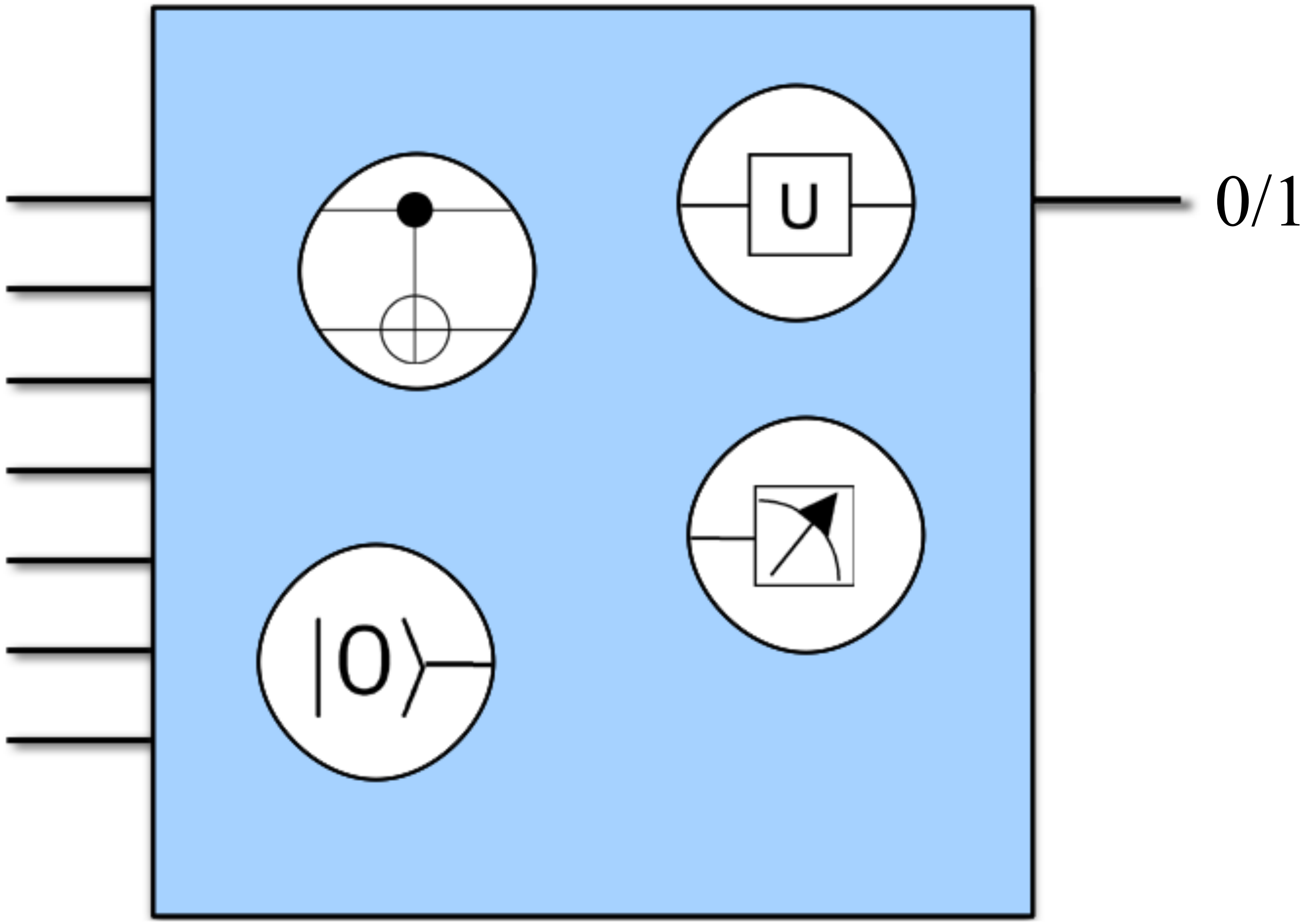
Fault-tolerance threshold for a distance-three quantum code

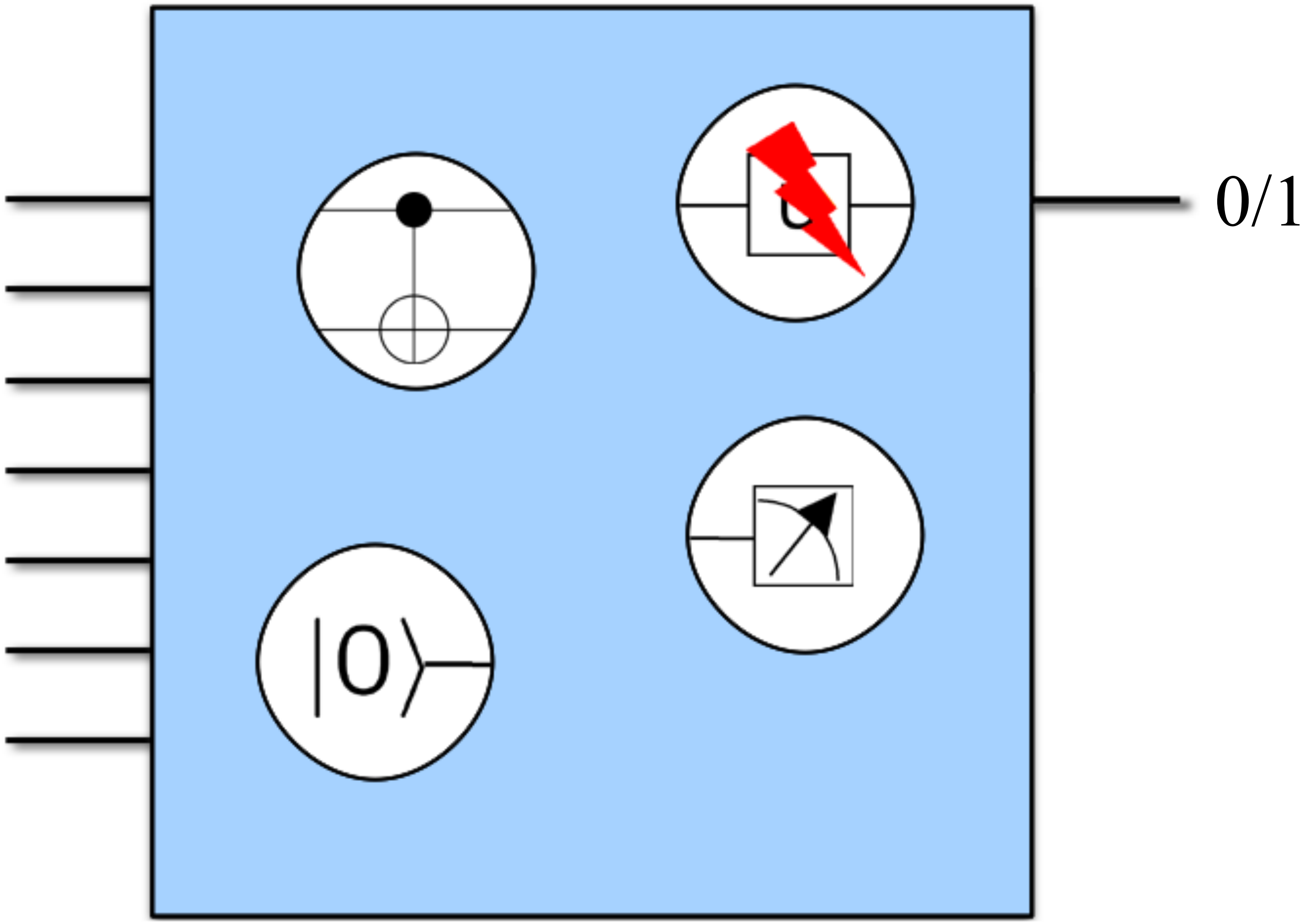
Ben Reichardt

UC Berkeley

N gate circuit

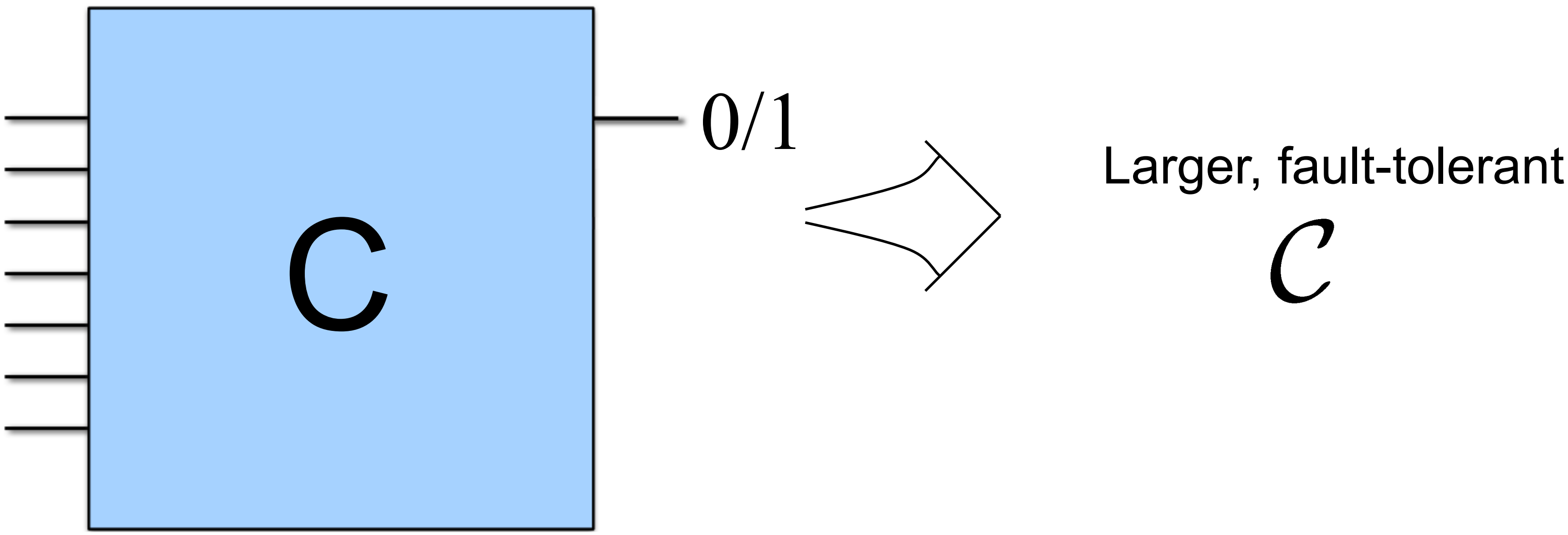


N gate circuit $\Rightarrow$ Need error $\ll 1/N$



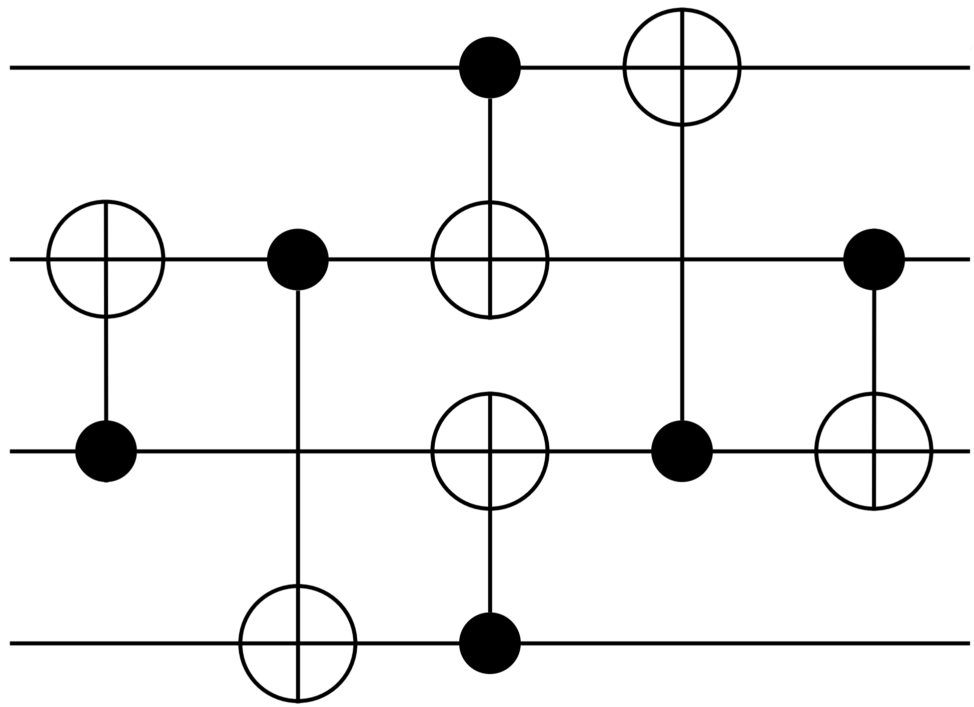
Quantum fault-tolerance problem

- Classical fault-tolerance: Von Neumann (1956)



Intuition

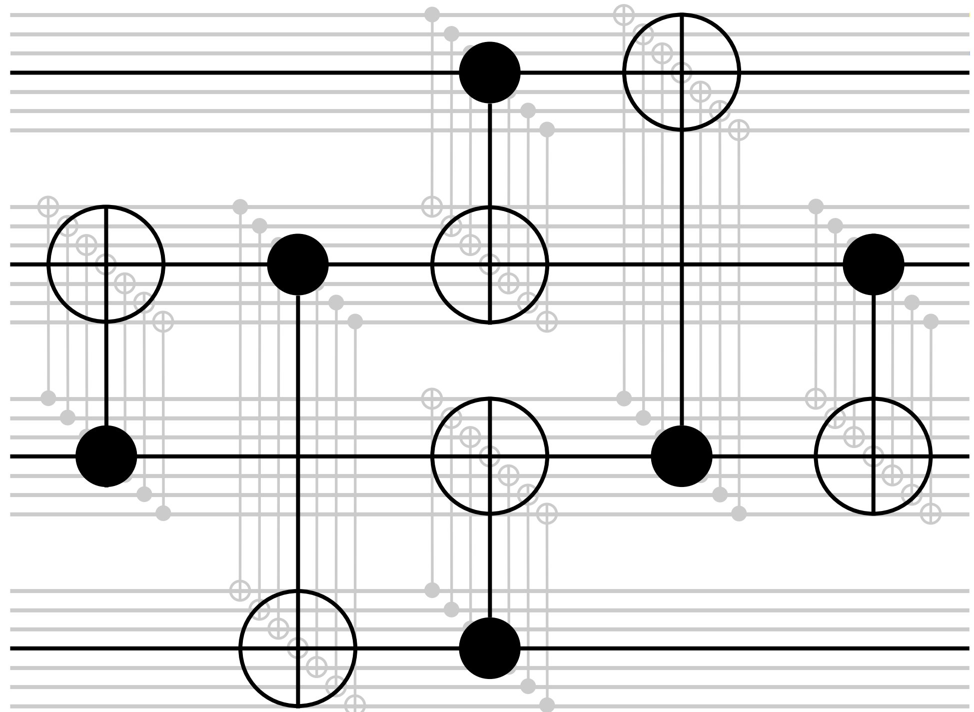
- Work on encoded data



• Quantum fault-tolerance: Shor (1996)

Intuition

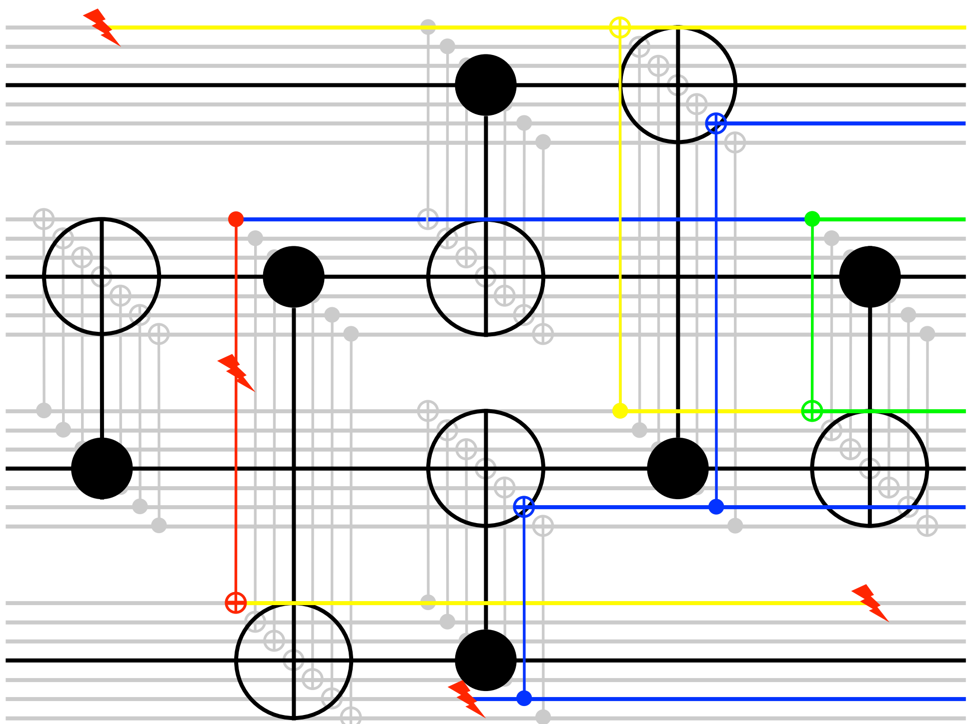
- Work on encoded data



- Quantum fault-tolerance: Shor (1996)

Intuition

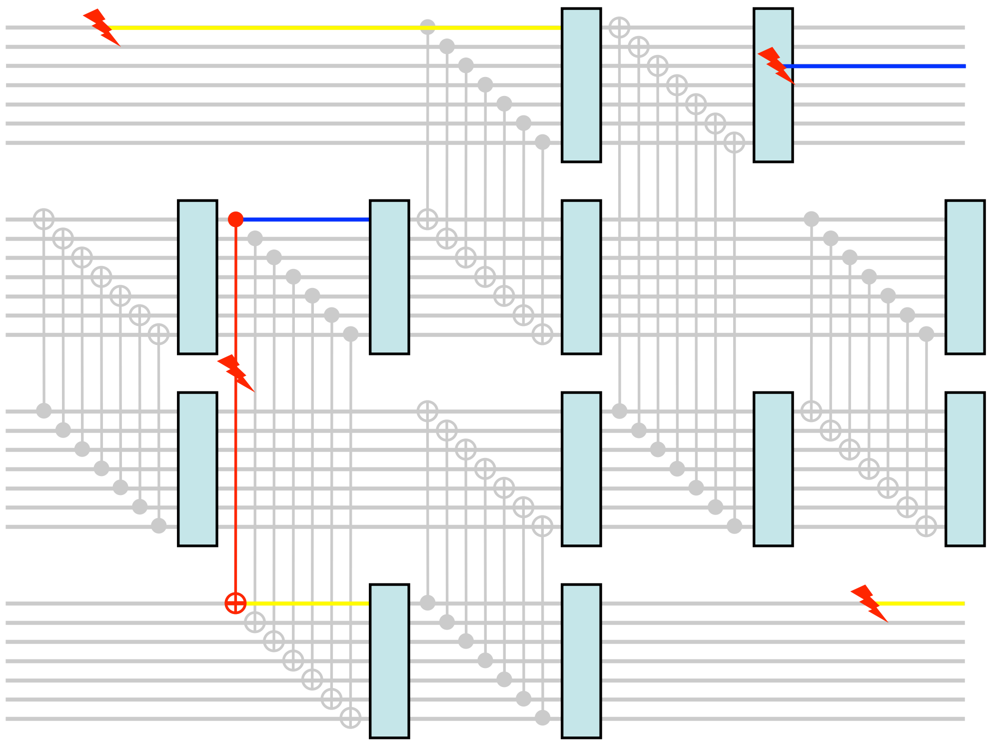
- Work on encoded data
- Correct errors to prevent spread



- Quantum fault-tolerance: Shor (1996)

Intuition

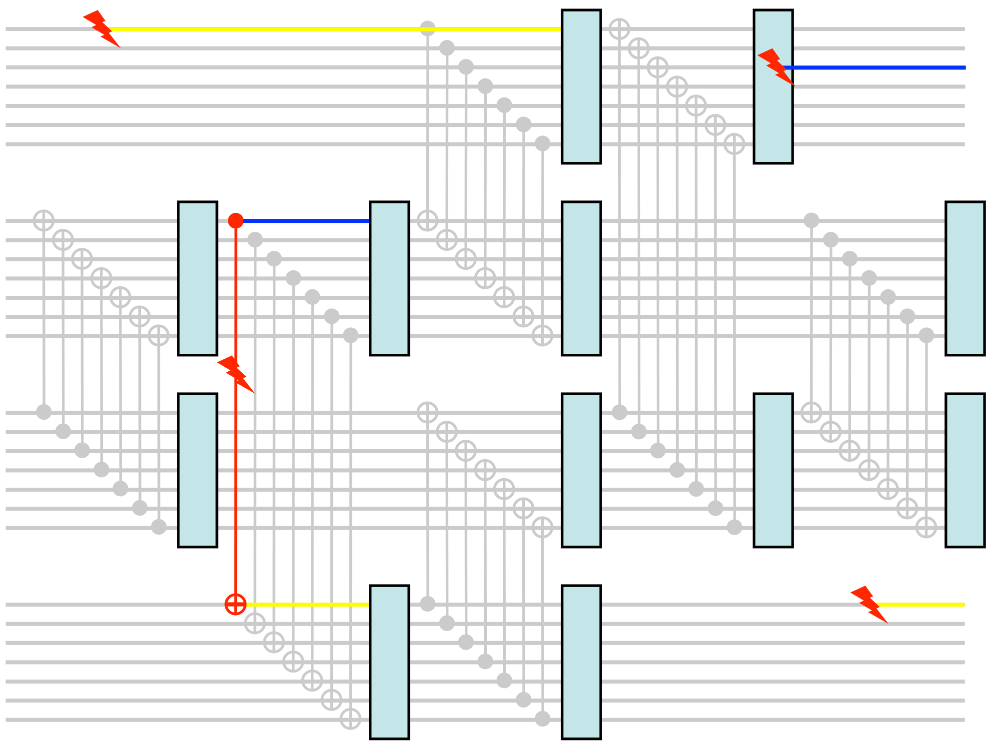
- Work on encoded data
- Correct errors to prevent spread



- Quantum fault-tolerance: Shor (1996)
  - – Using a poly(log N)-sized code, tolerate 1/poly(log N) gate error

Intuition

- Work on encoded data
- Correct errors to prevent spread
- Concatenate procedure for arbitrary reliability



- Shor (1996): poly(log N)-sized code to tolerate 1/poly(log N) gate error
- Aharonov & Ben-Or (1997), Kitaev (1997), Gotteman-Evslin-Kakade-Preskill (1997), Knill-Laflamme-Zurek (1997)

Threshold from concatenation

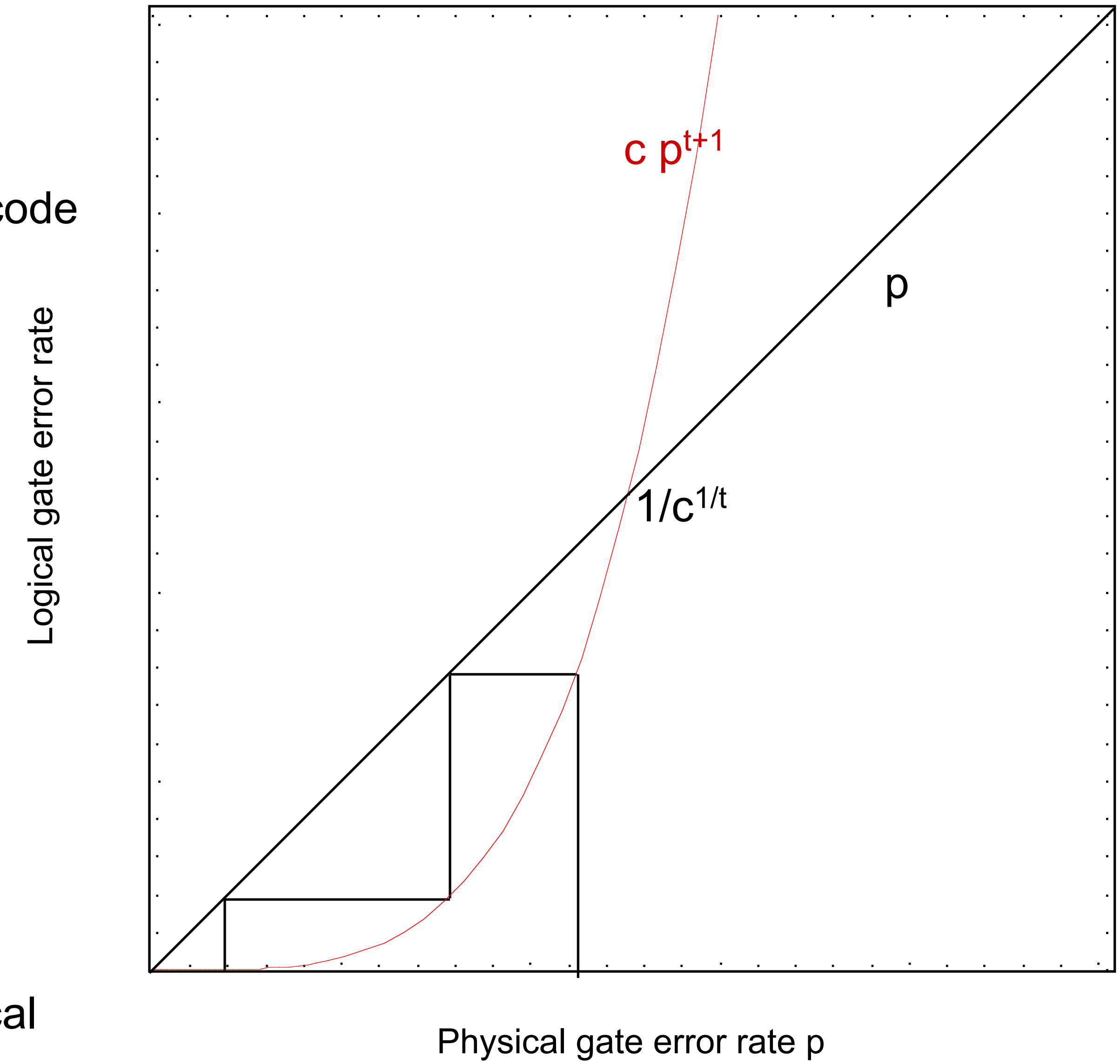
- N gate circuit

  $\Rightarrow$ Want error $\ll 1/N$

- m-qubit, t-error correcting code

Probability of error	Physical bits per logical bit
p	1
$c\,p^{t+1}$	m
$\sim p^{(t+1)^2}$	m^2
$p^{(t+1)^3}$	m^3

O(log log N) concatenations
poly(log N) physical bits / logical



Distance-3 code thresholds

- Basic estimates
  - Aharonov & Ben-Or (1997)
  - Knill-Laflamme-Zurek (1998)
  - Preskill (1998)
  - Gottesman (1997)
- Optimized estimates
  - Zalka (1997)
  - Reichardt (2004)
  - Svore-Cross-Chuang-Aho (2005)
- 2-dimensional locality constraint
  - Szkopek et al (2004)
  - Svore-Terhal-DiVincenzo (2005)

- But no constant threshold was even proven to exist for distance-3 codes!
  - Aharonov & Ben-Or proof only works for codes of distance at least 5
- Today: Threshold for distance-3 codes

Distance-2 code threshold

- Knill (2005) has highest threshold estimate ~5%
  - … Albeit with large constant overhead (1-3% more reasonable)
  - Again, no threshold has been proved to exist

- Gaps between proven and estimated thresholds
  - Estimates are as high as ~5%
  - But no proven lower-bounds (?)
  - Aliferis-Gottesman-Preskill (2005): 2.6×10^{-5}

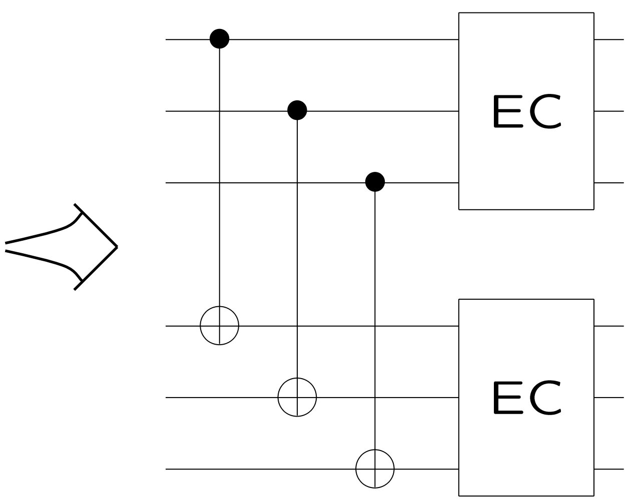
- Caveat on small versus large codes
  - Steane (2003) found 23-qubit Golay code had higher threshold (based on simulations), particularly with slow measurements
  - 23-qubit Golay code proven: 10^{-4}

Fault-tolerance for m-bit repetition code

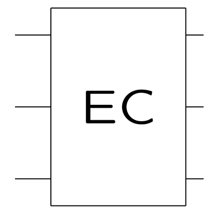
- **Encoding**
  - $0 \rightarrow 00\ldots0$
  - $1 \rightarrow 11\ldots1$

  } distance m

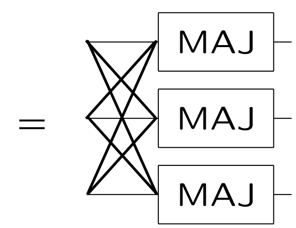
- **Gate compilation rule:** transverse, followed by error correction
  - e.g., $CNOT_L$



- **Error correction:**



  – e.g., classically, with majority gate and fan-out:



- **Concatenation…**

$$0_k = 0_{k-1}0_{k-1}0_{k-1} = \cdots = 0^{3^k}$$

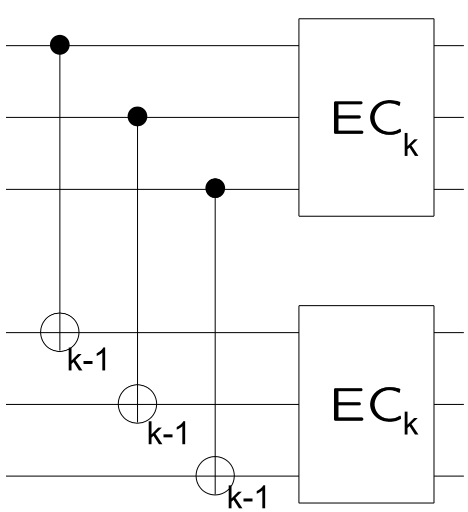
$$1_k = 1_{k-1}1_{k-1}1_{k-1} = \cdots = 1^{3^k}$$

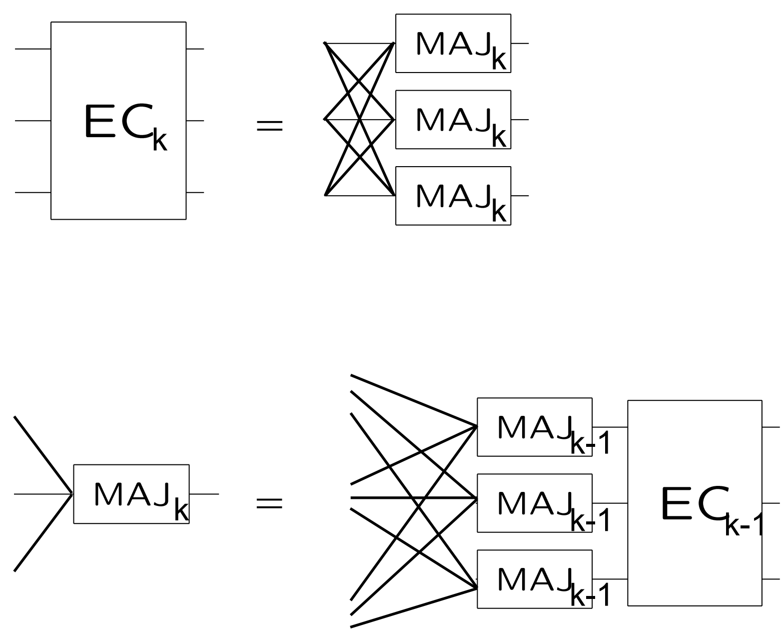
Classical fault-tolerance for repetition code

- **Concatenation:**

$$0_k = 0_{k-1}0_{k-1}0_{k-1} = \cdots = 0^{3^k}$$
$$1_k = 1_{k-1}1_{k-1}1_{k-1} = \cdots = 1^{3^k}$$

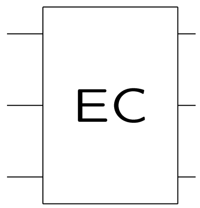
$CNOT_k =$



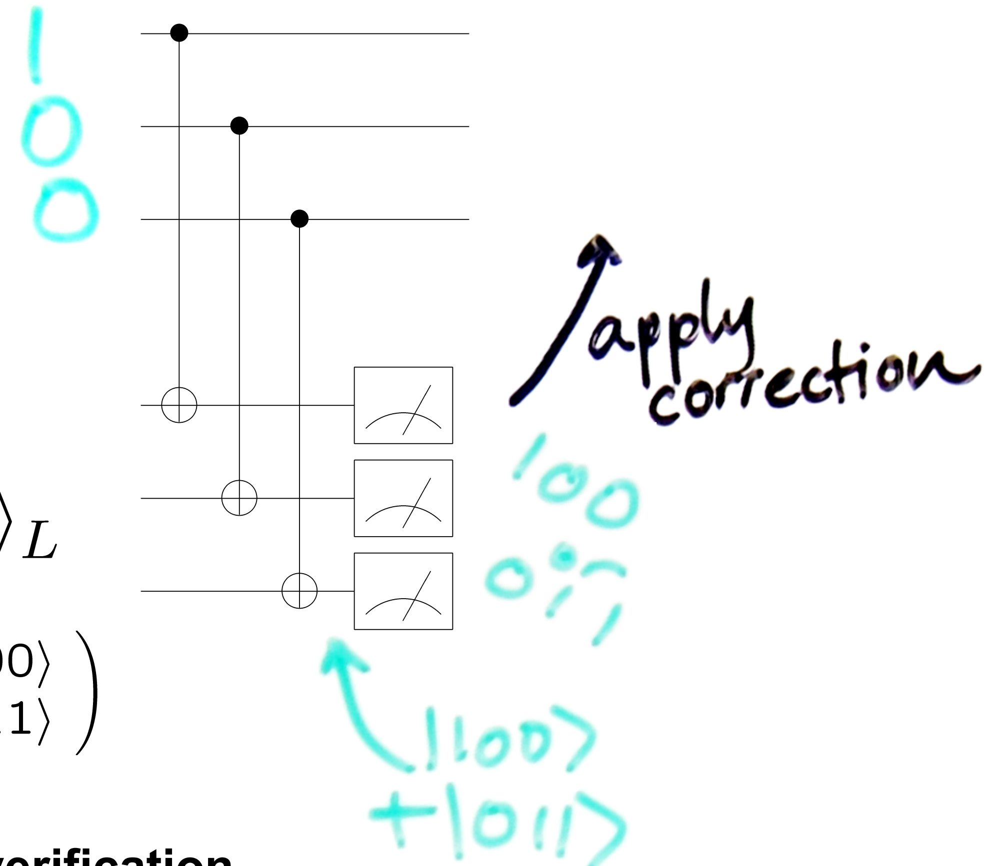


Quantum fault-tolerance for repetition code

- **Error correction**

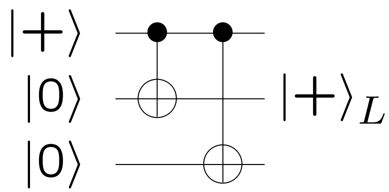


$=$

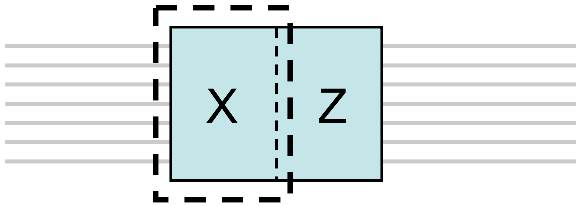


$$|+\rangle_L = \frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right)$$

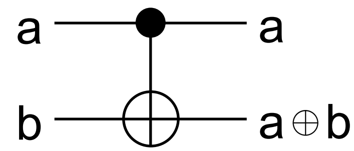
- **Ancilla preparation and verification**



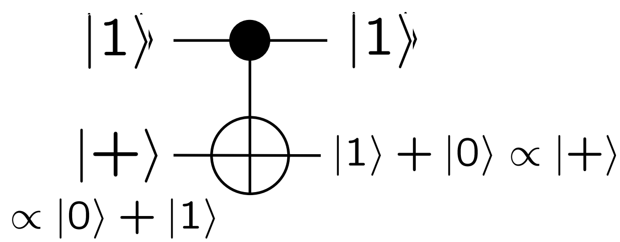
Quantum fault-tolerance scheme



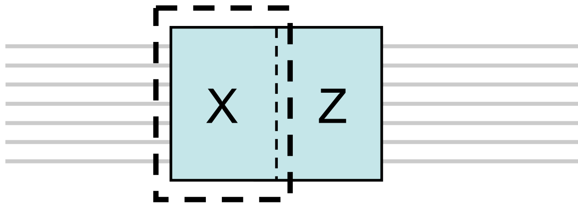
Def: CNOT



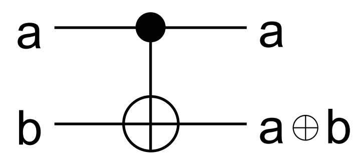
Fact 1:



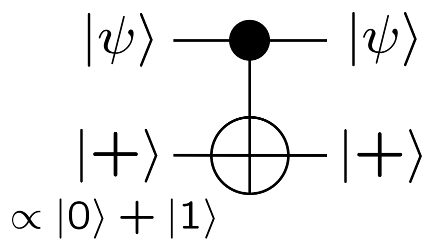
Quantum fault-tolerance scheme



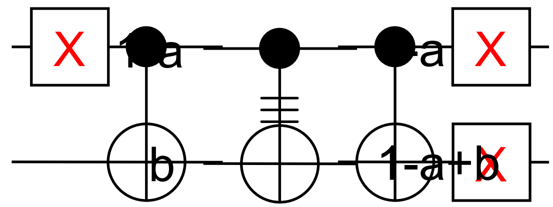
Def: CNOT



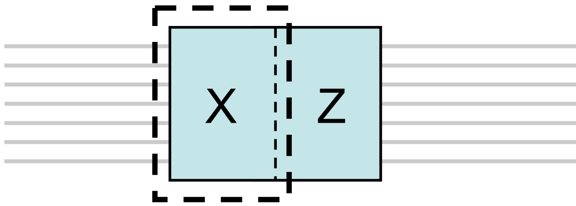
Fact 1:



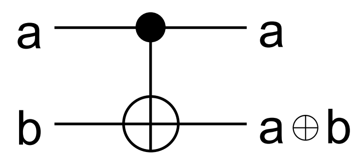
Fact 2:



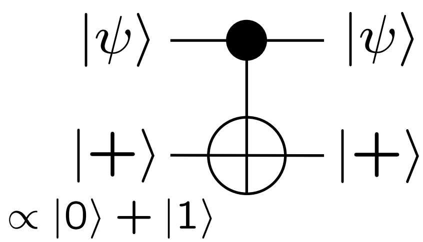
Quantum fault-tolerance scheme



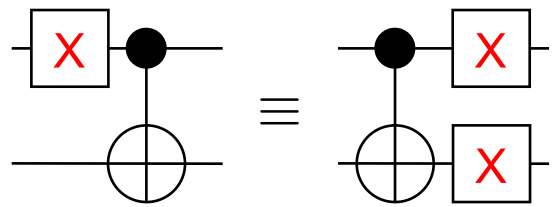
Def: CNOT

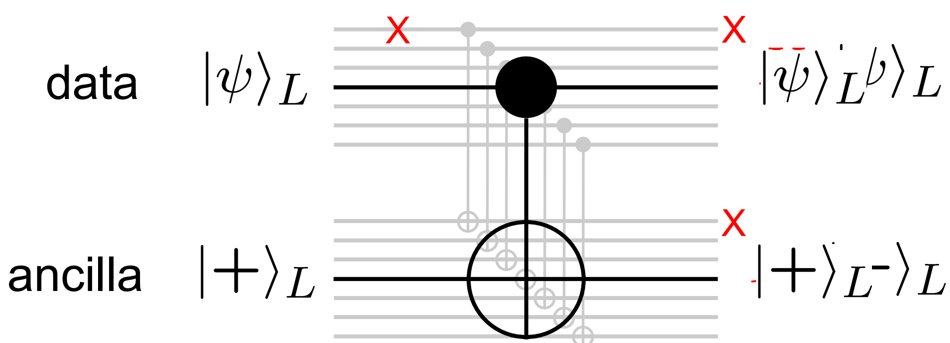


Fact 1:



Fact 2:



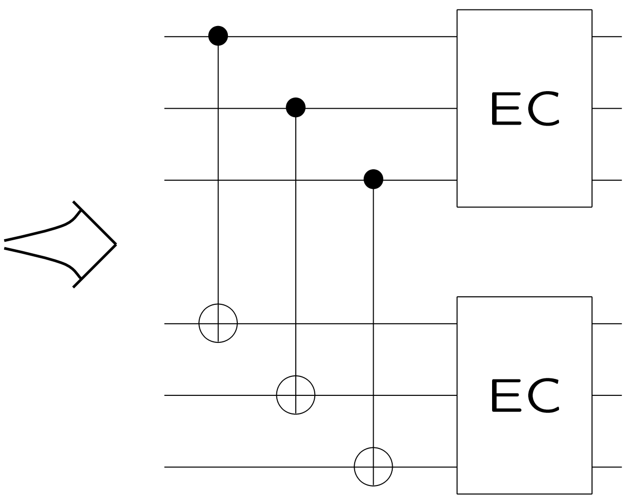


Fault-tolerance for m-bit repetition code

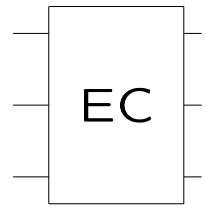
- **Encoding**
  - 0 → 00…0
  - 1 → 11…1

  distance m

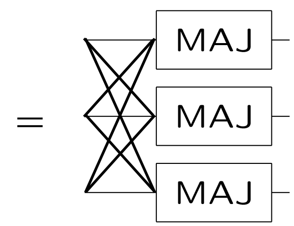
- **Gate compilation rule:** transverse, followed by error correction
  - e.g., CNOT_L



- **Error correction:**



  – e.g., classically, with majority gate and fan-out:



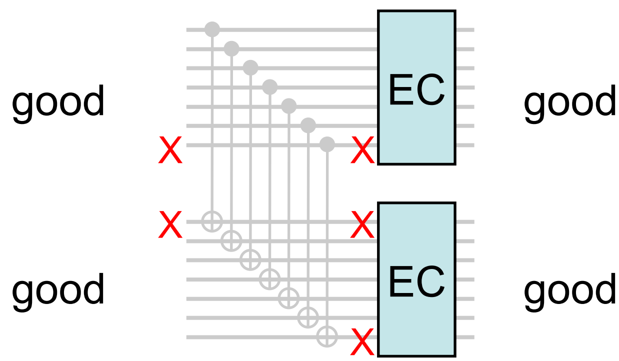
- **Concatenation…**

$$0_k = 0_{k-1}0_{k-1}0_{k-1} = \cdots = 0^{3^k}$$

$$1_k = 1_{k-1}1_{k-1}1_{k-1} = \cdots = 1^{3^k}$$

Aharonov & Ben-Or threshold proof intuition

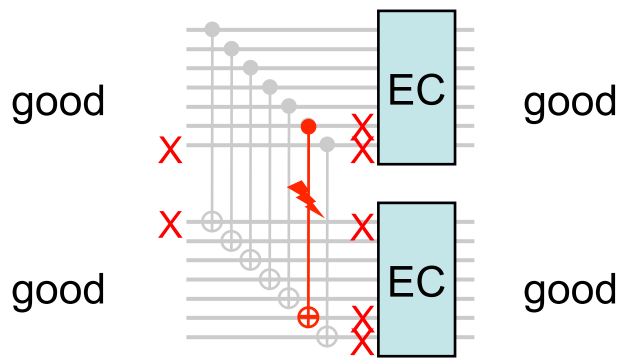
- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good "if it has at most one bad subblock.")



(assuming no level k-1 errors, m≥5)

Aharonov & Ben-Or threshold proof intuition

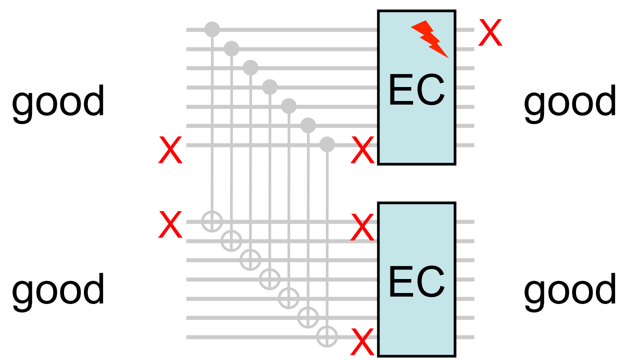
- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good “if it has at most one bad subblock.”)



(assuming one level k-1 error, m≥7)

Aharonov & Ben-Or threshold proof intuition

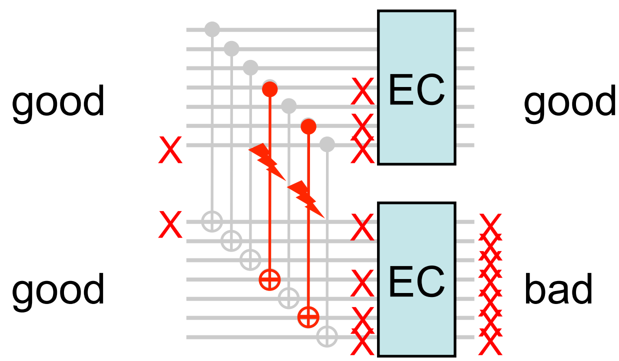
- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good "if it has at most one bad subblock.")



(assuming one level k-1 error, m≥7)

Aharonov & Ben-Or threshold proof intuition

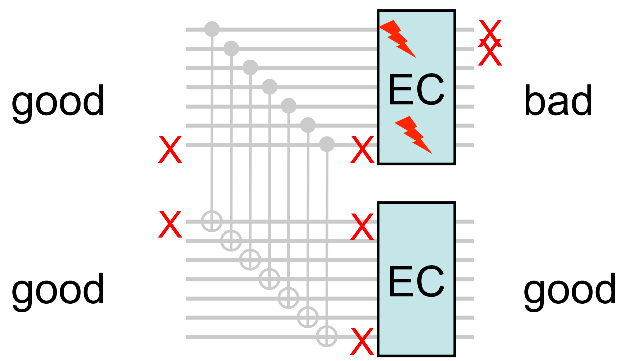
- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good “if it has at most one bad subblock.”)



(two level k-1 errors, m=7)

Aharonov & Ben-Or threshold proof intuition

- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good "if it has at most one bad subblock.")

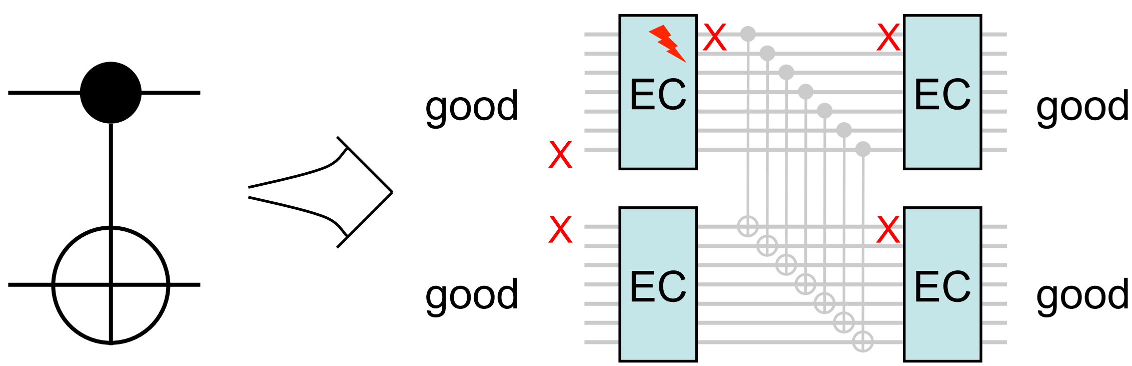


(two level k-1 errors)

Aharonov & Ben-Or threshold proof intuition

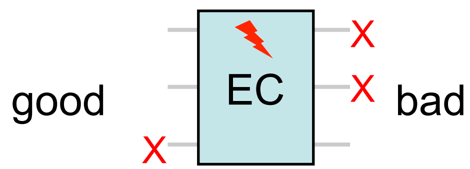
- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good "if it has at most one bad subblock.")

For distance-5 code:



Aharonov & Ben-Or threshold proof intuition

- **Idea:** Maintain inductive invariant of (1-)goodness. (A block is good "if it has at most one bad subblock.")
- Why not for distance-three codes?



(one level k-1 error is already too many)

- **New idea:** Most blocks should have no bad subblocks. Maintain inductive invariant of a controlled probability distribution of errors: "wellness." (A block is well "if it only rarely has a bad subblock.")

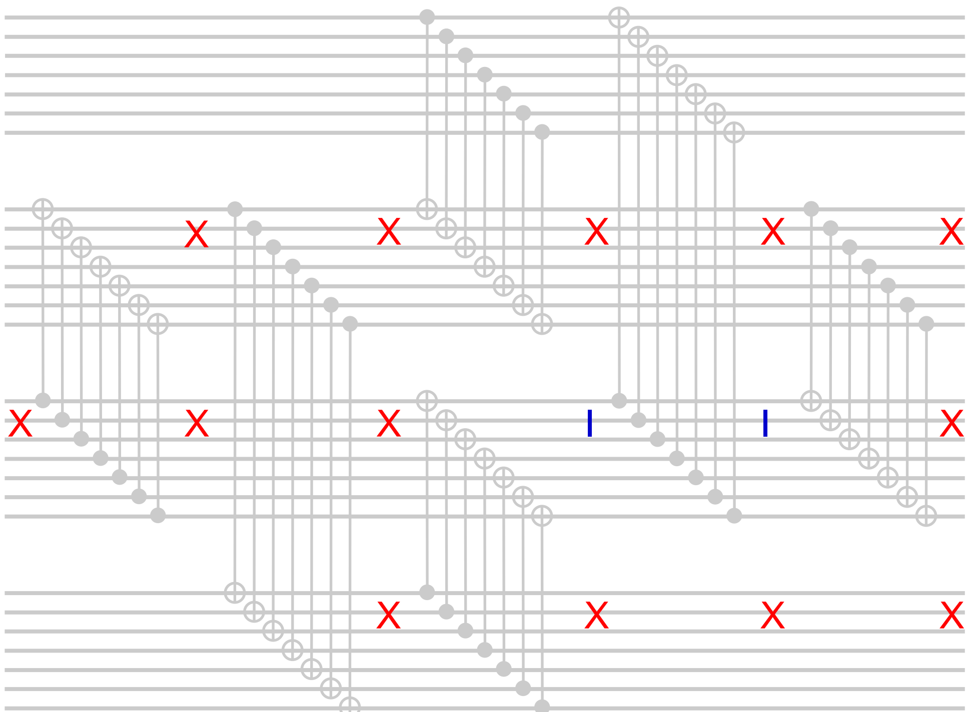
Concatenated distance-3 code proof overview

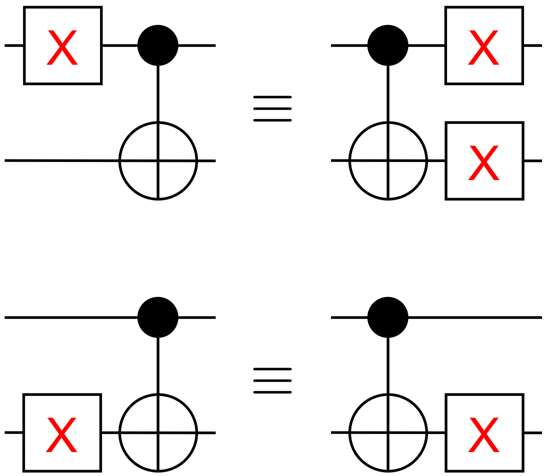
- Def: Error states
- Def: Relative error states
- **Def: 1-good block**
- Aharonov/Ben-Or threshold setup
- Def: Logical failure
- Aharonov/Ben-Or threshold proof
- **Def: “well” block**
- Distance-3 code threshold setup and proof for stabilizer operations

- Extension to **universality** via magic states distillation

Def: Error states

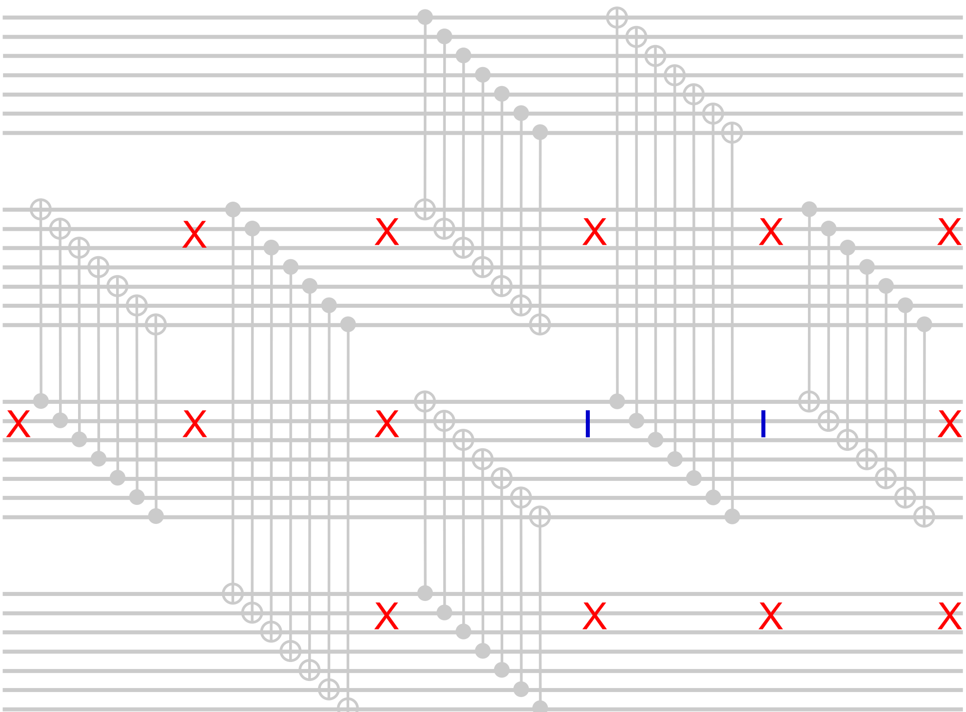
- **Tracking errors**



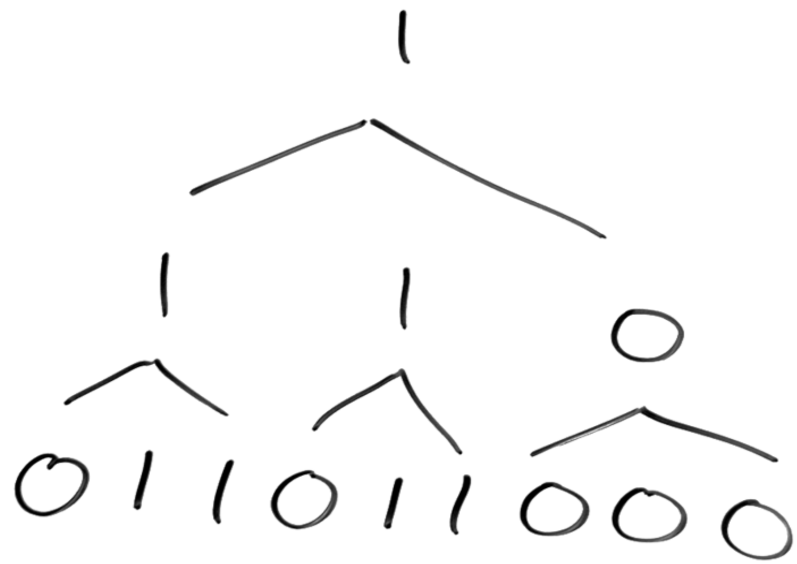


Def: Error states

- **Tracking errors**



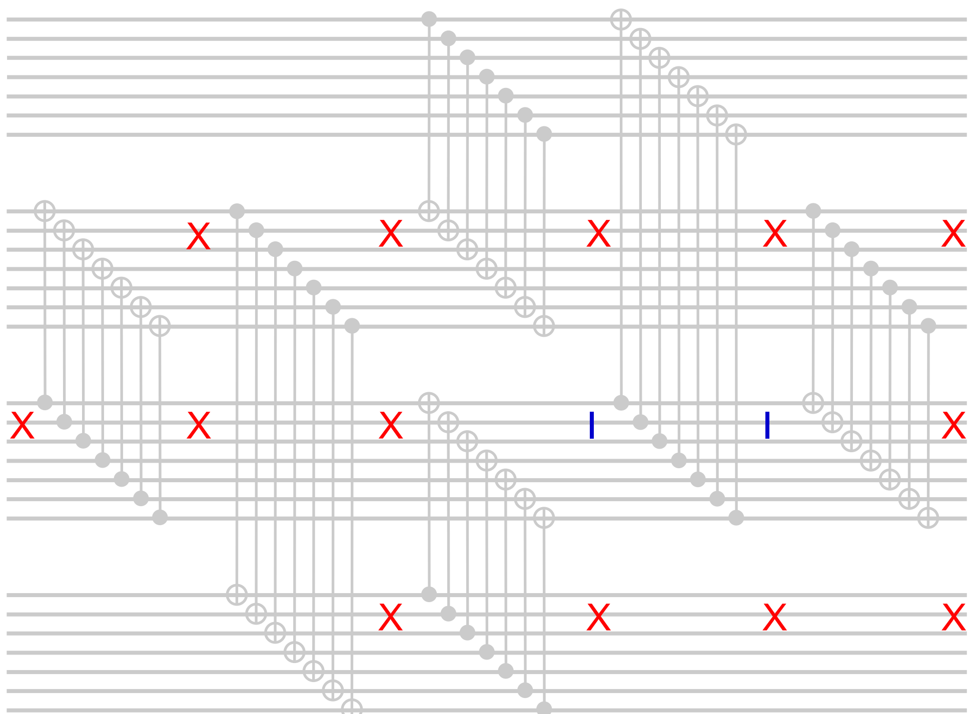
- **Block error states:** ideal recursive decoding



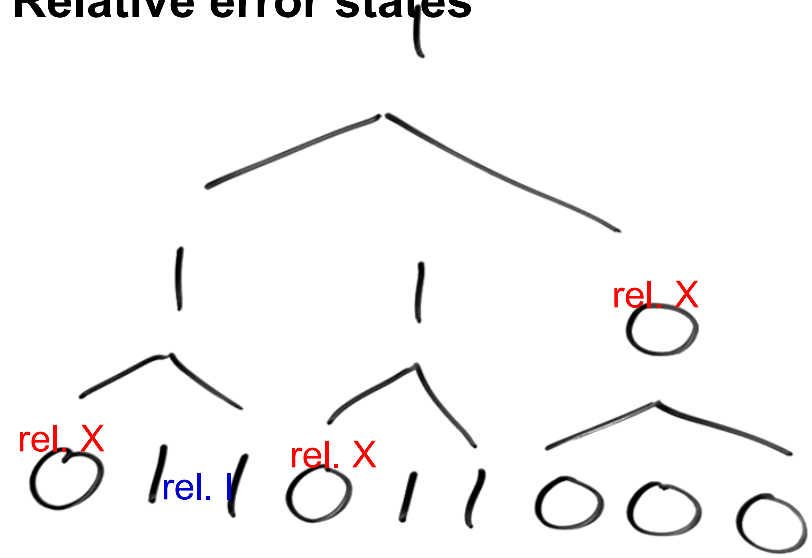
- Note: Block errors do not follow same rules as bit errors
  - e.g., 001 + 010 = 011

Def: *Relative* Error states

- **Tracking errors**



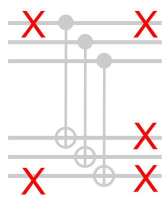
- **Block error states:** ideal recursive decoding
- **Relative error states**



- Note: relative state is relative to *state* of superblock, not superblock's relative state
- We can measure block relative states.

Def: good

- **Tracking errors**

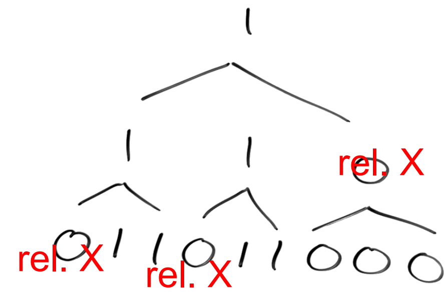


- **Block error states:** ideal recursive decoding
- **Relative error states**

- **Def:** A block$_k$ is relative (1-)good$_k$ if it has at most one subblock$_{k-1}$ either in relative error or not relative good$_{k-1}$ itself.
(Every bit [$\equiv$ block$_0$] is relative good$_0$.)

good examples

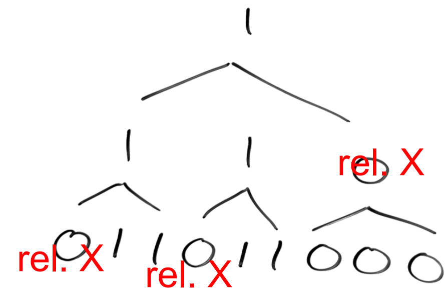
- **Relative error states** based on ideal recursive decoding
- A **good** block has at most one subblock either in relative error or bad.



good | bad

good examples

- **Relative error states** based on ideal recursive decoding
- A **good** block has at most one subblock either in relative error or bad.

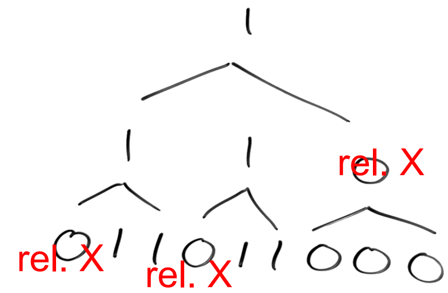


good

bad

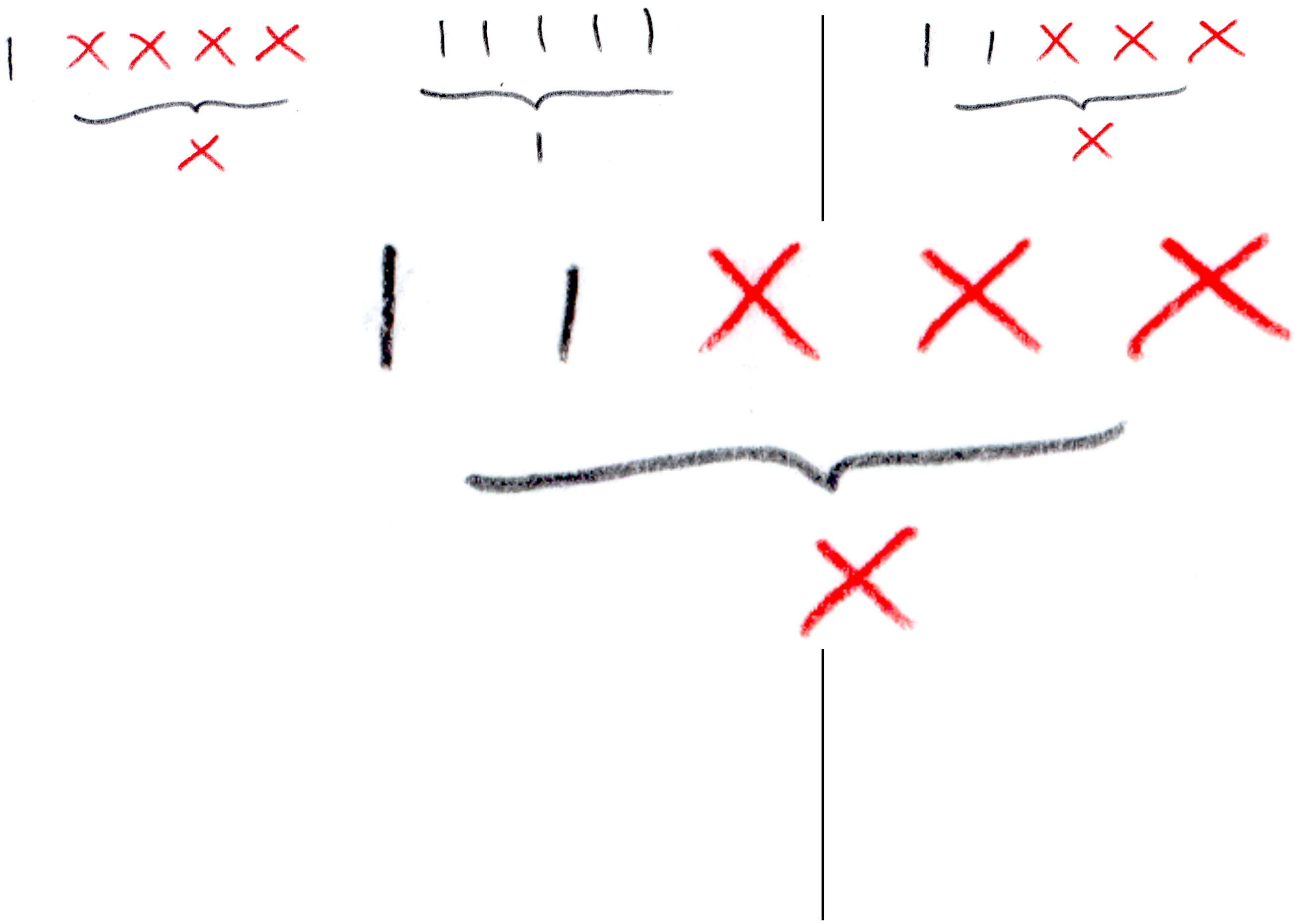
good examples

- **Relative error states** based on ideal recursive decoding
- A **good** block has at most one subblock either in relative error or bad.



good

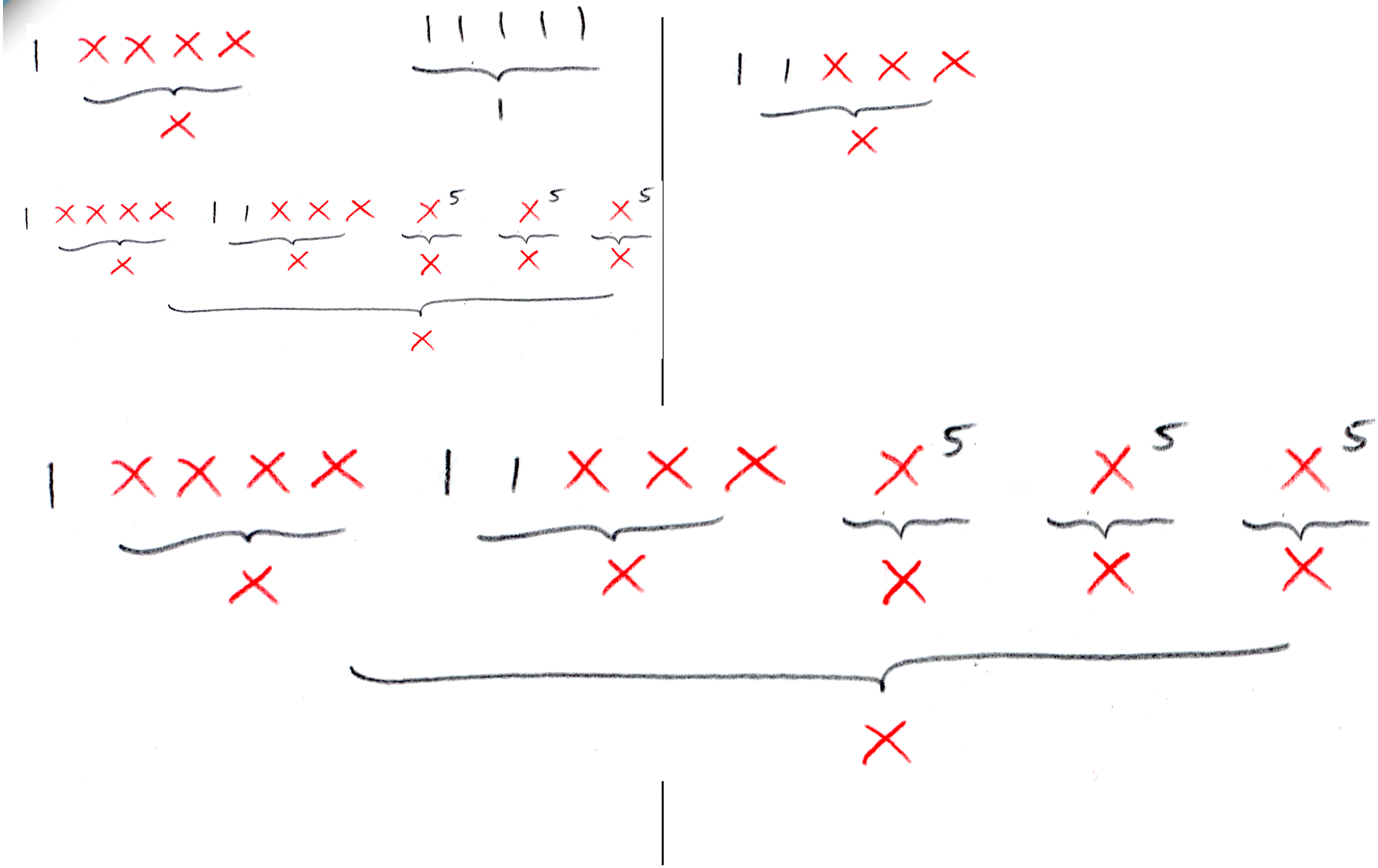
bad



good

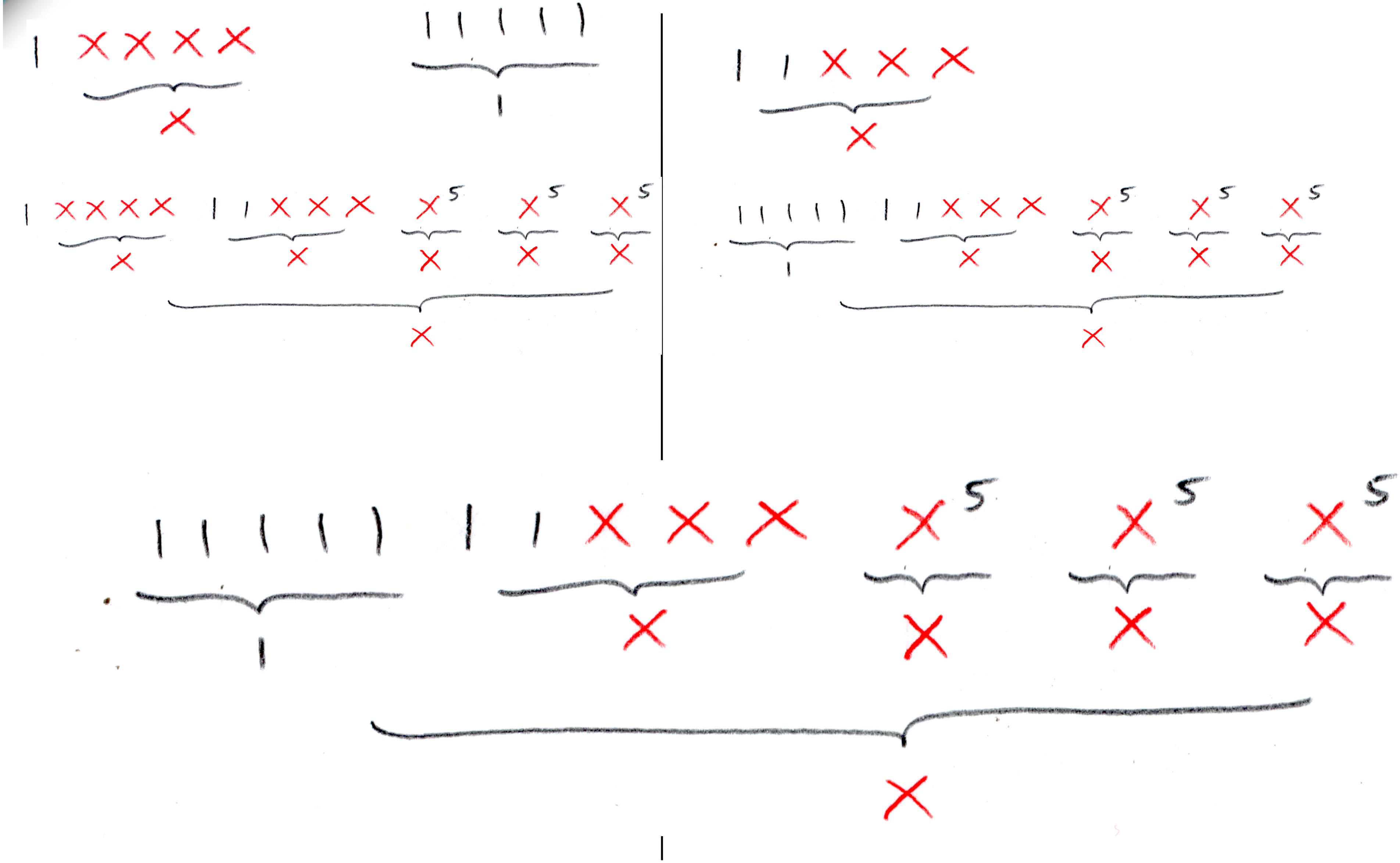
(at most one subblock either in relative error or bad)

bad



good | bad

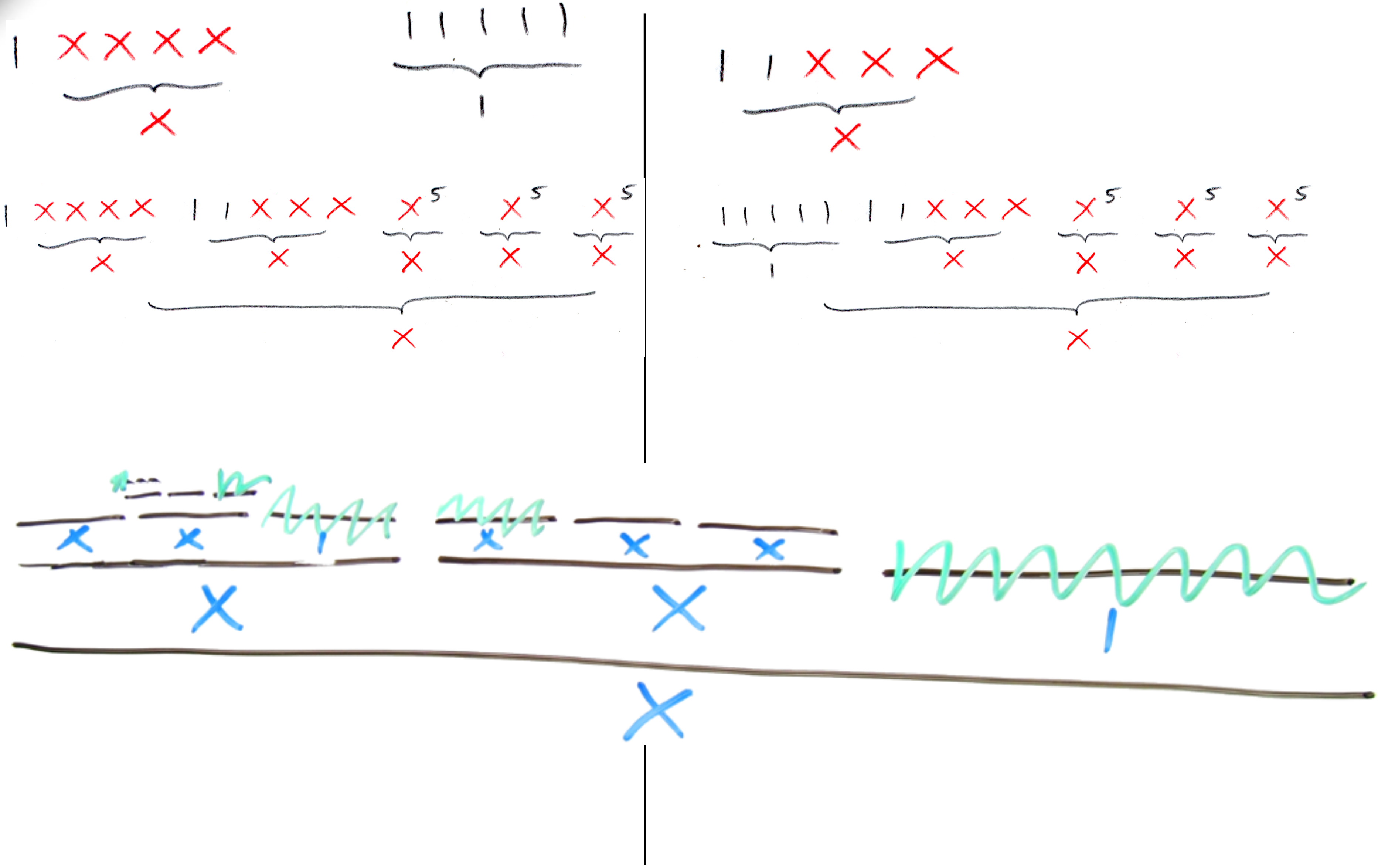
(at most one subblock either in relative error or bad)



good

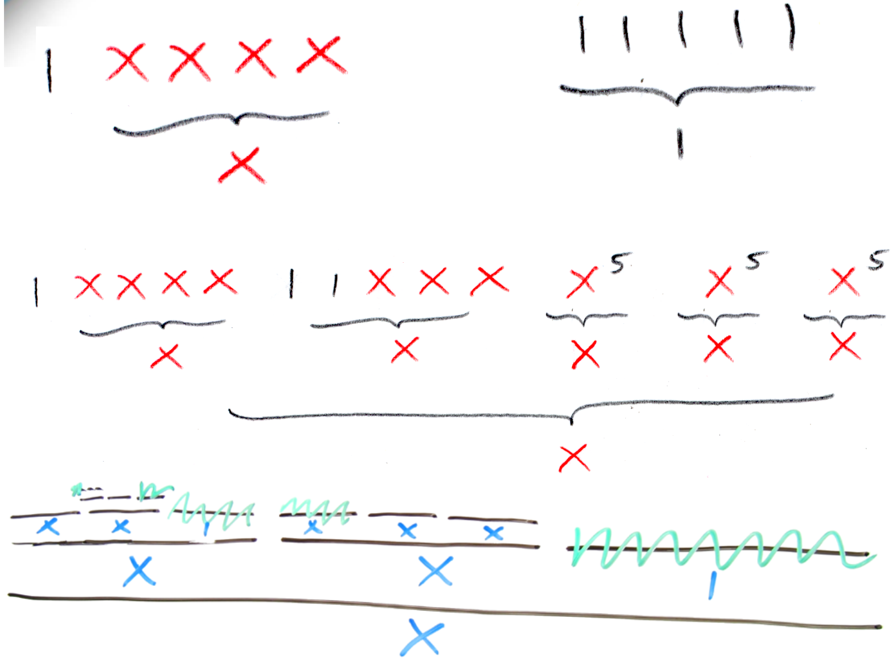
(at most one subblock either in relative error or bad)

bad

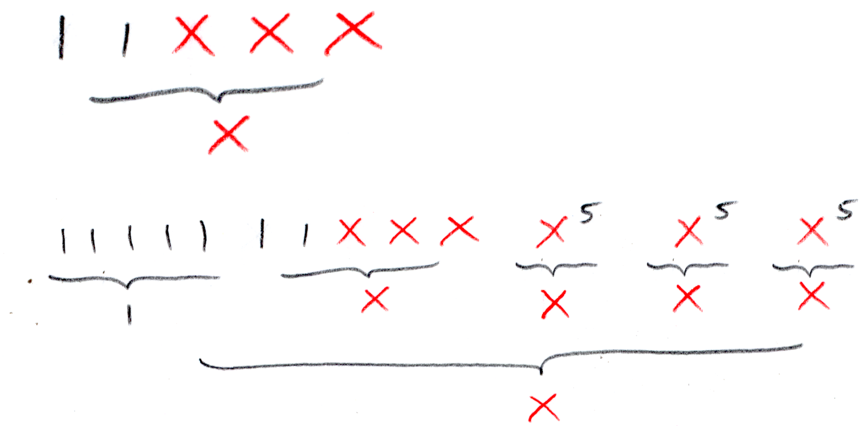


good

(at most one subblock either in relative error or bad)

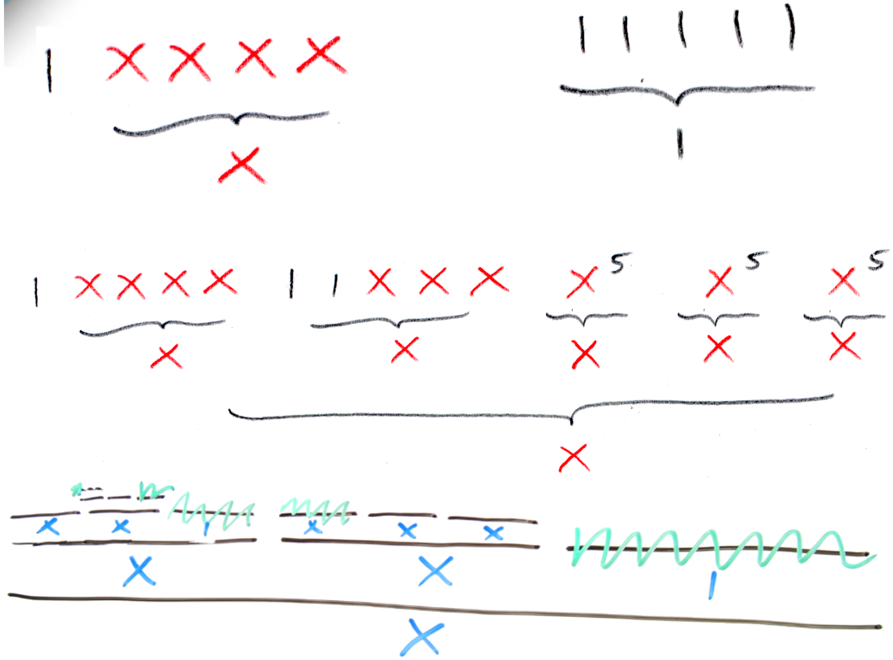


bad

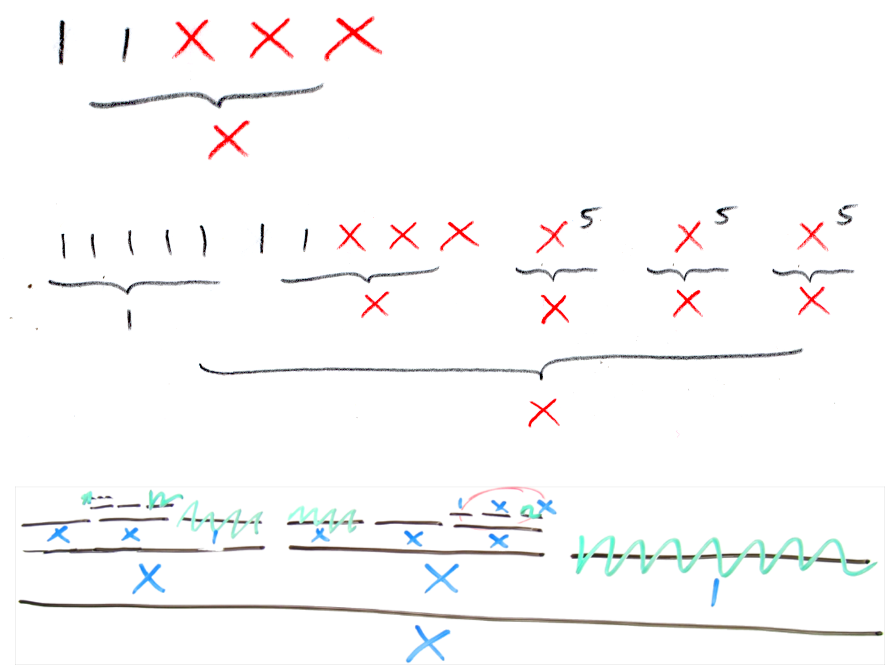


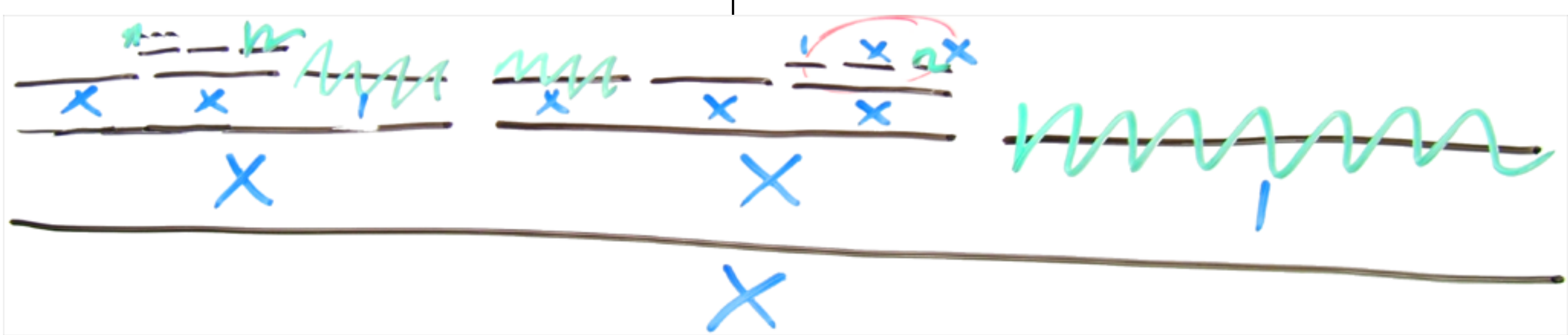
good

(at most one subblock either in relative error or bad)



bad

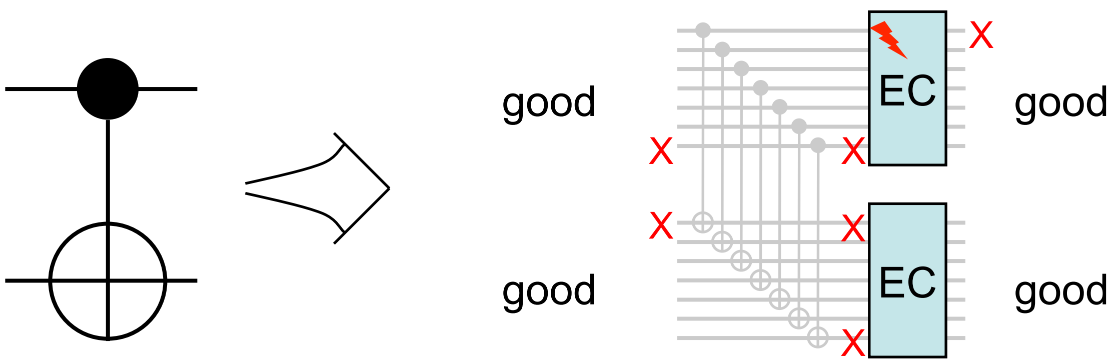




Aharonov/Ben-Or threshold setup

- **Base noise model:** $CNOT_0$ gates fail, giving X errors, independently w/ prob. p.
- **Claim C_k** ($CNOT_k$)**:** On success:
  - If the input blocks are $good_k$, then the output blocks are $good_k$, and a logical CNOT is applied.
  - On arbitrary inputs, the output $blocks_k$ are $good_k$ and a possibly incorrect logical effect is applied.

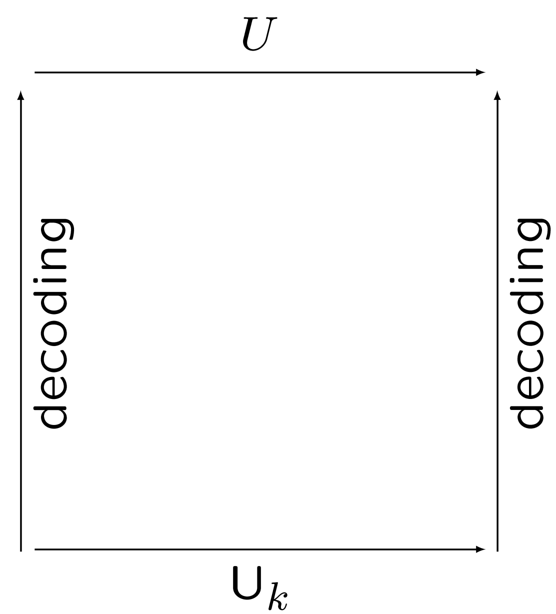
The failure probability is at most C_k ($C_0 = p$).



(one level k-1 error, d $\geq$ 7)

Def: Logical failure

- **Def:** Logical operation U_k on one or more $blocks_k$ has the correct logical effect if the diagram commutes:

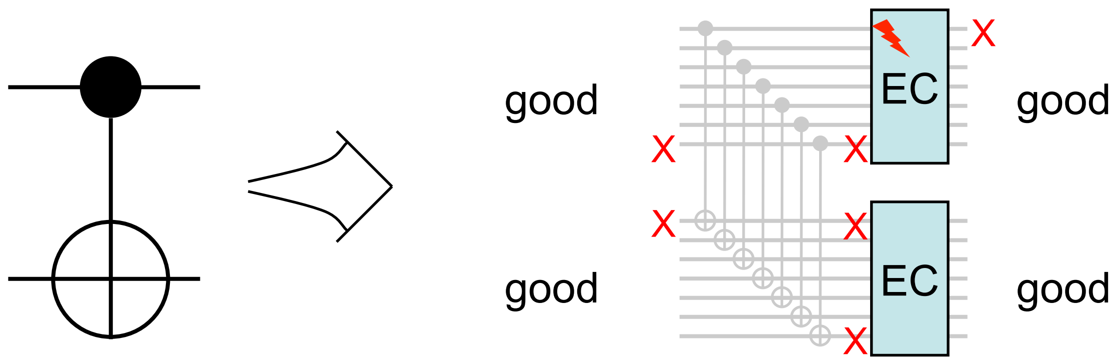


- U_k has a possibly incorrect logical effect if the same diagram commutes but with $P \circ U$ on the top arrow, where P is a Pauli operator or Pauli product on the involved blocks.

Aharonov/Ben-Or threshold setup

- **Claim C_k** (CNOT$_k$)**:** On success:
  - If the input blocks are good$_k$, then the output blocks are good$_k$, and a logical CNOT, the correct logical effect, is applied.
  - On arbitrary inputs, the output blocks$_k$ are good$_k$ and a possibly incorrect logical effect is applied.

The failure probability is at most C_k ($C_0 = p$).



- **Claim B_k** (Correction$_k$)**:** On success:
  - If the input block is good, then the output block is good and no logical effect is applied.
  - On arbitrary input, the output block is good.

The failure probability is at most B_k ($B_0 = 0$).

Aharonov/Ben-Or threshold proof

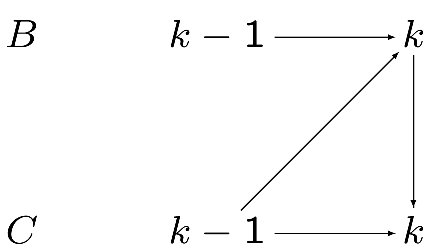
- **Two operations:**

  B. Error correction
  C. (Logical) CNOT gate

- **Two indexed claims:**

  B_k Error correction$_k$ success except w/ prob. $\leq B_k$
  C_k CNOT$_k$ success except w/ prob. $\leq C_k$

- **Proofs by induction:** Implications:



$$B_k = O\left((B_{k-1} + C_{k-1})^2\right)$$

$$C_k = O\left(B_k + C_{k-1}^2\right)$$

- **Base noise model:** CNOT$_0$ gates fail with X errors independently w/ prob. p

  $B_0 = 0 \qquad C_0 = p$

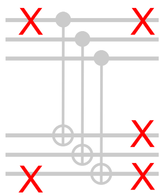
Proof overview

- Def: Error states
- Def: Relative error states
- **Def: 1-good block**
- Aharonov/Ben-Or threshold setup
- Def: Logical failure
- Aharonov/Ben-Or threshold proof
- **Def: “well” block**
- Distance-3 code threshold setup and proof for stabilizer operations

- Extension to **universality** via magic states distillation

Def: well

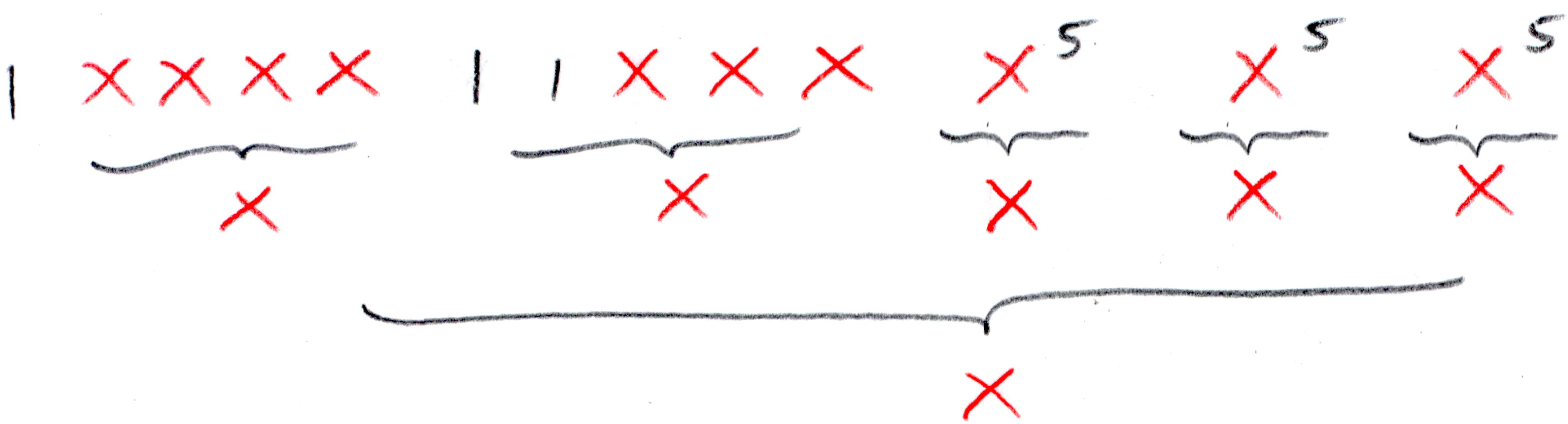
- **Tracking errors**



- **Block error states:** ideal recursive decoding
- **Relative error states**

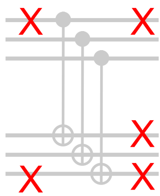
- **Def:** A block$_k$ is relative good$_k$ if it has at most one subblock$_{k-1}$ either in error or not relative good$_{k-1}$ itself. (Every bit [≡ block$_0$] is relative good$_0$.)

- **Def:** A block$_k$ is relative (1-)well$_k$(p$_1$,…,p$_k$) if at most one subblock$_{k-1}$ relative error or not relative well$_{k-1}$(p$_1$,…,p$_{k-1}$) itself. Additionally, the probability of such a subblock, conditioned on the block's state and the state of all bits in other blocks, is $\leq p_k$.
(Every bit [≡ block$_0$] is relative well$_0$.)



Def: well

- **Tracking errors**

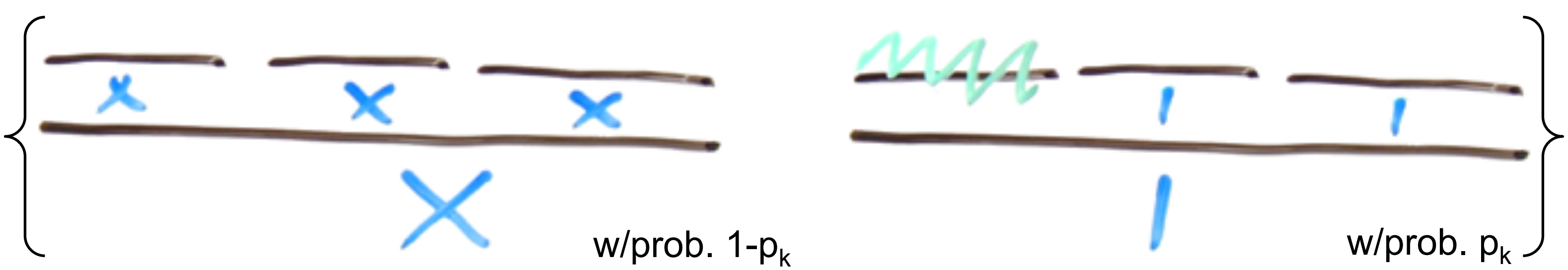


- **Block error states:** ideal recursive decoding

- **Relative error states**

- **Def:** A block_k is relative (1-)$\text{well}_k(p_1,\ldots,p_k)$ if it has at most one subblock_{k-1} either in relative error or not relative $\text{well}_{k-1}(p_1,\ldots,p_{k-1})$ itself.
  Additionally, the probability of such a subblock, conditioned on the block's state and the state of all bits in other blocks, is $\leq p_k$.
  (Every bit [$\equiv \text{block}_0$] is relative well_0.)

- **Note:** *Conditioned* on block's state, e.g.,

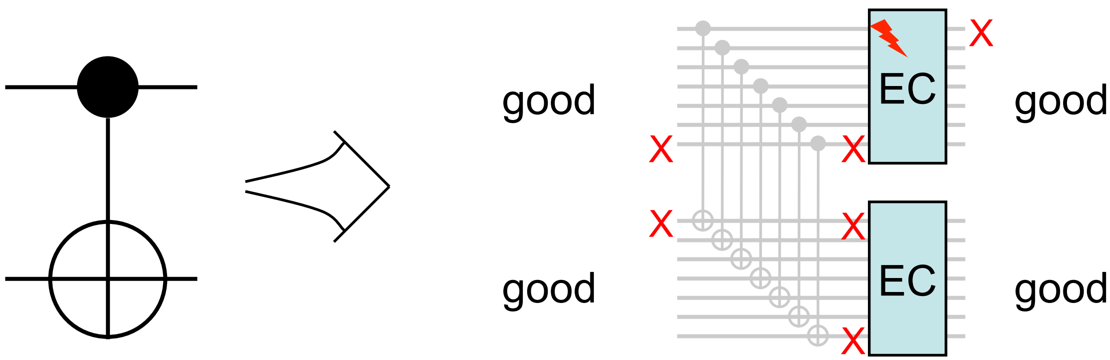


is *not* 1-well.

Aharonov/Ben-Or threshold setup

- **Claim C_k** ($CNOT_k$)**:** On success:
  - If the input blocks are $good_k$, then the output blocks are $good_k$, and a logical CNOT, the correct logical effect, is applied.
  - On arbitrary inputs, the output $blocks_k$ are $good_k$ and a possibly incorrect logical effect is applied.

The failure probability is at most C_k ($C_0 = p$).



- **Claim B_k** ($Correction_k$)**:** On success:
  - If the input blocks is good, then the output block is good and no logical effect is applied.
  - On arbitrary input, the output block is good.

The failure probability is at most B_k ($B_0 = 0$).

Aharonov/Ben-Or threshold setup

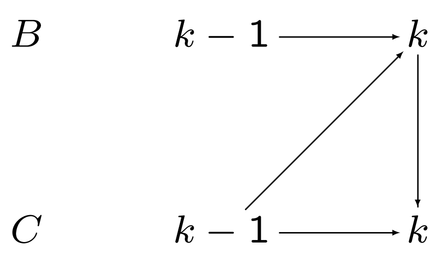
- **Two operations:**

  B. Error correction
  C. (Logical) CNOT gate

- **Two indexed claims:**

B_k	Error correction$_k$	success except w/ prob. $\leq B_k$
C_k	CNOT$_k$	success except w/ prob. $\leq C_k$

- **Proofs by induction:** Implications:



$$B_k = O\left((B_{k-1} + C_{k-1})^2\right)$$

$$C_k = O\left(B_k + C_{k-1}^2\right)$$

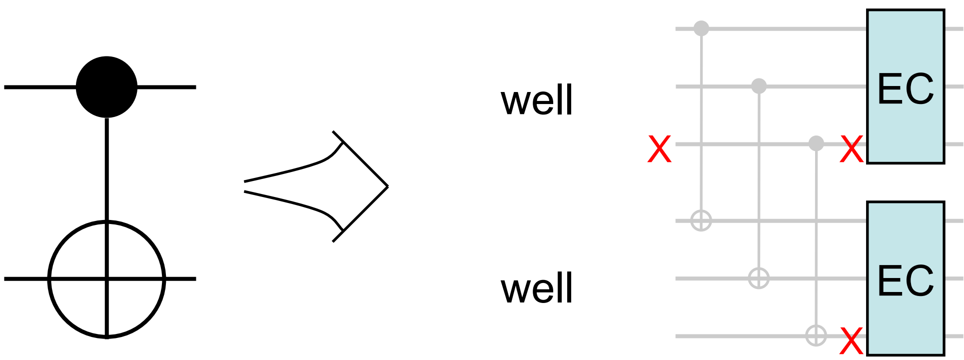
- **Base noise model:** CNOT$_0$ gates fail with X errors independently w/ prob. p

  $B_0 = 0 \qquad C_0 = p$

Dist 3 code setup

- **Claim C_k** (CNOT$_k$)**:** On success:
  - If the input blocks are well$_k$(b_1,…,b_k), then the output blocks are well$_k$(b_1, …,b_k), and a logical CNOT is applied.
  - On arbitrary inputs, the output blocks$_k$ are well$_k$(b_1,…,b_k) and a possibly incorrect logical effect is applied.

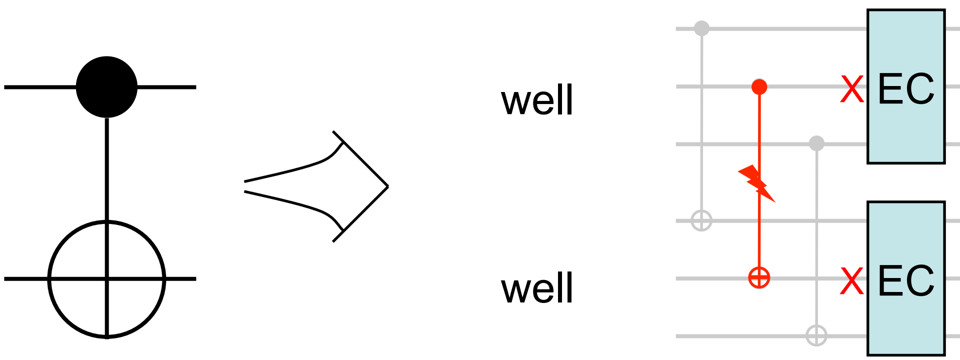
The failure probability is at most C_k (C_0 = p).



Dist 3 code setup

- **Claim C_k** (CNOT$_k$)**:** On success:
  - If the input blocks are well$_k$(b_1,…,b_k), then the output blocks are well$_k$(b_1, …,b_k), and a logical CNOT is applied.
  - On arbitrary inputs, the output blocks$_k$ are well$_k$(b_1,…,b_k) and a possibly incorrect logical effect is applied.

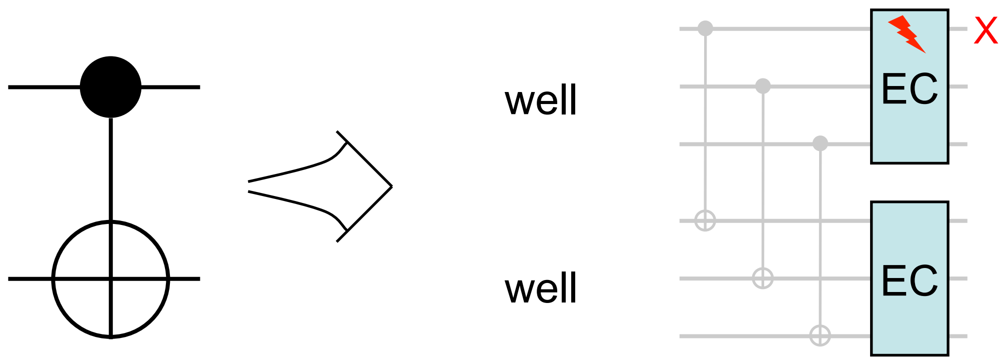
The failure probability is at most C_k (C_0 = p).



Dist 3 code setup

- **Claim C_k** (CNOT$_k$)**:** On success:
  - If the input blocks are well$_k$(b_1,…,b_k), then the output blocks are well$_k$(b_1, …,b_k), and a logical CNOT is applied.
  - On arbitrary inputs, the output blocks$_k$ are well$_k$(b_1,…,b_k) and a possibly incorrect logical effect is applied.

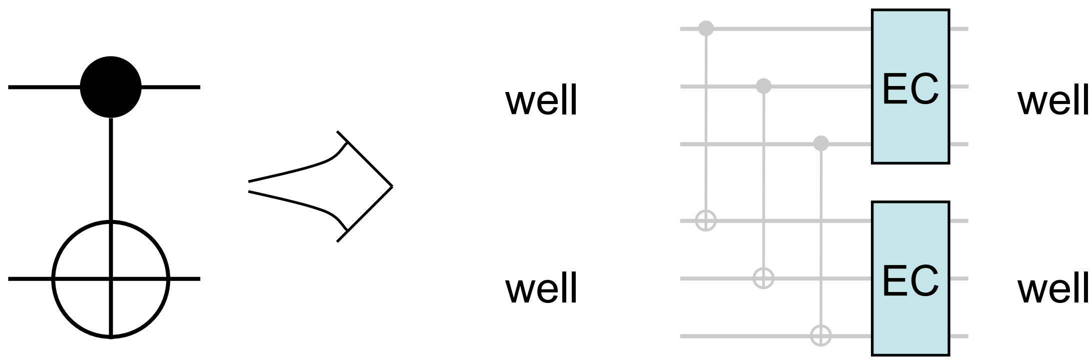
The failure probability is at most C_k (C_0 = p).



Dist 3 code setup

- **Claim C_k** (CNOT$_k$)**:** On success:
  - If the input blocks are well$_k$($b_1,\ldots,b_k$), then the output blocks are well$_k$(b_1, $\ldots,b_k$), and a logical CNOT is applied.
  - On arbitrary inputs, the output blocks$_k$ are well$_k$($b_1,\ldots,b_k$) and a possibly incorrect logical effect is applied.

  The failure probability is at most C_k (C_0 = p).



- **Claim B_k** (Correction$_k$)**:** On success:
  - If the input block is well$_k$($b_1,\ldots,b_k$), then the output block is well$_k$($b_1,\ldots,b_k$) and no logical effect is applied.
  - On arbitrary input, the output block is well$_k$($b_1,\ldots,b_k$).

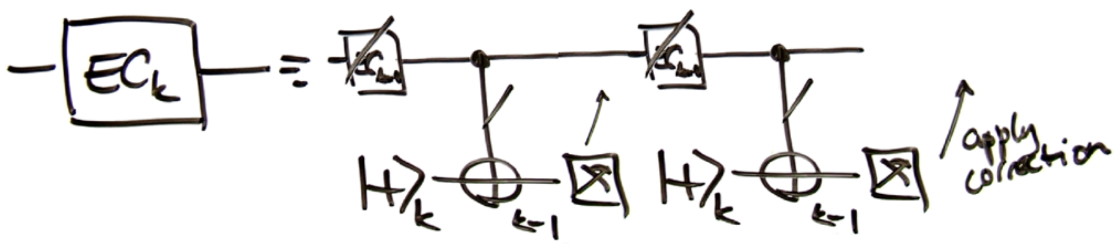
  The failure probability is at most B_k (B_0 = 0).

Additionally, if all but one of the input subblocks$_{k-1}$ are well$_{k-1}$($b_1,\ldots,b_{k-1}$), then with probability at least 1-B_k' there is no logical effect and the output is well$_k$($b_1,\ldots,b_k$).

Dist 3 threshold proof

- **Claim B_k** (Correction$_k$)**:** With probability at least $1 - B_k$ the output block$_k$ is well$_k$($b_1,\ldots,b_k$) and, if the input is well$_k$($b_1,\ldots,b_k$), there is no logical effect.
  Additionally, if all but one of the input subblocks$_{k-1}$ are well$_{k-1}$($b_1,\ldots,b_{k-1}$), then with probability at least $1 - B'_k$ there is no logical effect and the output is well$_k$($b_1,\ldots,b_k$).

Correction$_k$ proof:



Declare success if Ancillas$_k$ both succeed & at most one level k-1 failure.

$$B_k \equiv 2A_k + (b_k + 2a_k + 2nB_{k-1} + 2nC_{k-1})^2$$

$$B'_k \equiv 2A_k + 2a_k + 2nB_{k-1} + 2nC_{k-1}$$

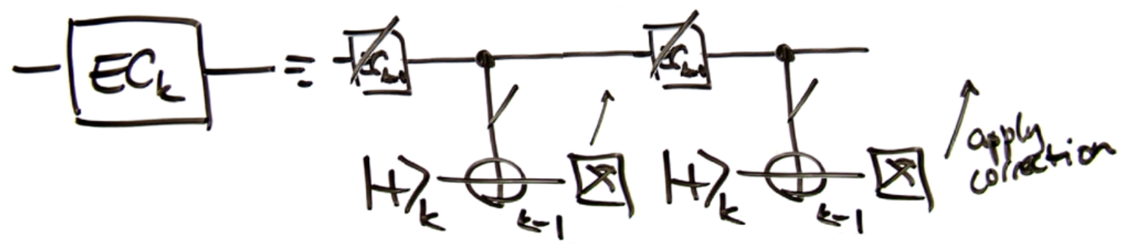
$$b_k \equiv (2a_k + 2nB_{k-1} + 2nC_{k-1}) / (1 - B'_k)$$

Proof is mostly similar to Aharonov/Ben-Or proof, with one exception…

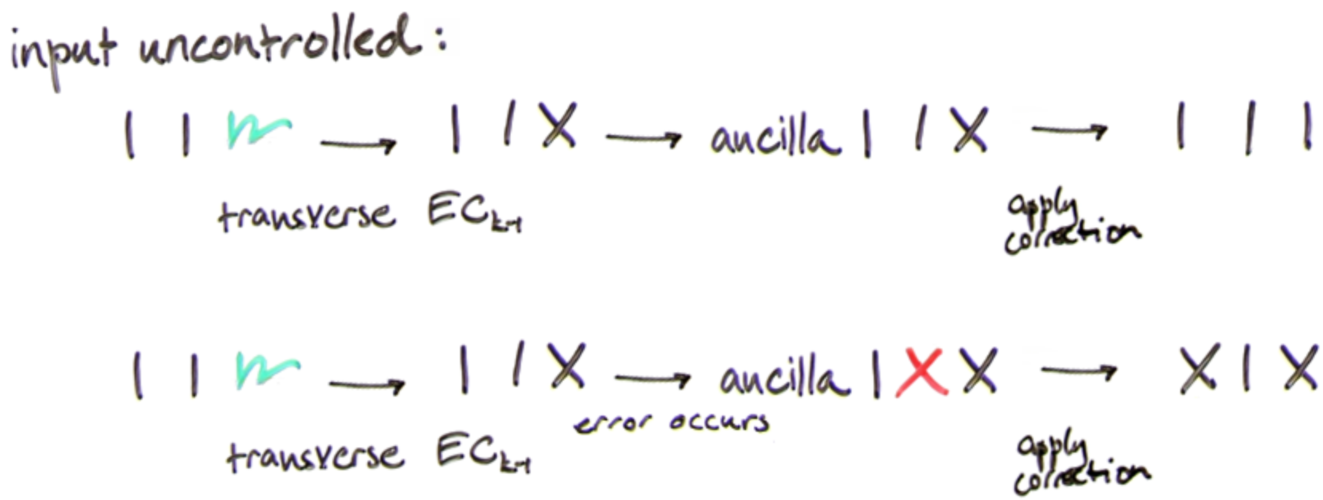
Dist 3 threshold proof

- **Claim B_k** (Correction$_k$)**:** With probability at least $1 - B_k$ the output block$_k$ is well$_k$($b_1,\ldots,b_k$) and, if the input is well$_k$($b_1,\ldots,b_k$), there is no logical effect.
  Additionally, if all but one of the input subblocks$_{k-1}$ are well$_{k-1}$($b_1,\ldots,b_{k-1}$), then with probability at least $1 - B'_k$ there is no logical effect and the output is well$_k$($b_1,\ldots,b_k$).

Correction$_k$ proof:



Proof is mostly similar to Aharonov/Ben-Or proof, with one exception…
Why are two syndrome extractions necessary?



□

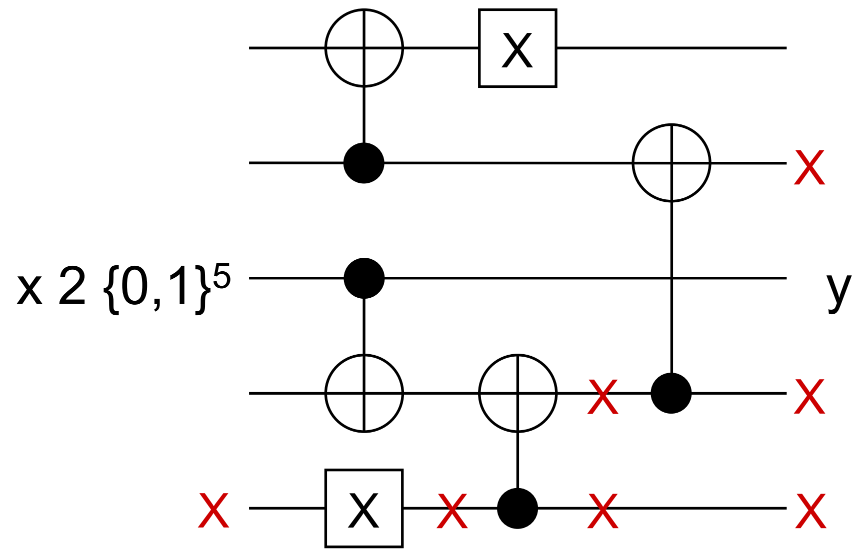
Stabilizer operations

Def: Stabilizer operations are

- Clifford group unitaries

$$\left\langle H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, K = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \text{CNOT} \right\rangle$$

- Preparation of $|0\rangle, |1\rangle$
- Measurement in $|0\rangle, |1\rangle$



$$y = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} + x \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Gottesman-Knill Theorem: Stabilizer operations are efficiently classically simulable.

Universality from stabilizer operations & repeated preparation of $\cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle$

Def: Stabilizer operations are

- Clifford group unitaries

$$\left\langle H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, K = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \text{CNOT} \right\rangle$$

- Preparation of $|0\rangle$, $|1\rangle$
- Measurement in $|0\rangle$, $|1\rangle$

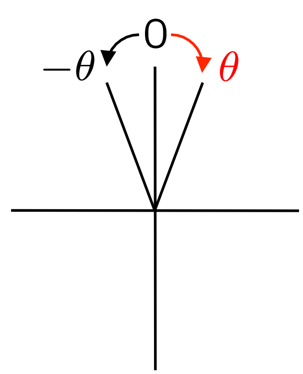
Theorem: [Shi'02] CNOT + any single-qubit gate not in Clifford group gives quantum universality.

Fact 1: Stab ops + prepare $|\psi\rangle = \begin{matrix} \cos\frac{\theta}{2}|0\rangle \\ + \sin\frac{\theta}{2}|1\rangle \end{matrix}$! universality.

(if $\theta \neq$ k π/2)

Proof: How to apply $U = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$

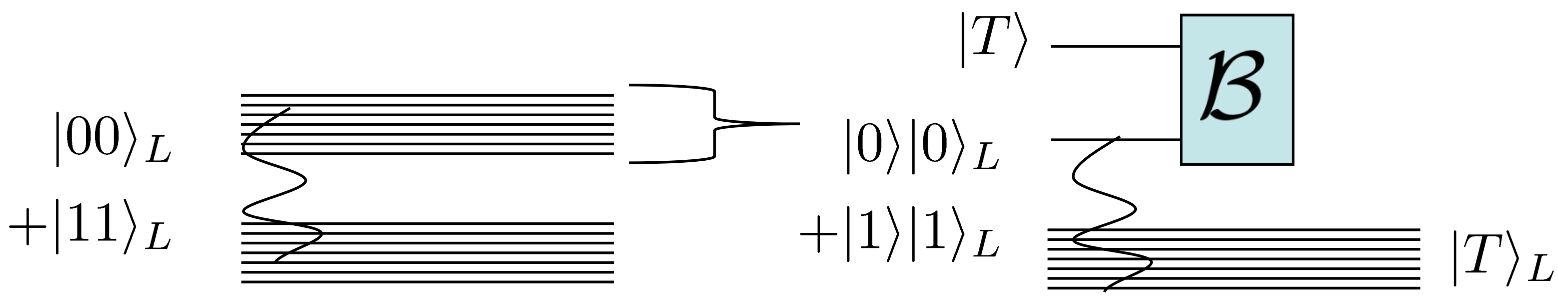
$$(\alpha|0\rangle + \beta|1\rangle)\,(|0\rangle + e^{i\theta}|1\rangle)$$
$$= \alpha|00\rangle + \beta e^{i\theta}|11\rangle$$
$$+\alpha e^{i\theta}|01\rangle + \beta|10\rangle$$



Application

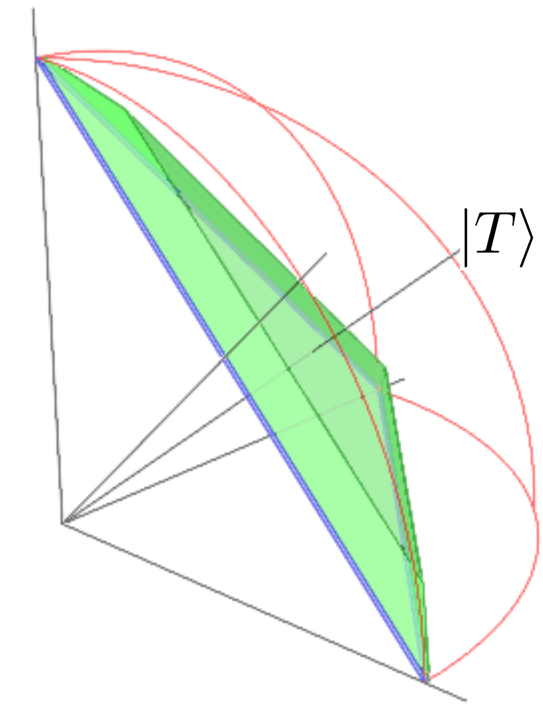
Theorem: [Shi'02] CNOT + any single-qubit gate not in Clifford group gives quantum universality.

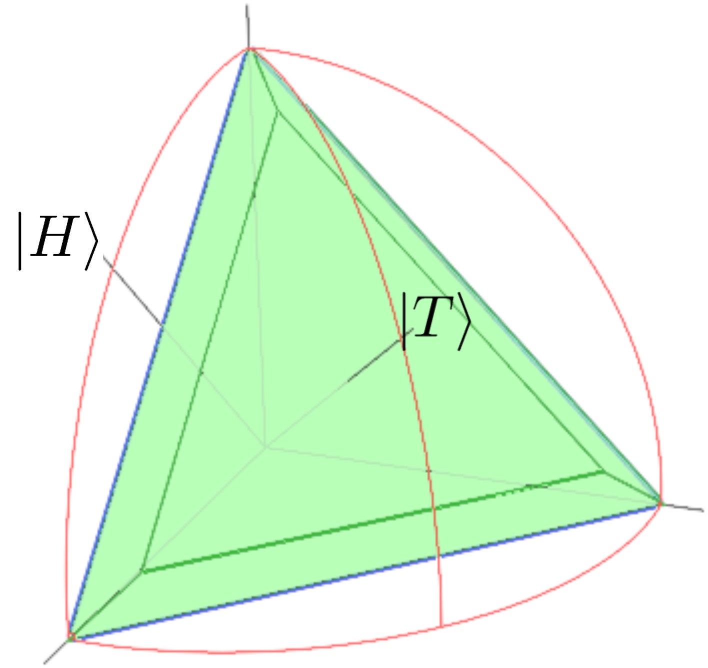
Fact 1: Stab ops + prepare $\cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle$! universality.



Stabilizer op. fault-tolerance $\rightsquigarrow$ Universal fault-tolerance

Q: Do $\left\{\begin{array}{l}\text{stabilizer operations,}\\ \text{prepare } \rho\end{array}\right\}$ form a universal set?



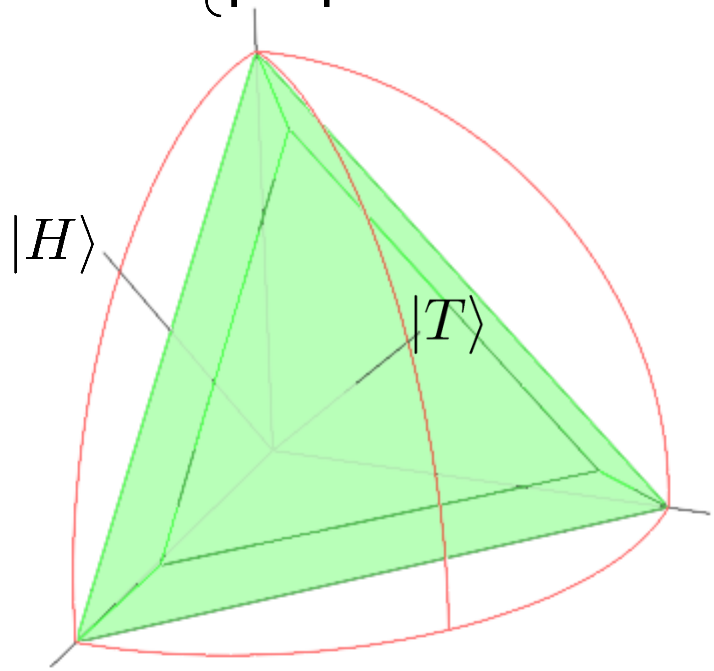


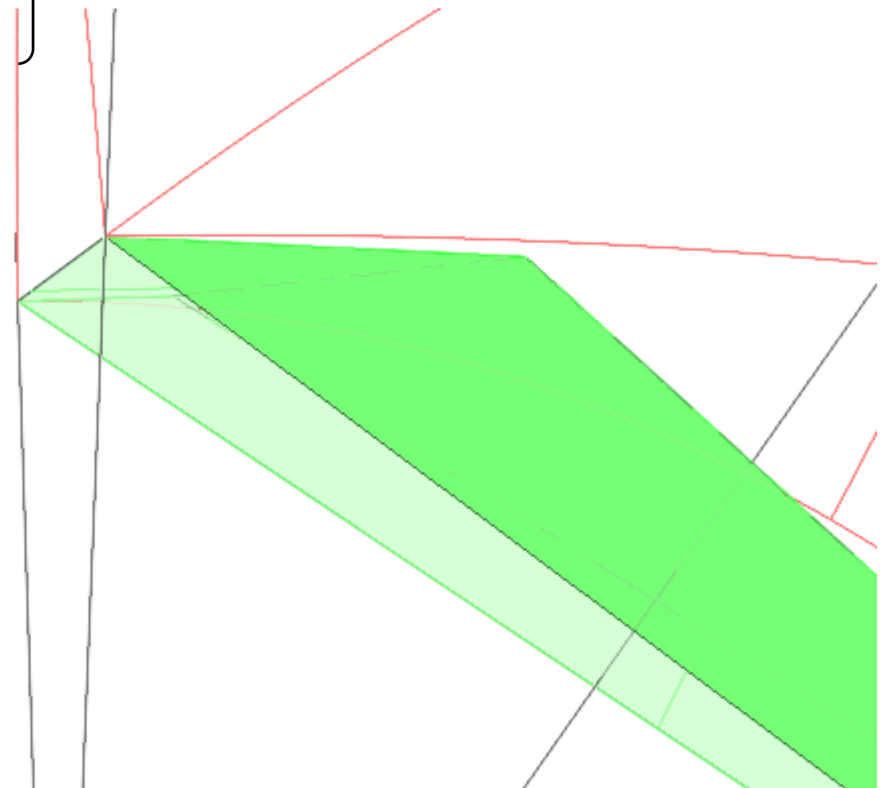
[Bravyi-Kitaev '04, Knill '04] Yes for $|H\rangle$ w/ <14.2% error

[Bravyi-Kitaev '04] Yes for $|T\rangle$ w/ <17.3% error

$\rotatebox{90}{=}\ \frac{1}{2}(1-\sqrt{\frac{3}{7}})$

Q: Do $\left\{ \begin{array}{l} \text{stabilizer operations,} \\ \text{prepare } \rho \end{array} \right\}$ form a universal set?



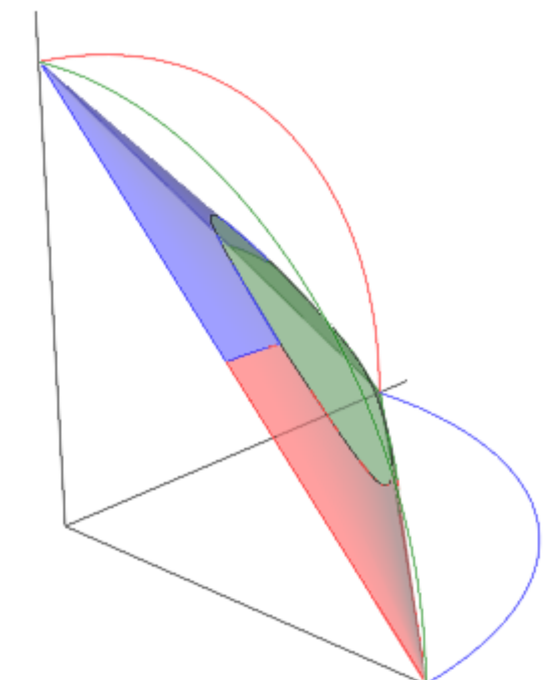


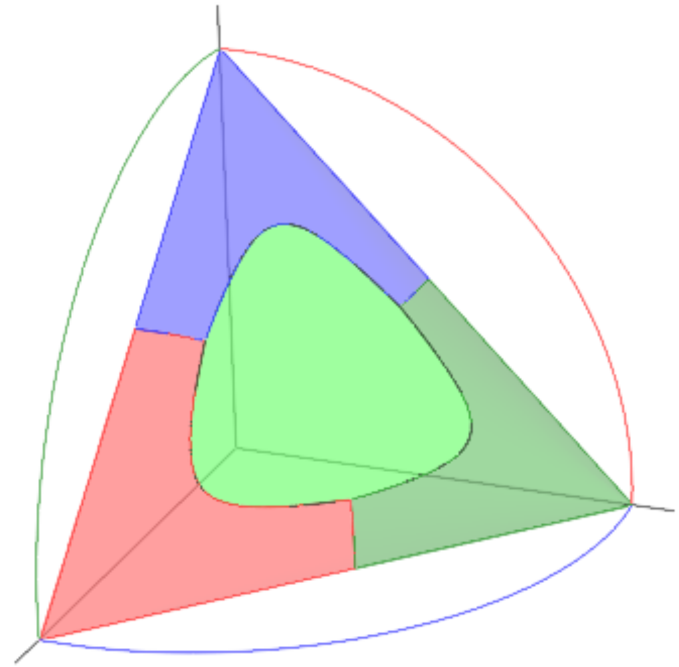
[Bravyi-Kitaev ‘04, Knill ‘04] Yes for $|H\rangle$ w/ <14.2% error

[Bravyi-Kitaev ‘04] Yes for $|T\rangle$ w/ <17.3% error $= \frac{1}{2}(1-\sqrt{\frac{3}{7}})$

Theorem: [R ‘04] Yes for $|H\rangle$ w/ <14.6% error $= \frac{1}{2}(1-\frac{1}{\sqrt{2}})$

Q: Do $\left\{\begin{array}{l}\text{stabilizer operations,}\\ \text{prepare } \rho\end{array}\right\}$ form a universal set?





[Bravyi-Kitaev ‘04, Knill ‘04] Yes for $|H\rangle$ w/ <14.2% error

[Bravyi-Kitaev ‘04] Yes for $|T\rangle$ w/ <17.3% error $= \frac{1}{2}(1-\sqrt{\frac{3}{7}})$

Theorem: [R ‘04] Yes for $|H\rangle$ w/ <14.6% error $= \frac{1}{2}(1-\frac{1}{\sqrt{2}})$

Conclusion

Proof overview:

- Def: Error states
- Def: Relative error states
- **Def: 1-good block**
- Aharonov/Ben-Or threshold setup
- Def: Logical failure
- Aharonov/Ben-Or threshold proof
- **Def: "well" block**
- Distance 3 code threshold setup and proof for stabilizer operations
- Extension to **universality** via magic states distillation

Open questions:

- Optimize proof for improving provable threshold
- Prove threshold for postselection-based fault-tolerance scheme
  - also for specialized error models
- Prove upper/lower bounds on magic states distillation, improve efficiency…