Fault-tolerance schemes and threshold theorems

Monday, August 18, 2014 4:30 PM

OUTLINE

Fault tolerance
- what it is ✓
- examples ✓
- carefully tracking errors

Threshold ½ for erasure errors ✓

Other fault-tolerance schemes ✓
- Steane-type error correction

AGP threshold theorem ✓
- malignant set counting

Other threshold existence theorems ✓

Overhead analysis of fault-tolerance schemes ✓

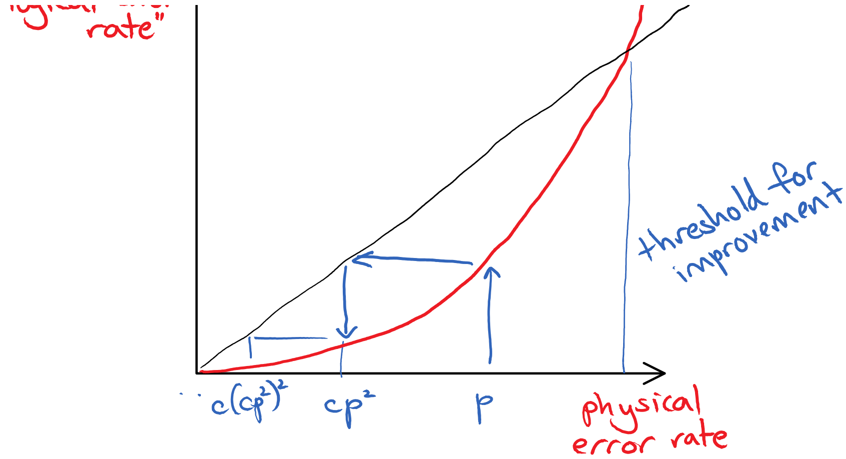
FAULT-TOLERANCE SCHEMES AND THRESHOLD THEOREMS

Ben Reichardt

RECALL:

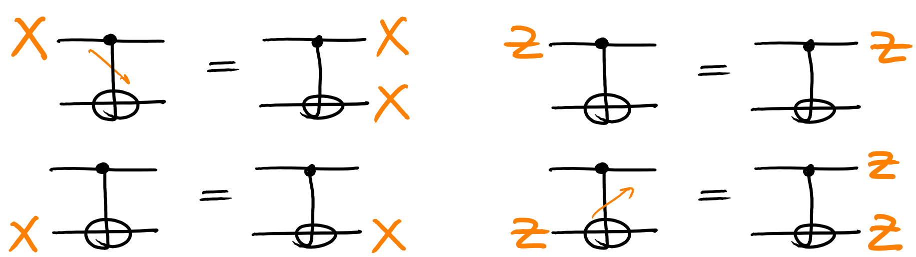
Noise threshold intuition

distance-3 code $\longrightarrow$ quadratic reduction in error rate



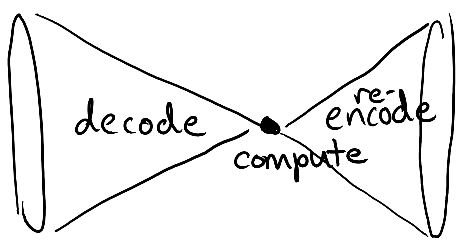
Concatenate the scheme for arbitrary reliability

CNOT gate: copies X forward, Z backward

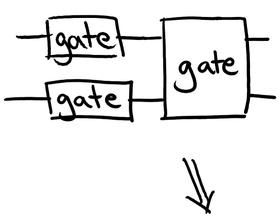


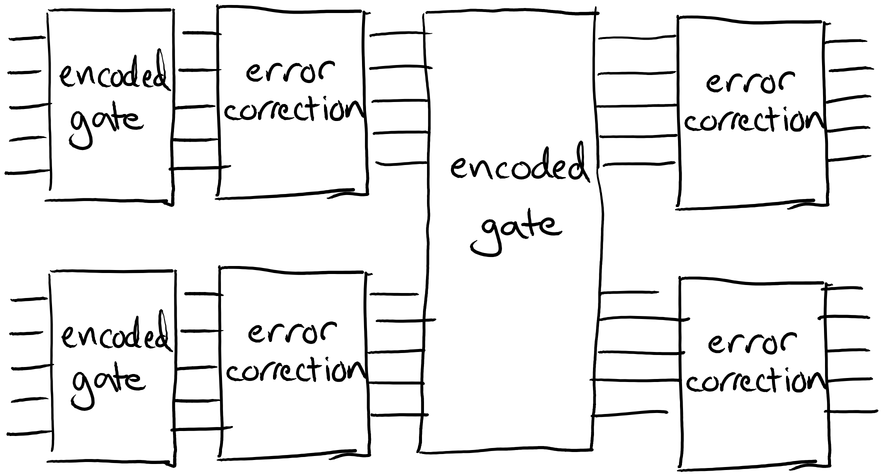
FAULT TOLERANCE

What we **don't** do:



We need to compute on the encoded data.





Error-correction procedures are themselves faulty, but applied periodically they hopefully keep noise under control.

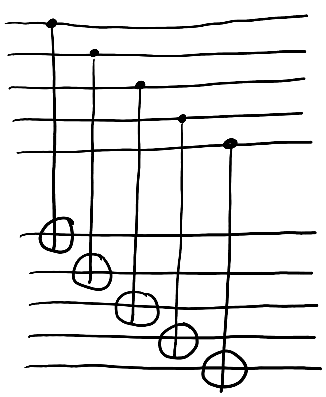
Fault-tolerant operations

1 CNOT gate

Start with a CSS code (all but one of the codes from last time)

$$\begin{pmatrix} X \text{ stabilizers} \\ \hdashline Z \text{ stabilizers} \end{pmatrix}$$

Claim: Transversal CNOTs between two code blocks implement logical CNOT.



– and they don't spread errors within either code block

1 faulty gate $\longrightarrow$ ≤ 1 error on each block

Why?

Block 1 | Block 2

(X stabilizers

Block 1 | Block 2

(X stabilizers | same X stabs

$$
\begin{array}{cc} \text{Block 1} & \text{Block 2} \end{array}
$$

$$
\begin{pmatrix} X \text{ stabilizers} & \\ \hdashline Z \text{ stabilizers} & \\ \hdashline & X \text{ stabilizers} \\ \hdashline & Z \text{ stabilizers} \end{pmatrix}
\rightarrow
\begin{pmatrix} X \text{ stabilizers} & \text{same } X \text{ stabs} \\ \hdashline Z \text{ stabilizers} & \\ \hdashline & X \text{ stabilizers} \\ \hdashline \text{same } Z \text{ stabs} & Z \text{ stabilizers} \end{pmatrix}
$$

$\wr$ equivalent to

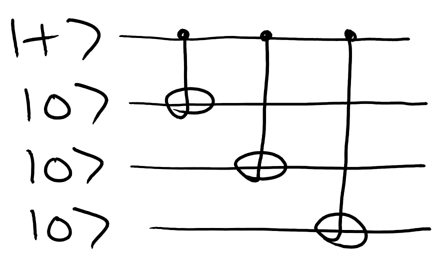
$$
\begin{pmatrix} X \text{ stabilizers} & \\ \hdashline Z \text{ stabilizers} & \\ \hdashline & X \text{ stabilizers} \\ \hdashline & Z \text{ stabilizers} \end{pmatrix}
$$

– so each block stays in the codespace

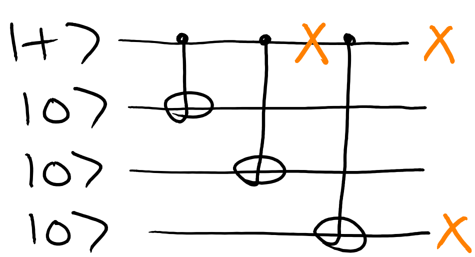
and logical X operators are copied forward (as they should be)
logical Z operators copied backward ✓

② Cat state preparation

$|0000\rangle + |1111\rangle$



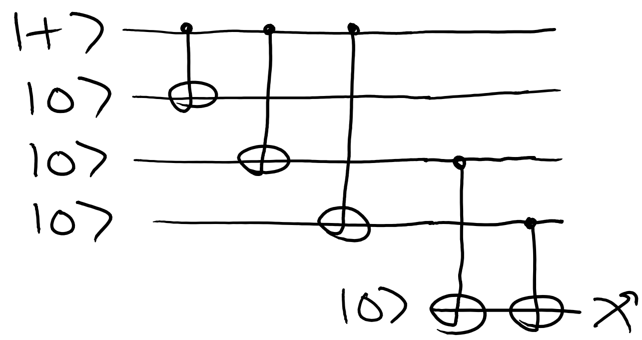
This is not fault tolerant; errors can spread



$|1001\rangle + |0110\rangle$

one fault → 2 errors on output

Solution: Check for correlated errors.



with the above $X\,I\,I\,X$ error, this bit will flip to 1

Note that this will not catch $I\,I\,X\,X$, but that can't happen w/ 1st-order probability

Moral: We only need to check for some correlated errors.

Fault-tolerant state preparation:

weight-k errors should have probability $O(p^k)$.

(for k up to obvious limit, e.g., $k \leq 2$ for a distance-3 code)

?: Why do we only check for correlated X errors?

Any two Zs are a stabilizer for $|0000\rangle + |1111\rangle$.

③ Syndrome extraction (optional)

error correction

To correct errors, we need to know each stabilizer's sign, ± 1.

E.g., repetition code

code stabilizers	syndromes
$Z\,Z\,I\,I$	-1

Code stabilizers	syndrome
Z Z I I	−1
I Z Z I	−1
I I Z Z	+1

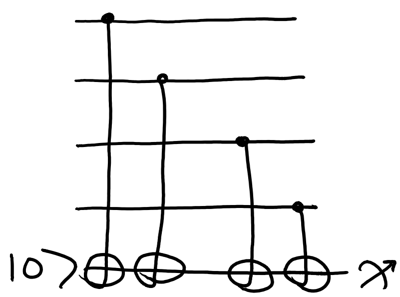
$\Rightarrow$ there is an X error on qubit 2

How do we measure these error syndromes, without collapsing the quantum state?

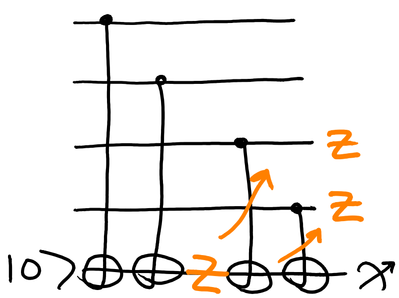
Example: Consider a code with a ZZZZ stabilizer.

(a) Measuring each qubit & adding their parities gives the syndrome — but also collapses the state! It measures too much.

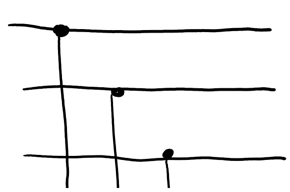
(b) Measure **only** the parity we want:

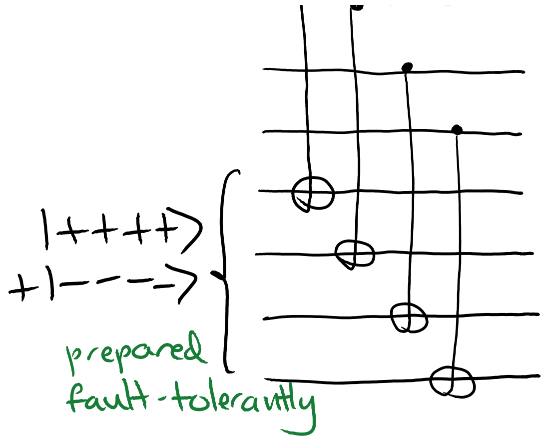


as desired, this copies X errors down — but Z errors are copied back!



(c) **Fault-tolerant** parity measurement





stabilizers

$$\begin{array}{cccc} Z & Z & Z & Z \\ X & X & I & I \\ I & X & X & I \\ I & I & X & X \end{array}$$

X errors are copied down, but two Xs are a stabilizer $\Rightarrow$ get the parity

④ **Exercise**: Give a stabilizer circuit to prepare $|0\rangle$ encoded into the 7-qubit Steane code.

What correlated errors can the circuit create, i.e., single faults $\longrightarrow$ errors of weight ≥ 2 ?

stabilizers:

$$\left.\begin{array}{ccccccc} I & I & I & X & X & X & X \\ I & X & X & I & I & X & X \\ X & I & X & I & X & I & X \\ I & I & I & Z & Z & Z & Z \\ I & Z & Z & I & I & Z & Z \\ Z & I & Z & I & Z & I & Z \end{array}\right\} \text{code stabilizers}$$

$$\begin{array}{ccccccc} Z & Z & Z & Z & Z & Z & Z \end{array} \text{ — code logical } Z$$

THRESHOLD ½ FOR ERASURE ERRORS

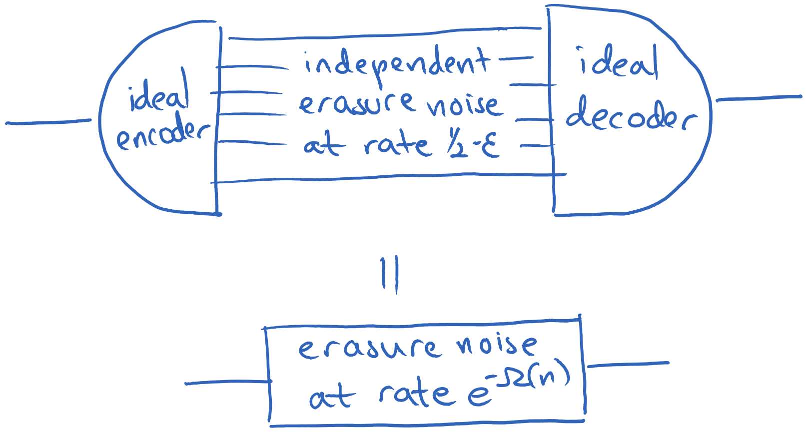
Theorem: The threshold for erasure noise is ½.

[Knill, q-ph/0312190]

Proof sketch:

① **Lemma**: For any $\varepsilon > 0$, there exists a (CSS, stabilizer) QECC that with high probability corrects $\frac{1}{2} - \varepsilon$

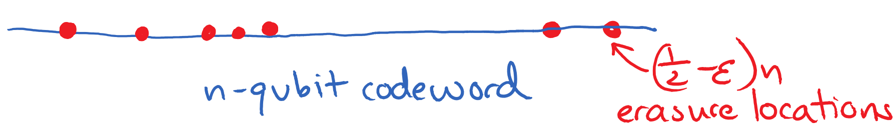
probability erasure errors.



Proof sketch:

Let C be a uniformly random n-qubit stabilizer code (i.e., pick n-1 pairwise commuting random Pauli stabilizers).

$$\underset{\substack{\text{over choice of C} \\ \text{and the random error E}}}{\mathbb{P}\text{rob}} \left[\begin{array}{l} \text{code C incorrectly} \\ \text{decodes error E} \end{array} \right] \stackrel{?}{=} 2^{-\Omega(n)} .$$



The actual error gives some syndrome $\sigma \in \{+1, -1\}^{n-1}$

$$\mathbb{P}\left[\begin{array}{l} \text{another Pauli error on the same} \\ \text{qubits gives the same syndrome} \end{array} \right]$$

$$\leq \frac{\#\text{ of other Pauli errors on same qubits} = 4^{(\frac{1}{2}-\varepsilon)n}}{2^{n-1}}$$

since any given nontrivial Pauli error's syndrome on a random stabilizer is +1 or -1 with equal prob. ½

$$= 2^{1-2\varepsilon n} = 2^{-\Omega(n)} \quad \checkmark$$

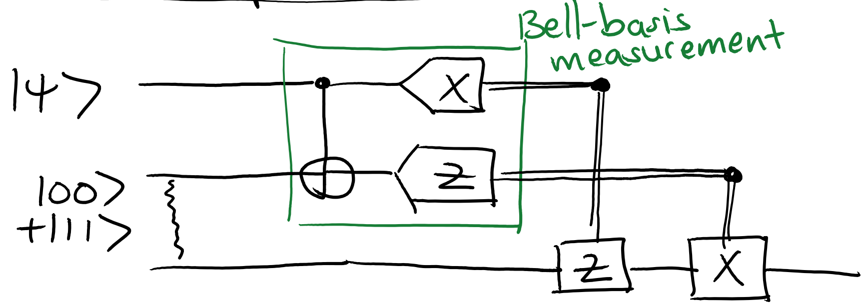
$$2^{-\Omega(n)} = \mathbb{P}_{C,E}\left[\text{C fails on E}\right] = \sum_C \mathbb{P}[C] \cdot \mathbb{P}_E[\text{C fails on E}]$$

so a good code exists. $\square$

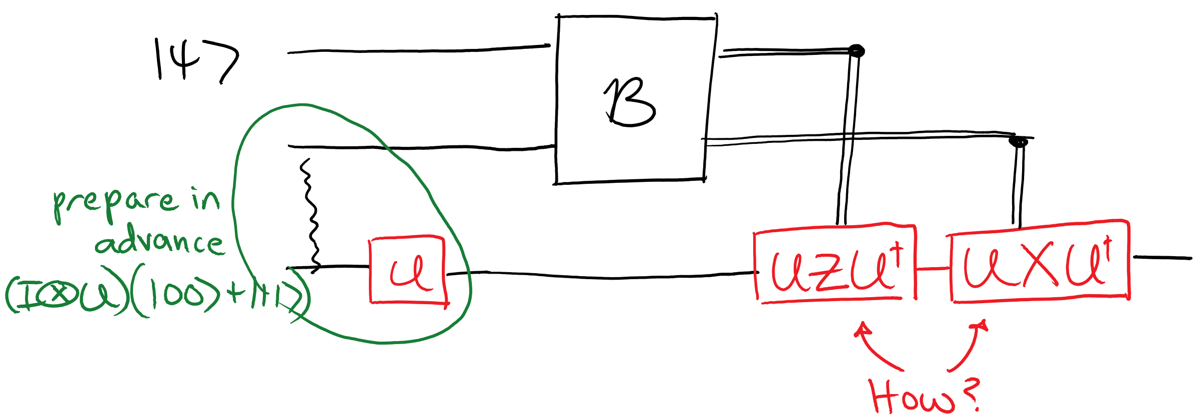
2 Computation by teleportation:

② Computation by teleportation:

Standard teleportation:



+Computation: Want $|\psi\rangle \mapsto \mathcal{U}|\psi\rangle$



Fact: The Clifford group (generated by CNOT, H, T), and

$$P = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$$

form a **universal** gate set.

- For $\mathcal{U} \in$ Clifford, $\mathcal{U}Z\mathcal{U}^\dagger$ and $\mathcal{U}X\mathcal{U}^\dagger$ are Paulis ✓
- For $\mathcal{U} = P$, $PZP^\dagger = Z$

  $PXP^\dagger = e^{-i\pi/4}\begin{pmatrix} 0 & 1 \\ i & 0 \end{pmatrix}$ ← a Clifford!

③ Putting it together:

We get universal quantum computation from:

- ability to apply Paulis X, Z

- preparation of states
  $(I\otimes H)(|00\rangle+|11\rangle)$, $(I\otimes T)(|00\rangle+|11\rangle)$
  $\mathrm{CNOT}_{2,3}(|00\rangle+|11\rangle)^{\otimes 2}$
  $(I\otimes P)(|00\rangle+|11\rangle)$
- Bell basis measurements

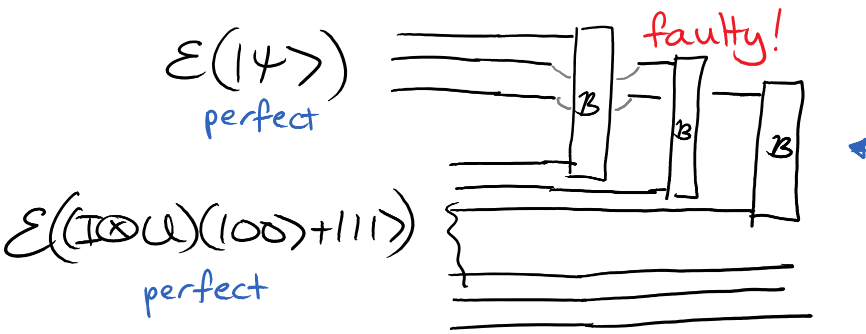
with adaptive classical control.

Fault-tolerance scheme: Do it at the encoded level...

- Prepare states $\mathcal{E}\big((I\otimes H)(|00\rangle+|11\rangle)\big)$, ...

  If an error is detected, throw it away!

- Teleport to compute:



use the Bell measurements to determine the syndrome, decode the logical Bell measurement $\longrightarrow \mathcal{E}(U|\psi\rangle)$

Question: Is this scheme efficient?
What is the overhead?

$\Rightarrow$ Erasure error threshold is $\geq 1/2$.

Remark: This is tight. Why? (no-cloning) $\square$

Morals:

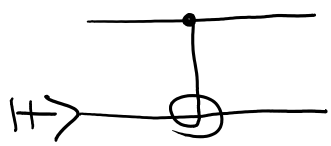
Morals:

- Detected errors are **much** nicer than undetected errors
- Large QECCs can be very efficient
- "Ancilla factories": preparing large encoded states is a key problem
  - for initialization, error correction, computation by teleportation
- Decoding efficiency is important
- The **overhead** is just as important as the noise threshold
  - overhead is highest just below the threshold but drops rapidly with lower noise rates

OTHER FAULT-TOLERANCE SCHEMES

Steane-style error correction

Observe:

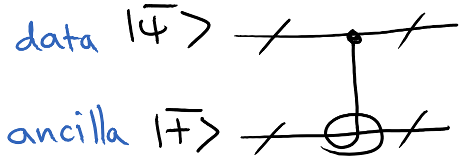


has no effect

$X|+\rangle = |+\rangle$

$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$\therefore$ on codewords for a CSS code,



has no logical effect

but it copies X errors from the data to the ancilla

— measure ancilla (in Z basis) to determine errors

Since it uses transversal gates, this is fault tolerant for any CSS code, provided the encoded ancilla is prepared fault tolerantly

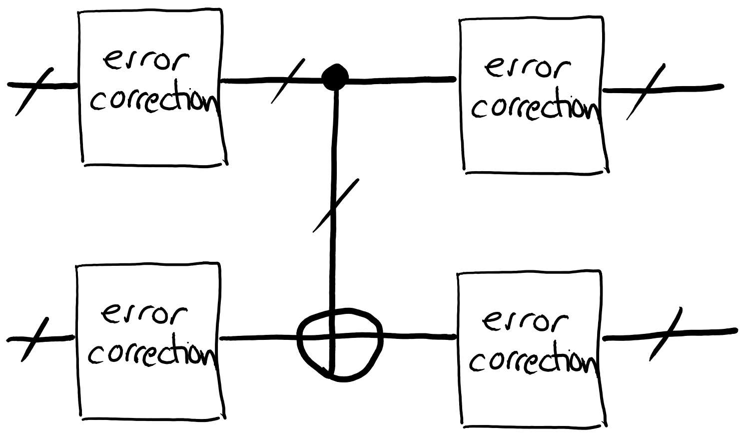
(in an ancilla factory... typically you prepare lots of ancillas)

(and check them to catch correlated errors)

AGP THRESHOLD THEOREM & malignant set counting

[Aliferis, Gottesman, Preskill, http://arxiv.org/abs/quant-ph/0504218]

- The easiest way of lower-bounding your fault-tolerance scheme's noise threshold.



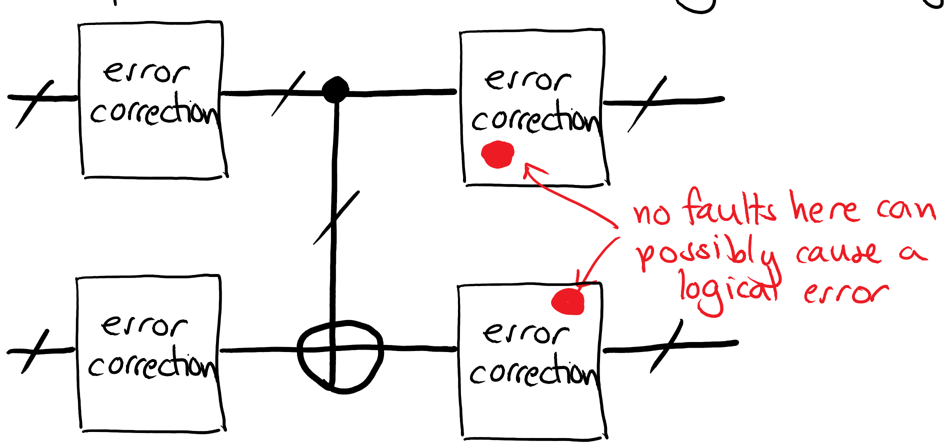
N error locations, fault tolerant, distance 3

$\Rightarrow$ effective error rate $\leq \binom{N}{2} p^2$

$\Rightarrow$ tolerable noise threshold $\geq \frac{1}{\binom{N}{2}}$

Better lower bound: "Malignant set counting"

not all pairs of locations can cause a logical error, e.g.,



$\Rightarrow$ Solve

$$p = (\#\text{ malignant pairs}) \cdot p^2 + \binom{N}{3} p^3$$

to determine the threshold.

for every set of 2 locations, see if $XX, XY, \ldots, ZZ$ errors can lead to any logical error

Example: [Aliferis & Cross, http://arxiv.org/abs/quant-ph/0610063]

Code	FTEC	locs.	ε_0 ($\times 10^{-4}$)	$\varepsilon_0^{\mathrm{MC}}$ ($\times 10^{-4}$)
Steane [[7,1,3]]	Steane	575	0.27	
$\mathcal{C}_{BS}^{(3)}$ [[9,1,3]]	Steane	297	1.21	1.21 ± 0.06
	Knill	297	1.26	1.26 ± 0.05
$\mathcal{C}_{BS}^{(5)}$ [[25,1,5]]	Steane	1,185	**1.94**	1.92 ± 0.02
	Knill	1,185		$\mathbf{2.07 \pm 0.03}$
Golay [[23,1,7]]	Steane	7,551		≈ 1
$\mathcal{C}_{BS}^{(7)}$ [[49,1,7]]	Steane	2,681		1.74 ± 0.01
	Knill	2,681		1.91 ± 0.01

TABLE I: Rigorous lower bounds on the accuracy threshold, ε_0, for adversarial stochastic noise with the concatenated Bacon-Shor code of varying block size and comparison with prior rigorous lower bounds using the concatenated Steane [[7,1,3]] code [14] and Golay [[23,1,7]] code [16]. The third column gives the number of locations in the CNOT extended rectangle [14]. The forth column gives exact lower bounds on ε_0; the results are obtained using a computer-assisted combinatorial analysis. The fifth column is the Monte-Carlo estimate for ε_0 with 1σ uncertainties. Bold fonts indicate the best results in each column.

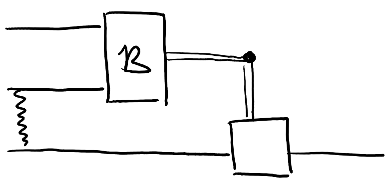
Remark: Can also count malignant triples, etc., or can sample random sets to estimate malignant set counts (especially useful at very low noise rates).

Highest proven threshold: $\sim 10^{-4}$ depolarizing noise per gate

Threshold Existence Theorems

① Leakage errors

$$\underbrace{|0\rangle, |1\rangle}_{\text{computational space}}, \underbrace{|2\rangle, |3\rangle, \ldots}_{\text{leaks}}$$

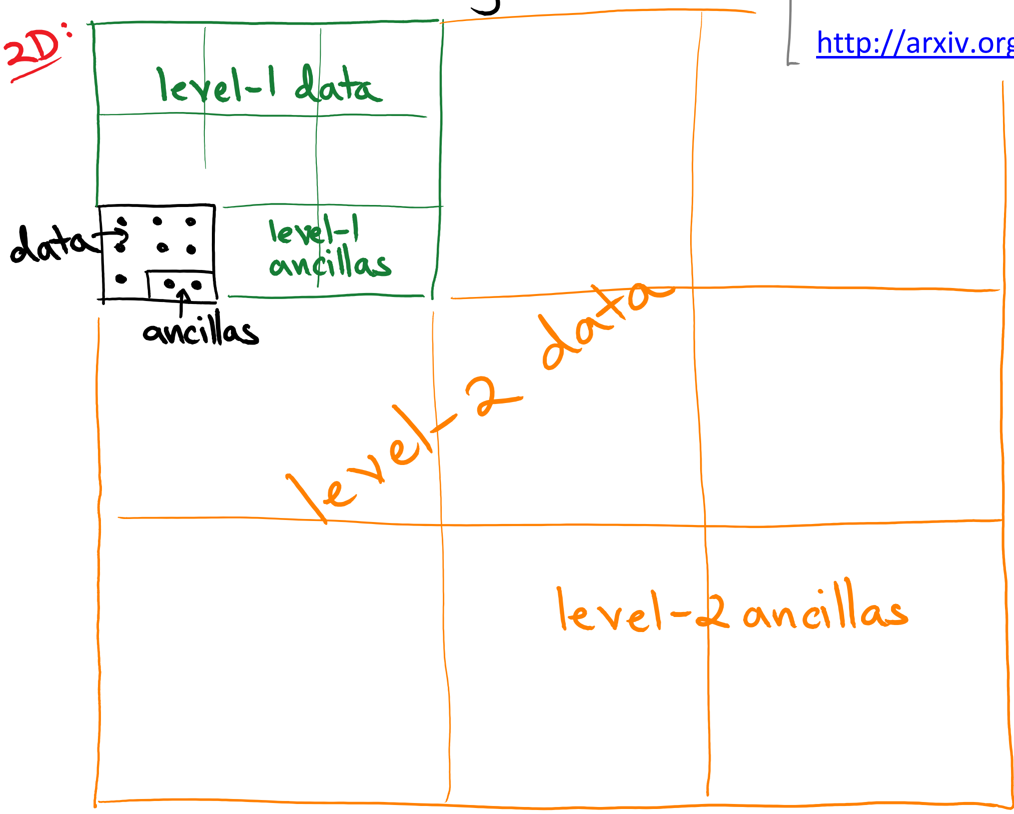


teleportation eliminates leaks (leaks → erasure errors)

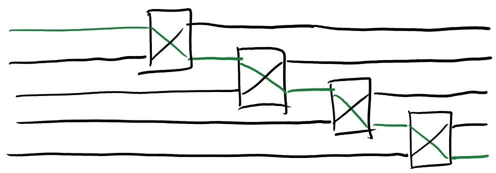
② Geometric locality constraints

[Gottesman http://arxiv.org/abs/quant-ph/9903099]

2D:



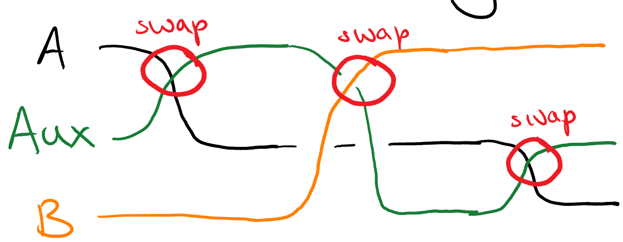
1D:



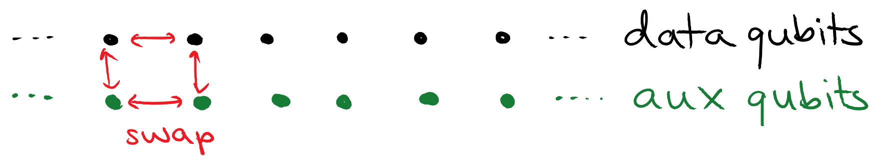
Problem: Swaps are not fault tolerant
(get weight-2 errors in code block w/ 1st-order probability)

Solutions:

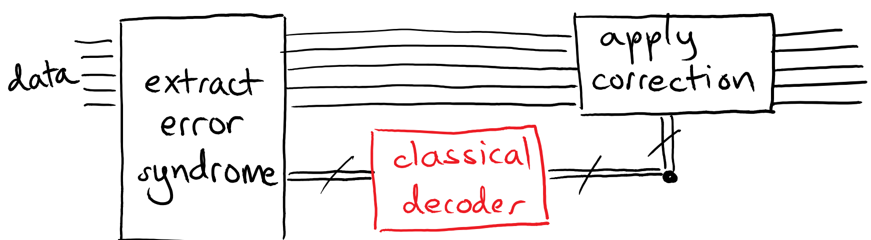
- Allow next-nearest neighbor interactions



- Almost-1D architecture



③ All-unitary control



↑ this can be run inside the quantum computer, but then it is not fault tolerant

Trivial solution: Run the above circuit quantumly, but only apply the correction to qubit 1. (Then do it all again for qubit 2, etc.)

Moral: Classical control is very helpful, allows tolerating much more noise.

④ Non-Markovian noise

$$\mathcal{H} = \mathcal{H}_S + \mathcal{H}_B + \mathcal{H}_{SB}$$

system bath interaction

1. If SB only touches interacting data qubits
$$\mathcal{E} = \max \| H_{SB}(t) \| \cdot t_0$$

2. For noise coupling all data qubits, decaying in space
$$\mathcal{E}^2 = \max_i \sum_j \| H_{ijB} \| \cdot t_0$$

[Aharonov, Kitaev, Preskill, http://arxiv.org/abs/quant-ph/0510231]

Problem: Hamiltonian norms are not measurable, may be infinite (e.g., harmonic oscillator)

[Ng & Preskill, http://arxiv.org/abs/0810.4953]

Challenges:

- Rigorous threshold lower bounds are far below simulation-based threshold estimates (especially for the surface code). Which is right?
- Rigorous thresholds for Hamiltonian noise are generally ~quadratically lower than for stochastic noise.

  Is this an artifact of the proofs?
  Simulations can't help.
- What are the thresholds for real noise models? Do they even exist? [Alicki, http://arxiv.org/abs/1310.8457]

Do they even exist? http://arxiv.org/abs/1310.8457

Overhead Analysis of Fault-Tolerance Schemes

What is the overhead? What are the bottlenecks?

See, for example,

Comparing the Overhead of Topological and Concatenated Quantum Error Correction http://arxiv.org/abs/1312.2316

Martin Suchara, Arvin Faruque, Ching-Yi Lai, Gerardo Paz, Frederic T. Chong, John Kubiatowicz

Table 3. Logical gate count for Shor's algorithm factoring a 1024-bit number. A conservative estimate of parallelization factors shown.

Gate	Occurences	Parallelism
$CNOT$	1.18×10^9	1
H	3.36×10^8	1
T or $T^\dagger$	1.18×10^9	2.33

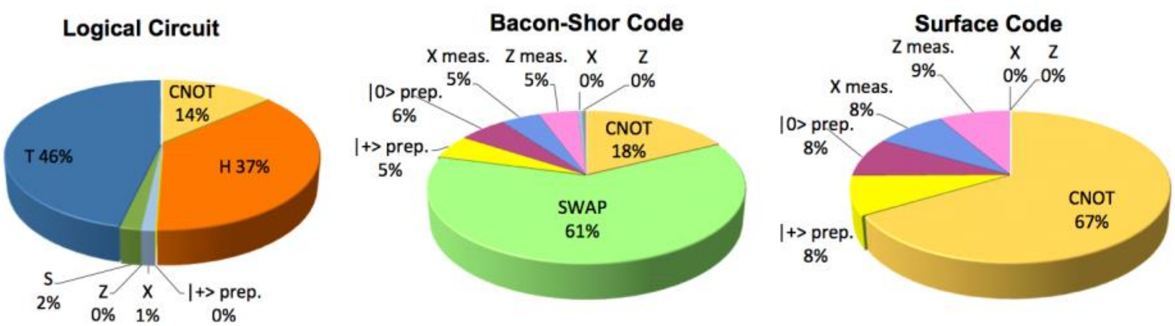


Fig. 13. The gate types used in a typical logical circuit and a typical fault-tolerant circuit that uses the Bacon-Shor and Surface codes all differ.

Table 4. The resources needed to factor a 1024-bit number with Shor's algorithm. Results shown for the surface and Bacon-Shor codes on three technologies.

Technology	Neutral Atoms	Supercond. Qubits	Ion Traps
Gate error	1×10^{-3}	1×10^{-5}	1×10^{-9}
Avg. gate time	19,000 ns	25 ns	32,000 ns
Execution time	2.62 years	10.81 hours	2.22 years
No. qubits	5.29×10^8	4.57×10^7	1.44×10^8
No. gates	1.02×10^{21}	2.55×10^{19}	5.10×10^{19}
Dominant gate	*CNOT*	*CNOT*	*CNOT*
Code distance	17	5	3
Logical gate error	4.99×10^{-11}	2.95×10^{-11}	4.92×10^{-15}
Logical gate time	1.29×10^5 ns	2.10×10^2 ns	5.96×10^5 ns
No. qubits per logical	3.73×10^4	3.23×10^3	1.16×10^3
No. gates per logical	1.11×10^5	9.60×10^3	3.46×10^3
Execution time	N/A	5.10 years	57.98 days
No. qubits	N/A	2.65×10^{12}	4.60×10^5
No. gates	N/A	1.16×10^{32}	4.07×10^{18}
Dominant gate	N/A	*SWAP*	*CNOT*
Code concatenations	N/A	5	1
Logical gate error	N/A	3.42×10^{-15}	5.09×10^{-14}
Logical gate time	N/A	1.42×10^7 ns	7.27×10^5 ns
No. qubits per logical	N/A	2.82×10^8	49
No. gates per logical	N/A	1.18×10^{11}	79

Surface code
better at high error rates

Bacon-Shor code
better at low error rates

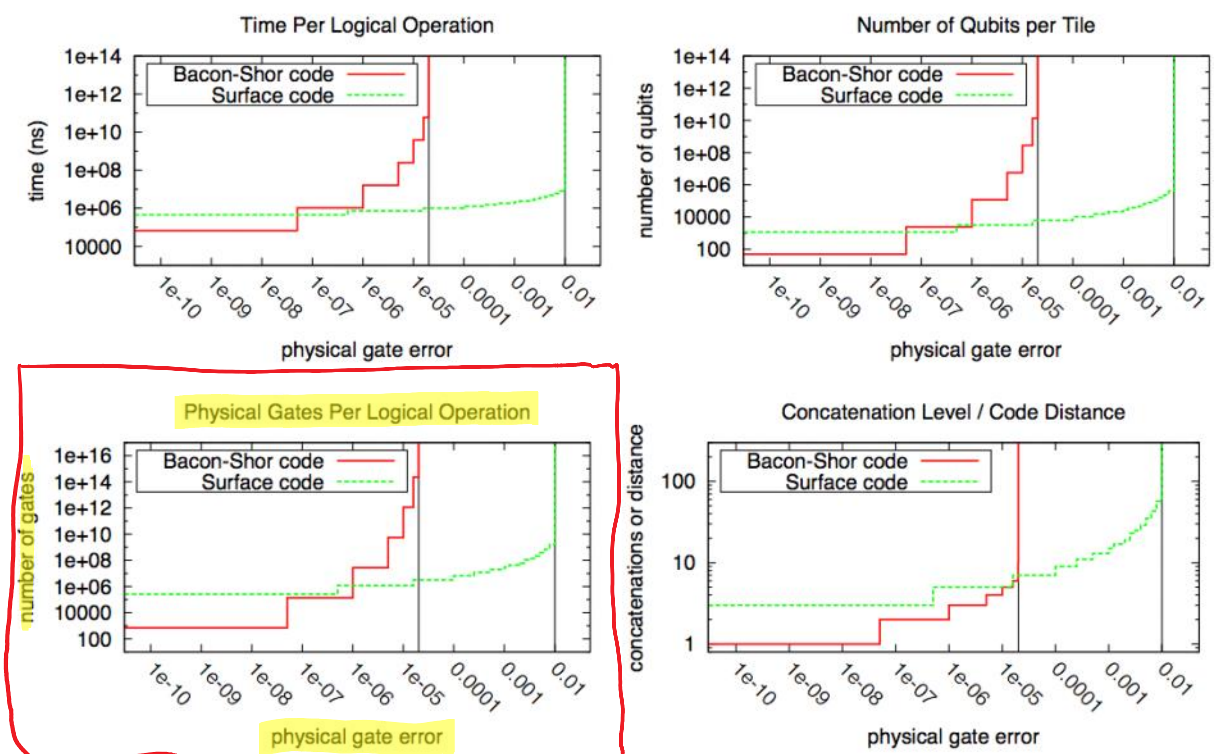


Fig. 10. Properties of error correction in an abstract quantum technology with physical gate error varying between 1×10^{-10} and 1×10^{-2}. Vertical lines indicate the error correction threshold of the Bacon-Shor and Surface error-correcting codes. The target error rate for a logical operation was chosen to be 1×10^{-10}.

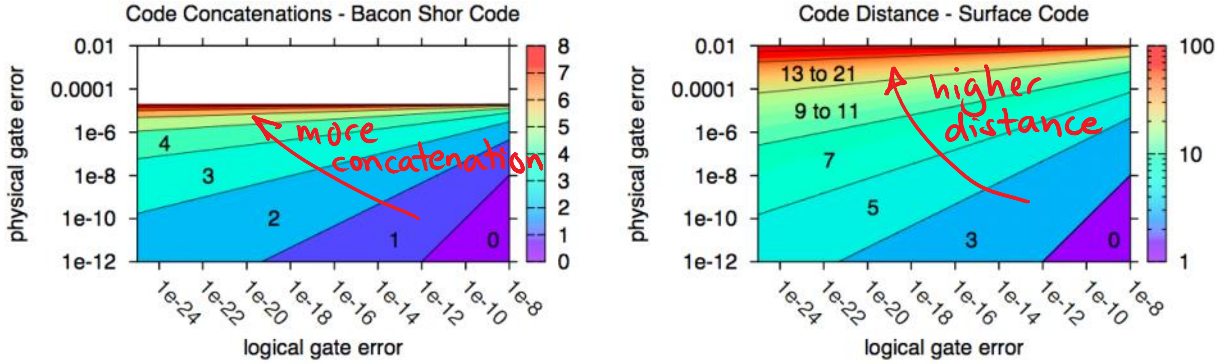


Fig. 11. The required concatenation level and code distance of the Bacon Shor and surface codes increase with increasing gate error of the physical technology and decreasing desired logical gate error.