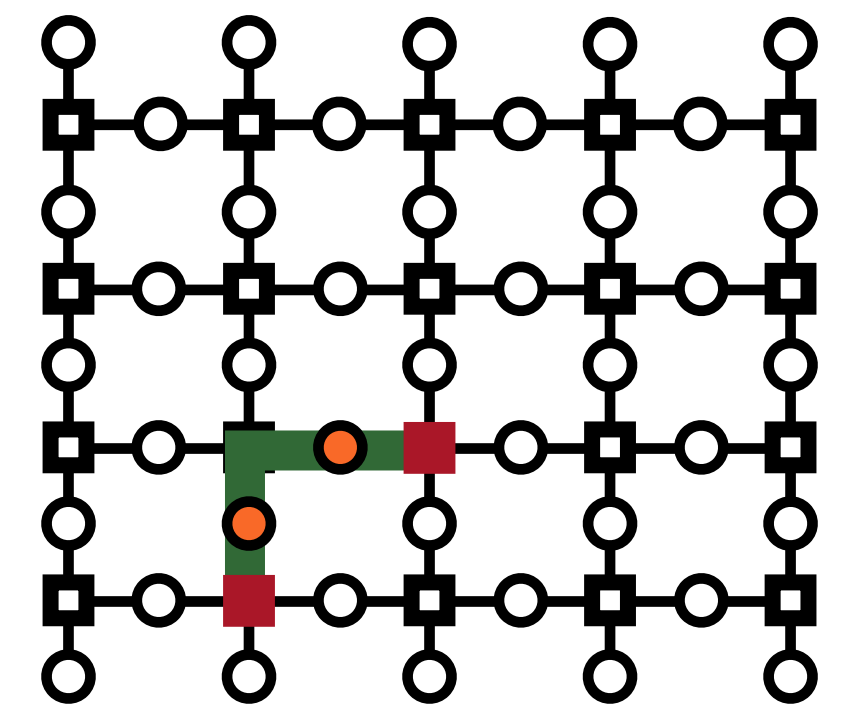
where W is the total weight of error $E = Z^\eta$ on graph G as

$$W(E) = \sum_{e \in 1[\eta]} w_e \text{ where } w_e = \log((1-p)/p)$$

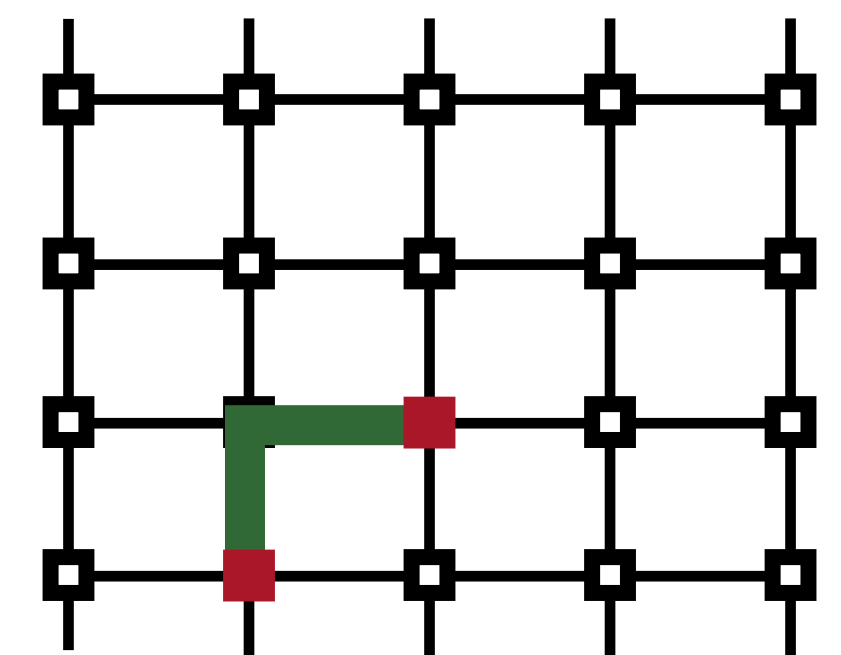
G : graph with Edges = {Elementary Error}

Vertices = {Parity check ancilla}

Tanner graph

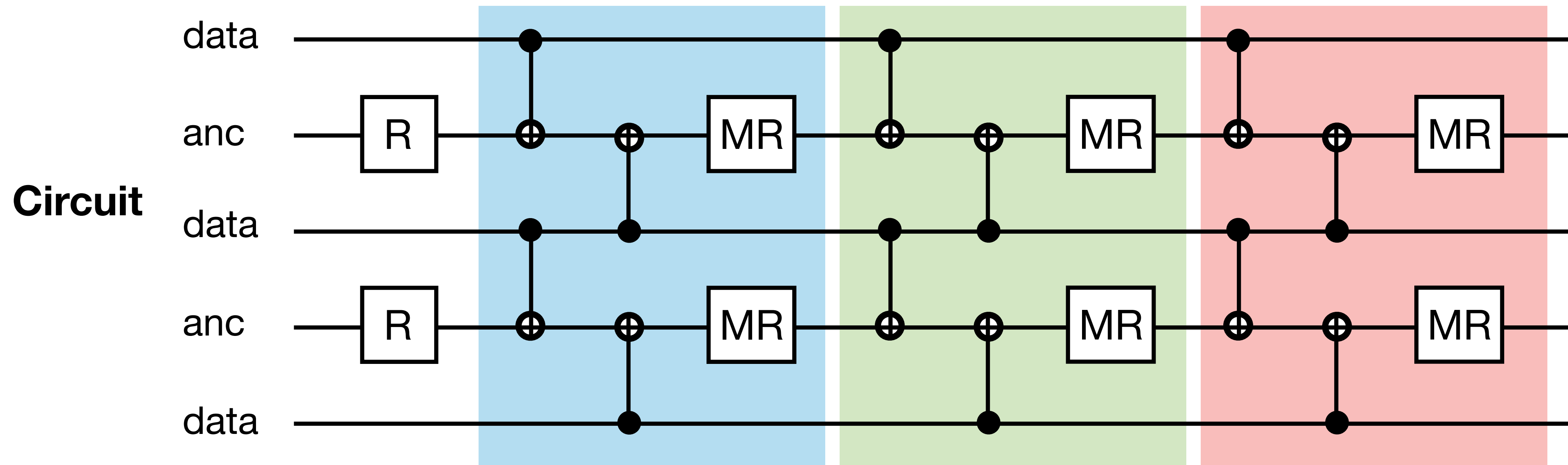


Matching graph



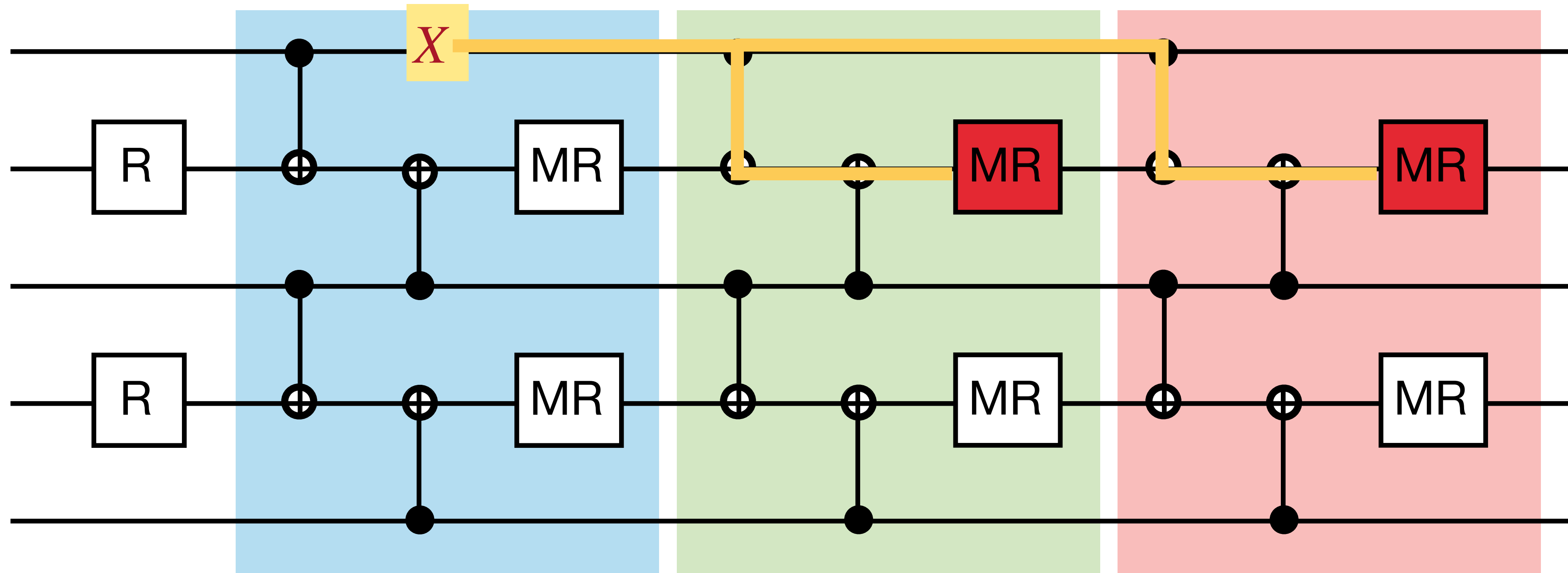
Circuit-level decoding graph in rep. code

In the circuit-level, we must consider the correlations between multiple rounds



Circuit-level decoding graph in rep. code

Circuit

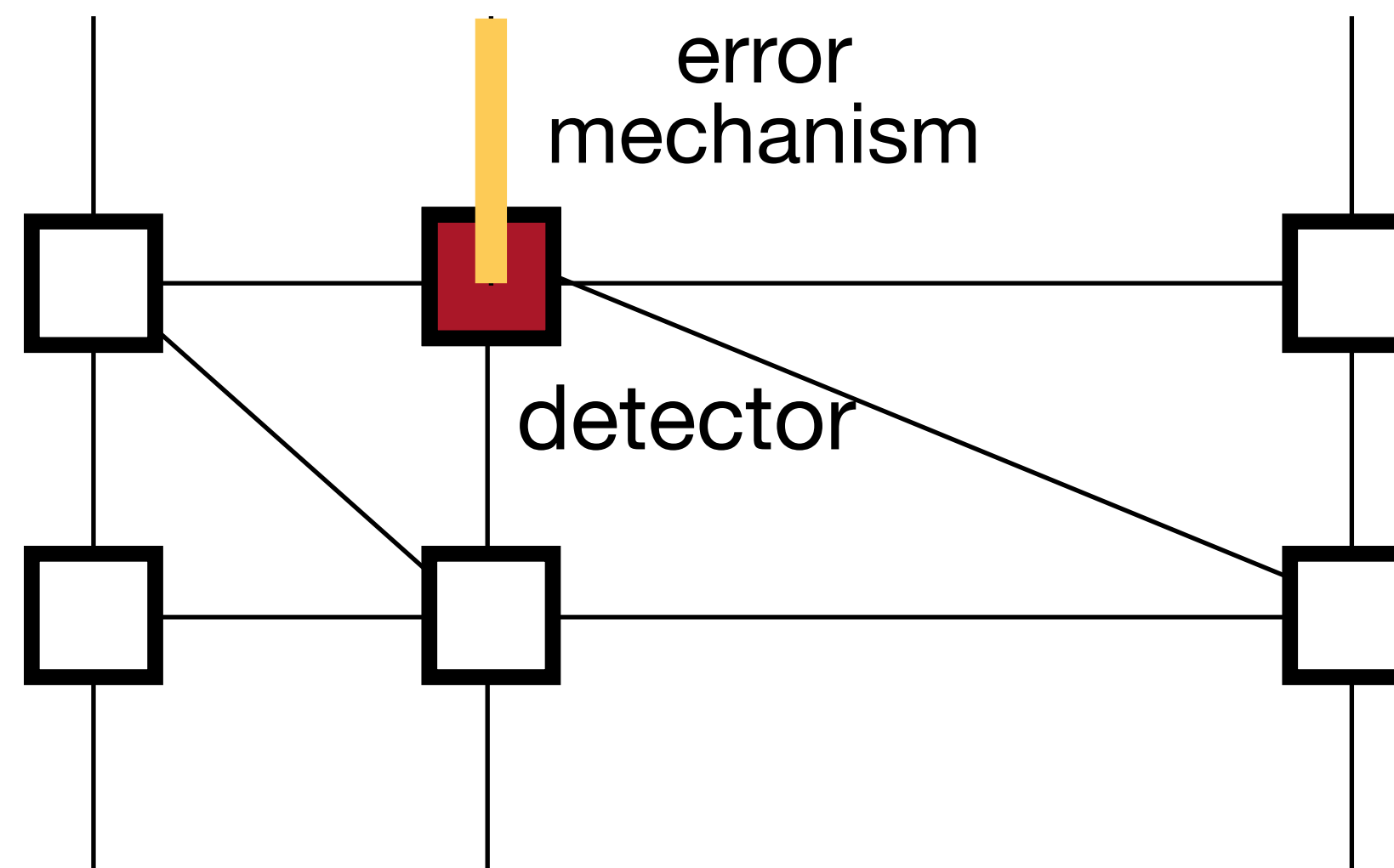


Detector Error Model

- First M is detector
- XOR is taken for subsequent

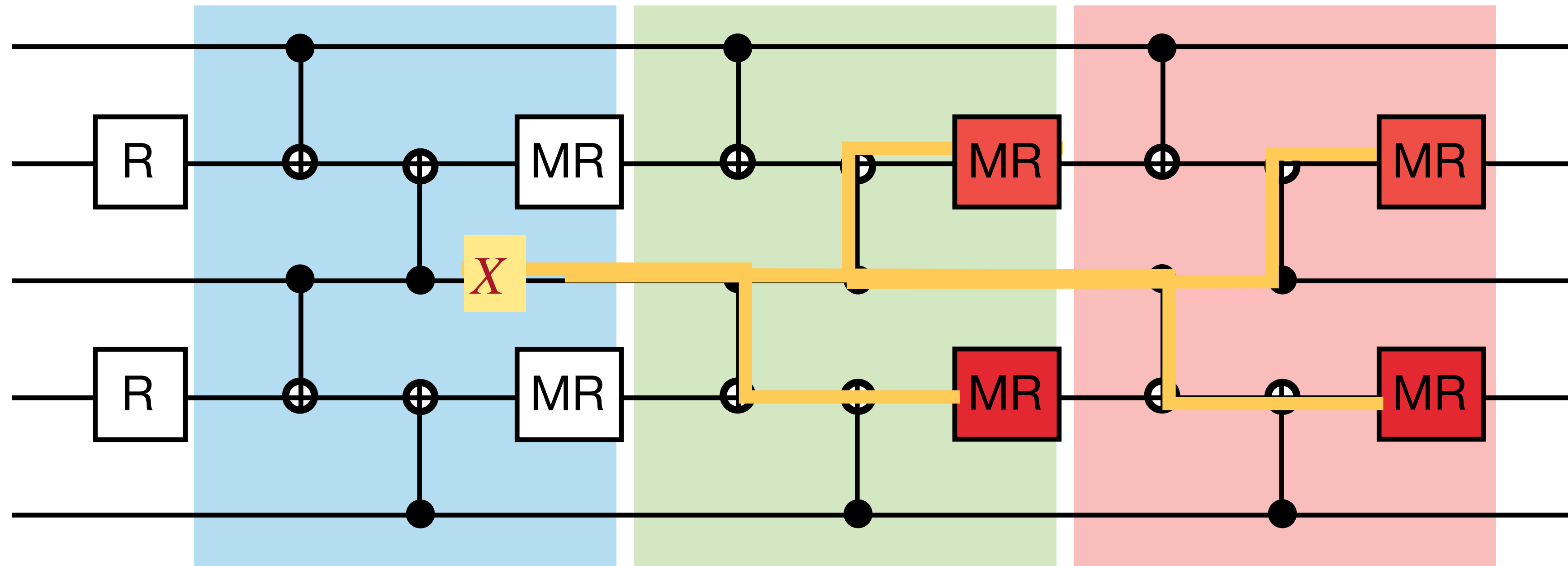
$$d_i = m_i \oplus m_{i-1}$$

- Edge indicates error mech.



Circuit-level decoding graph in rep. code

Circuit

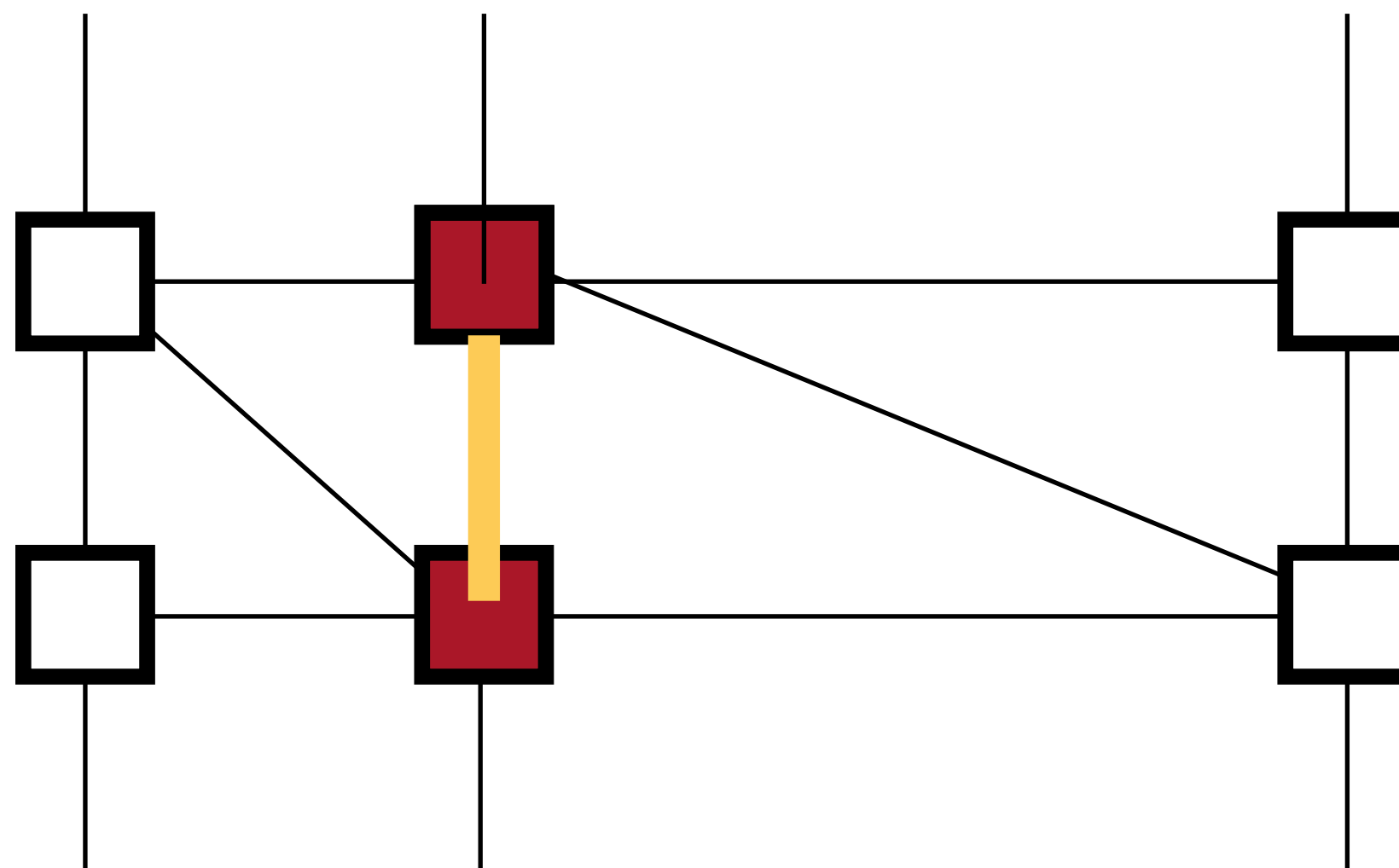


Detector Error Model

- First M is detector
- XOR is taken for subsequent

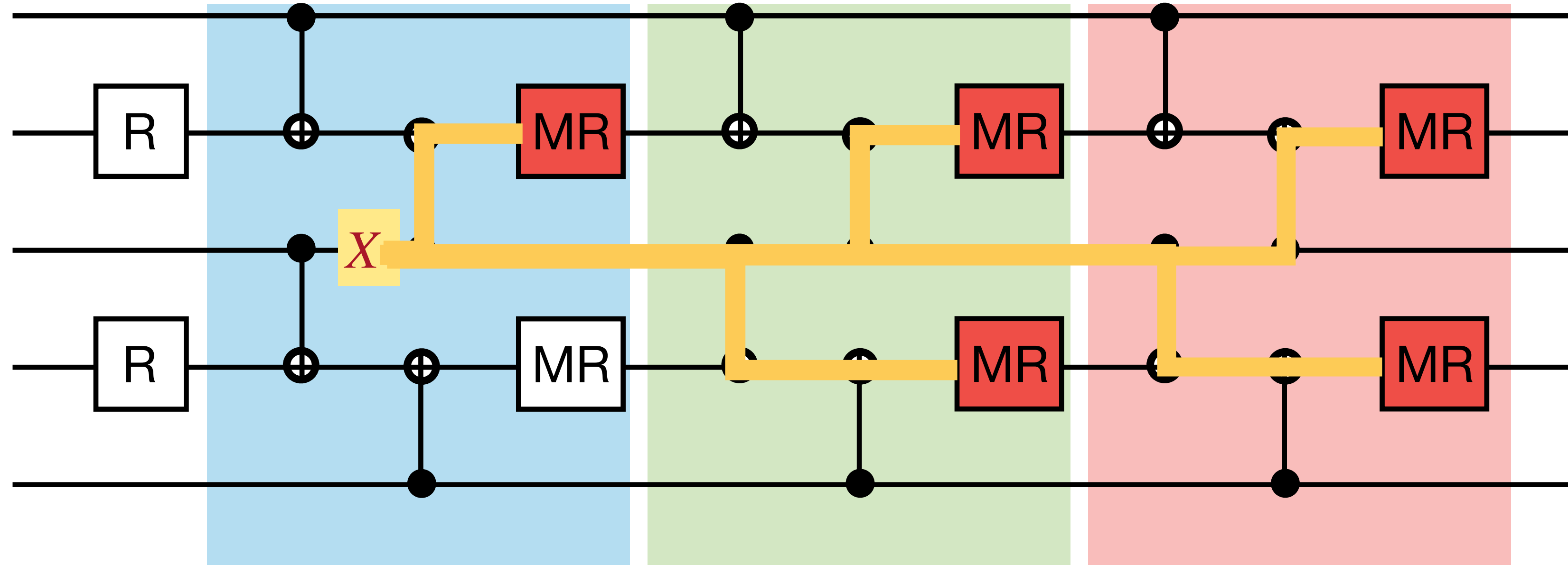
$$d_i = m_i \oplus m_{i-1}$$

- Edge indicates error mech.



Circuit-level decoding graph in rep. code

Circuit

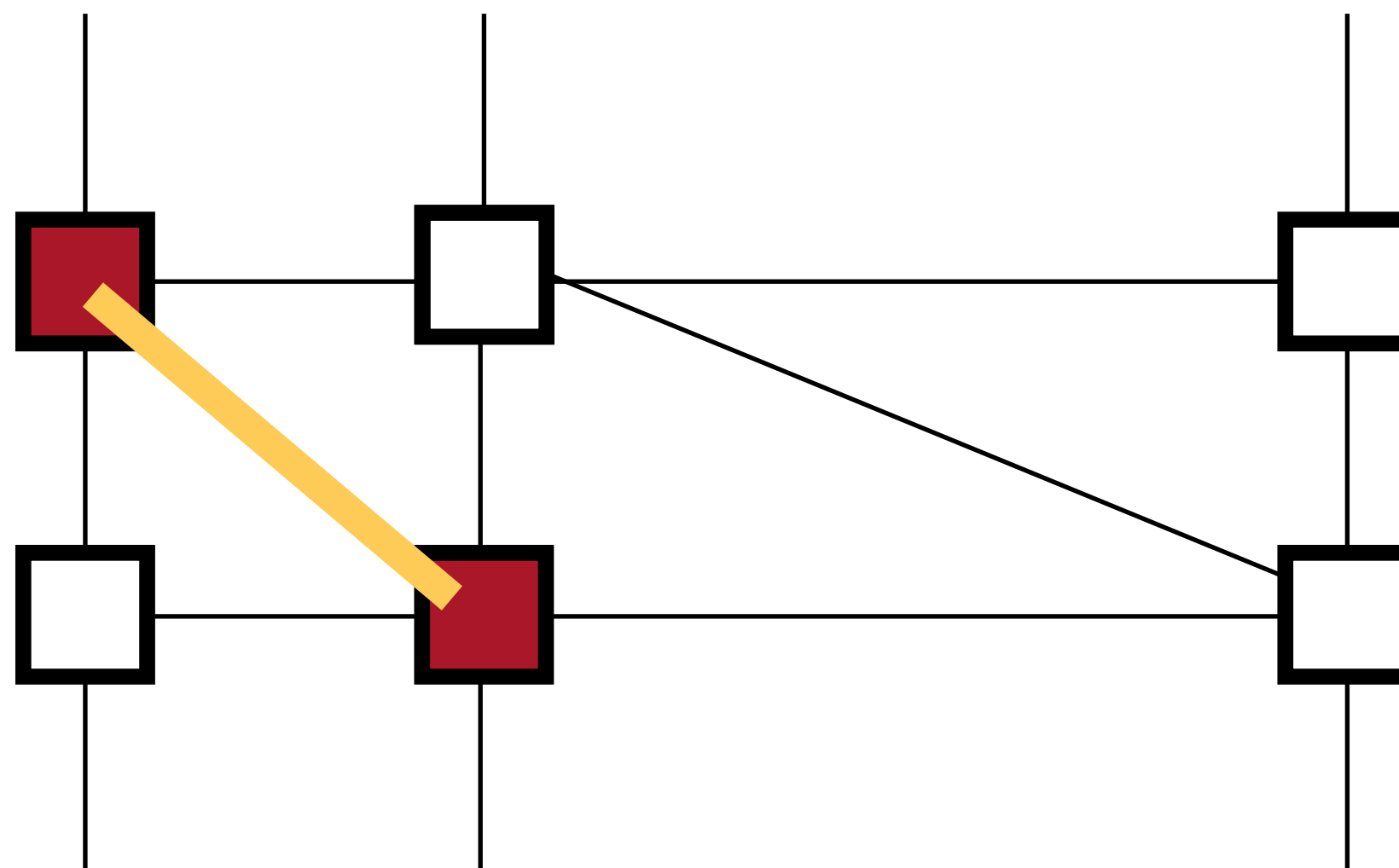


Detector Error Model

- First M is detector
- XOR is taken for subsequent

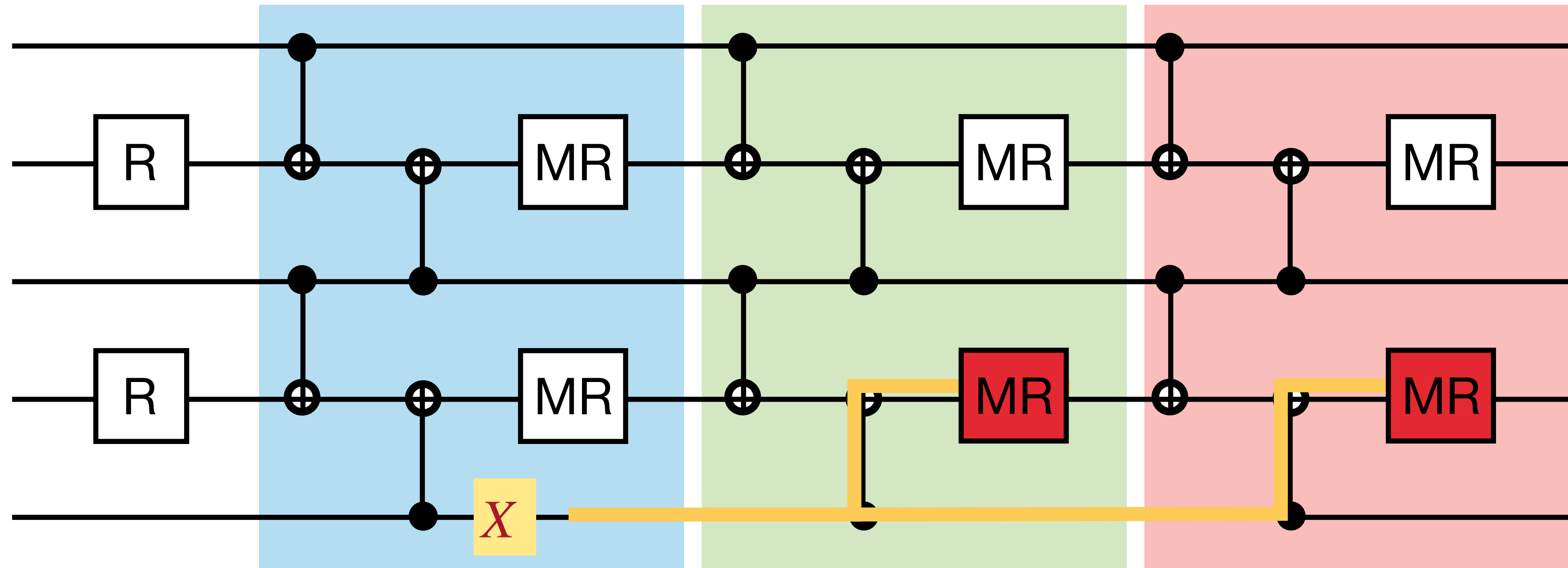
$$d_i = m_i \oplus m_{i-1}$$

- Edge indicates error mech.



Circuit-level decoding graph in rep. code

Circuit

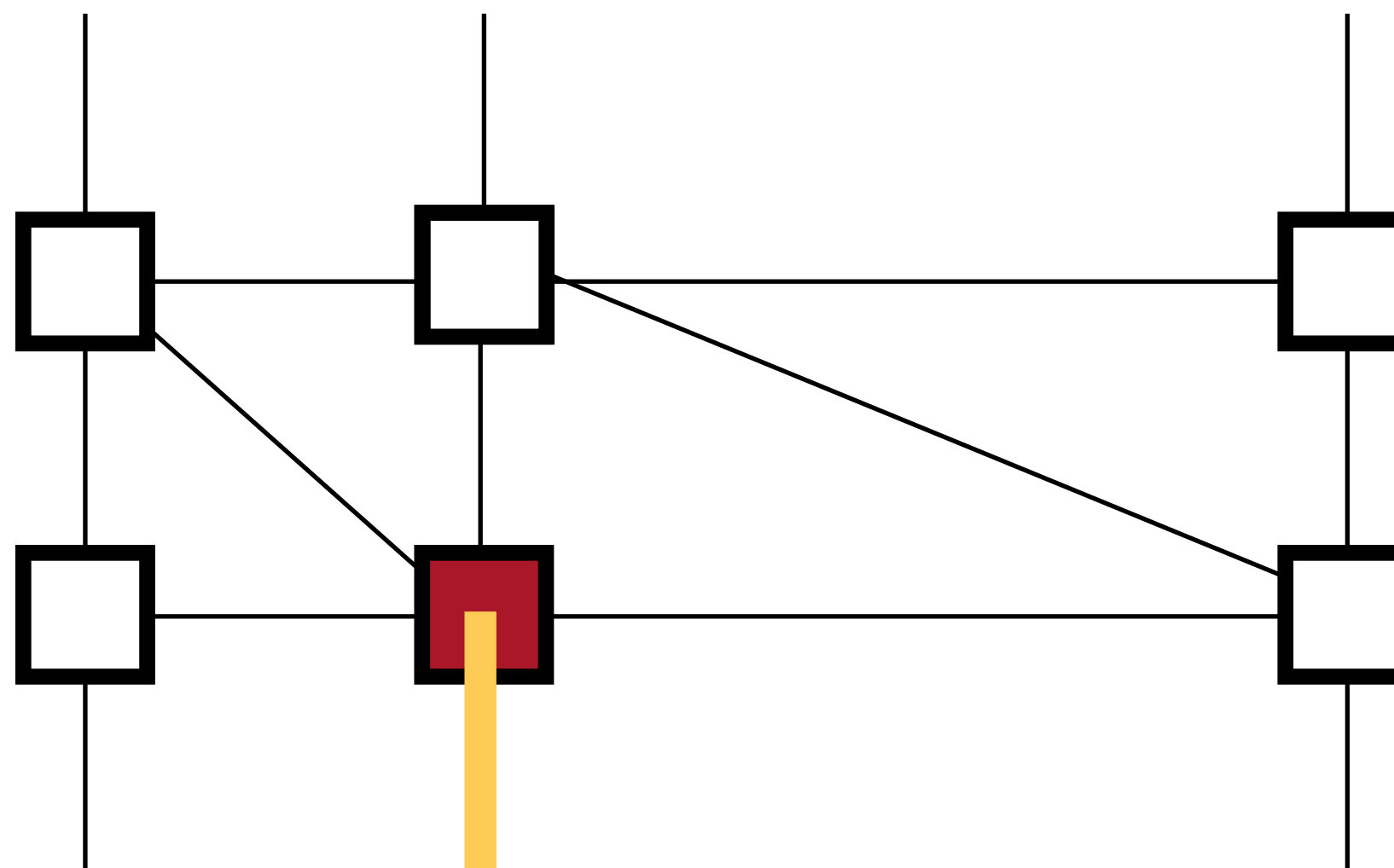


Detector Error Model

- First M is detector
- XOR is taken for subsequent

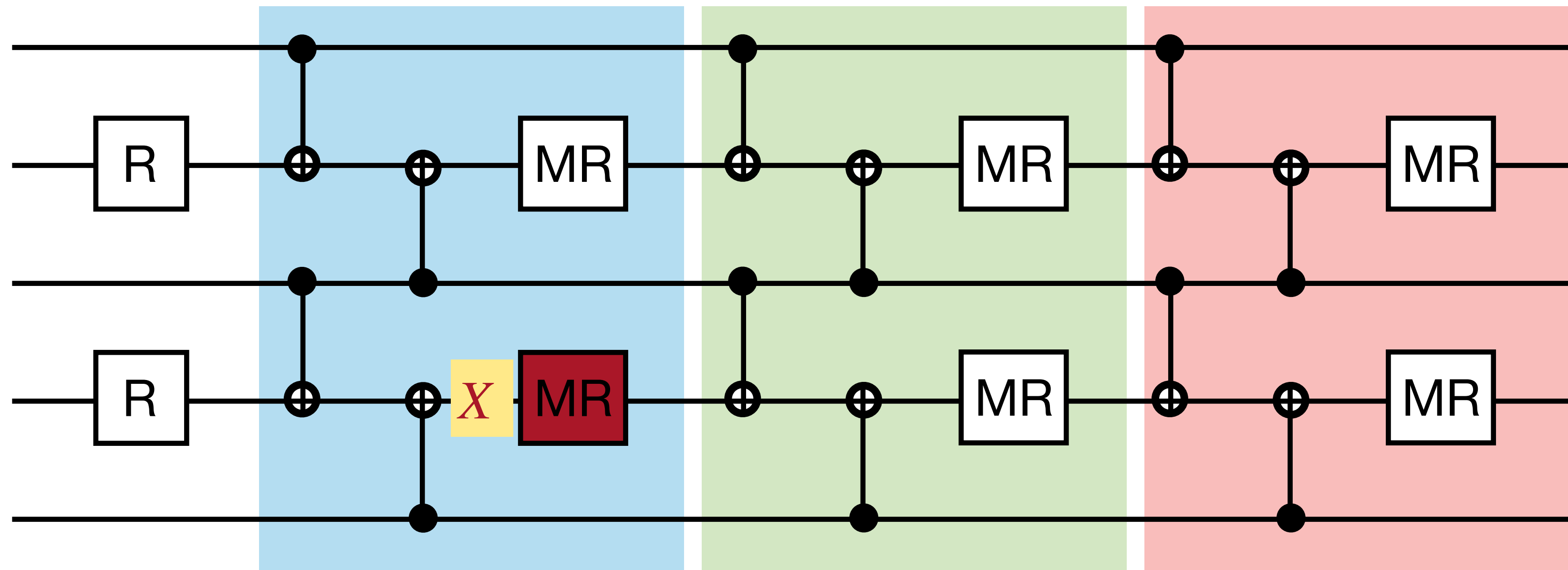
$$d_i = m_i \oplus m_{i-1}$$

- Edge indicates error mech.



Circuit-level decoding graph in rep. code

Circuit

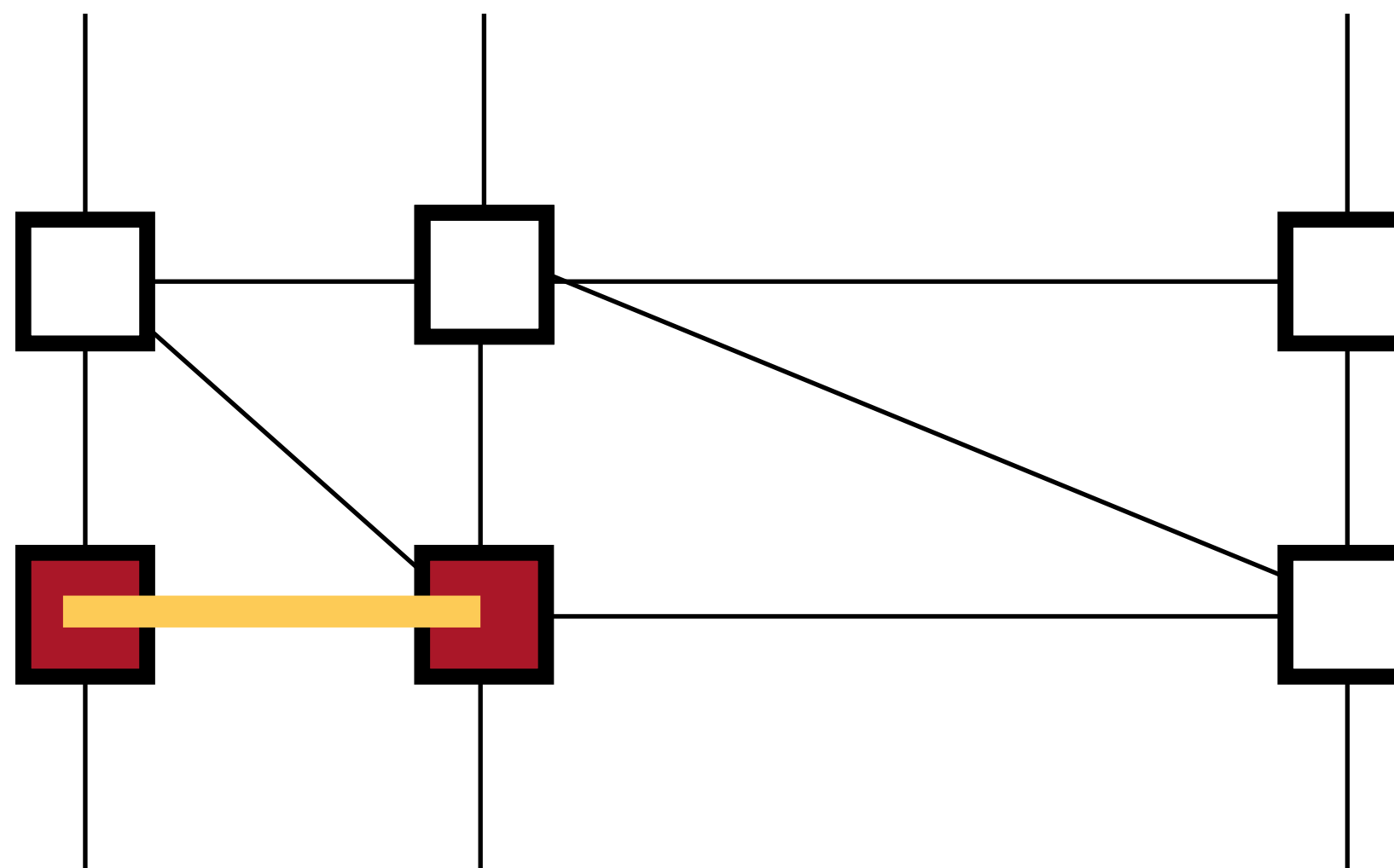


Detector Error Model

- First M is detector
- XOR is taken for subsequent

$$d_i = m_i \oplus m_{i-1}$$

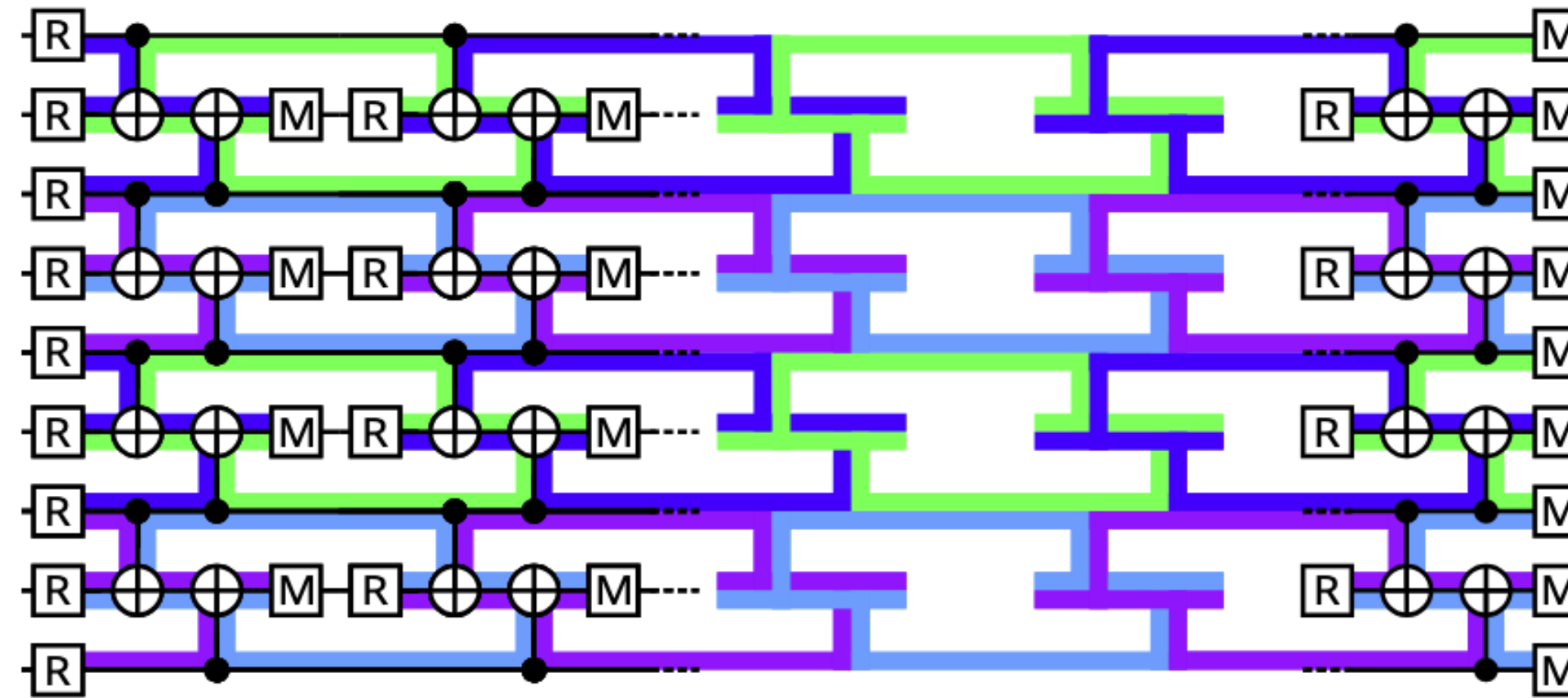
- Edge indicates error mech.



Circuit-level decoding graph in rep. code

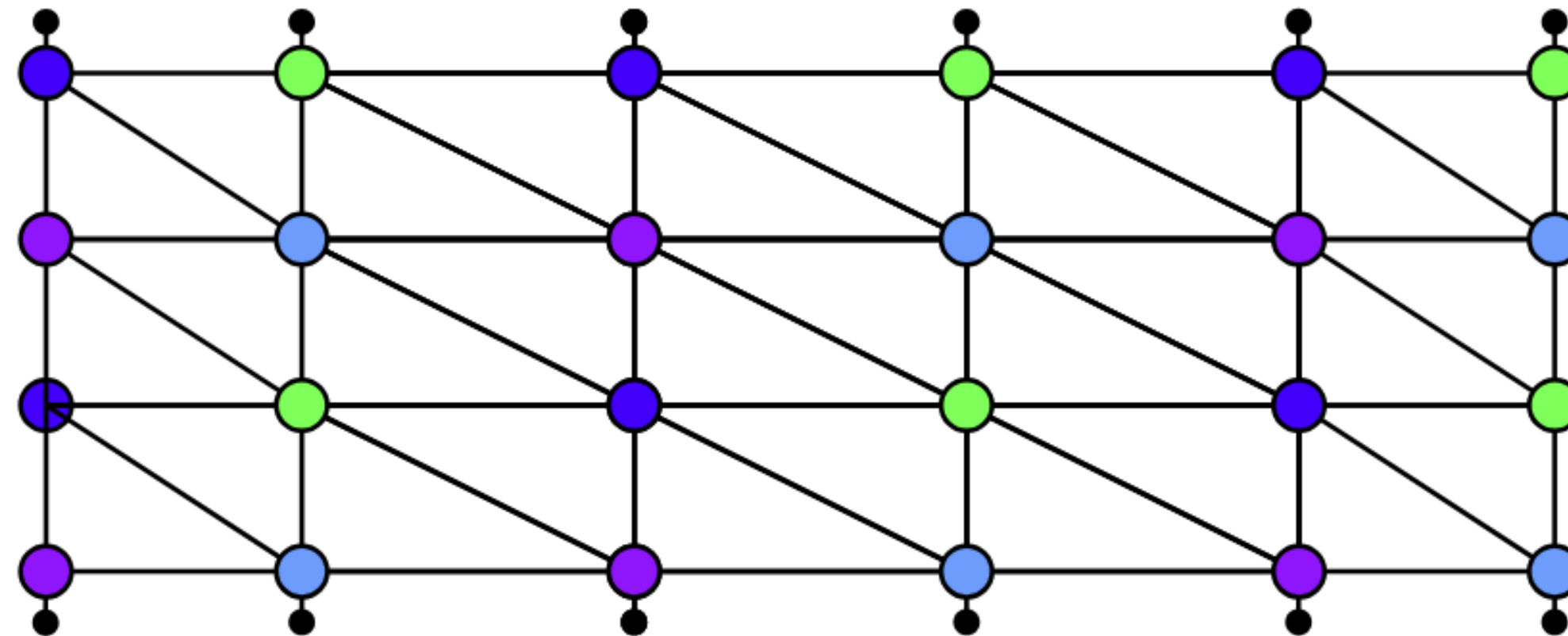
b) All detecting regions for a distance-5 code run for 4 cycles

Circuit

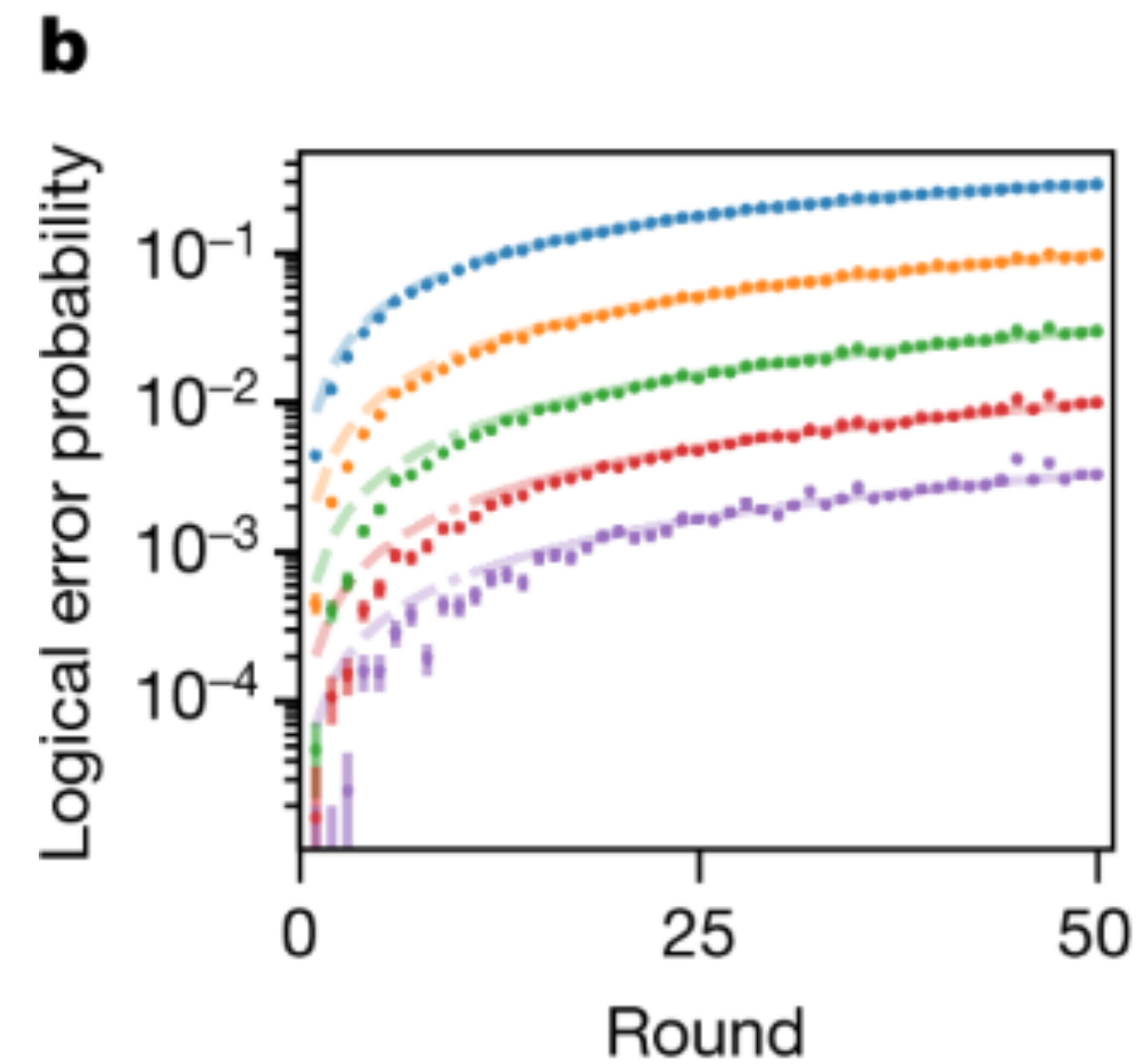
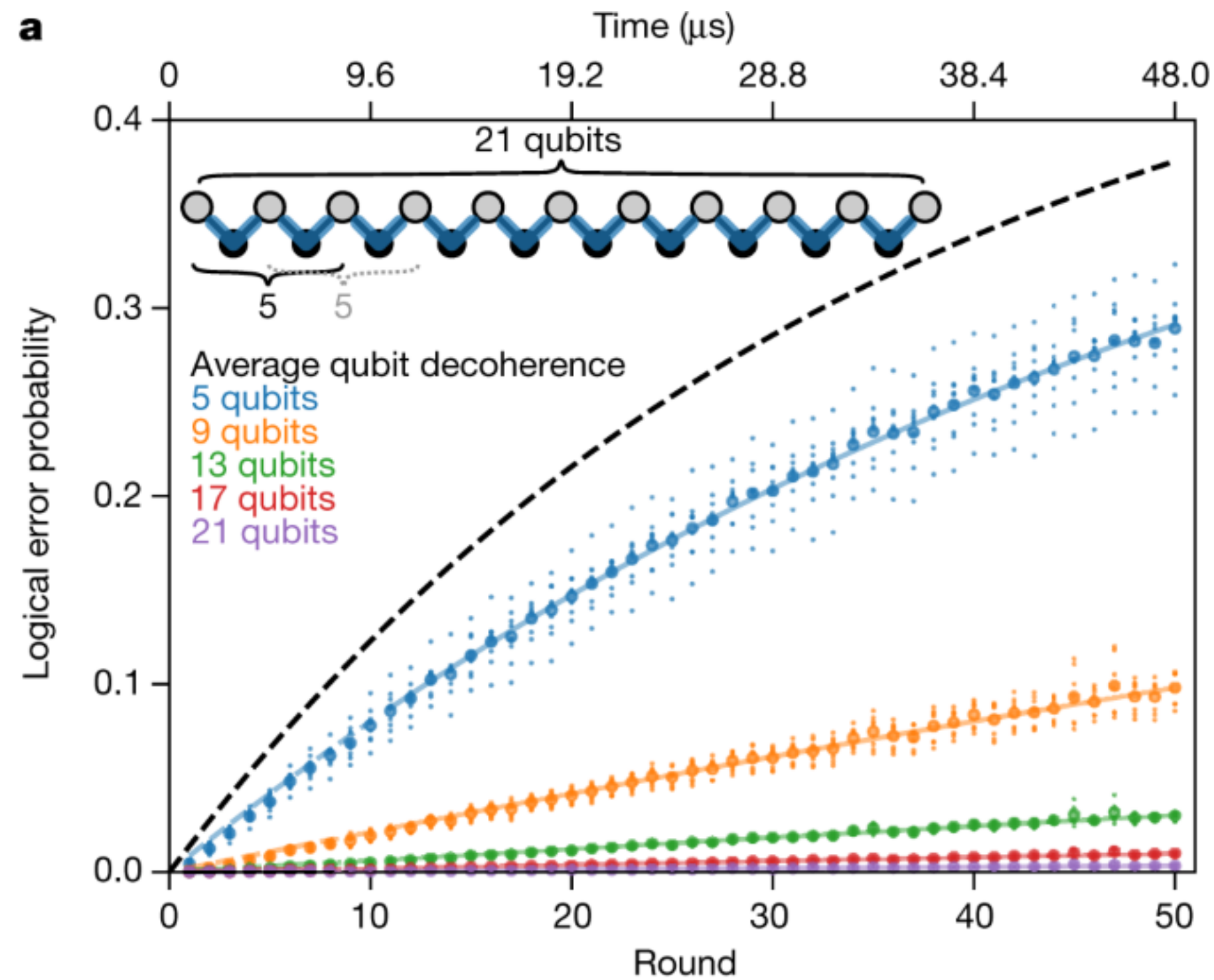


c) Error graph under an independent bit-flip error model

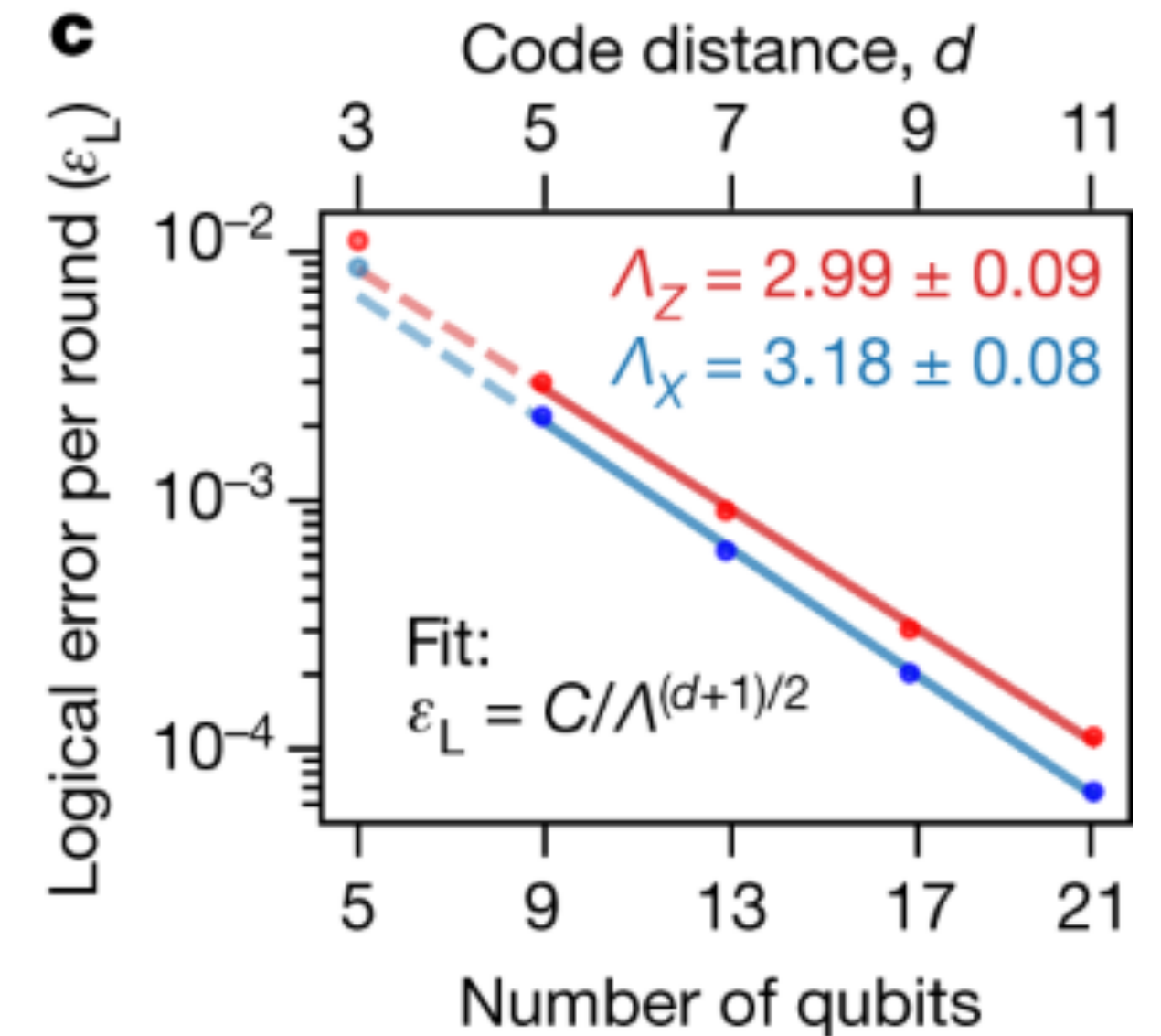
Detector Error Model



Circuit-level decoding graph in rep. code



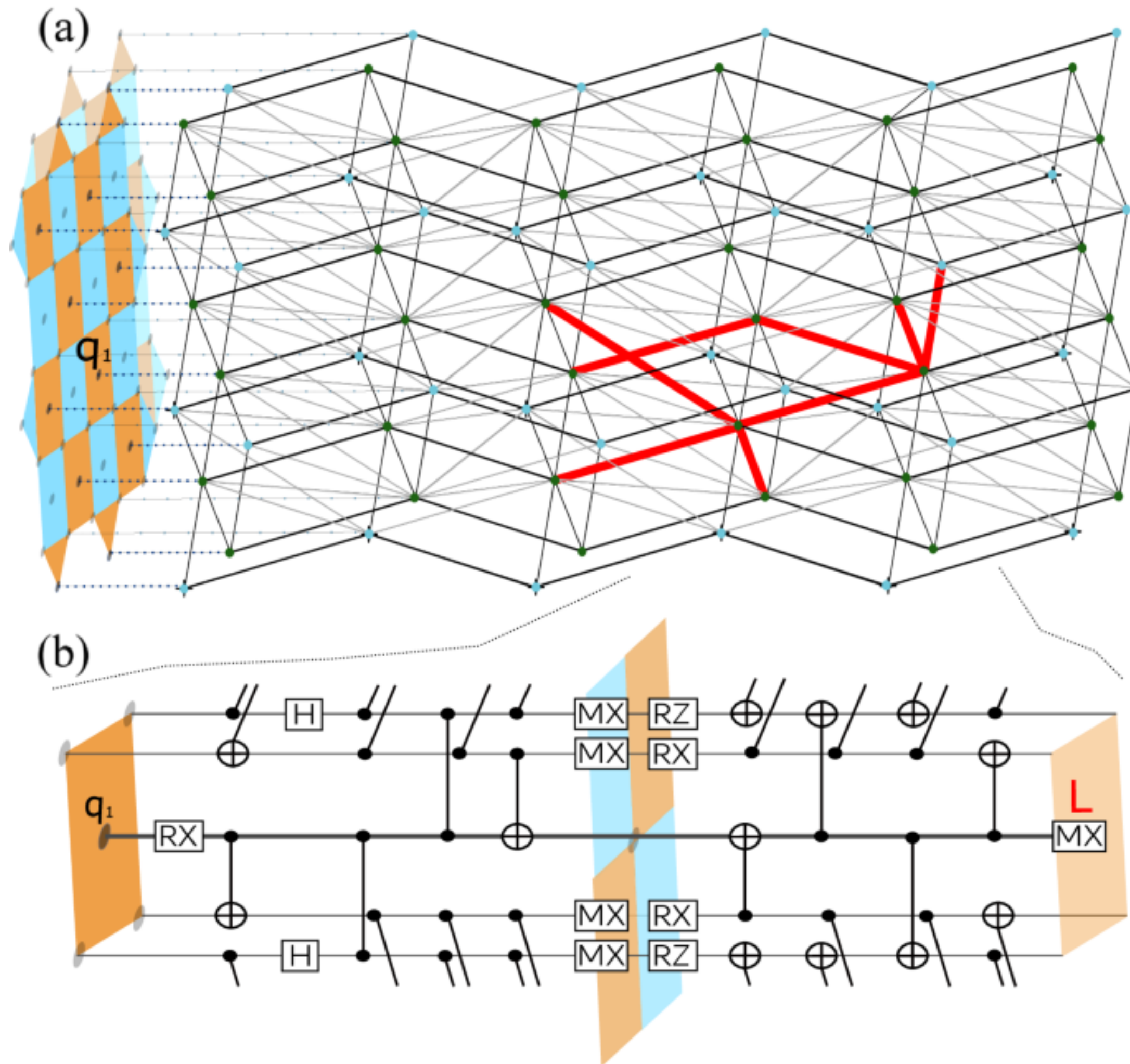
Google Quantum AI, Nature ('21)



- Exponential suppression of X/Z noise observed in repetition code (decoded with matching)

$$\epsilon_L \propto 1/\Lambda^{\lfloor \frac{d+1}{2} \rfloor}$$

Circuit-level decoding graph in surface code



For surface code, spacetime structure is reflected in 3D graph for DEM

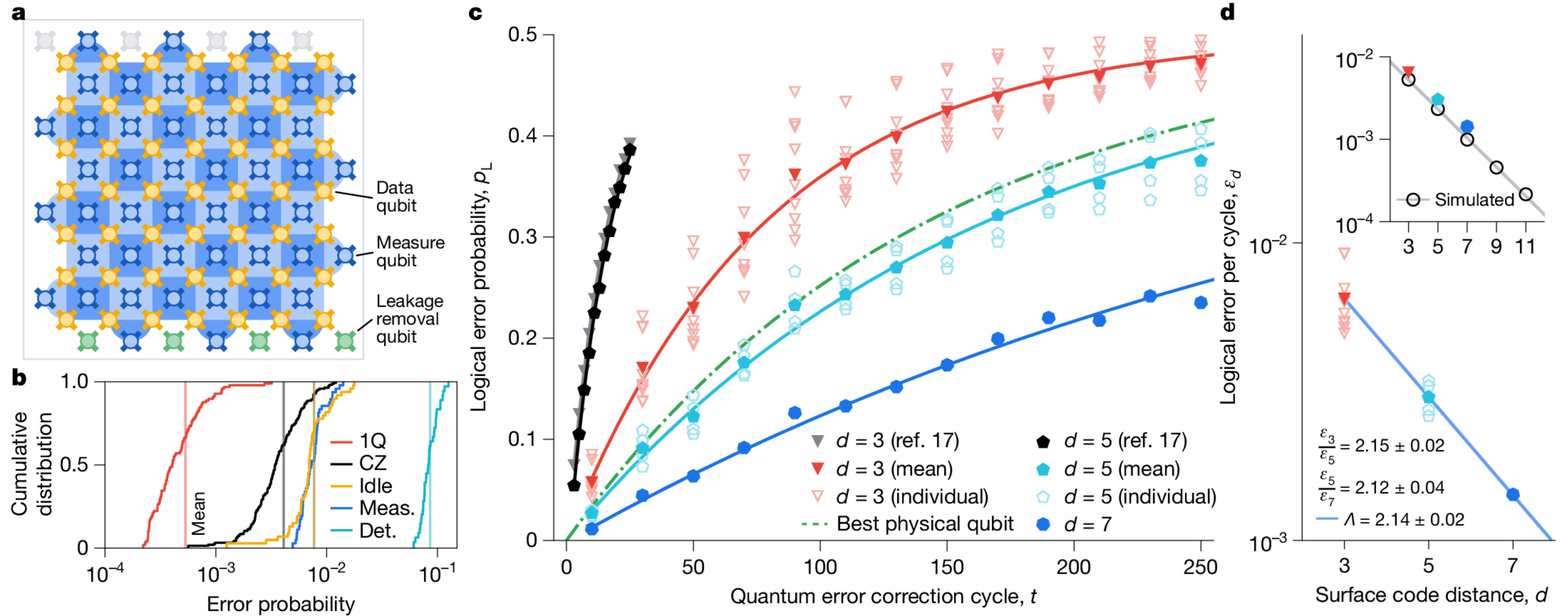
Usually Y errors are decomposed; decoders run independently for X and Z graph.

In exchange with computational cost, one may also run correlated matching.

	Uncorr., MWPM	Corr., MWPM
Threshold p_{th}	0.82%	0.94%

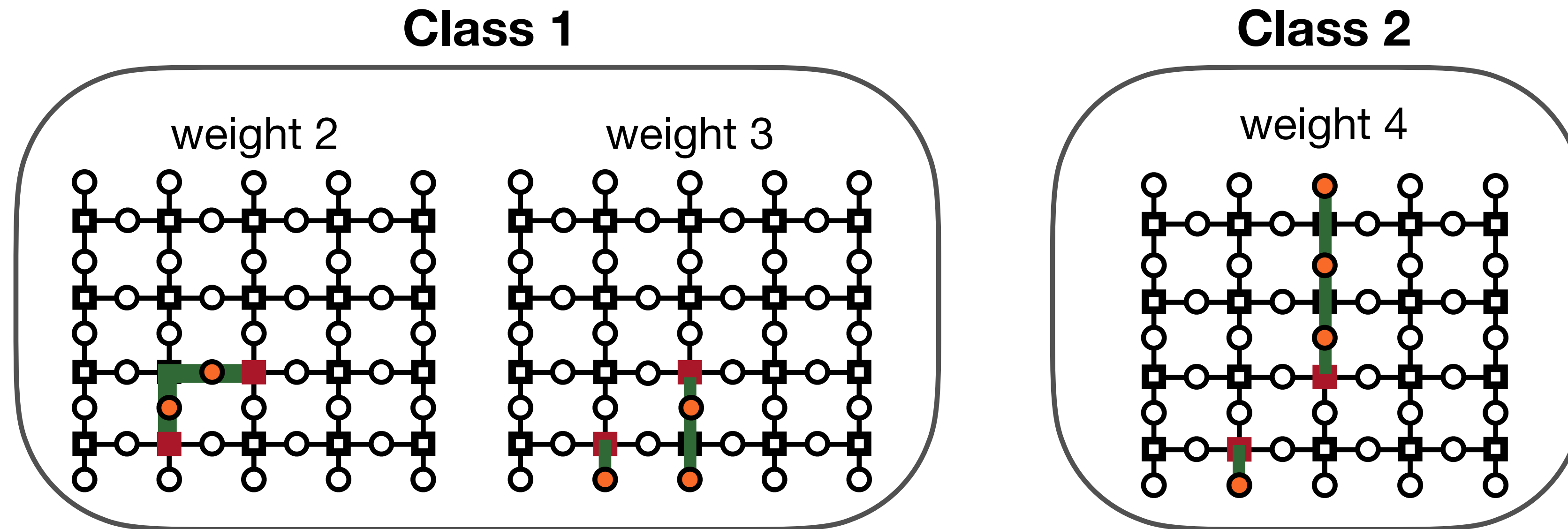
Higgott et al., PRX ('23)

Memory experiment for surface code



Decoder confidence for postselection

Decoder confidence



$\Pr(E)$

$$O(p^2)$$

$$O(p^3)$$

$$O(p^4)$$

$L(E)$

$$2 \log(1/p)$$

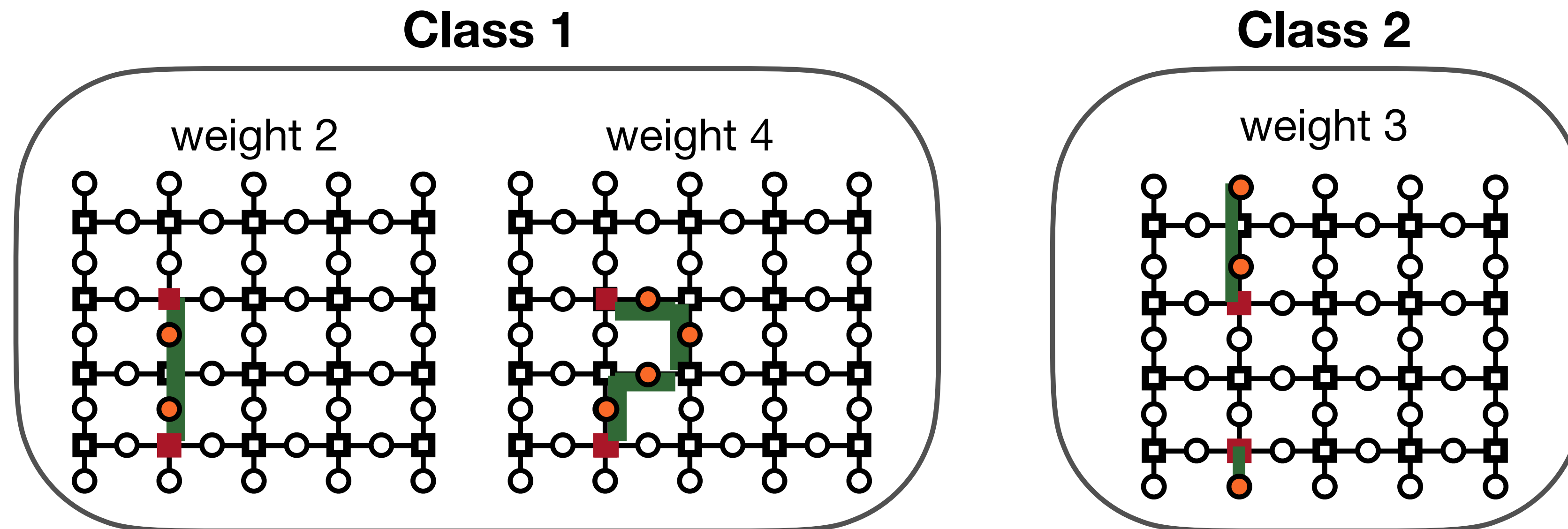
$$3 \log(1/p)$$

$$4 \log(1/p)$$

Most likely

Conditional error rate is $O(p^2)$ in this case

Decoder confidence



$\Pr(E)$

$O(p^2)$

$O(p^4)$

$O(p^3)$

$L(E)$

$2 \log(1/p)$

$4 \log(1/p)$

$3 \log(1/p)$

Most likely

Conditional error rate is $\underline{O(p)}$ in this case

The gap of log likelihood can be interpreted as decoder confidence

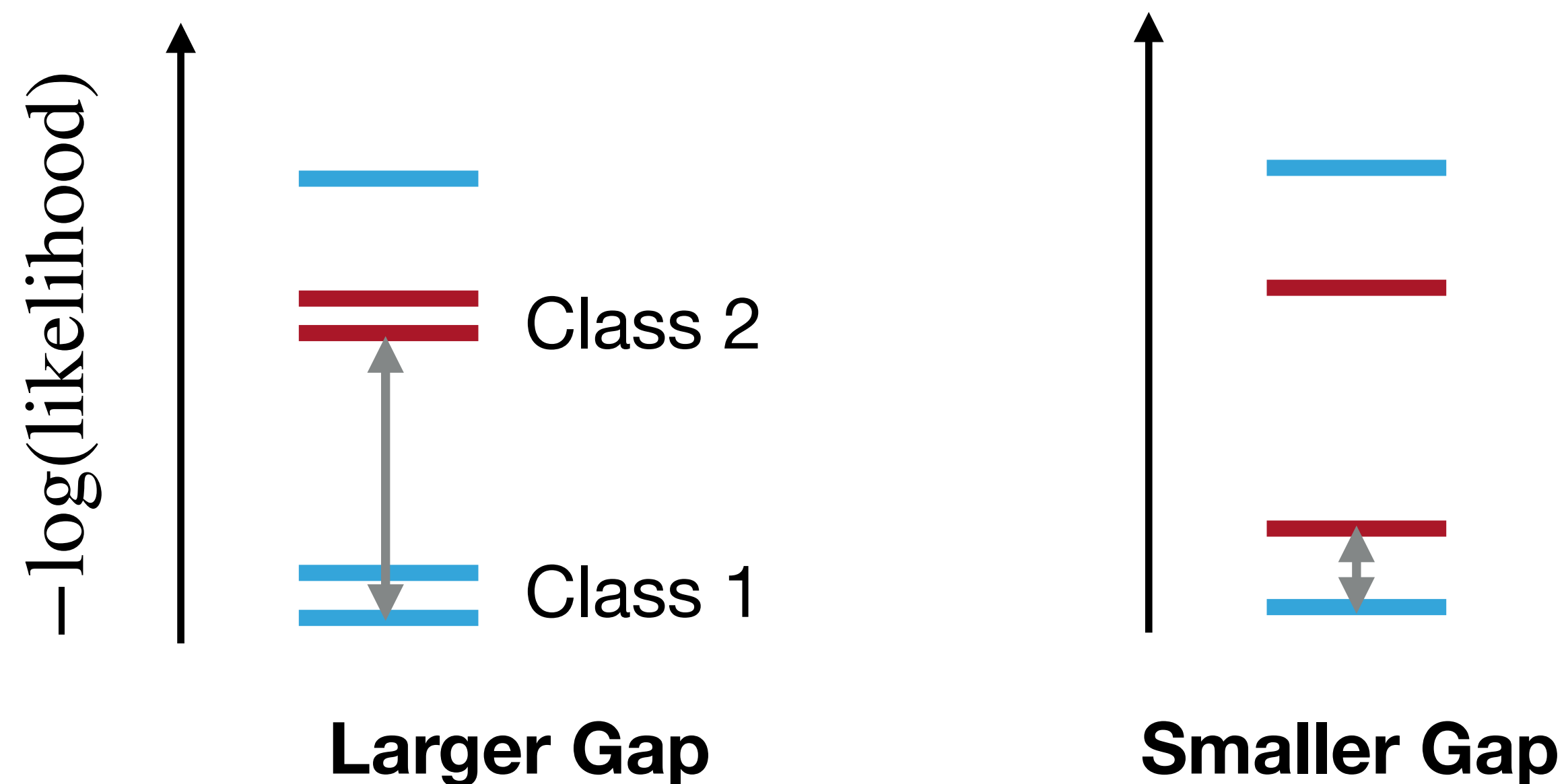
Complementary gap

Negative log likelihood of Error E :

$$L(E) = -\log P(E) = \sum_{i=1}^N \eta_i \log \left(\frac{1-p}{p} \right) + \text{const.}$$

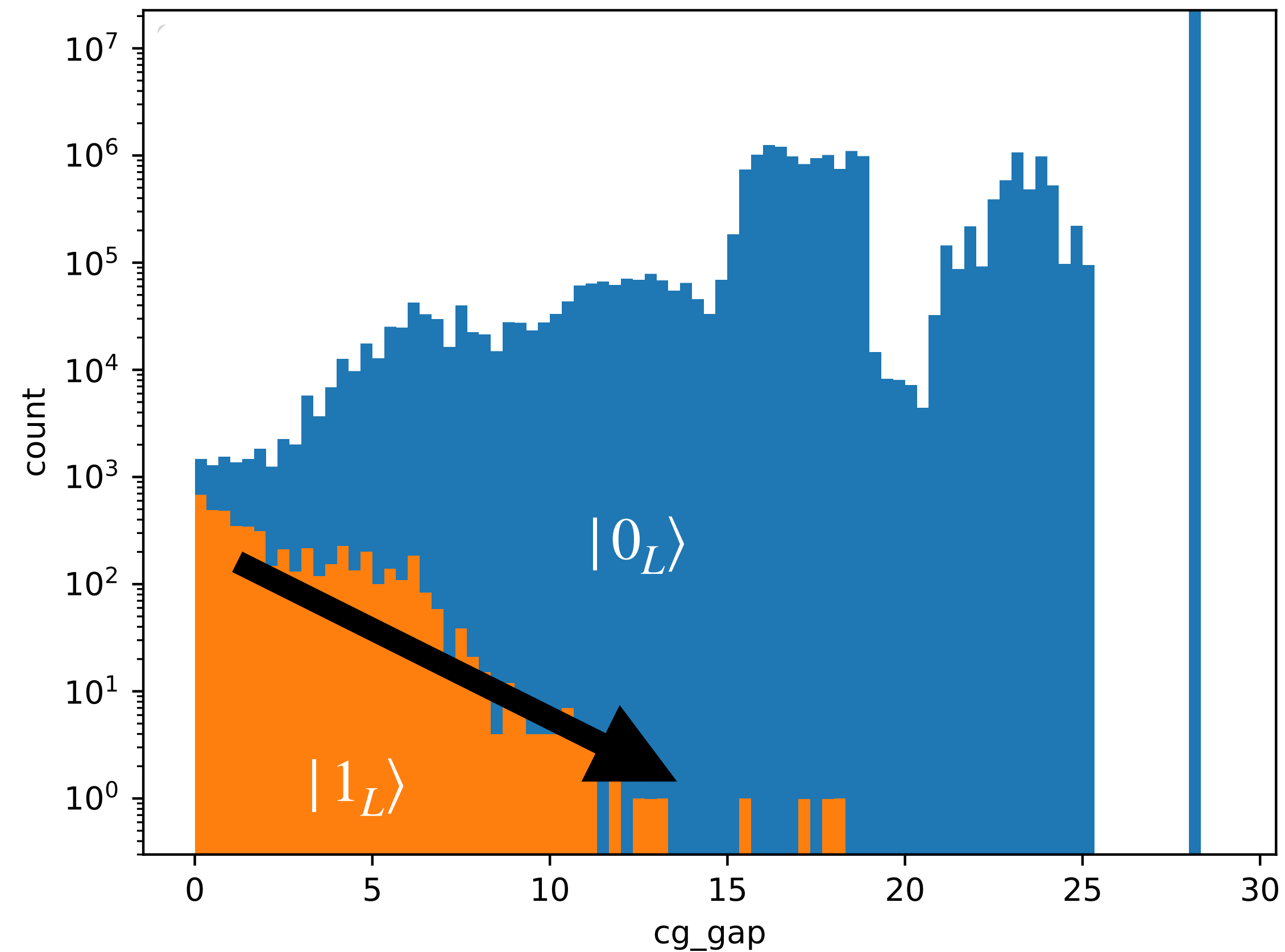
Difference of log likelihoods with different logical cosets

$$(\text{Gap}) = \left| \min_{E \in \mathcal{E}_1} L(E) - \min_{E \in \mathcal{E}_2} L(E) \right|$$



Distribution of complementary gap

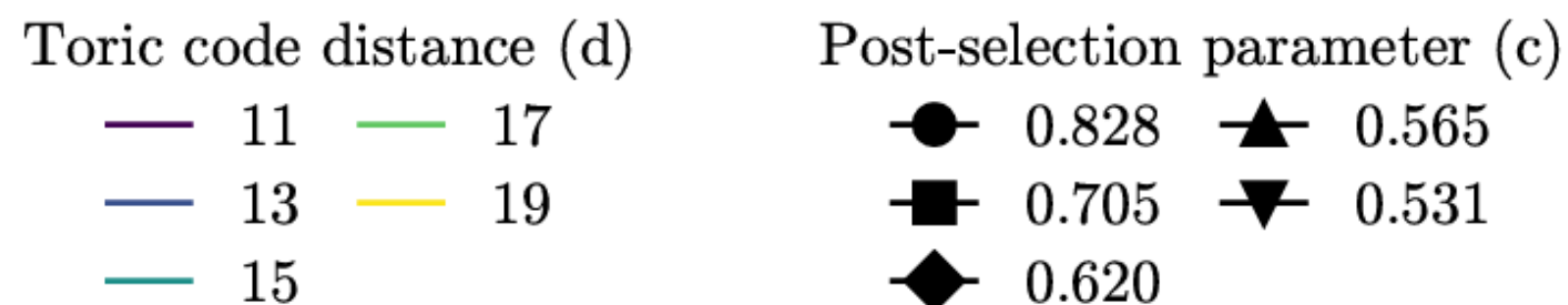
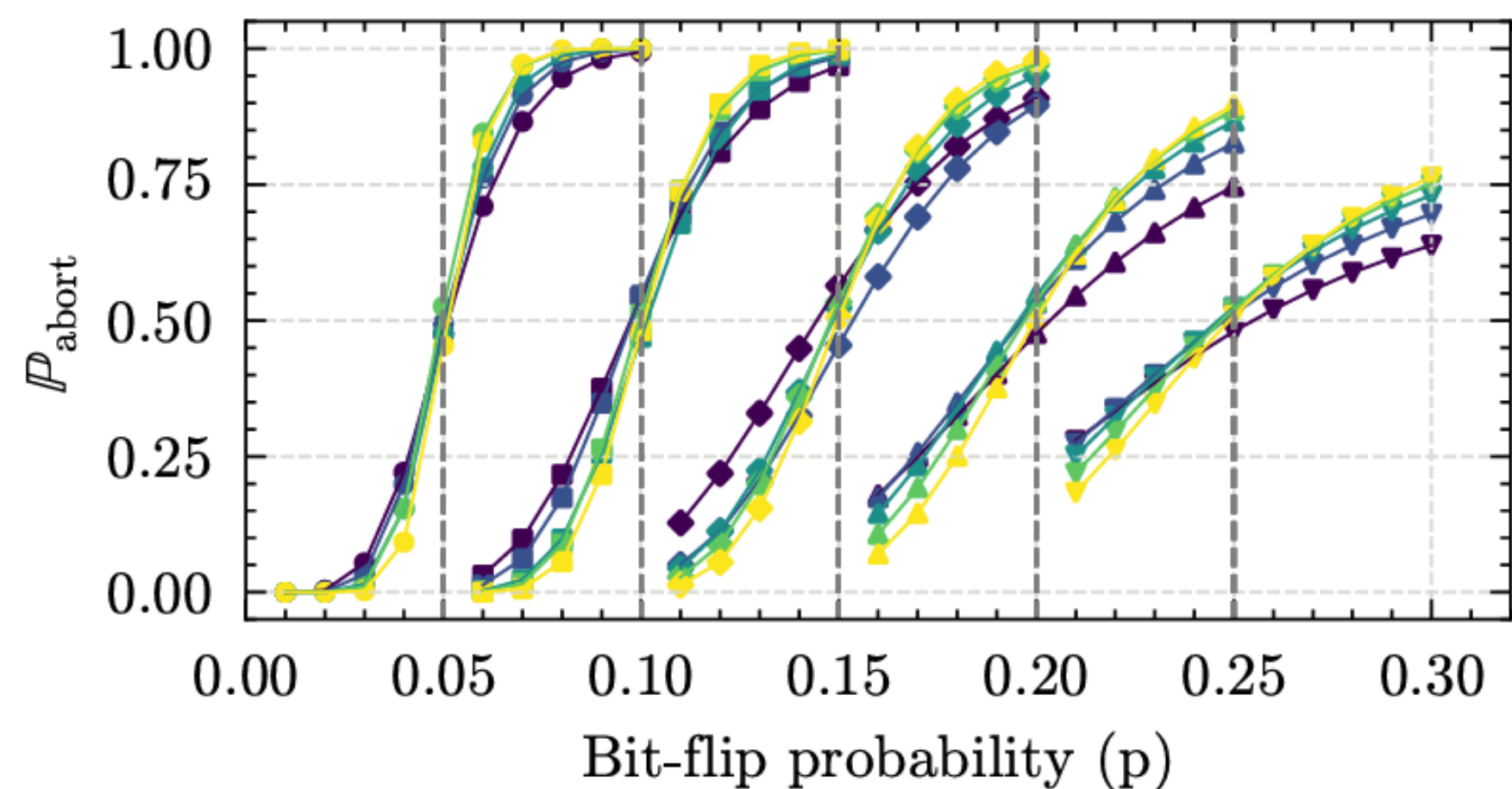
Memory experiment for $|0_L\rangle$
Surface code, $d=5$, 2 rounds, $p = 10^{-3}$



Logical error with larger CG is exponentially unlikely

Distribution of complementary gap

Theorem 1. *Post-selection that aborts if MLD returns a maximum coset probability less than some $\gamma \in [0, 1]$ is optimal in the following sense. Such post-selection partitions the errors \mathcal{E} into an abort set $\mathcal{E}_{\text{abort}}$ of some cardinality $|\mathcal{E}_{\text{abort}}| = r$. This partition strictly upper bounds \mathbb{P}_{succ} for any abort set with equivalent cardinality r .*



Problem of complementary gap

For data patch with k logical qubits, one must run the decoding for 4^k logical coset

There are various situations where this becomes a problem:

- Multi-patch lattice surgery for surface code

- High-rate codes

 - BB code family such as $[[144, 12, 12]]$

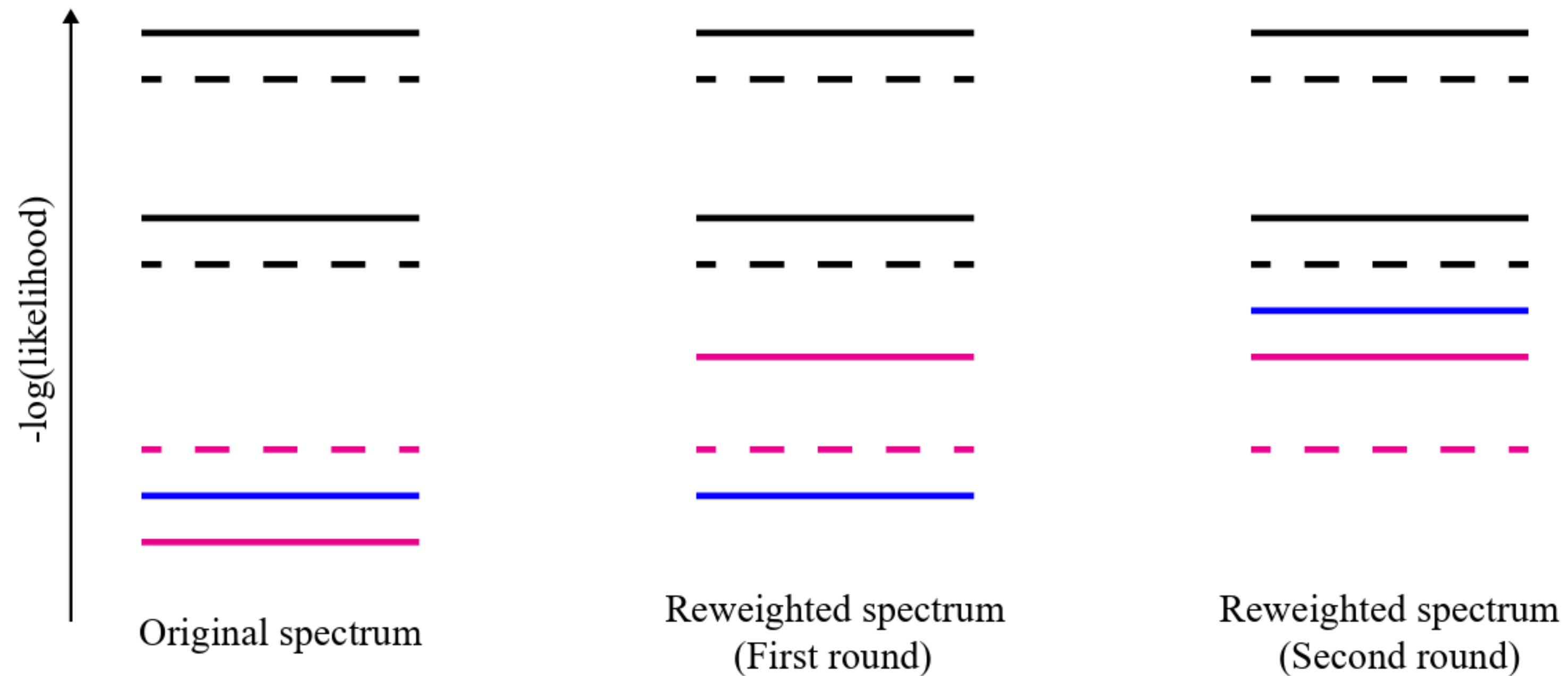
 - LP code family such as $[[2610, 744, \leq 16]]$

Surrogate way for postselection

Argument reweighting

Xie, NY, Tsubouchi, Li, arXiv:2601.17757

Run the decoding $O(1)$ times with perturbed likelihood to ensure that decoding is robust.



Surrogate way for postselection

Argument reweighting

Xie, NY, Tsubouchi, Li, arXiv:2601.17757

Run the decoding $O(1)$ times with perturbed likelihood to ensure that decoding is robust.

Physical-error criterion (PEC)

Input: Likelihood function L_1 , Number of rounds R

- Obtain correction $\hat{E}_1 = \arg \min L(E)$ from ordinary decoding.
- Construct a new function L_2 by penalizing error location \hat{E}_1
- Run the decoding, obtain $\hat{E}_2 = \arg \min L_2(E)$.

Output: REJECT if $\hat{E}_1 \neq \hat{E}_2$, otherwise ACCEPT

Surrogate way for postselection

Argument reweighting

Xie, NY, Tsubouchi, Li, arXiv:2601.17757

Run the decoding $O(1)$ times with perturbed likelihood to ensure that decoding is robust.

Logical-error criterion (LEC)

Input: Likelihood function L_1 , Number of rounds R

Run the ordinary decoding. Obtain correction $\hat{E}_1 = \arg \min L(E)$

for $1 \leq r \leq R$

Construct a new function L_r by penalizing error location \hat{E}_{r-1}

Run the decoding, obtain $\hat{E}_r = \arg \min L_r(E)$.

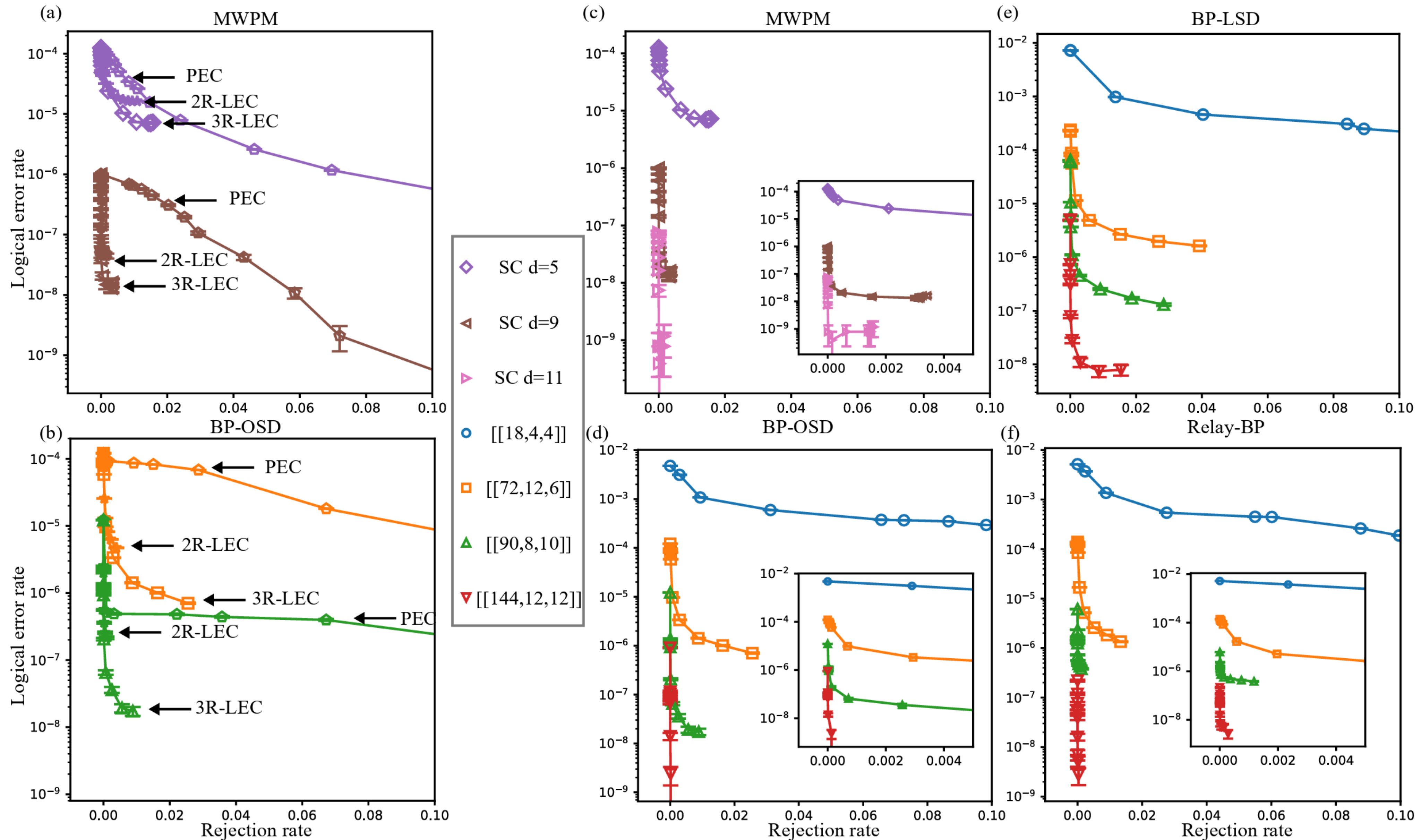
return REJECT if $C(\hat{E}_r) \neq C(\hat{E}_{r-1})$ C : logical coset

Surrogate way for postselection

Decoder	App.	Decoder	App.
MWPM [23, 24]	✓	BP-OSD [26, 27]	✓
Union Find [32, 33]	✓	BP-LSD [34]	✓
BP [25]	○	BP-AC [35]	○
Mem-BP [36]	○	Tesseract [37]	○
BP-GD [31, 38]	○	DTDs [39]	○
Relay-BP [13]	✓	Neural Network [12]	✗

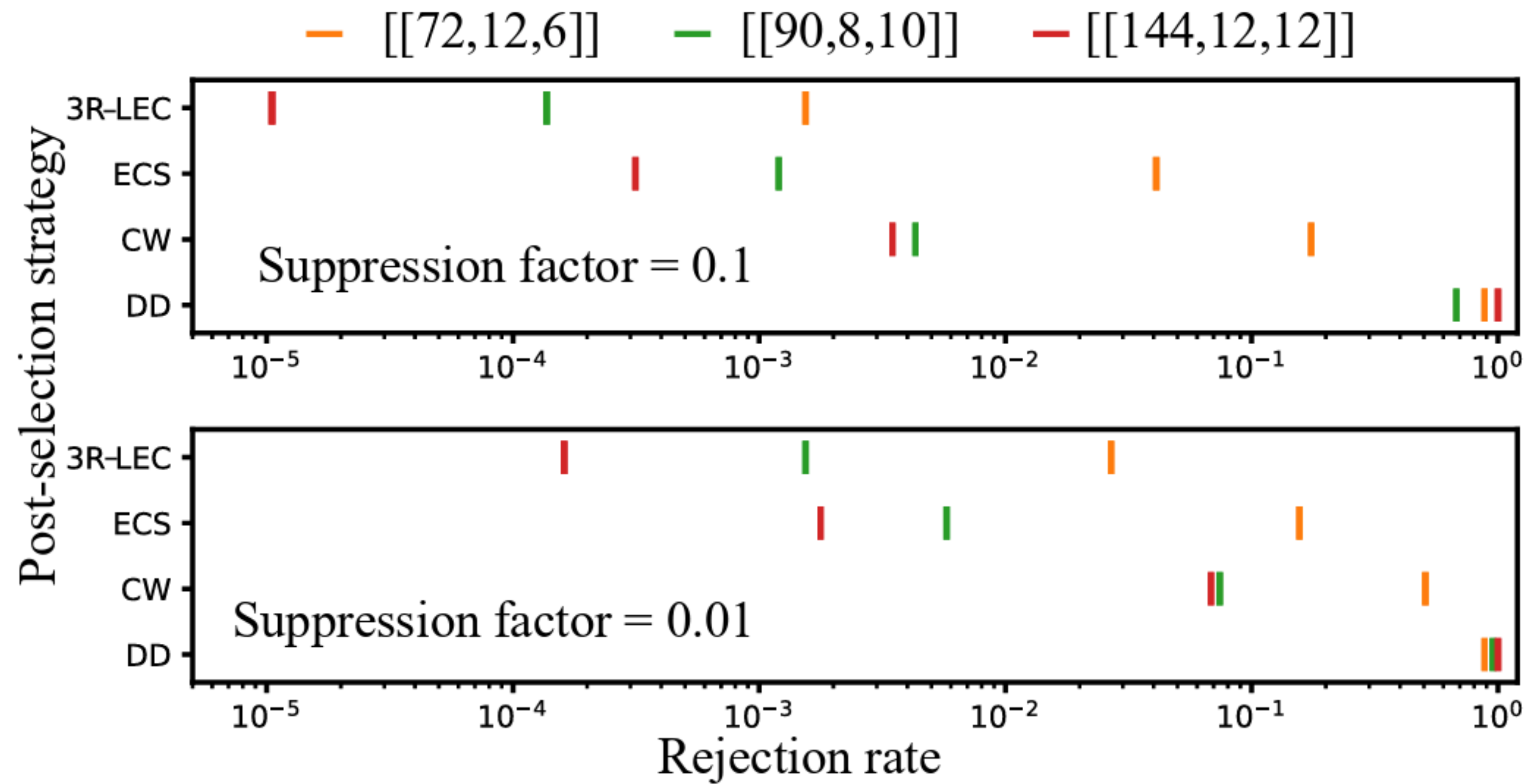
Numerical demonstration of argument reweighting

Xie, NY, Tsubouchi, Li, arXiv:2601.17757



Numerical demonstration of argument reweighting

Xie, NY, Tsubouchi, Li, arXiv:2601.17757



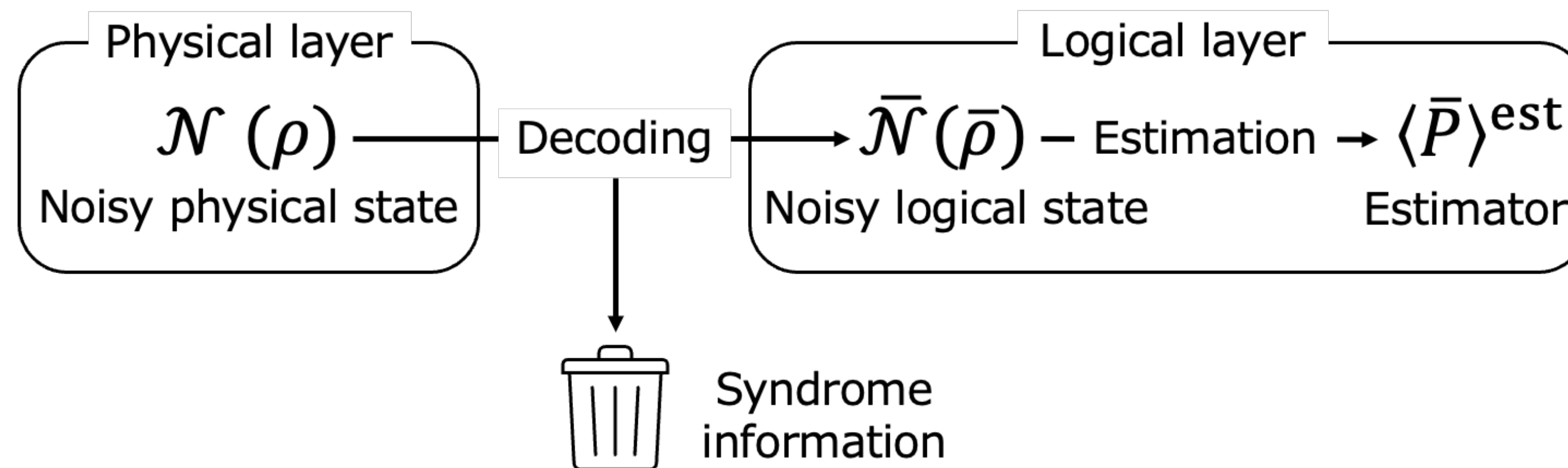
Open problems

- What is the fundamental limit of improvement by postselection?
- Currently the argument reweighting is not as good as complementary gap (MLD) for MWPM.
How severe is the performance difference for qLDPC?
- How does argument reweighting behave under neural network decoders?
Infinite-width networks shall allow analysis via neural tangent kernel

Syndrome-aware estimation

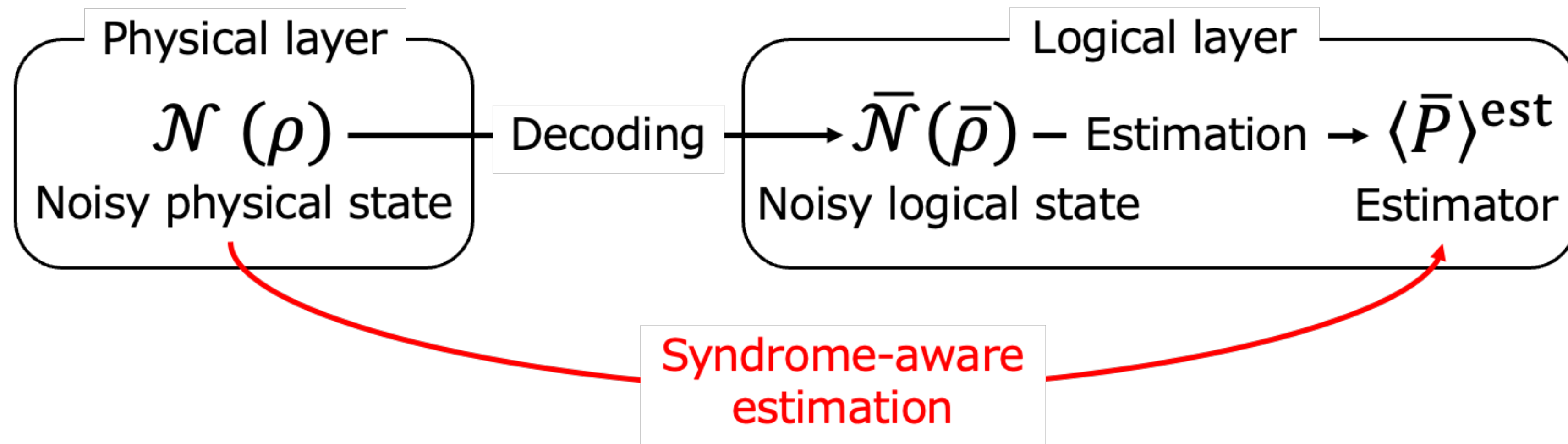
Syndrome for estimation

In usual setting, syndrome information is not used for logical estimation.



Syndrome for estimation

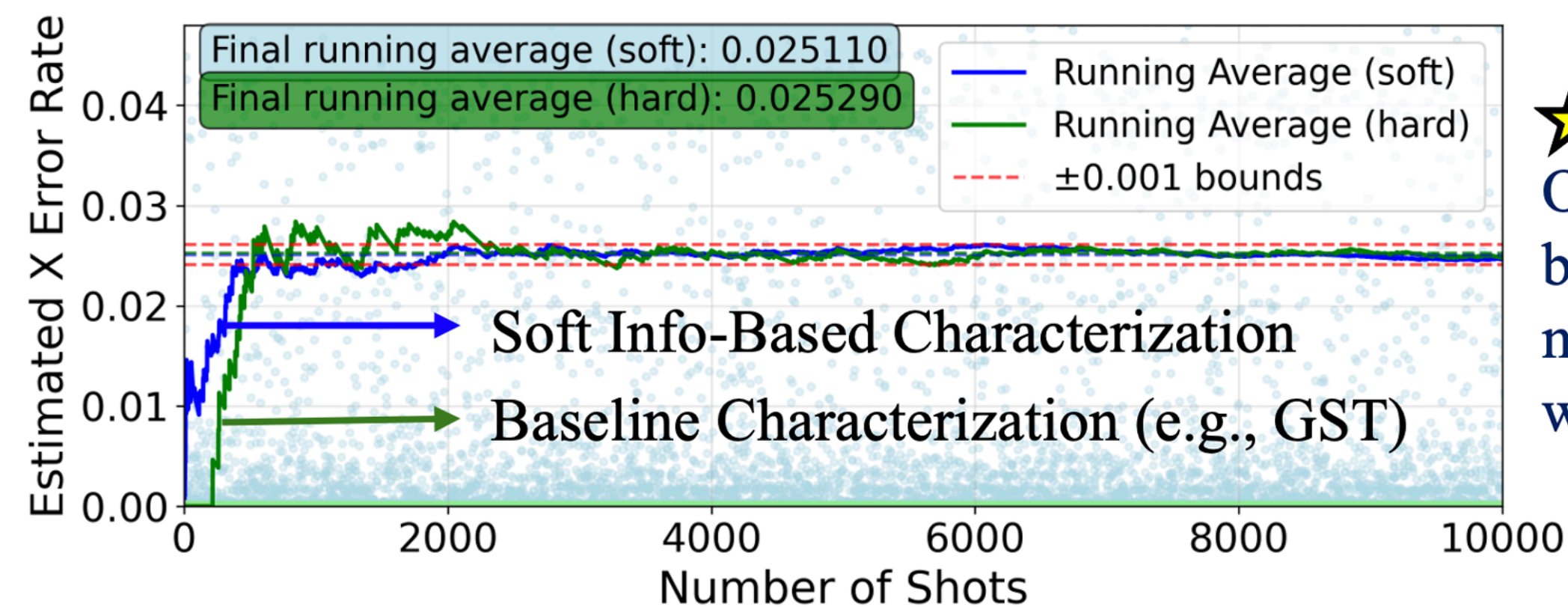
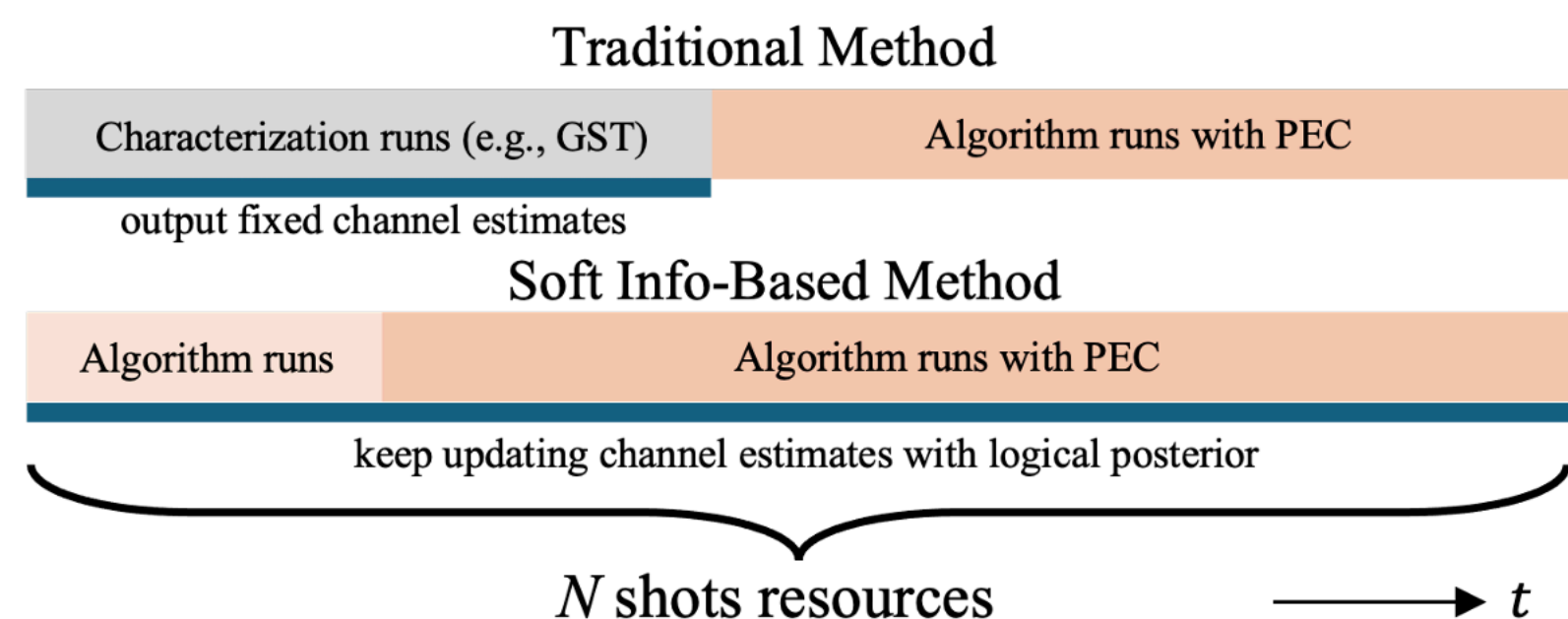
What if we perform observables estimation with the help of syndromes?



Examples

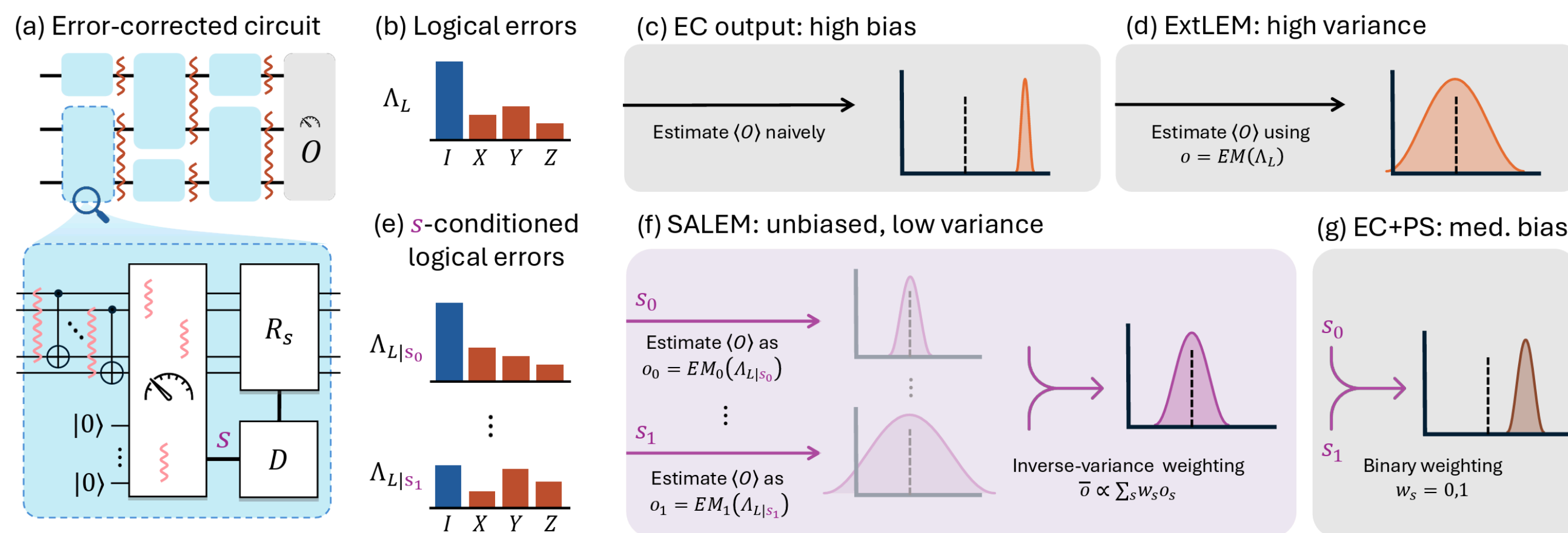
Syndrome info for skipping noise characterization for PEC

Zhou et al., arXiv:2512.09863

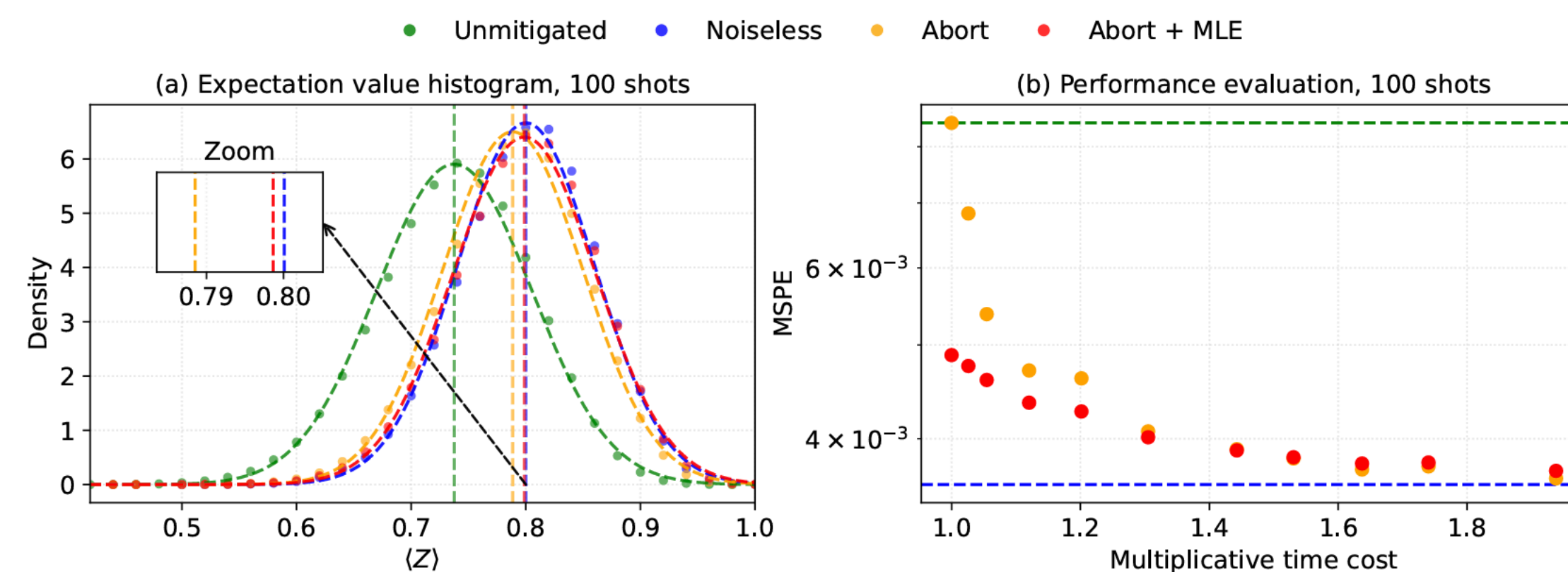


★ Advantage:
 Only the first batch of our method needs warm-up shots.

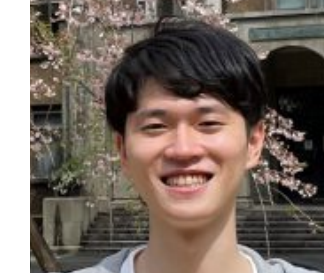
Confidence-aware ZNE



Aharonov et al., arXiv:2512.23810

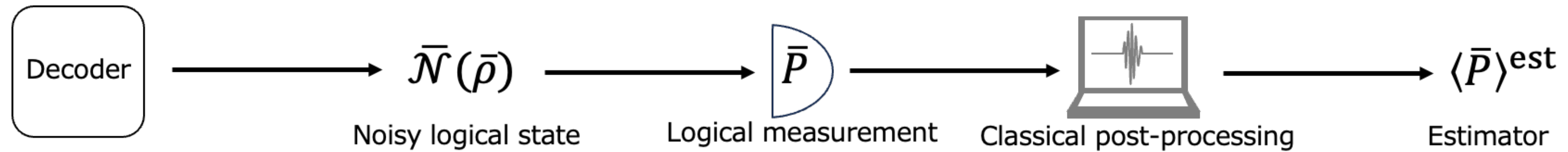


Dinca, Chan, Benjamin, arXiv:2512.15689



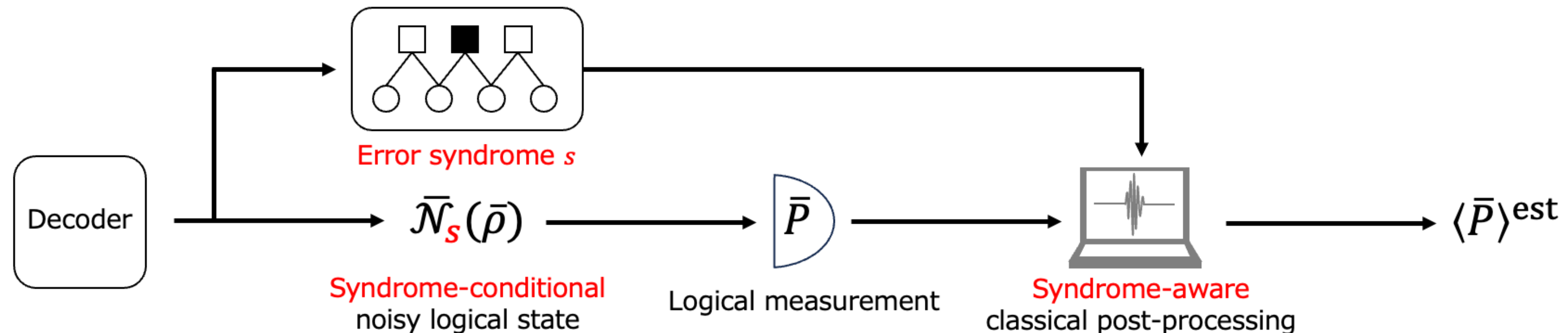
Syndrome-agnostic estimation

syndrome is only used by decoder.

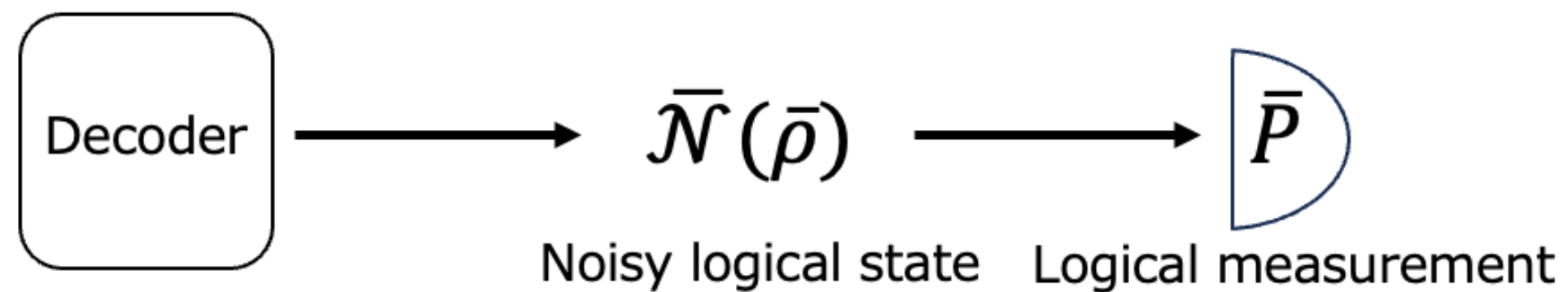


Classical syndrome-aware estimation

syndrome is used for postprocessing



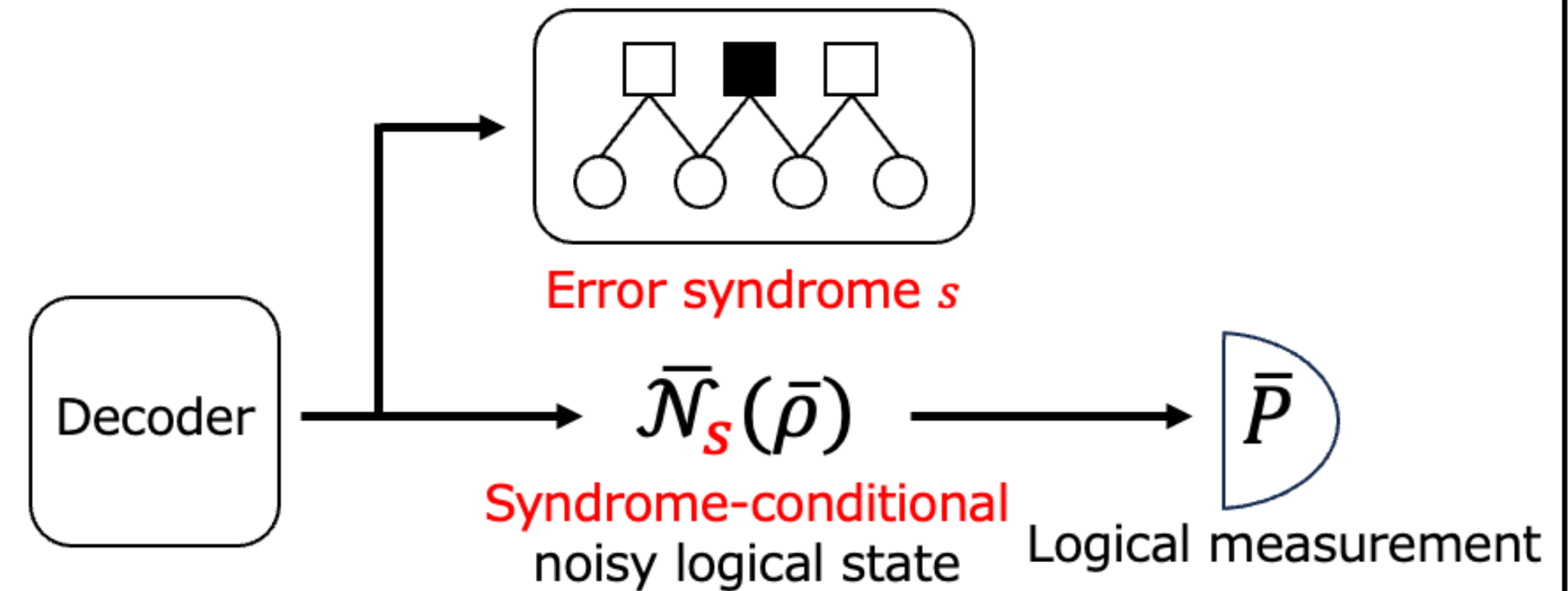
■ Syndrome-agnostic protocol



$$\Pr(x|\theta) = \frac{1 + x(1 - 2\epsilon)\theta}{2}$$

$\theta = \text{tr}[\bar{\rho}\bar{P}]$: Target expectation value
 $x = \pm 1$: Logical measurement result
 ϵ : Logical error rate

■ Classical syndrome-aware protocol



$$\Pr(x, s|\theta) = p_s \frac{1 + x(1 - 2\epsilon_s)\theta}{2}$$

s : Observed error syndrome
 p_s : Probability of obtaining error syndrome s
 ϵ_s : Syndrome-conditioned logical error rate
 Note that $\epsilon = \sum_s p_s \epsilon_s$

■ Classical Cramér-Rao bound:

- The accuracy limit of the estimator is bounded by **classical Fisher information** F_θ .

$$\boxed{\text{Var}[\theta^{\text{est}}]} \geq \frac{1}{\boxed{N}} \boxed{F_\theta^{-1}}$$

Accuracy Sampling overhead Classical Fisher information

Theorem: The effective logical error rate is lower bounded as

$$\frac{1 - \theta^2}{2} \epsilon \leq \epsilon^{\text{cSynd}} \leq \epsilon.$$

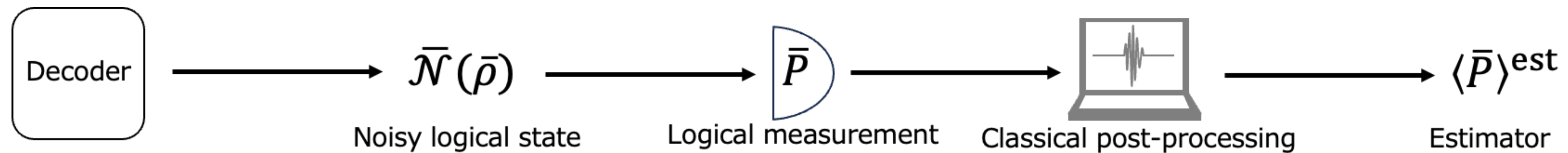
Especially for Haar-random target logical state, $\mathbb{E}_{\text{Haar}}[\theta^2] = 2^{-k}$, so we have

$$\frac{1}{2} \epsilon \lesssim \mathbb{E}_{\text{Haar}}[\epsilon^{\text{cSynd}}] \leq \epsilon.$$

Syndrome information can improve the effect of logical errors by at most a factor of two.

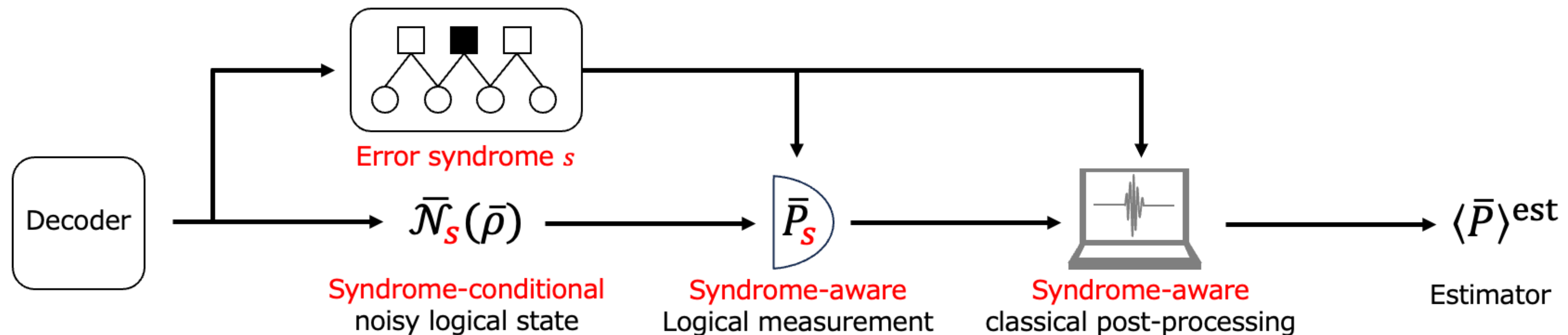
Syndrome-agnostic estimation

syndrome is only used by decoder.



Syndrome-aware quantum estimation

Measurement basis is optimized so that Fisher information is maximized.

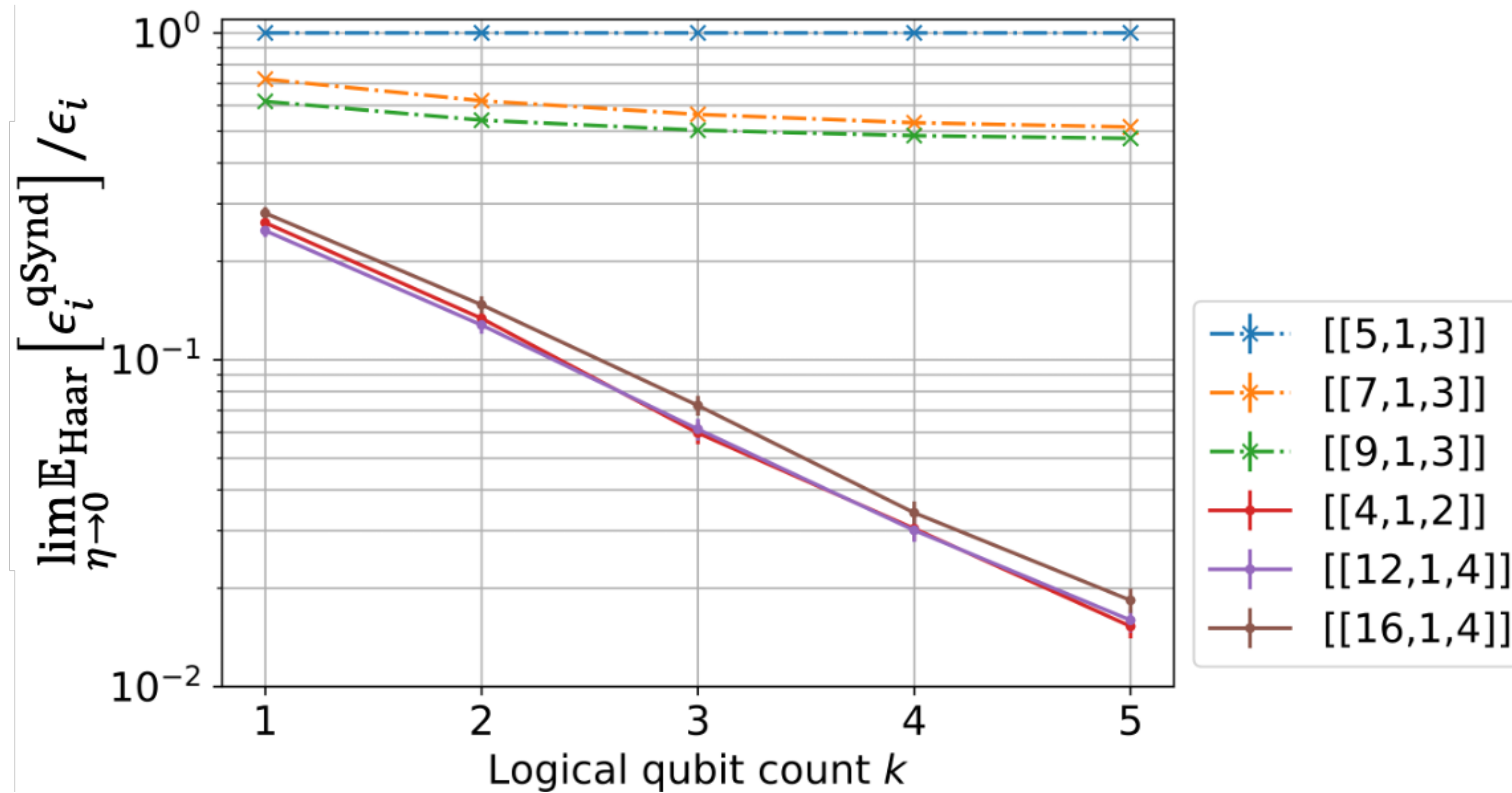


Theorem: There exists an even-distance code and a local noise model such that the ratio of the Haar-averaged effective logical error rate $\epsilon_i^{\text{qSynd}}$ to the original error rate ϵ_i decays exponentially with the number of logical qubits k in the limit of vanishing physical error rate η :

$$\lim_{\eta \rightarrow 0} \frac{\mathbb{E}_{\text{Haar}} \left[\epsilon_i^{\text{qSynd}} \right]}{\epsilon_i} = \frac{1}{2^{k+1}}.$$

- C.f. For the classical protocol, $\epsilon_i^{\text{cSynd}} / \epsilon_i \geq 1/2$.

Syndrome-conditioned quantum measurement allows one to exponentially reduce the effect of logical errors.



For instance, under a single erasure Q error:

$$(1 - p) |0\rangle\langle 0| \otimes \bar{\rho}(\boldsymbol{\theta}) + p |1\rangle\langle 1| \otimes \left(\frac{1}{2} \bar{\rho}(\boldsymbol{\theta}) + \frac{1}{2} \bar{Q} \bar{\rho}(\boldsymbol{\theta}) \bar{Q} \right)$$

$$= (1 - p) |0\rangle\langle 0| \otimes \bar{\rho}(\boldsymbol{\theta}) + p |1\rangle\langle 1| \otimes \bar{\rho}(\Lambda_{\bar{Q}} \boldsymbol{\theta}),$$

Optimal observable for measurement is

$$|0\rangle\langle 0| \otimes \frac{1}{1-p} (\bar{P}_i - p \bar{O}_i(\boldsymbol{\theta})) + |1\rangle\langle 1| \otimes \bar{O}_i(\boldsymbol{\theta}) \quad \text{where} \quad \bar{O}_i(\boldsymbol{\theta}) = \mathbf{e}_i^T \mathbf{C}(\boldsymbol{\theta})_{-+} (\mathbf{C}(\boldsymbol{\theta})_{++})^{-1} \bar{\mathbf{P}}_+$$

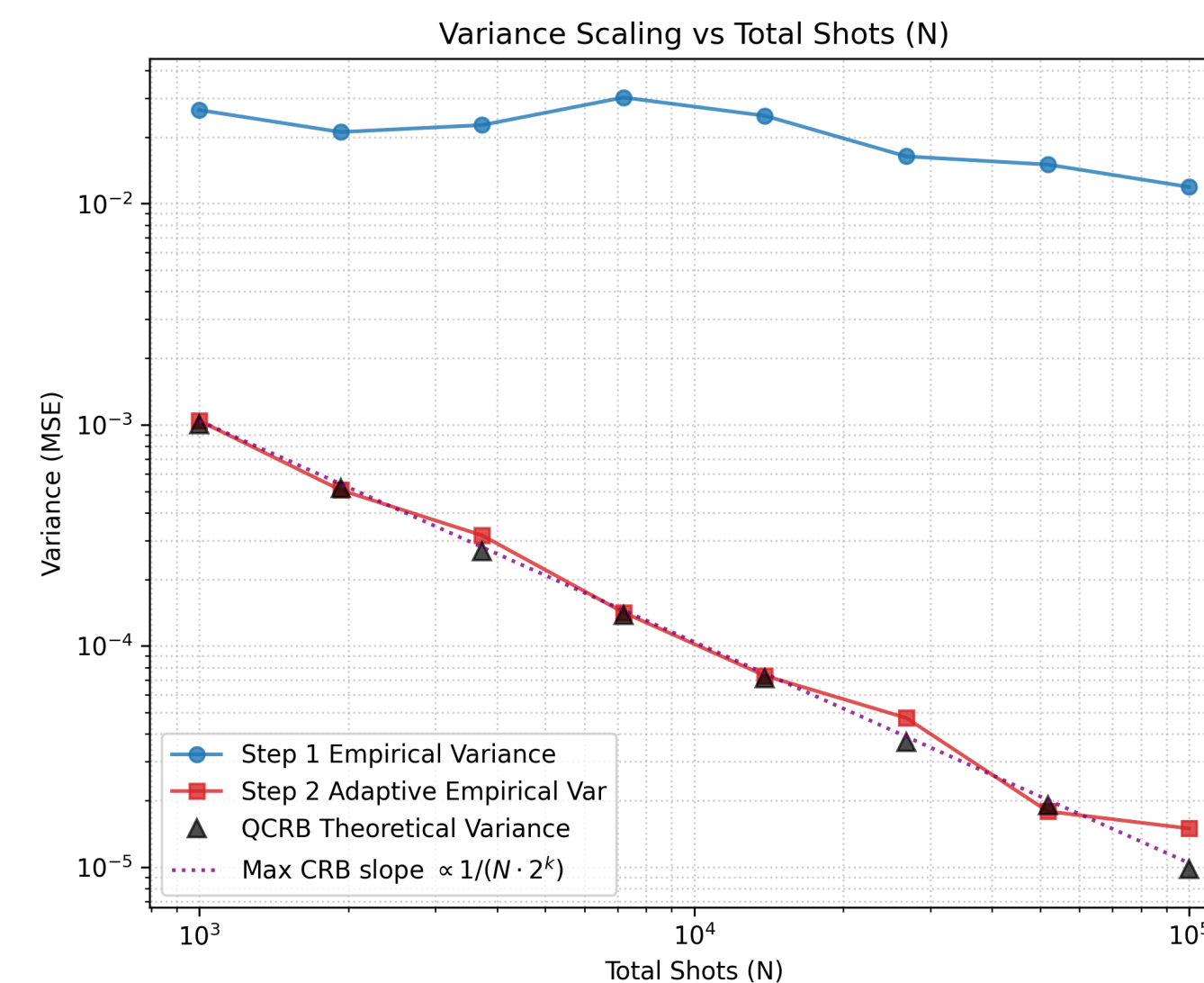
(weighted sum of Paulis that commute with Q)

Step 1: Run rough tomography using \sqrt{N} shots

construct the measurement basis to maximize QFI

Step 2: Run fine measurement using $N - \sqrt{N}$ shots

This is unbiased and achieves Cramer-Rao bound



Open problems

- How stable is the quantum advantage under fewer shots?
- What are class of quantum states that allow even more efficient learning?
- Is the two-step protocol the optimal, or even better protocol?
- Does advantage get sharper by extending to spacetime codes?

Summary

Lecture 2 : Spacetime cost in early FTQC

Decoders for surface code

Minimum-weight perfect matching

Decoder confidence for postselection

Scalable and reliable postselection criteria to be still developed

Syndrome-aware estimation

Exponential separation between classical v.s. quantum estimation