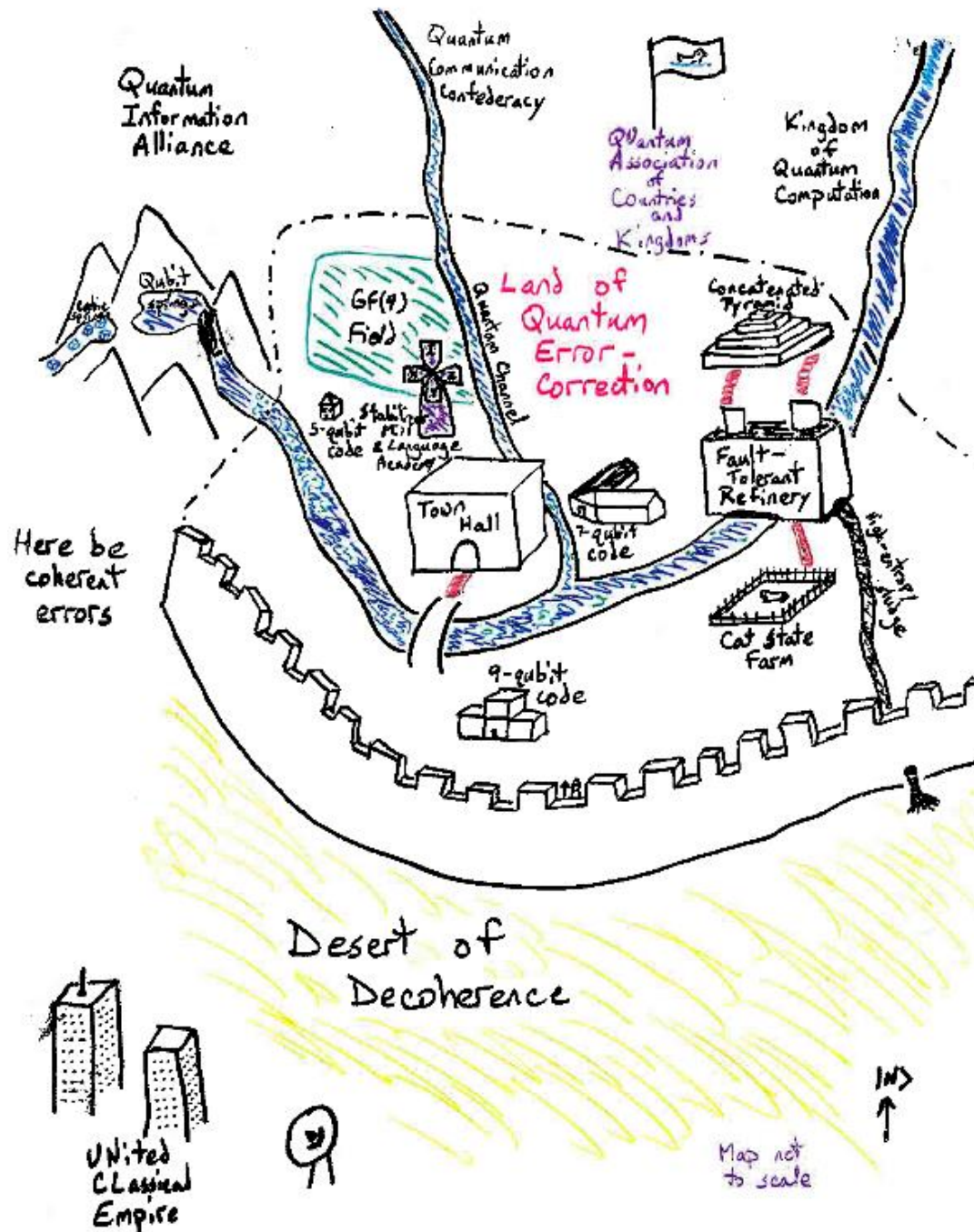


Quantum Error Correction

Daniel Gottesman

Perimeter Institute

The Classical and Quantum Worlds



Quantum Errors

A general quantum error is a superoperator:

$$\rho \rightarrow \sum A_k \rho A_k^\dagger$$

Examples of single-qubit errors:

Bit Flip X : $X|0\rangle = |1\rangle, X|1\rangle = |0\rangle$

Phase Flip Z : $Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle$

Complete dephasing: $\rho \rightarrow (\rho + Z\rho Z^\dagger)/2$
(decoherence)

Rotation: $R_\theta|0\rangle = |0\rangle, R_\theta|1\rangle = e^{i\theta}|1\rangle$

Classical Repetition Code

To correct a single bit-flip error for classical data, we can use the repetition code:

$$0 \rightarrow 000$$

$$1 \rightarrow 111$$

If there is a single bit flip error, we can correct the state by choosing the majority of the three bits, e.g. $010 \rightarrow 0$. When errors are rare, one error is more likely than two.

No-Cloning Theorem

There is no device that will copy an unknown quantum state:

$$|0\rangle \rightarrow |0\rangle |0\rangle, \quad |1\rangle \rightarrow |1\rangle |1\rangle$$

By linearity,

$$\begin{aligned} \alpha |0\rangle + \beta |1\rangle &\rightarrow \alpha |0\rangle |0\rangle + \beta |1\rangle |1\rangle \\ &\neq (\alpha |0\rangle + \beta |1\rangle)(\alpha |0\rangle + \beta |1\rangle) \end{aligned}$$

(Related to Heisenberg Uncertainty Principle)

Barriers to Quantum Error Correction

1. Measurement of error destroys superpositions.
2. No-cloning theorem prevents repetition.
3. Must correct multiple types of errors (e.g., bit flip and phase errors).
4. How can we correct continuous errors and decoherence?

Measurement Destroys Superpositions?

Let us apply the classical repetition code to a quantum state to try to correct a single bit flip error:

$$\alpha |0\rangle + \beta |1\rangle \rightarrow \alpha |000\rangle + \beta |111\rangle$$

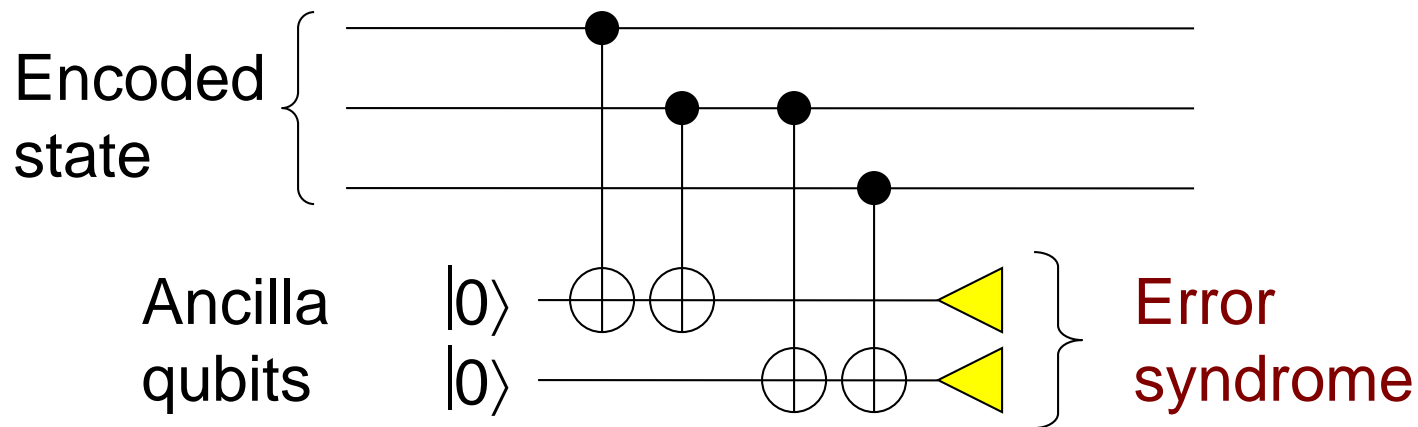
Bit flip error (X) on 2nd qubit:

$$\alpha |010\rangle + \beta |101\rangle$$

2nd qubit is now **different** from 1st and 3rd. We wish to measure that it is different without finding its actual value.

Measure the Error, Not the Data

Use this circuit:



1st bit of error syndrome says whether the first two bits of the state are the same or different.

2nd bit of error syndrome says whether the second two bits of the state are the same or different.

Measure the Error, Not the Data

With the information from the error syndrome, we can determine whether there is an error and where it is:

E.g., $\alpha |010\rangle + \beta |101\rangle$ has syndrome 11, which means the second bit is different. Correct it with a X operation on the second qubit. Note that the syndrome does not depend on α and β .

We have learned about the error without learning about the data, so superpositions are preserved!

Redundancy, Not Repetition

This encoding does not violate the no-cloning theorem:

$$\alpha |0\rangle + \beta |1\rangle \rightarrow \alpha |000\rangle + \beta |111\rangle \\ \neq (\alpha |0\rangle + \beta |1\rangle)^{\otimes 3}$$

We have repeated the state only in the computational basis; superposition states are spread out (redundant encoding), but not repeated (which would violate no-cloning).

Update on the Problems

- ✓ 1. Measurement of error destroys superpositions.
- ✓ 2. No-cloning theorem prevents repetition.
- 3. Must correct multiple types of errors (e.g., bit flip and phase errors).
- 4. How can we correct continuous errors and decoherence?

Correcting Just Phase Errors

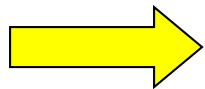
Hadamard transform H exchanges bit flip and phase errors:

$$H (\alpha |0\rangle + \beta |1\rangle) = \alpha |+\rangle + \beta |-\rangle$$

$$X |+\rangle = |+\rangle, X |-\rangle = -|-\rangle \text{ (acts like phase flip)}$$

$$Z |+\rangle = |-\rangle, Z |-\rangle = |+\rangle \text{ (acts like bit flip)}$$

Repetition code corrects a bit flip error



Repetition code in Hadamard basis corrects a phase error.

$$\alpha |+\rangle + \beta |-\rangle \rightarrow \alpha |+++ \rangle + \beta |--- \rangle$$

Nine-Qubit Code

To correct both bit flips and phase flips, use both codes at once:

$$\alpha |0\rangle + \beta |1\rangle \rightarrow$$

$$\alpha(|000\rangle + |111\rangle)^{\otimes 3} + \beta(|000\rangle - |111\rangle)^{\otimes 3}$$

Repetition 000, 111 corrects a bit flip error,
repetition of phase +++, --- corrects a phase error

Actually, this code corrects a bit flip **and** a phase, so it also corrects a Y error:

$$\mathbf{Y = iXZ}: Y|0\rangle = i|1\rangle, Y|1\rangle = -i|0\rangle \quad (\text{global phase irrelevant})$$

Update on the Problems

- ✓ 1. Measurement of error destroys superpositions.
- ✓ 2. No-cloning theorem prevents repetition.
- ✓ 3. Must correct multiple types of errors (e.g., bit flip and phase errors).
- 4. How can we correct continuous errors and decoherence?

Correcting Continuous Rotation

Let us rewrite continuous rotation

$$R_\theta |0\rangle = |0\rangle, R_\theta |1\rangle = e^{i\theta} |1\rangle$$

$$\begin{aligned} R_\theta &= \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix} = e^{i\theta/2} \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix} \\ &= \cos(\theta/2) I - i \sin(\theta/2) Z \end{aligned}$$

$$R_\theta^{(k)} |\psi\rangle = \cos(\theta/2) |\psi\rangle - i \sin(\theta/2) Z^{(k)} |\psi\rangle$$

($R_\theta^{(k)}$ is R_θ acting on the k th qubit.)

Correcting Continuous Rotations

How does error correction affect a state with a continuous rotation on it?

$$R_{\theta}^{(k)} |\psi\rangle = \cos(\theta/2) |\psi\rangle - i \sin(\theta/2) Z^{(k)} |\psi\rangle$$
$$\longrightarrow \cos(\theta/2) |\psi\rangle |I\rangle - i \sin(\theta/2) Z^{(k)} |\psi\rangle |Z^{(k)}\rangle$$

Error syndrome

Measuring the error syndrome collapses the state:

Prob. $\cos^2(\theta/2)$: $|\psi\rangle$ (no correction needed)

Prob. $\sin^2(\theta/2)$: $Z^{(k)} |\psi\rangle$ (corrected with $Z^{(k)}$)

Correcting All Single-Qubit Errors

Theorem: If a quantum error-correcting code (QECC) corrects errors A and B , it also corrects $\alpha A + \beta B$.

Any 2x2 matrix can be written as $\alpha I + \beta X + \gamma Y + \delta Z$.

A general single-qubit error $\rho \rightarrow \sum A_k \rho A_k^\dagger$ acts like a mixture of $|\psi\rangle \rightarrow A_k |\psi\rangle$, and A_k is a 2x2 matrix.

Any QECC that corrects the single-qubit errors X , Y , and Z (plus I) corrects every single-qubit error.

Correcting all t -qubit X , Y , Z on t qubits (plus I) corrects all t -qubit errors.

The Pauli Group

Define the Pauli group P_n on n qubits to be generated by X , Y , and Z on individual qubits. Then P_n consists of all tensor products of up to n operators I , X , Y , or Z with overall phase ± 1 , $\pm i$.

Any pair M , N of Pauli operators either commutes ($MN = NM$) or anticommutes ($MN = -NM$).

The **weight** of $M \in P_n$ is the number of qubits on which M acts as a non-identity operator.

The Pauli group spans the set of all n -qubit errors.

Small Error on Every Qubit

Suppose we have a small error U_ε on every qubit in the QECC, where $U_\varepsilon = I + \varepsilon E$.

Then

$$U_\varepsilon^{\otimes n} |\psi\rangle = |\psi\rangle + \varepsilon(E^{(1)} + \dots + E^{(n)}) |\psi\rangle + O(\varepsilon^2).$$

If the code corrects one-qubit errors, it corrects the sum of the $E^{(i)}$ s. Therefore it corrects the $O(\varepsilon)$ term, and **the state remains correct to order ε^2 .**

A code correcting t errors keeps the state correct to order ε^{t+1} .

QECC is Possible

- ✓ 1. Measurement of error destroys superpositions.
- ✓ 2. No-cloning theorem prevents repetition.
- ✓ 3. Must correct multiple types of errors (e.g., bit flip and phase errors).
- ✓ 4. How can we correct continuous errors and decoherence?

QECC Conditions

Thm.: A QECC can correct a set \mathcal{E} of errors iff

$$\langle \psi_i | E_a^\dagger E_b | \psi_j \rangle = C_{ab} \delta_{ij}$$

where $\{ |\psi_i\rangle \}$ form a basis for the codewords,
and $E_a, E_b \in \mathcal{E}$.

Note: The matrix C_{ab} does not depend on i and j .

As an example, consider $C_{ab} = \delta_{ab}$. Then we can make a measurement to determine the error.

If C_{ab} has rank $<$ maximum, code is **degenerate**.

Erasure Errors

Suppose we know the location of an error, but not its type (I, X, Y, or Z). This is called an **erasure**.

By QECC conditions $\langle \psi_i | E_a^\dagger E_b | \psi_j \rangle = C_{ab} \delta_{ij}$:

Correct t general errors \longleftrightarrow Correct $2t$ erasures

For erasures, QECC conditions become:

$\text{Tr}_A \rho$ does not depend on encoded state $|\psi\rangle$,
where A is set of qubits which are not erased.

That is, erased qubits have no info. about $|\psi\rangle$.

Stabilizer Codes

Error Syndromes Revisited

Let us examine more closely the error syndrome for the classical repetition code.

For correctly-encoded state 000 or 111: first two bits have even parity (an even number of 1's), and similarly for the 2nd and 3rd bits.

For state with error on one of the first two bits: odd parity for the first two bits.

We can rephrase this by saying a codeword is a +1 eigenvector of $Z \otimes Z \otimes I$ and a state with an error on the 1st or 2nd bit is a -1 eigenvector of $Z \otimes Z \otimes I$.

Error Syndromes Revisited

For the three-qubit phase error correcting code, a codeword has eigenvalue +1 for $X \otimes X \otimes I$, whereas a state with a phase error on one of the first two qubits has eigenvalue -1 for $X \otimes X \otimes I$.

Measuring $Z \otimes Z$ detects bit flip (X) errors, and measuring $X \otimes X$ detects phase (Z) errors.

Error syndrome is formed by measuring enough operators to determine location of error.

Stabilizer for Nine-Qubit Code

We can write down all the operators determining the syndrome for the nine-qubit code.

M_1	Z	Z								
M_2		Z	Z							
M_3				Z	Z					
M_4					Z	Z				
M_5							Z	Z		
M_6								Z	Z	
M_7	X	X	X	X	X	X				
M_8				X	X	X	X	X	X	

These generate a group, the **stabilizer** of the code, consisting of all Pauli operators M with the property that $M|\psi\rangle = |\psi\rangle$ for all encoded states $|\psi\rangle$.

Properties of a Stabilizer

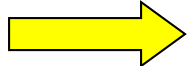
The stabilizer is a group:

If $M |\psi\rangle = |\psi\rangle$ and $N |\psi\rangle = |\psi\rangle$, then $MN |\psi\rangle = |\psi\rangle$.

The stabilizer is Abelian:

If $M |\psi\rangle = |\psi\rangle$ and $N |\psi\rangle = |\psi\rangle$, then

$$(MN - NM) |\psi\rangle = MN |\psi\rangle - NM |\psi\rangle = 0$$

(For Pauli matrices)  $MN = NM$

Given any Abelian group S of Pauli operators, define a code space $T(S) = \{ |\psi\rangle \text{ s.t. } M |\psi\rangle = |\psi\rangle \ \forall M \in S \}$.

Then $T(S)$ encodes k logical qubits in n physical qubits when S has $n-k$ generators (so size 2^{n-k}).

Stabilizer Elements Detect Errors

Suppose $M \in S$ and Pauli error E anticommutes with M . Then:

$$M (E |\psi\rangle) = - EM |\psi\rangle = - E |\psi\rangle,$$

so $E |\psi\rangle$ has eigenvalue -1 for M .

Conversely, if M and E commute for all $M \in S$,

$$M (E |\psi\rangle) = EM |\psi\rangle = E |\psi\rangle \quad \forall M \in S,$$

so $E |\psi\rangle$ has eigenvalue +1 for all M in the stabilizer.

The eigenvalue of an operator M from the stabilizer detects errors which anticommute with M .

Distance of a Stabilizer Code

Let S be a stabilizer, and let $T(S)$ be the corresponding QECC. Define

$$N(S) = \{N \in P_n \text{ s.t. } MN=NM \ \forall \ M \in S\}.$$

Then the **distance d** of $T(S)$ is the weight of the smallest Pauli operator N in $N(S) \setminus S$.

The code **detects any error not in $N(S) \setminus S$** (i.e., errors which commute with the stabilizer are not detected).

Why minus S ? “Errors” in S leave all codewords fixed, so are not really errors. (**Degenerate** QECC.)

Error Syndromes and Stabilizers

To **correct** errors, we must accumulate enough information about the error to figure out which one occurred.

The **error syndrome is the list of eigenvalues of the generators of S** : If the error E commutes with $M \in S$, then M has eigenvalue $+1$; if E and M anticommute, M has eigenvalue -1 .

We can then correct a set of possible errors if they all have distinct error syndromes.

Stabilizer Codes Correct Errors

Thm.: The code corrects errors for which $E^\dagger F \notin N(S) \setminus S$ for all possible pairs of errors (E, F) .

E and F have same error syndrome \longleftrightarrow

E and F commute with same elements of S

$\longleftrightarrow E^\dagger F \in N(S)$

$E^\dagger F \in S \longleftrightarrow E^\dagger F |\psi\rangle = |\psi\rangle \longleftrightarrow F |\psi\rangle = E |\psi\rangle$

E and F act the same, so we need not distinguish.

A stabilizer code with distance d corrects $\lfloor (d-1)/2 \rfloor$ errors (i.e., to correct t errors, we need $d = 2t+1$):

Stabilizer Codes Summary

- Choose an Abelian subgroup of the Pauli group. This will be the **stabilizer** S of the QECC.
- The codewords: $\{ |\psi\rangle \text{ s.t. } M |\psi\rangle = |\psi\rangle \quad \forall M \in S \}$
- If S has r generators on n qubits, the QECC has $k = n - r$ encoded qubits.
- The codes corrects errors if $E^\dagger F \notin N(S) \setminus S$ for all pairs (E, F) of possible errors. The **distance** d is the **minimum weight** of $N(S) \setminus S$.

Application: 5-Qubit Code

We can generate good codes by picking an appropriate stabilizer. For instance:

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

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

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

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

$n = 5$ physical qubits

- 4 generators of S

$k = 1$ encoded qubit

Distance d of this code is 3.

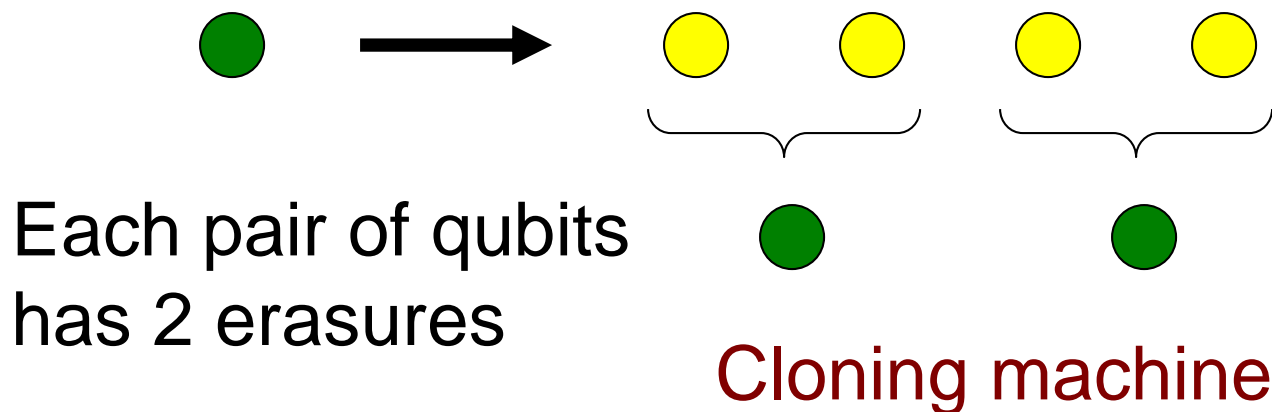
Notation: $[[n,k,d]]$ for a QECC encoding k logical qubits in n physical qubits with distance d . The five-qubit code is a **non-degenerate $[[5,1,3]]$** QECC.

The Smallest QECC

Can we do better than a $[[5,1,3]]$ QECC? **No.**

Recall that correcting 1 general error is equivalent to correcting 2 erasure errors.

Suppose we had a 4-qubit code which could correct 2 erasure errors:



Classical Linear Codes

A large and useful family of classical error-correcting codes can be defined similarly, using a **parity check matrix**. Let H be a $(n-k) \times n$ binary matrix, and define a classical error-correcting code C of n -bit vectors by

$$v \in C \iff Hv = 0.$$

C is linear: $v, w \in C \Rightarrow v+w \in C$. Also, let the **distance** d of C be the weight (# of non-zero entries) of the smallest non-zero $v \in C$. Then **a code with distance $2t+1$ corrects t errors**: the error syndrome of error e is He , and $He = Hf$ only if $e+f \in C$.

Classical Hamming Codes

Define a parity check matrix whose columns are all vectors of length r . E.g., for $r=3$:

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

This code has distance 3: if error e has weight 1, the error syndrome He specifies its location. Thus, the Hamming code for r is an **$[n=2^r-1, k=2^r-r-1, d=3]$** ECC (with k logical bits encoded in n physical bits and distance 3).

E.g., for $r=3$, we have a $[7,4,3]$ code.

Linear Codes and Stabilizers

The classical parity check matrix H is analogous to the stabilizer S of a quantum error-correcting code. Indeed, if we **replace all of the 1's of H with Z operators**, we get a stabilizer S defining exactly the same classical code. In particular, it can correct the same number of bit-flip errors.

E.g., Stabilizer of the $[7,4,3]$ Hamming code:

$$\begin{array}{cccccccc} Z & \otimes & Z & \otimes & Z & \otimes & Z & \otimes & I & \otimes & I & \otimes & I \\ Z & \otimes & Z & \otimes & I & \otimes & I & \otimes & Z & \otimes & Z & \otimes & I \\ Z & \otimes & I & \otimes & Z & \otimes & I & \otimes & Z & \otimes & I & \otimes & Z \end{array}$$

CSS Codes

We can then define a quantum error-correcting code by choosing two classical linear codes C_1 and C_2 , and replacing the parity check matrix of C_1 with Z's and the parity check matrix of C_2 with X's.

E.g.:

[[7,1,3]]
QECC

Z	⊗	Z	⊗	Z	⊗	Z	⊗	I	⊗	I	⊗	I
Z	⊗	Z	⊗	I	⊗	I	⊗	Z	⊗	Z	⊗	I
Z	⊗	I	⊗	Z	⊗	I	⊗	Z	⊗	I	⊗	Z
X	⊗	X	⊗	X	⊗	X	⊗	I	⊗	I	⊗	I
X	⊗	X	⊗	I	⊗	I	⊗	X	⊗	X	⊗	I
X	⊗	I	⊗	X	⊗	I	⊗	X	⊗	I	⊗	X

C_1 : [7,4,3]
Hamming

C_2 : [7,4,3]
Hamming

Which CSS Codes Are Possible?

Not all pairs C_1 and C_2 are possible: the stabilizer must be Abelian.

The **dual** C^\perp of a classical code C is the set of vectors w s.t. $v \cdot w = 0$ for all $v \in C$. The rows of the parity check matrix for C generate C^\perp .

If $v \in C_1^\perp$ and $w \in C_2^\perp$, the corresponding Pauli operators commute iff $v \cdot w = 0$. Thus, $w \in C_2^\perp$ is also in $(C_1^\perp)^\perp = C_1$.

To make a CSS code, we require $C_2^\perp \subseteq C_1$.

Properties of CSS Codes

The parameters of a CSS code made from C_1 , a $[n, k_1, d_1]$ code, and C_2 , a $[n, k_2, d_2]$ code, are

$$[[n, k_1 + k_2 - n, d']] \quad \text{with } d' \geq \min(d_1, d_2).$$

Why \geq ? Because of degeneracy (e.g., 9-qubit code).

Codewords of a CSS code are superpositions of classical codewords: For $v \in C_1$,

$$|\bar{v}\rangle = \sum_{w \in C_2^\perp} |v+w\rangle$$

If $v-v' \in C_2^\perp$, $|\bar{v}\rangle$ and $|\bar{v}'\rangle$ are the same state, so v should run over C_1/C_2^\perp . (Recall $C_2^\perp \subseteq C_1$.)

Summary: Stabilizer Codes

- We can describe a quantum stabilizer code by giving its stabilizer, an Abelian subgroup of the Pauli group.
- By looking at the stabilizer, we can learn all of the most interesting properties of a QECC, including the set of errors it can correct.
- One interesting and useful class of stabilizer codes is the family of CSS codes, derived from two classical codes. The 7-qubit code is the smallest example.

Quantum Error Correction Sonnet

We cannot clone, perforce; instead, we split
Coherence to protect it from that wrong
That would destroy our valued quantum bit
And make our computation take too long.

Correct a flip and phase - that will suffice.
If in our code another error's bred,
We simply measure it, then God plays dice,
Collapsing it to X or Y or Zed.

We start with noisy seven, nine, or five
And end with perfect one. To better spot
Those flaws we must avoid, we first must strive
To find which ones commute and which do not.

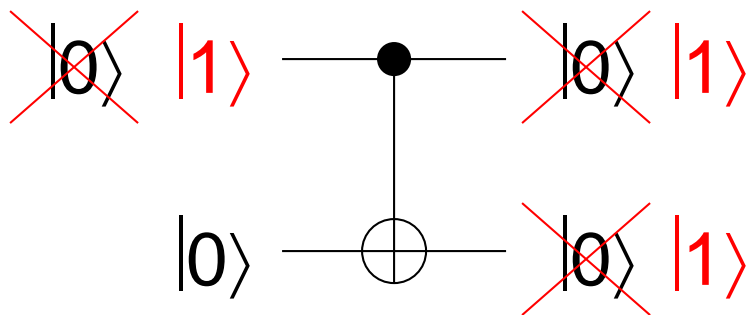
With group and eigenstate, we've learned to fix
Your quantum errors with our quantum tricks.

Fault-Tolerant Quantum Computation

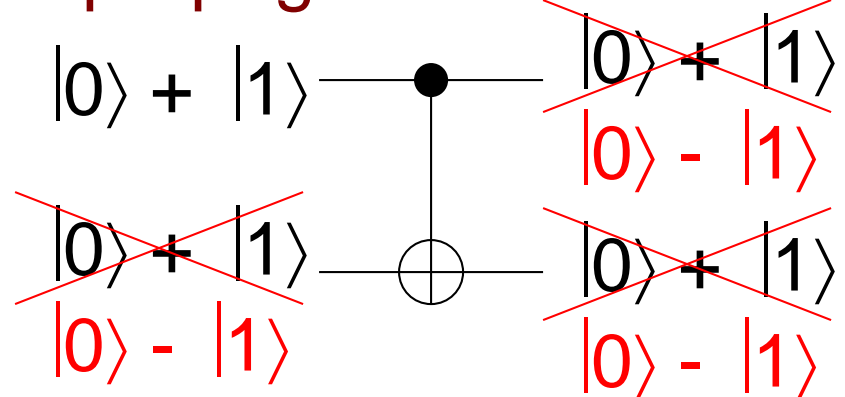
Error Propagation

When we perform multiple-qubit gates during a quantum computation, any existing errors in the computer can propagate to other qubits, even if the gate itself is perfect.

CNOT propagates
bit flips forward:



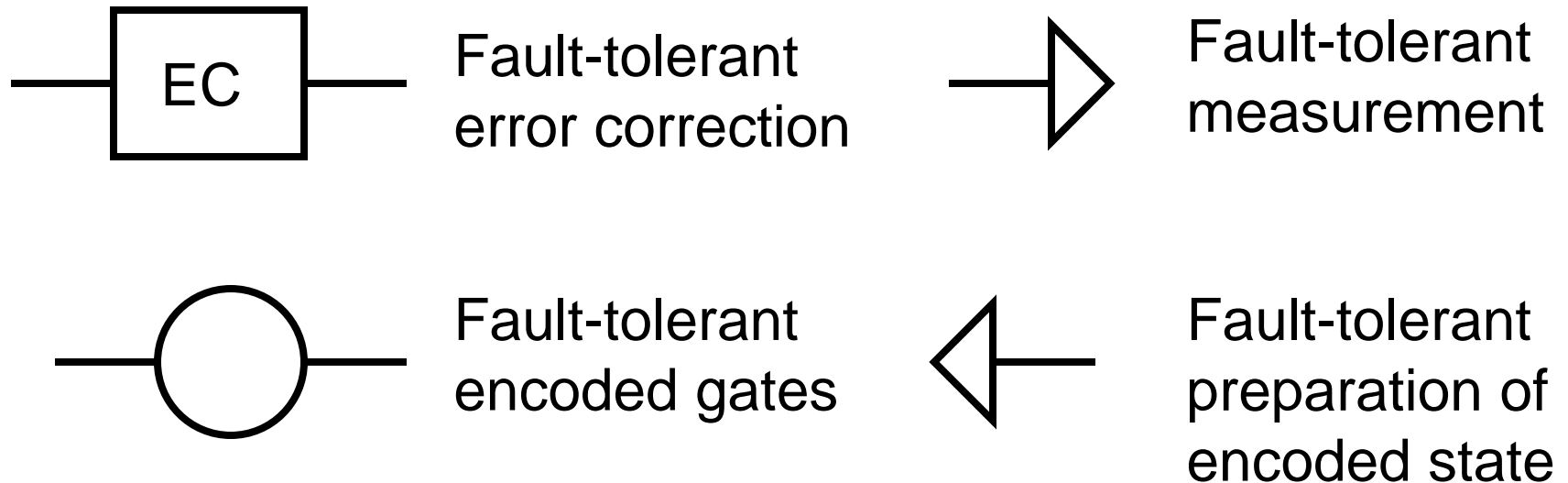
Phase errors
propagate backwards:



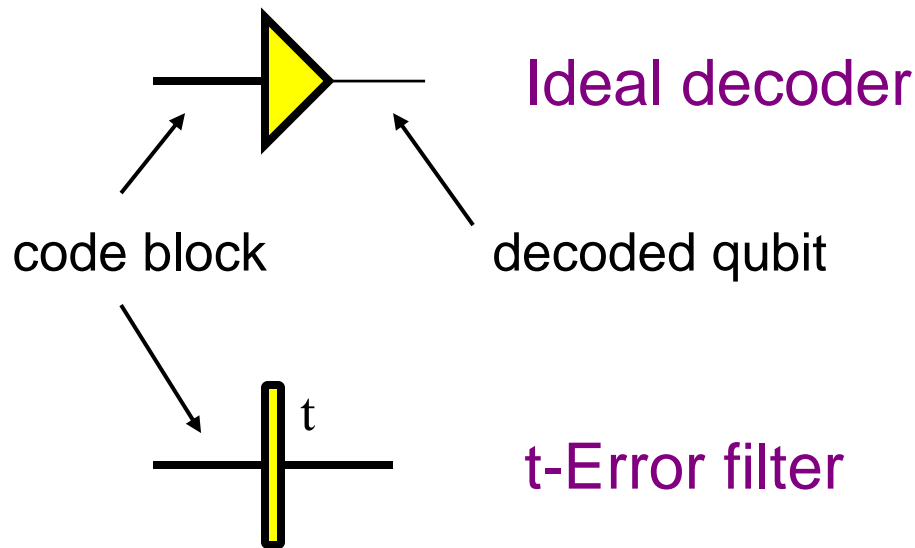
Fault-Tolerance

We need to avoid error propagation, and also need to perform gates without decoding the QECC.

We need:



Ideal Decoder and t-Filter

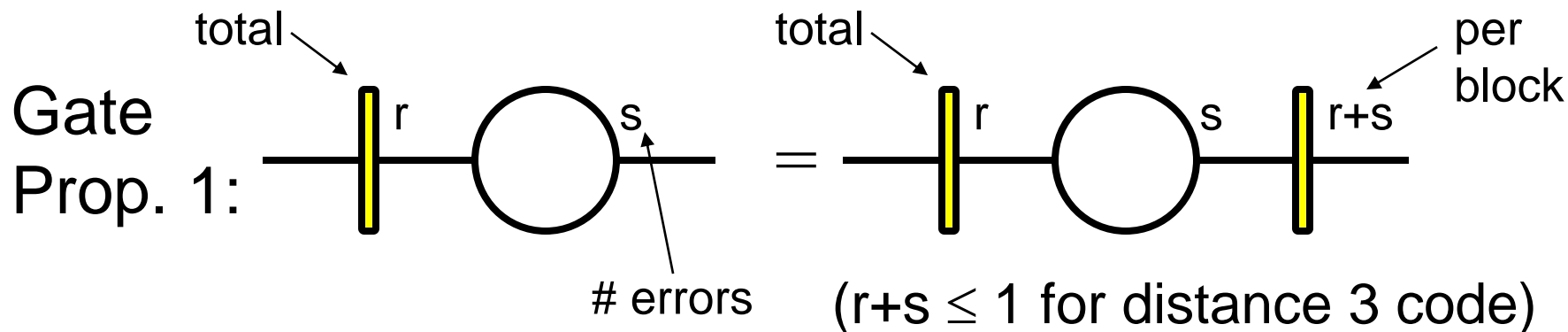


Corrects errors and decodes state producing an unencoded qubit.

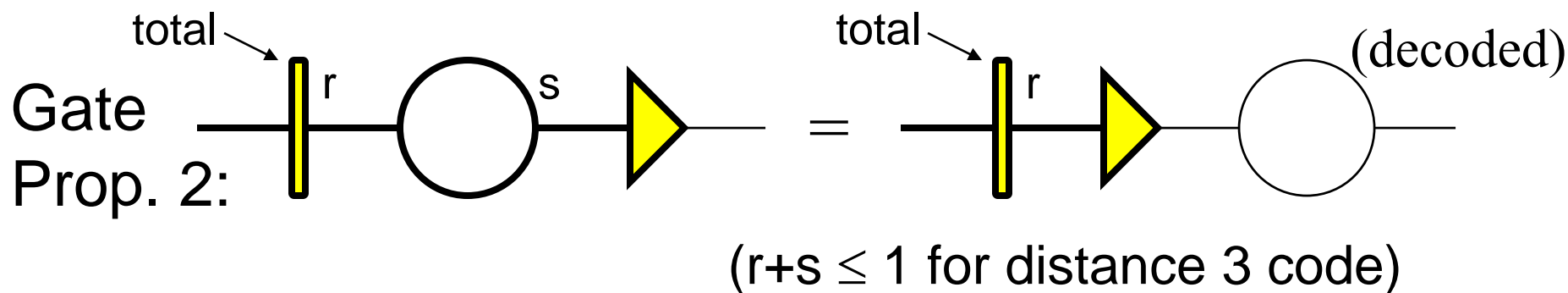
Projects on states that are within t errors of a valid codeword

These operations cannot be performed using real gates, but they are useful for defining and proving fault-tolerance.

Properties of FT gates



Errors propagate benignly. (Similarly for preparation.)

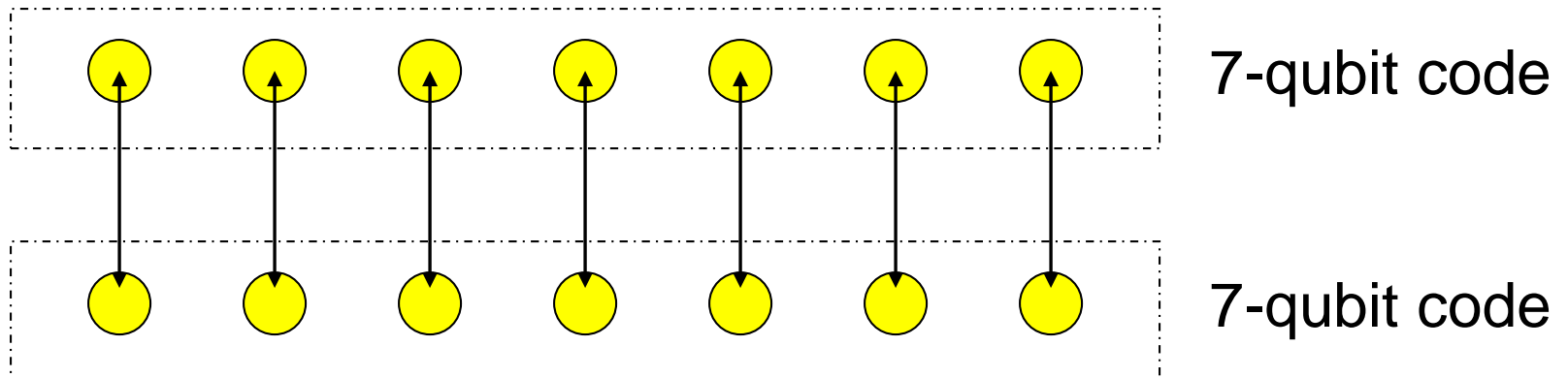


A good FT gate performs the right gate on the encoded state. (Similarly for preparation and measurement.)

Transversal Operations

Error propagation is only a serious problem if it propagates errors within a block of the QECC. Then one wrong gate could cause the whole block to fail.

The solution: **Perform gates transversally** - i.e. only between corresponding qubits in separate blocks.



Encoded X and Z

Operations which commute with a stabilizer, but are not in it, perform encoded operations: We can identify them as **logical Pauli gates**. Also, they are transversal.

$$\begin{array}{cccccccc}
 Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I \otimes I \\
 Z \otimes Z \otimes I \otimes I \otimes Z \otimes Z \otimes I \\
 Z \otimes I \otimes Z \otimes I \otimes Z \otimes I \otimes Z \\
 X \otimes X \otimes X \otimes X \otimes I \otimes I \otimes I \\
 X \otimes X \otimes I \otimes I \otimes X \otimes X \otimes I \\
 X \otimes I \otimes X \otimes I \otimes X \otimes I \otimes X
 \end{array}$$

\overline{X}
 \overline{Z}

$$\begin{array}{cccccccc}
 X \otimes X \otimes X \otimes X \otimes X \otimes X \otimes X \\
 Z \otimes Z \otimes Z \otimes Z \otimes Z \otimes Z \otimes Z
 \end{array}$$

Logical Clifford Group Gates

For the **7-qubit code**, CNOT, Hadamard H, and phase gate $P = R_{\pi/2} = \text{diag}(1, i)$ can all be done with **transversal gates**. (They generate the **Clifford group**, which can be efficiently simulated classically.)

For instance, for CNOT:

$$\begin{aligned} \text{CNOT}^{\otimes 7} \sum |v+w\rangle \sum |v'+w'\rangle \\ &= \sum |v+w\rangle \sum |(v+v')+(w+w')\rangle \\ &= (\text{logical CNOT}) \bar{|v\rangle} \bar{|v'\rangle} \end{aligned}$$

FT Preparation and Measurement

FT measurement for a CSS code:

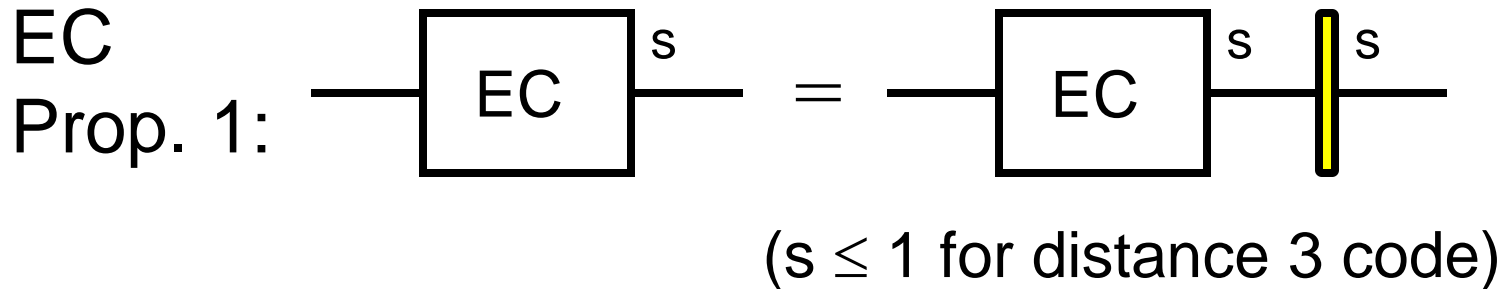
$$|\bar{v}\rangle = \sum_{w \in C_2^\perp} |v+w\rangle$$

If we measure each qubit, we get a classical codeword v from C_1 , shifted by a random codeword from C_2^\perp . Use classical EC to find v .

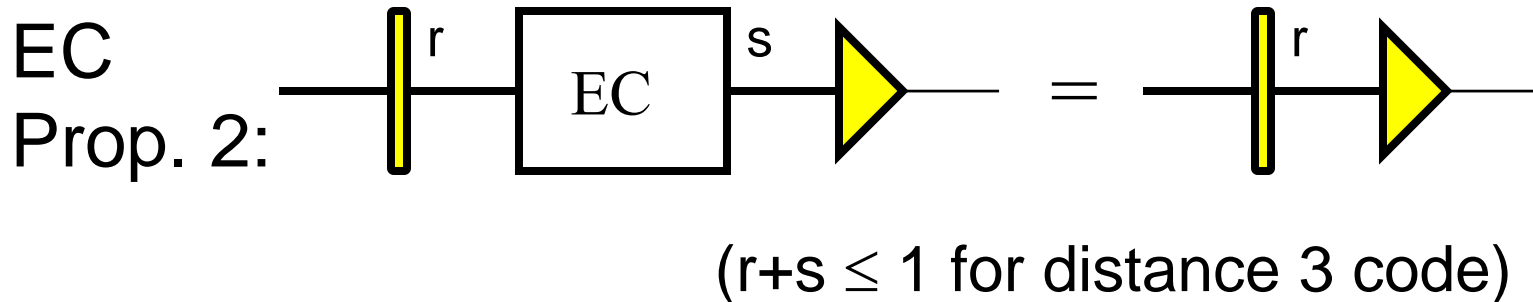
FT preparation:

1. Encode using any circuit.
2. Perform FT error correction.
3. Test encoded state.

Properties of FT Error Correction



Good error correction corrects any pre-existing errors.

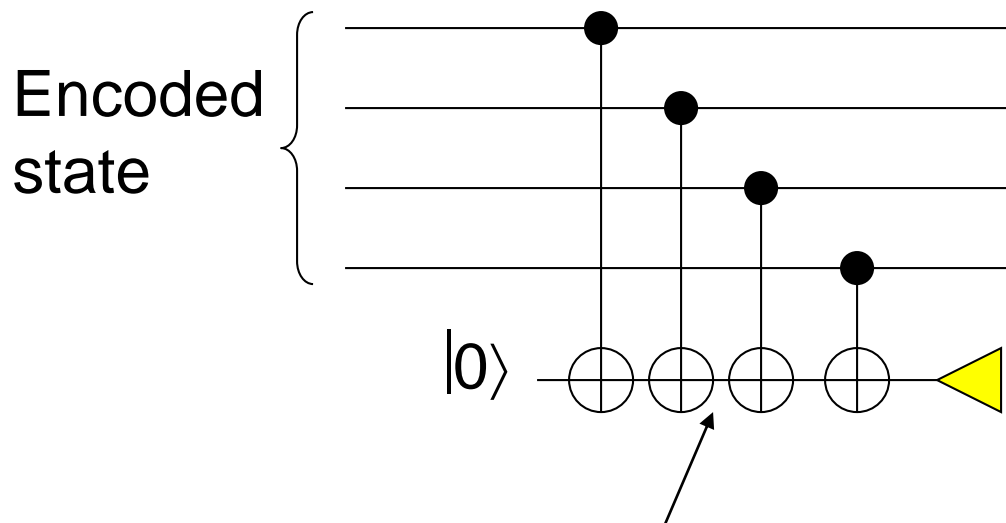


Good error correction leaves the encoded state alone.

Fault-Tolerant Error Correction

How can we perform error correction transversally?

Non-fault-tolerant measurement of $Z \otimes Z \otimes Z \otimes Z$:

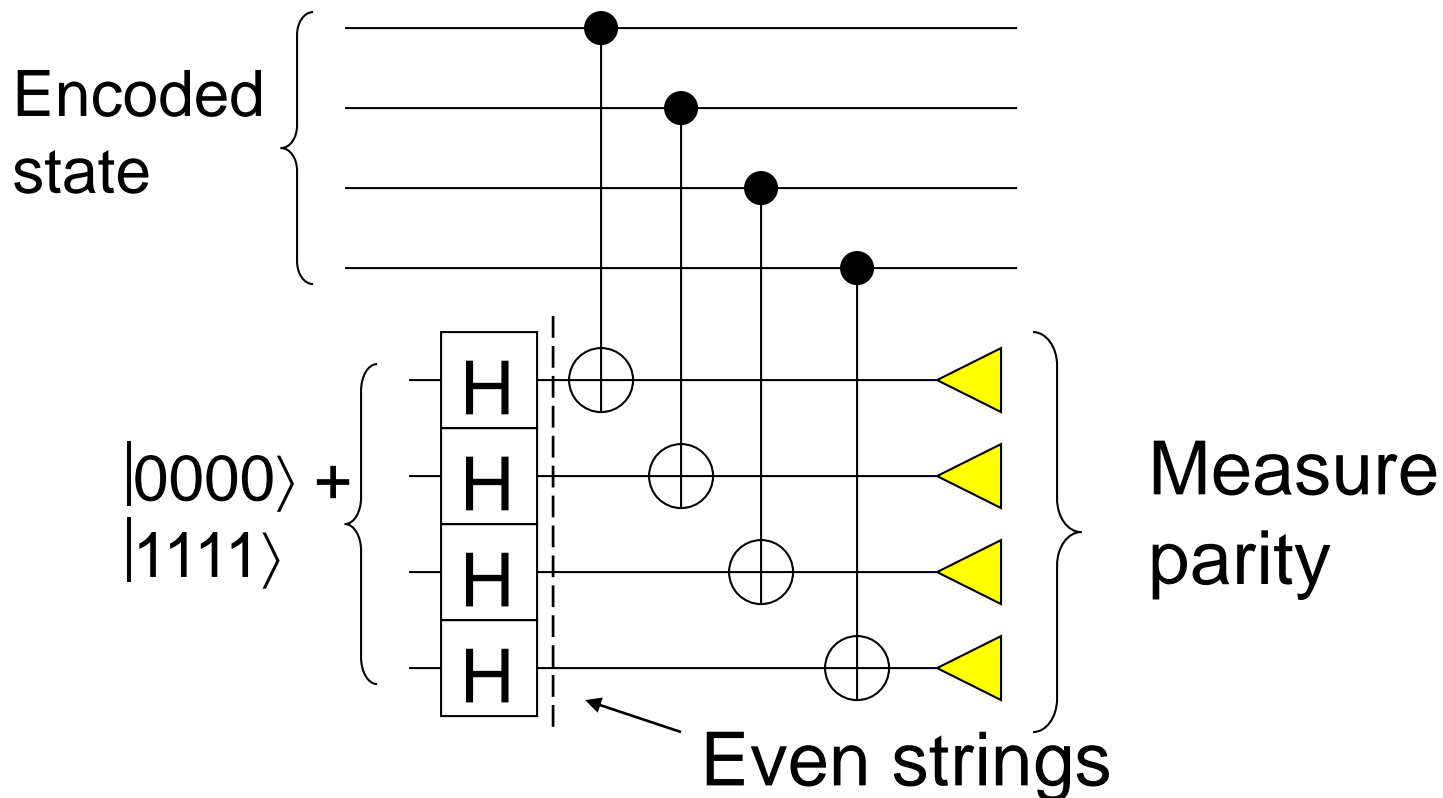


An error here could propagate to two qubits.

Fault-Tolerant Error Correction

How can we perform error correction transversally?

Fault-tolerant measurement of $Z \otimes Z \otimes Z \otimes Z$:



Fault-Tolerant Error Correction

Shor fault-tolerant error correction:

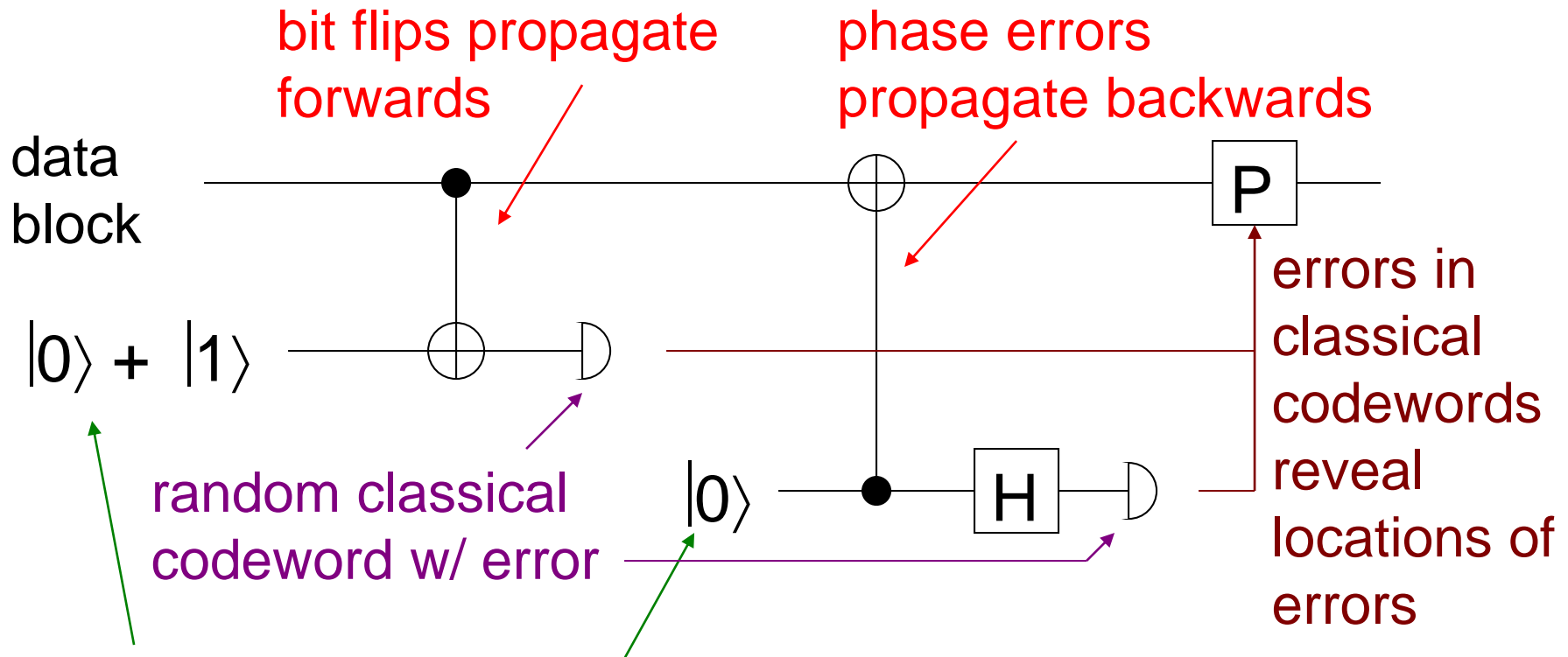
- Create and verify “cat” states $|0000\rangle + |1111\rangle$.
- Measure stabilizer generators to learn syndrome.
- Repeat for more confidence.

More advanced FT syndrome measurement techniques use more complicated ancillas, but fewer operations on the data:

- **Steane error correction** (uses encoded states)
- **Knill error correction** (based on teleportation)

Steane Error Correction

For CSS codes:



Ancilla blocks are encoded using same code
(verify them before using)

Universal Fault-Tolerant QC

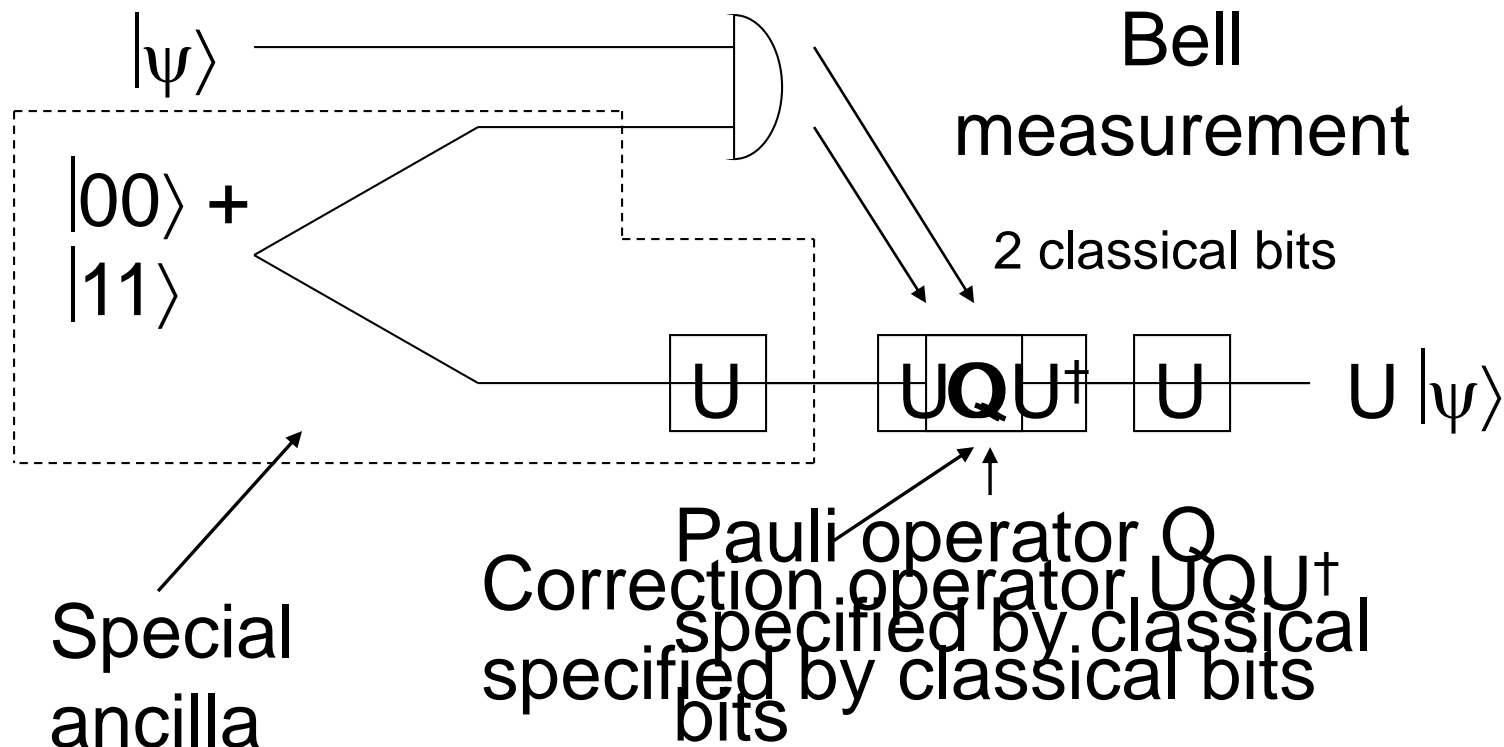
To complete a universal set of gates, we need some **additional gate outside the Clifford group**, for instance the **$\pi/8$ -gate**: $R_{\pi/4} |0\rangle = e^{-i\pi/8} |0\rangle$, $R_{\pi/4} |1\rangle = e^{i\pi/8} |1\rangle$.

We cannot perform it transversally, so we will need a more complicated construction:

- Create special ancilla state
- Perform teleportation-like procedure
- Perform Clifford group logical “correction”

Teleporting Quantum Gates

Quantum teleportation followed by U :



Fault-Tolerant $\pi/8$ -Gate

$\pi/8$ gate has a special and useful property:

$$R_{\pi/4} X R_{\pi/4}^\dagger = e^{-i\pi/4} PX, \quad R_{\pi/4} Z R_{\pi/4}^\dagger = Z$$

The correction required after teleportation is a Clifford group gate, for which we already know a fault-tolerant procedure!

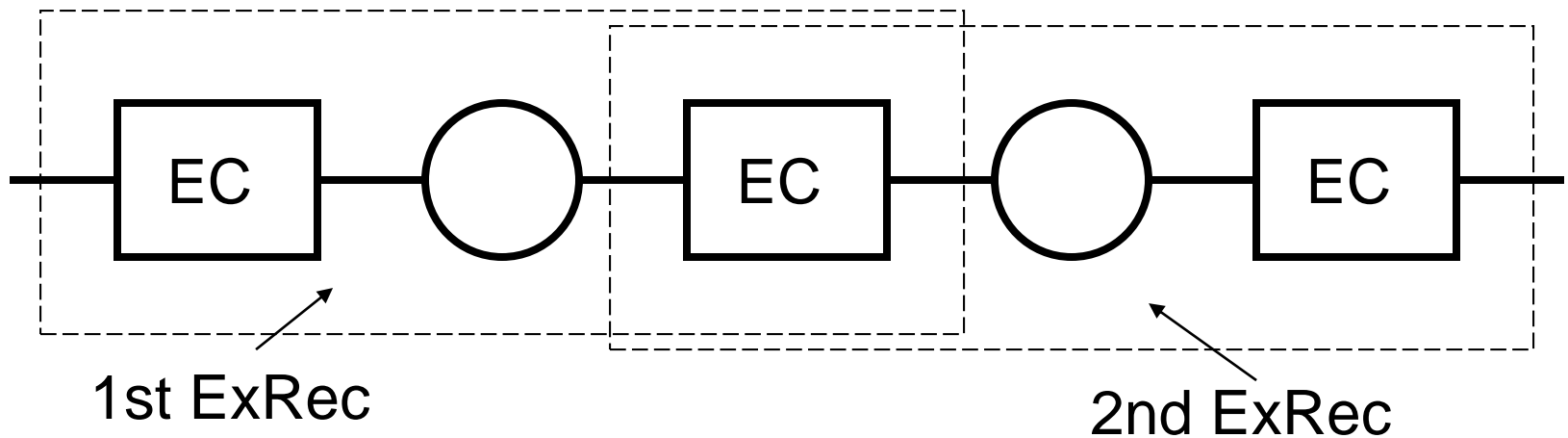
- Create ancilla state: encoded $|00\rangle + e^{i\pi/4} |11\rangle$
- Perform teleportation-like procedure
- Perform Clifford group logical “correction”

Extended Rectangles

Definition: An “extended rectangle” (or “ExRec”) consists of an EC step (“leading”), followed by an encoded gate, followed by another EC step (“trailing”).

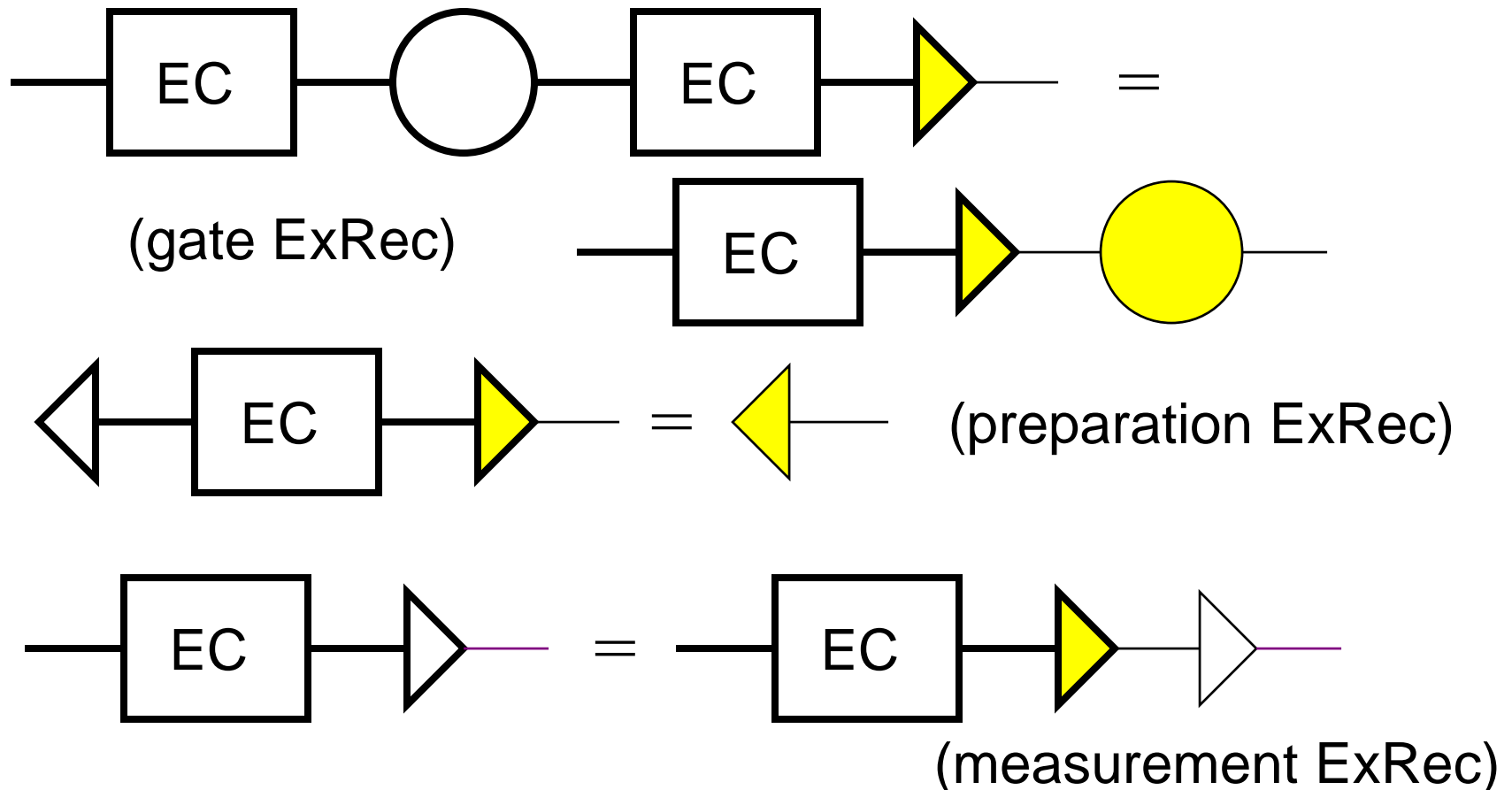
Definition: An ExRec is “good” if it contains at most one error (roughly speaking) in a gate in the ExRec.

Note: Extended rectangles overlap with each other.



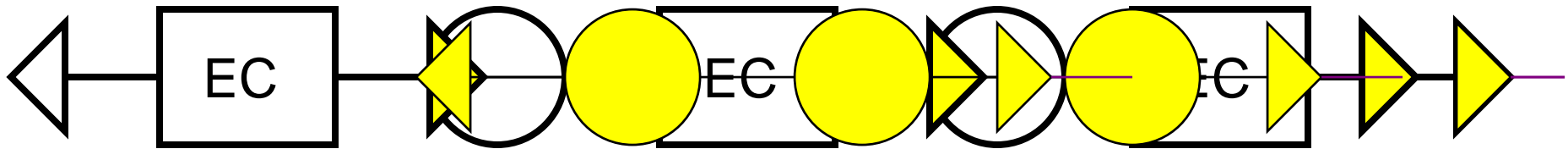
Good Circuits are Correct

Lemma [ExRec-Cor]: An ideal decoder can be pulled back through a good ExRec to just after the leading EC.



Correct Means What It Ought To

Suppose we have a circuit consisting of only good ExRecs. Then its action is equivalent to that of the corresponding ideal circuit:



1. Use ExRec-Cor for measurement to introduce an ideal decoder before the final measurement.
2. Use ExRec-Cor for gates to push the ideal decoder back to just after the very first EC step.
3. Use ExRec-Cor for preparation to eliminate the decoder.

Fault-Tolerant Simulation

A fault-tolerant circuit **simulates** the behavior of a circuit on the encoded qubits. We can replace each ExRec with the corresponding gate. Suppose the ExRec has A gates, and each gate is wrong with probability p :

Good ExRec:

- If there are 0 or 1 errors in the ExRec, replace the ExRec with the **ideal gate**.

Bad ExRec:

- With prob. $\sim (A^2/2)p^2$ the ExRec is bad, and we replace it with a **faulty gate**.

For small enough p , error rate decreases.

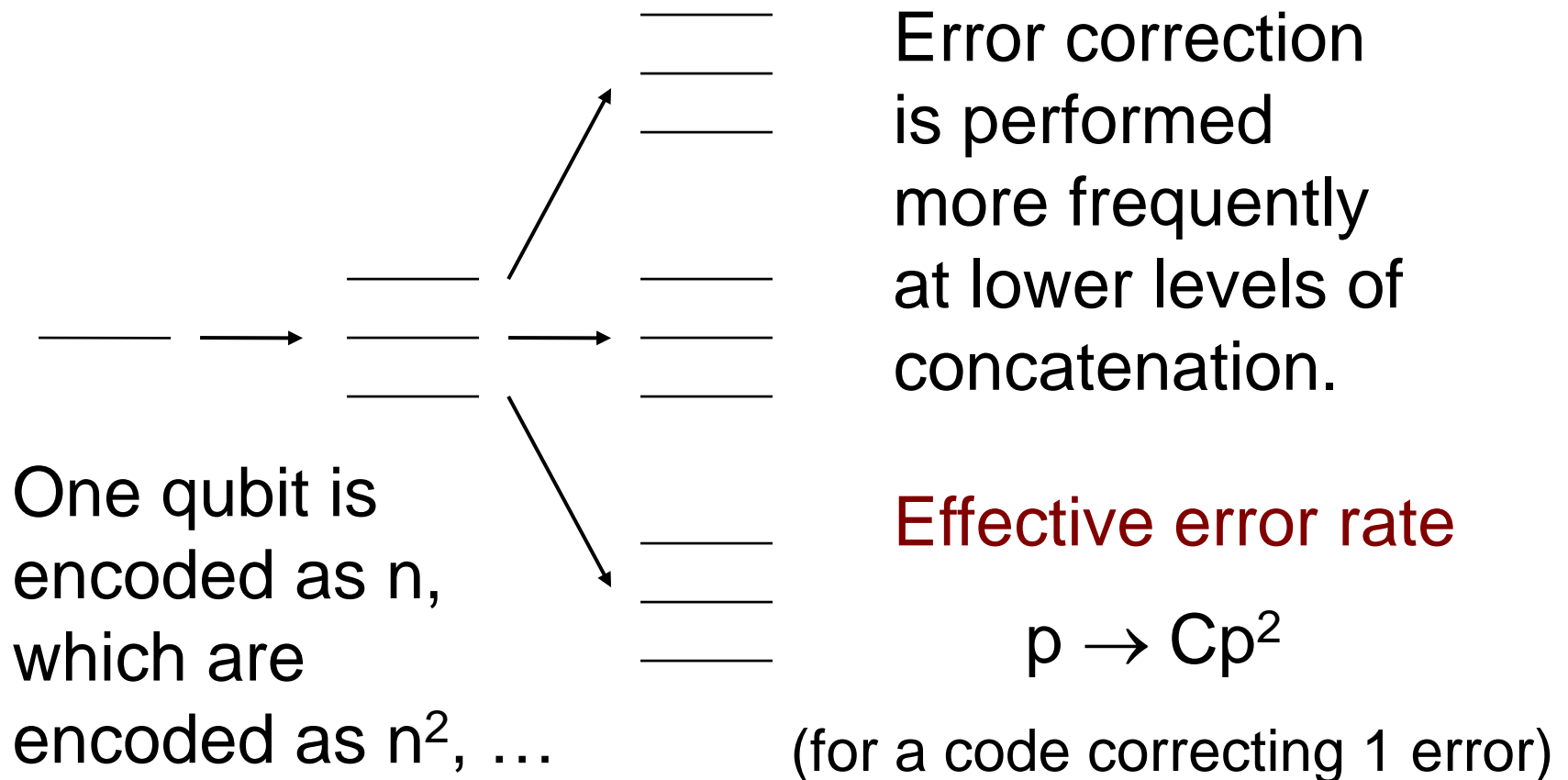
Summary of Fault Tolerance

- We can create fault-tolerant circuits from any stabilizer code, but the 7-qubit code is particularly straightforward.
- We perform many logical gates by transversal operations to avoid error propagation within blocks.
- Not all operations can be performed this way - the other trick is to create special ancillas and work hard to verify them.

Threshold for Fault Tolerance

Concatenated Codes

Threshold for fault-tolerance proven using concatenated error-correcting codes.



Threshold for Fault-Tolerance

Theorem: There exists a threshold p_t such that, if the error rate per gate and time step is $p < p_t$, arbitrarily long quantum computations are possible.

Proof sketch: Each level of concatenation changes the effective error rate $p \rightarrow p_t (p/p_t)^2$. The effective error rate p_k after k levels of concatenation is then



and for a computation of length T , we need only $\log(\log T)$ levels of concatenation, requiring $\text{polylog}(T)$ extra qubits, for sufficient accuracy.

History of the Threshold

Golden

Age (1996)

Shor (1996) - FT protocols

Aharonov, Ben-Or
(1996) - threshold
proof

Knill, Laflamme (1996)
storage threshold

Kitaev (1996-...)
topological FT,
threshold

Zalka (1996)
simulation

K, L, Zurek (1996)
threshold

Dark
Ages

Other
simulations

G, Preskill
higher value

Dennis et al. (2001)
topological threshold

Aliferis, G, Preskill
(2005) - simple proof

Knill (2004),
Reichardt (2004)
very high threshold

Reichardt (2005)
d=3 proof

Renaissance (2004-)

Local gates, specific systems, ...

The DiVincenzo Criteria

1. A scalable physical system with well-characterized qubits.
2. The ability to initialize the state of the qubits to a simple fiducial state, such as $|0\dots 0\rangle$.
3. Long relevant decoherence times, much longer than the gate operation time.
4. A “universal” set of quantum gates.
5. A qubit-specific measurement capability.
6. The ability to interconvert stationary and flying qubits.
7. The ability to faithfully transmit flying qubits between specified locations.

Requirements for Fault-Tolerance

1. Low gate error rates.
2. Ability to perform operations in parallel.
3. A way of remaining in, or returning to, the computational Hilbert space.
4. A source of fresh initialized qubits during the computation.
5. Benign error scaling: error rates that do not increase as the computer gets larger, and no large-scale correlated errors.

Additional Desiderata

1. Ability to perform gates between distant qubits.
2. Fast and reliable measurement and classical computation.
3. Little or no error correlation (unless the registers are linked by a gate).
4. Very low error rates.
5. High parallelism.
6. An ample supply of extra qubits.
7. Even lower error rates.

Tradeoffs Between Desiderata

The mandatory requirements for fault-tolerance are not too strenuous -- many physical systems will satisfy them.

However, we will probably need at least **some of the desiderata** in order to actually make a fault-tolerant quantum computer.

It is difficult, perhaps impossible, to find a physical system which satisfies **all** desiderata. Therefore, we need to study **tradeoffs: which sets of properties will allow us to perform fault-tolerant protocols?** For instance, if we only have nearest-neighbor gates, what error rate do we need?

The Meaning of Error Rates

Cited error rates are error **probabilities**; that is, the probability of projecting onto the correct state after one step.

E.g.: Rotation by angle θ has error probability $\sim \theta^2$.

- **Gate errors**: errors caused by an imperfect gate.
- **Storage errors**: errors that occur even when no gate is performed.

Error rates are for a particular universal gate set.

Determining the Threshold Value

There are three basic methodologies used to determine the value of the threshold:

- **Numerical simulation**: Randomly choose errors on a computer, see how often they cause a problem. Tends to give high threshold value, but maybe this is an overestimate; only applies to simple error models.
- **Rigorous proof**: Prove a certain circuit is fault-tolerant for some error rate. Gives the lowest threshold value, but everything is included (up to proof's assumptions).
- **Analytic estimate**: Guess certain effects are negligible and calculate the threshold based on that. Gives intermediate threshold values.

Threshold Values

Computed threshold value depends on **error-correcting code**, **fault-tolerant circuitry**, **analysis technique**. Assume for now that all additional desiderata are satisfied.

- Concatenated 7-qubit code, standard circuitry:
 - Threshold $\sim 10^{-3}$ (various simulations)
 - Threshold $\sim 3 \times 10^{-5}$ (proof: Aliferis, Gottesman, Preskill, quant-ph/0504218; also Reichardt, quant-ph/0509203)
- Best known code: 25-qubit Bacon-Shor code
 - Threshold $\sim 2 \times 10^{-4}$ (proof: Aliferis, Cross, quant-ph/0610063)

Ancilla Factories

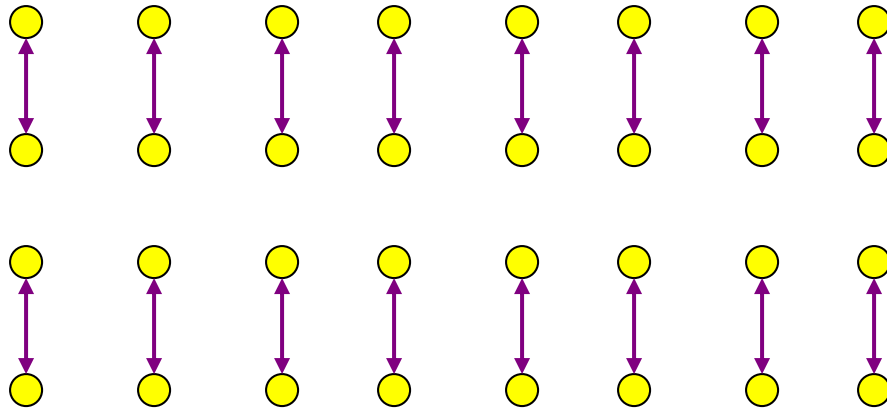
Best methods trade extra ancilla qubits for error rate: **Ancilla factories** create complex ancilla states to substitute for most gates on the data. Errors on ancillas are less serious, since bad ancillas can be discarded safely (Steane, quant-ph/9611027).

Extreme case: Create all states using error-detecting codes, ensuring a low basic error rate but very high overheads (e.g. 10^6 or more physical qubits per logical qubit) -- Knill, quant-ph/0404104, Reichardt, quant-ph/0406025.

- **Simulations:** threshold $\sim 1\%$ or higher.
- **Provable threshold** $\sim 10^{-3}$ (Reichardt, quant-ph/0612004, Aliferis, Gottesman, and Preskill, quant-ph/0703264)

Parallel Operations

Fault-tolerant gates are easily parallelized.

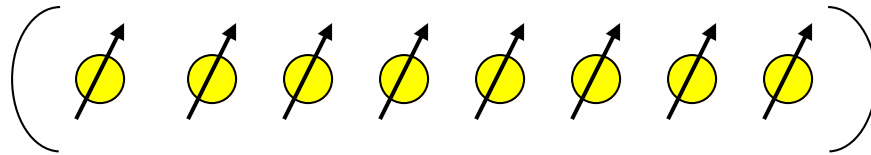


Error correction operations should be applied in parallel, so we can correct all errors before decoherence sets in.

Threshold calculations assume full parallelism.

Erasure Errors

For instance: loss of atoms



Losing one is not too serious,
but losing all is fatal.

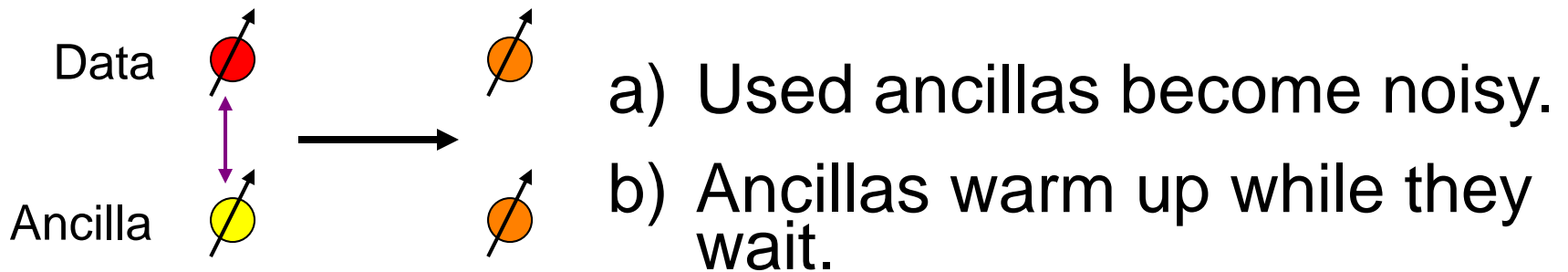
Erasures are an issue for:

- Quantum cellular automata
- Encoded universality

Fresh Ancilla States

We need a constant source of fresh blank qubits to perform error correction.

Thermodynamically, noise introduces **entropy** into the system. Error correction pumps entropy into **cold** ancilla states.



Fresh Ancilla States

Used ancillas can be replaced by new ancillas, but we must ensure ancillas do not wait too long: otherwise, there is an exponential loss of purity.

In particular:

- It is not sufficient to initialize all qubits at the start of computation.

For instance, this is a problem for liquid-state NMR.

Large-Scale Error Rates

The error rate for a given qubit should not increase when we add more qubits to the computer.

For instance:

- Long-range crosstalk (such as $1/r^2$ Coulomb coupling)

Short-range crosstalk is OK, since it stops increasing after neighbors are added.

(See Aharonov, Kitaev, Preskill, quant-ph/0510231.)

Correlated Errors

Small-scale correlations are acceptable:

We can choose an error-correcting code which corrects multiple errors.

Large-scale correlations are fatal:

A large fraction of the computer fails with reasonable probability.

Note: This type of error is rare in most weakly-coupled systems.

Correlated Errors

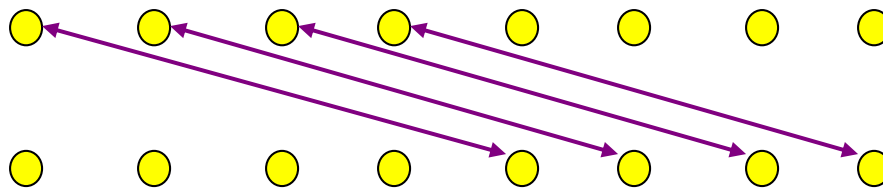
Small-scale correlations are not fatal, but are still better avoided.

We assume correlated errors can occur when a gate interacts two qubits. Any other source of multiple-qubit errors is an additional error rate not included in the threshold calculations.

The worst case is correlated errors within a block of the code, but the system can be designed so that such qubits are well separated.

Long-Range Gates

Most calculated thresholds assume we can perform gates between qubits at arbitrary distances.



If not, **threshold still exists**, but we need better error rates to get a threshold, since we use additional gates to move data around during error correction.

Local Gates

Proof that threshold still exists with local gates: Gottesman, quant-ph/9903099; Aharonov, Ben-Or, quant-ph/9906129.

We are starting to understand the value of the threshold in this case:

- With concatenation, in 2D, proven threshold of $\sim 10^{-5}$ (Svore, Terhal, DiVincenzo, quant-ph/0604090)
- Almost 2D, w/ topological codes & cluster states, simulated threshold of $\sim 6 \times 10^{-3}$ (Raussendorf, Harrington, quant-ph/0610082)
- Almost 1D: proven threshold of $\sim 10^{-6}$ (Stephens, Fowler, Hollenberg, quant-ph/0702201)

Fast Classical Processing

Fast measurement and classical processing is very useful for error correction to compute the actual type and location of errors.

We can implement the classical circuit with quantum gates if necessary, but this adds overhead: the classical circuit must be made classically fault-tolerant.

May not matter much for threshold? (The classical repetition code is very robust.)

(Szkopek et al., quant-ph/0411111.)

Other Error Models

- Coherent errors: Not serious; could add amplitudes instead of probabilities, but this worst case will not happen in practice (unproven).
- Restricted types of errors: Generally not helpful; tough to design appropriate codes. (But other control techniques might help here.)
- Non-Markovian errors: Allowed; when the environment is weakly coupled to the system, at least for bounded system-bath Hamiltonians.
(Terhal, Burkhard, quant-ph/0402104, Aliferis, Gottesman, Preskill, quant-ph/0504218, Aharonov, Kitaev, Preskill, quant-ph/0510231.)

Summary

- Quantum error-correcting codes exist which can correct very general types of errors on quantum systems.
- A systematic theory of QECCs allows us to build many interesting quantum codes.
- Fault-tolerant protocols enable us to accurately perform quantum computations even if the physical components are not completely reliable, provided the error rate is below some threshold value.

Further Information

- Short intro. to QECCs: quant-ph/0004072
- Short intro. to fault-tolerance: quant-ph/0701112
- Chapter 10 of Nielsen and Chuang
- Chapter 7 of John Preskill's lecture notes:
<http://www.theory.caltech.edu/~preskill/ph229>
- Threshold proof & fault-tolerance: quant-ph/0504218
- My Ph.D. thesis: quant-ph/9705052
- Complete course on QECCs:
<http://perimeterinstitute.ca/personal/dgottesman/QECC2007>

The Future of Fault-Tolerance

Industrial Age

Experimental FT

Ancilla
factories

Understanding
of resource
tradeoffs

Efficient fault-
tolerance

Large quantum
computers!

Quantum
Information Age

