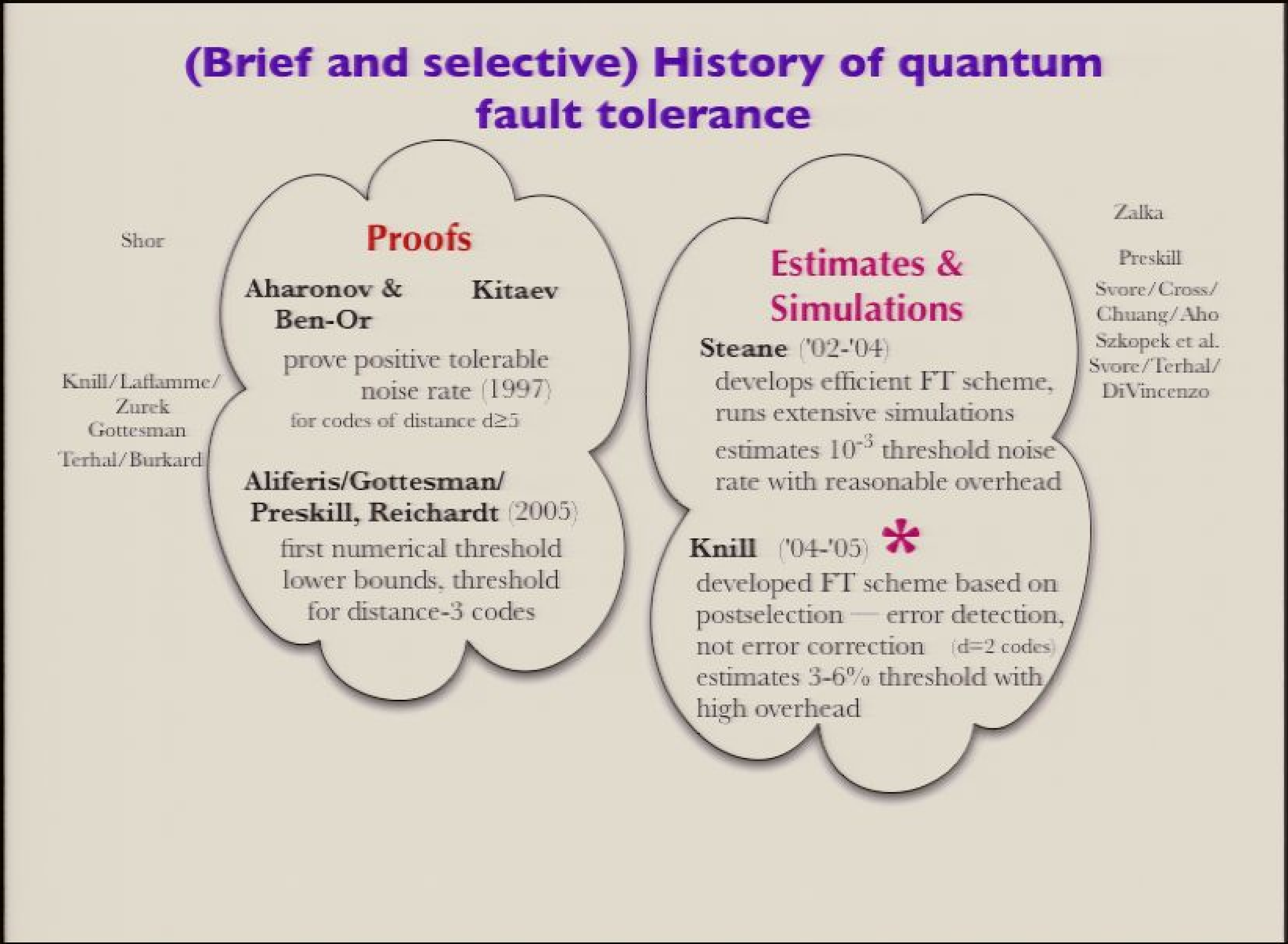
Title: The Postselection Threshold Proof

Date: Jun 15, 2007 03:30 PM

URL: http://pirsa.org/07060056

Abstract:



(Brief and selective) History of quantum fault tolerance

Shor

Knill/Laflamme/
Zurek
Gottesman
Terhal/Burkard

Proofs

Aharonov & Ben-Or **Kitaev**

prove positive tolerable noise rate (1997)

for codes of distance d≥5

Aliferis/Gottesman/ Preskill, Reichardt (2005)

first numerical threshold lower bounds, threshold for distance-3 codes

Estimates & Simulations

Steane ('02-'04)

develops efficient FT scheme, runs extensive simulations

estimates 10^{-3} threshold noise rate with reasonable overhead

Knill ('04-'05) *

developed FT scheme based on postselection — error detection, not error correction (d=2 codes)

estimates 3-6% threshold with high overhead

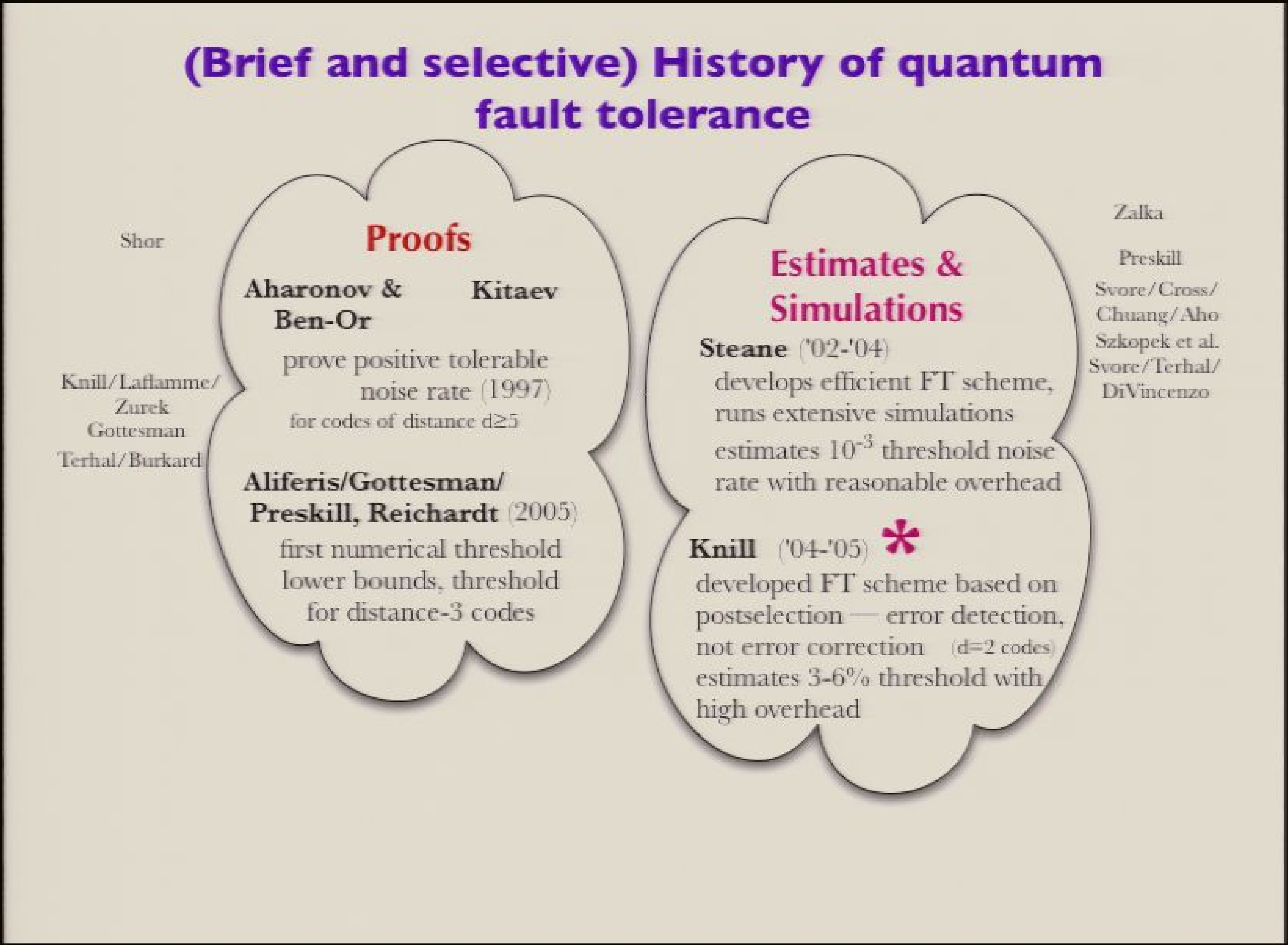
Zalka

Preskill

Svore/Cross/
Chuang/Aho

Szkopek et al.

Svore/Terhal/
DiVincenzo



(Brief and selective) History of quantum fault tolerance

Shor

Knill/Laflamme/
Zurek
Gottesman
Terhal/Burkard

Proofs

Aharonov & Ben-Or **Kitaev**

prove positive tolerable noise rate (1997)

for codes of distance d≥5

Aliferis/Gottesman/ Preskill, Reichardt (2005)

first numerical threshold lower bounds, threshold for distance-3 codes

Estimates & Simulations

Steane ('02-'04)

develops efficient FT scheme, runs extensive simulations

estimates 10^{-3} threshold noise rate with reasonable overhead

Knill ('04-'05) *

developed FT scheme based on postselection — error detection, not error correction (d=2 codes)

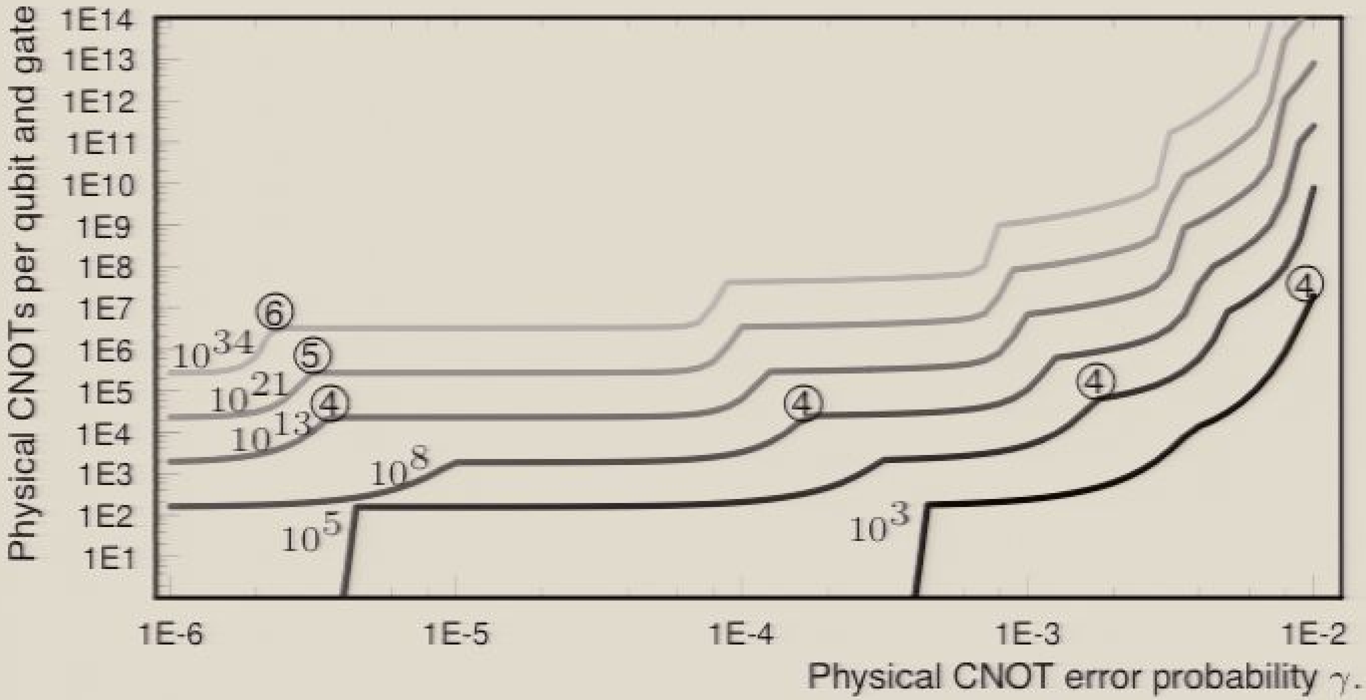
estimates 3-6% threshold with high overhead

Zalka

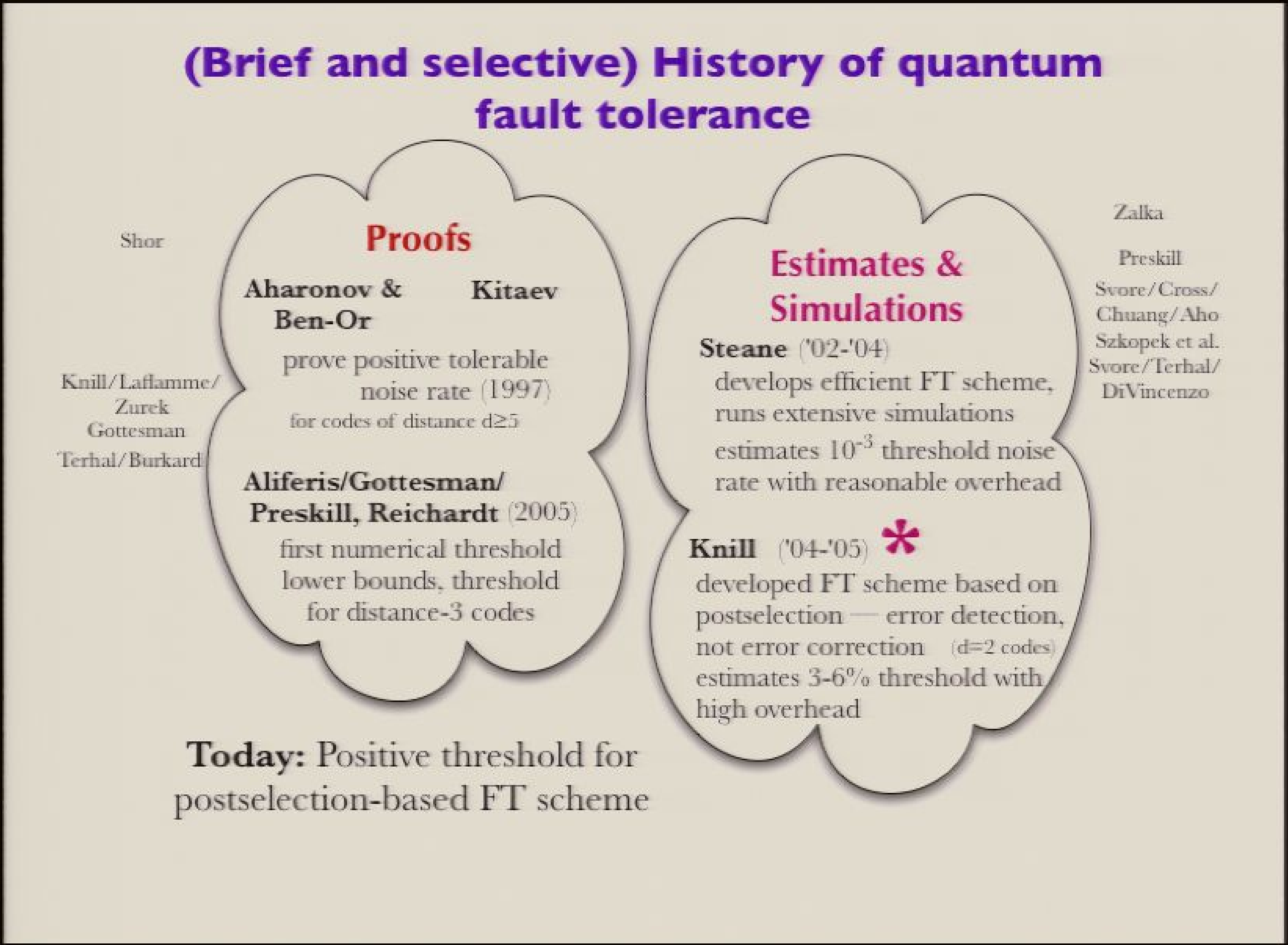
Preskill

Svore/Cross/
Chuang/Aho
Szkopek et al.
Svore/Terhal/
DiVincenzo

Error-detection-based threshold scheme resource requirements



[Knill]



(Brief and selective) History of quantum fault tolerance

Shor

Knill/Laflamme/
Zurek
Gottesman
Terhal/Burkard

Proofs

Aharonov & Ben-Or **Kitaev**

prove positive tolerable noise rate (1997)

for codes of distance $d \geq 5$

Aliferis/Gottesman/ Preskill, Reichardt (2005)

first numerical threshold lower bounds, threshold for distance-3 codes

Estimates & Simulations

Steane ('02-'04)

develops efficient FT scheme, runs extensive simulations

estimates 10^{-3} threshold noise rate with reasonable overhead

Knill ('04-'05) *

developed FT scheme based on postselection — error detection, not error correction (d=2 codes)

estimates 3-6% threshold with high overhead

Zalka

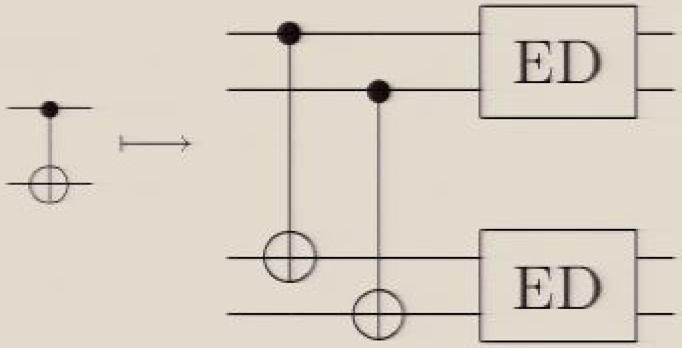
Preskill

Svore/Cross/
Chuang/Aho
Szkopek et al.
Svore/Terhal/
DiVincenzo

Today: Positive threshold for postselection-based FT scheme

Results

- *Existence* of tolerable noise rates for many fault-tolerance schemes, including:
    - Schemes based on error-*detecting* codes, not just ECCs (Knill-type)
    - Fibonnacci-type thresholds
- Tolerable threshold *lower bounds**
    - 0.1% simultaneous depolarization noise†
    - 1.1%, if error model known *exactly*



* Subject to minor numerical caveats

† Versus .02% best lower bound for error-correction-based FT scheme [Aliferis, Cross 2006]

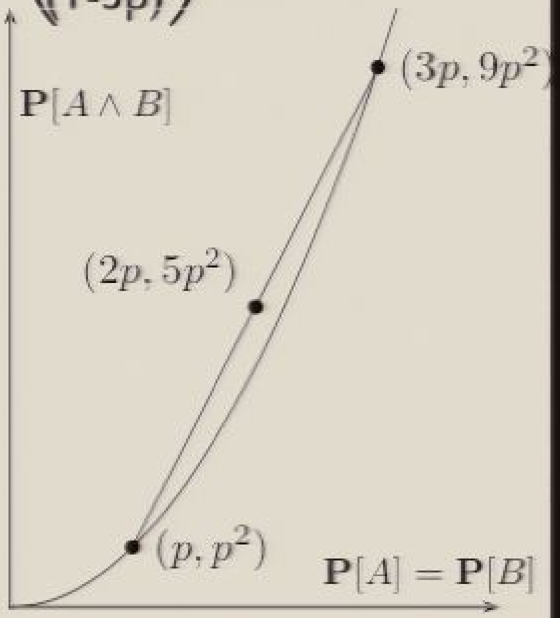
Technique: Mixing of distributions

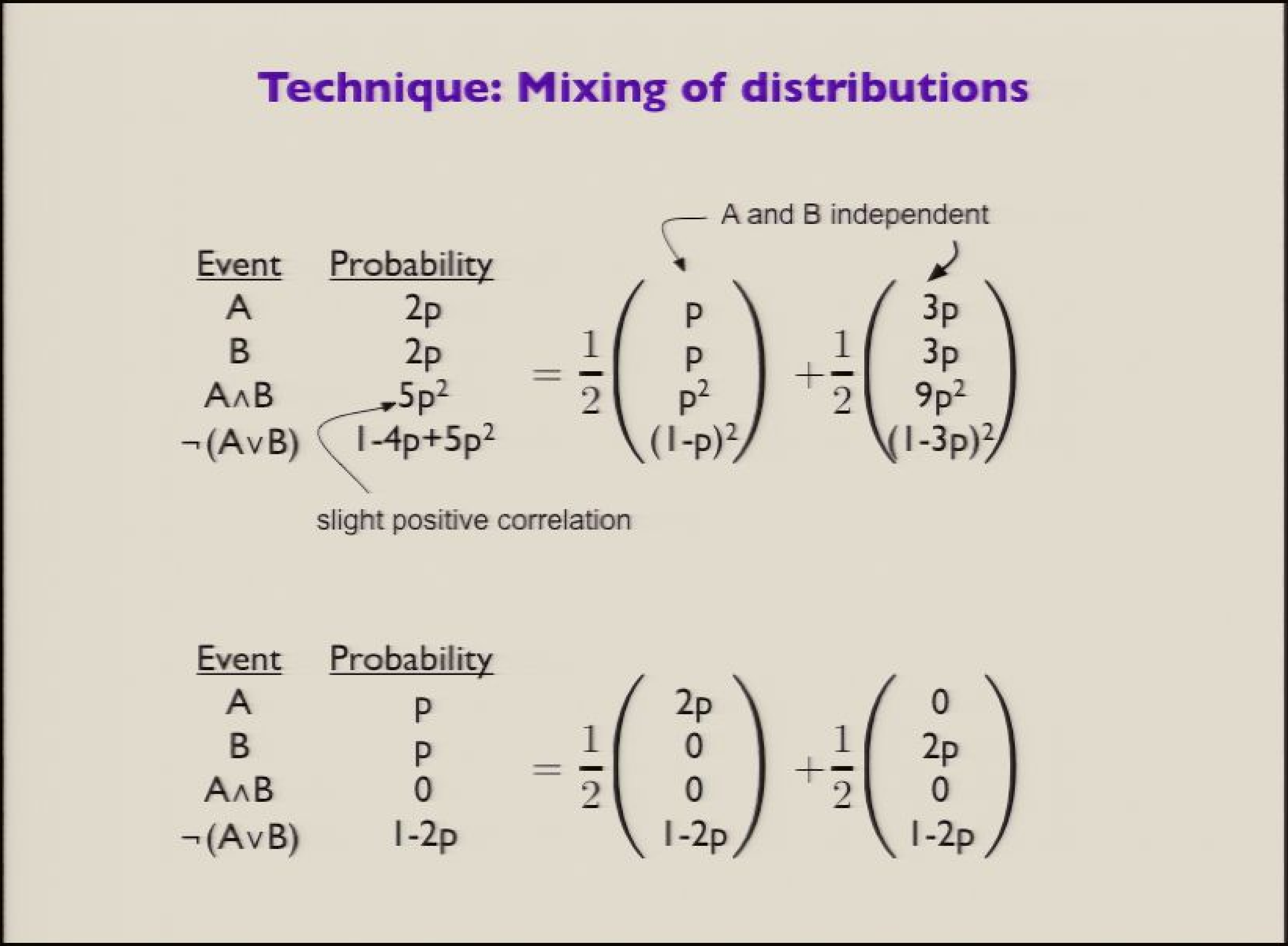
A and B independent

Event	Probability
A	2p
B	2p
A∧B	5p²
¬(A∨B)	1-4p+5p²

$$= \frac{1}{2}\begin{pmatrix} p \\ p \\ p^2 \\ (1-p)^2 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 3p \\ 3p \\ 9p^2 \\ (1-3p)^2 \end{pmatrix}$$

slight positive correlation





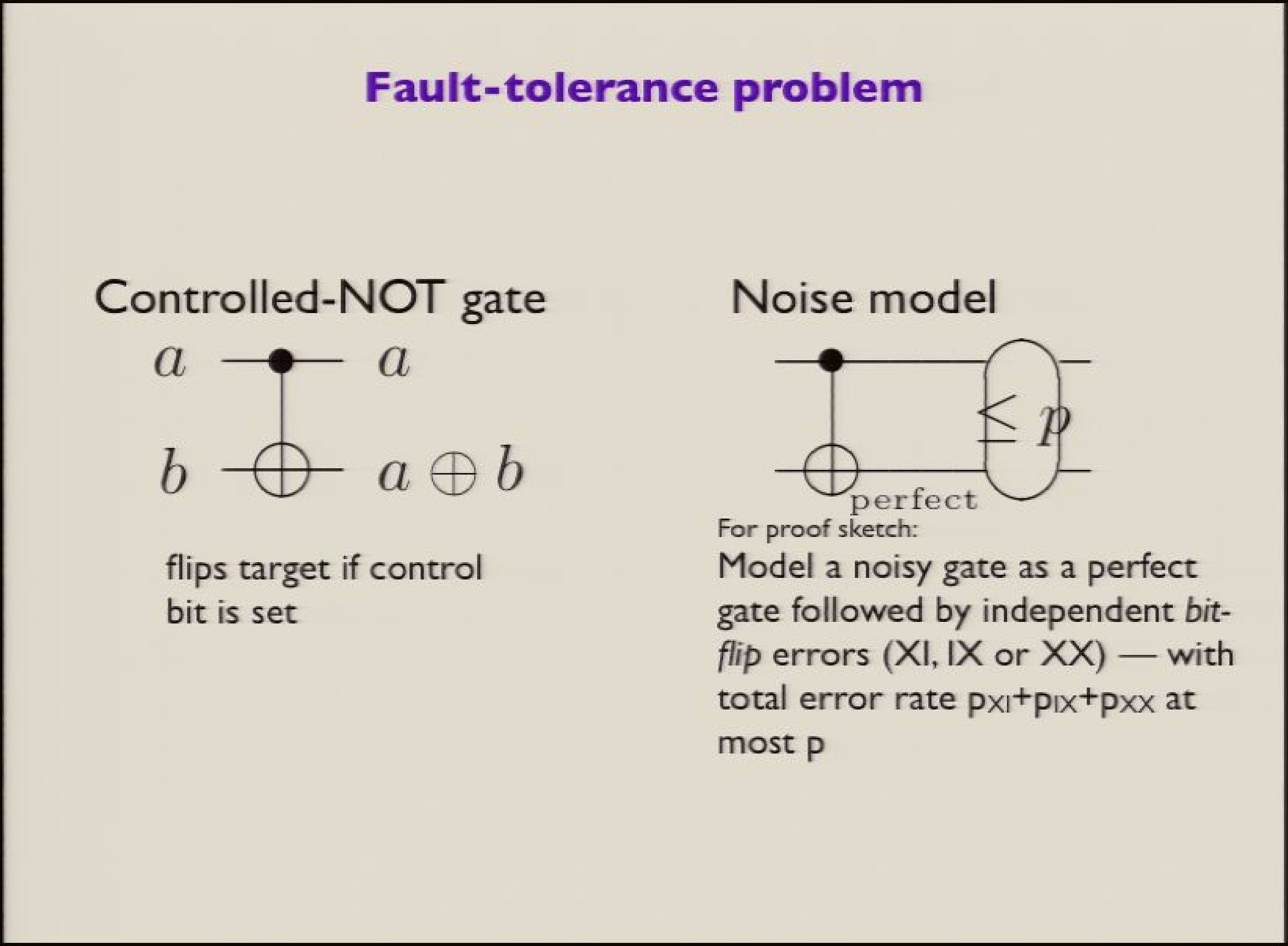
Technique: Mixing of distributions

A and B independent

$$\begin{array}{cc} \underline{\text{Event}} & \underline{\text{Probability}} \\ A & 2p \\ B & 2p \\ A\wedge B & 5p^2 \\ \neg(A\vee B) & 1\text{-}4p+5p^2 \end{array} = \frac{1}{2}\begin{pmatrix} p \\ p \\ p^2 \\ (1\text{-}p)^2 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 3p \\ 3p \\ 9p^2 \\ (1\text{-}3p)^2 \end{pmatrix}$$

slight positive correlation

$$\begin{array}{cc} \underline{\text{Event}} & \underline{\text{Probability}} \\ A & p \\ B & p \\ A\wedge B & 0 \\ \neg(A\vee B) & 1\text{-}2p \end{array} = \frac{1}{2}\begin{pmatrix} 2p \\ 0 \\ 0 \\ 1\text{-}2p \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 \\ 2p \\ 0 \\ 1\text{-}2p \end{pmatrix}$$



Fault-tolerance problem

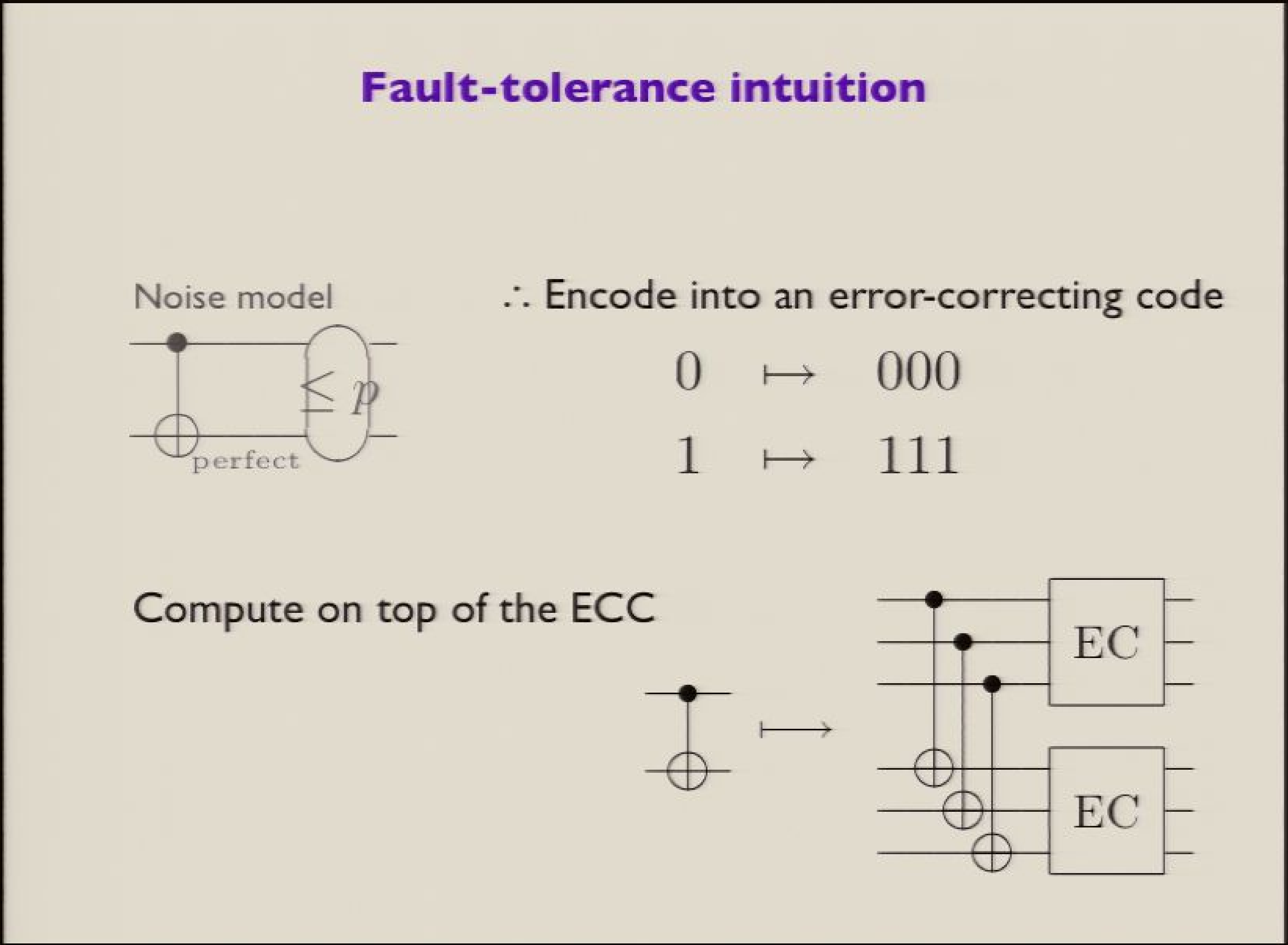
Controlled-NOT gate

flips target if control bit is set

Noise model

For proof sketch:

Model a noisy gate as a perfect gate followed by independent *bit-flip* errors (XI, IX or XX) — with total error rate $p_{XI}+p_{IX}+p_{XX}$ at most p



Fault-tolerance intuition

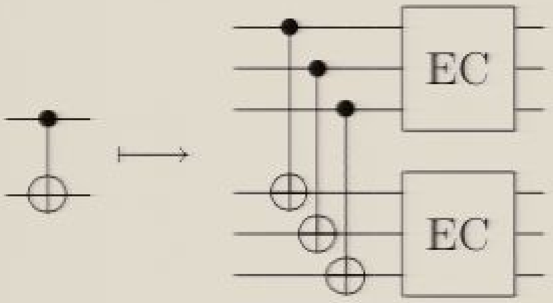
Noise model

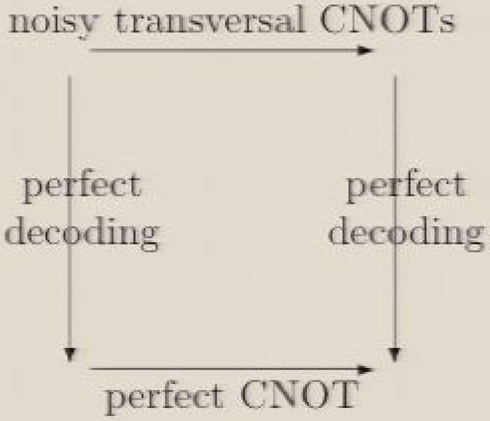
$\therefore$ Encode into an error-correcting code

$$\begin{aligned} 0 &\mapsto 000 \\ 1 &\mapsto 111 \end{aligned}$$

Compute on top of the ECC

Fault-tolerance intuition





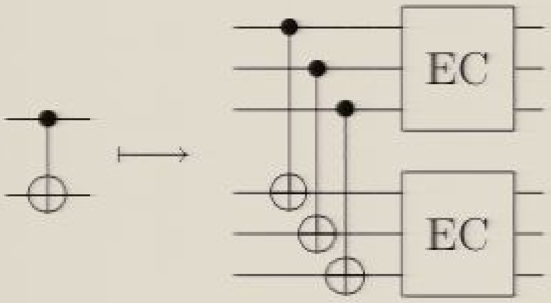
noisy transversal CNOTs

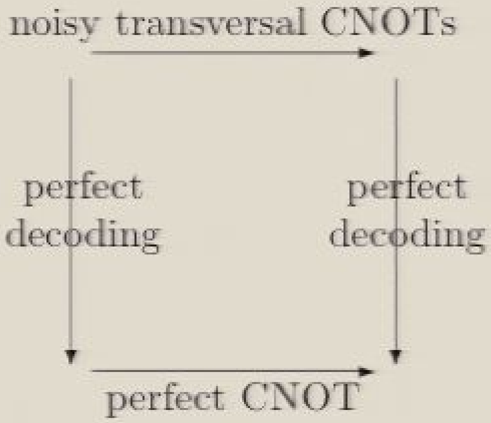
perfect decoding

perfect decoding

perfect CNOT

Fault-tolerance intuition





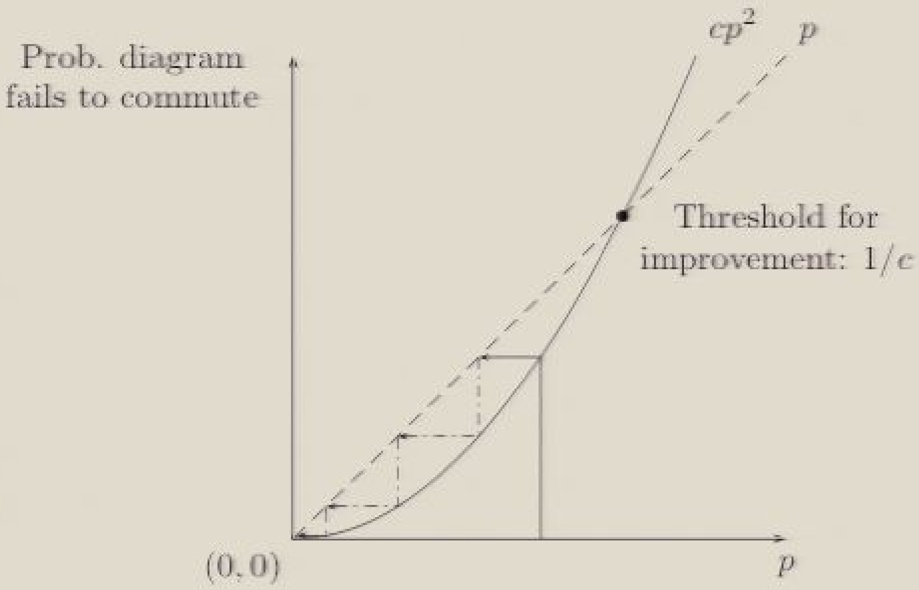
noisy transversal CNOTs

perfect decoding

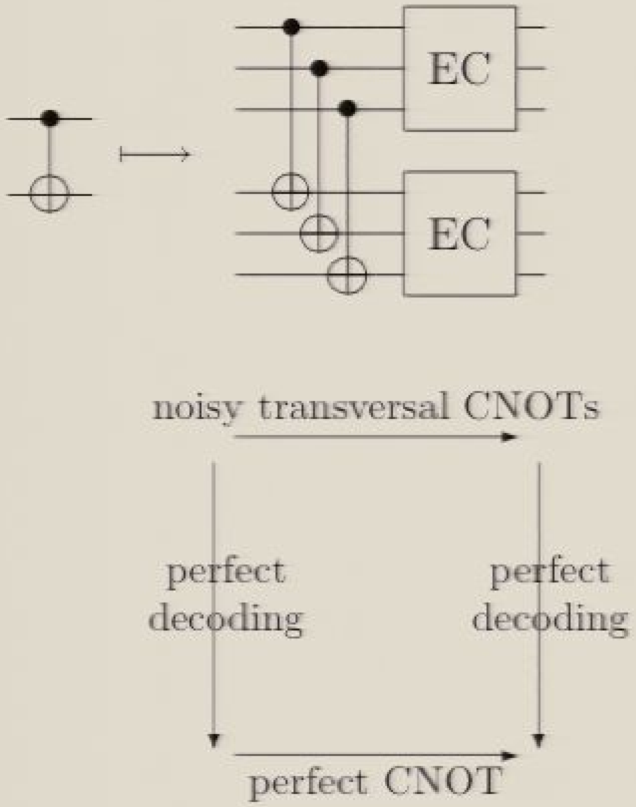
perfect decoding

perfect CNOT

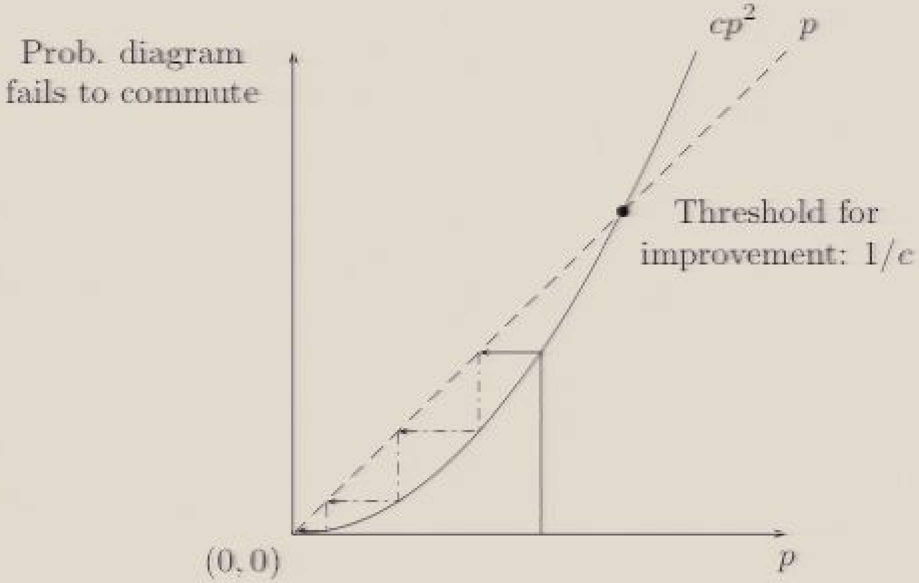
Improved reliability beneath constant tolerable noise threshold



Fault-tolerance intuition

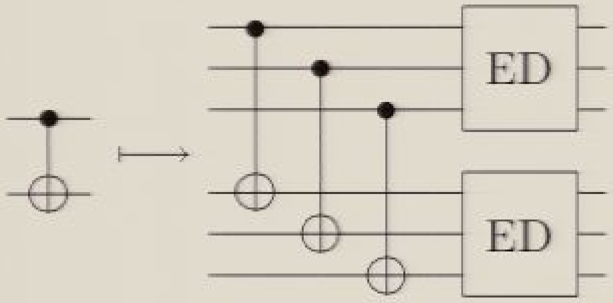


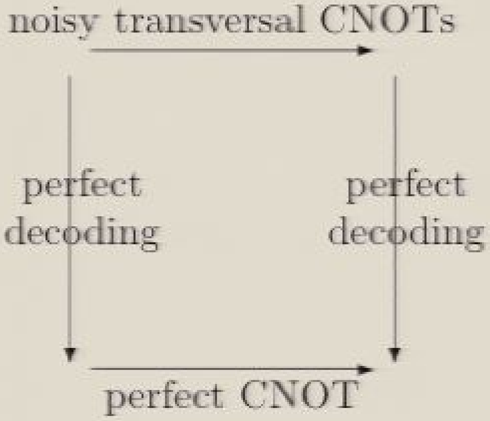
Improved reliability beneath constant tolerable noise threshold



Repeat for arbitrarily improved reliability.

Error-detection-based FT intuition





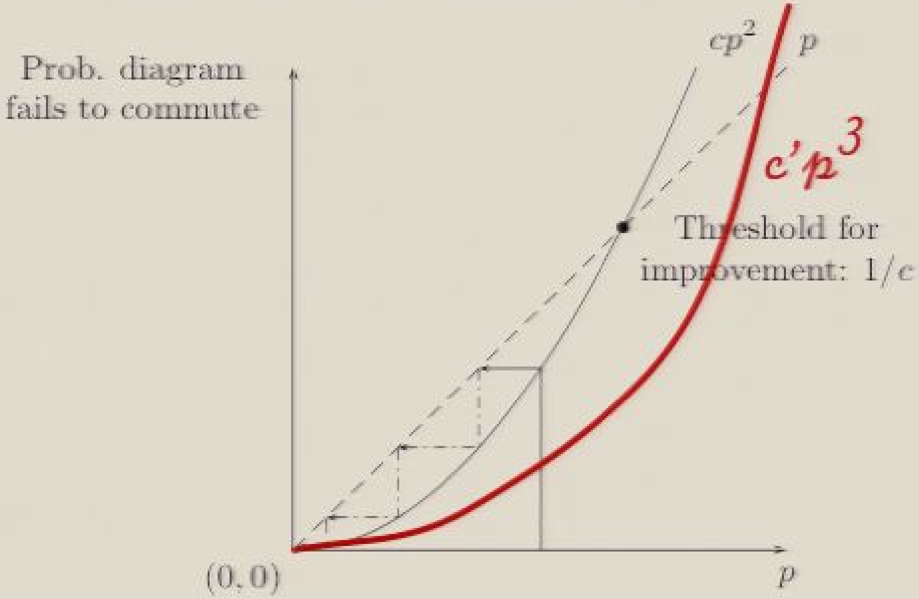
noisy transversal CNOTs

perfect decoding

perfect decoding

perfect CNOT

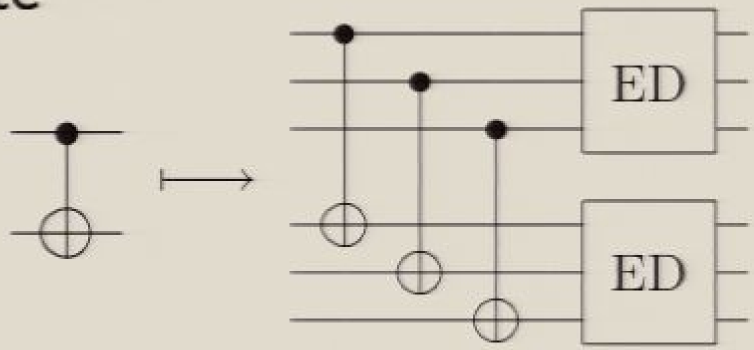
Improved reliability beneath constant tolerable noise threshold



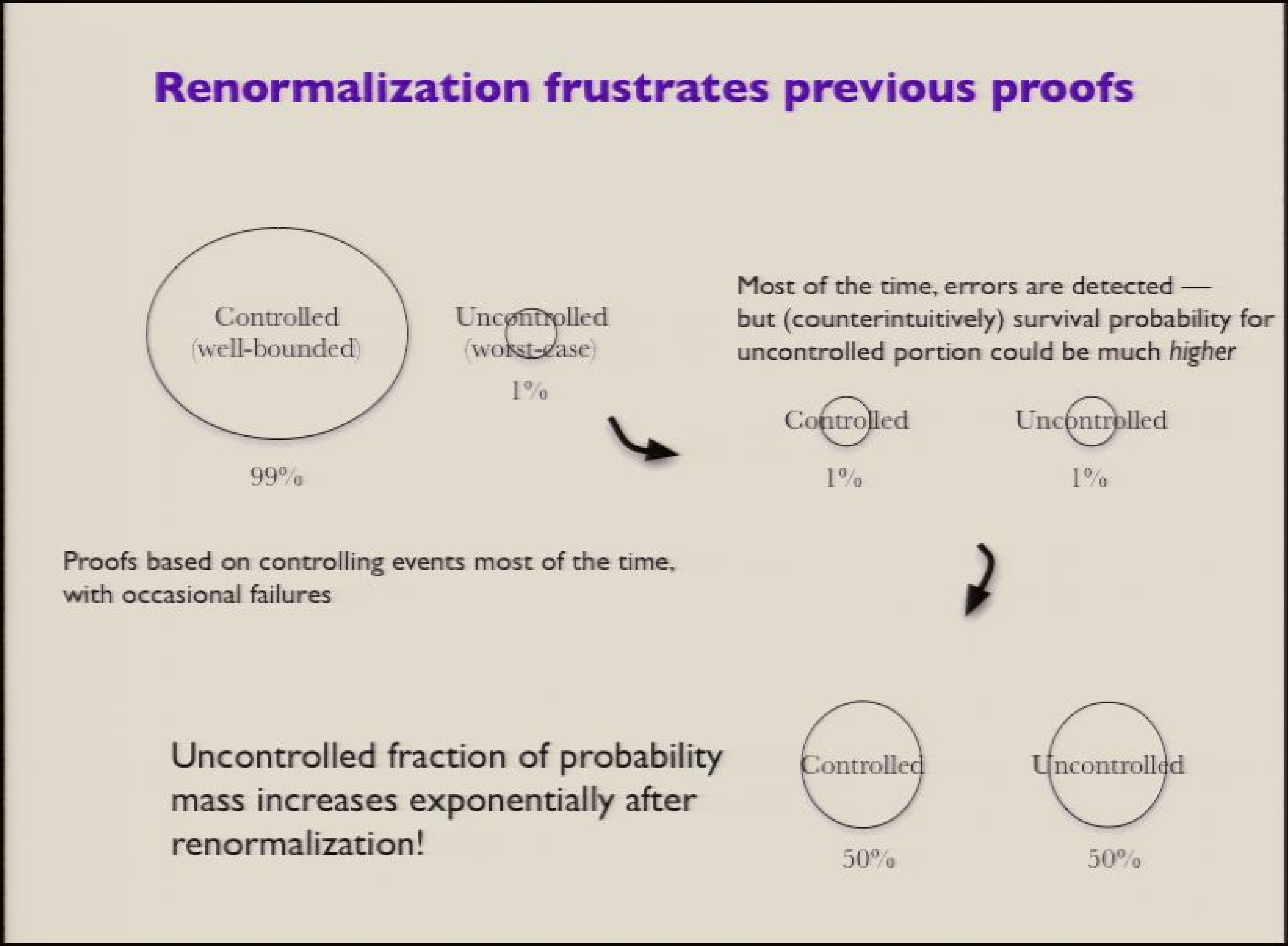
Repeat for arbitrarily improved reliability.

Fault-tolerance based on error *detection*

CNOT gate



- In simulations, tolerates 10x higher noise rates than error-correction-based FT schemes
- But previously, no proven positive threshold at all!
- Note: Overhead is substantial, but theoretically efficient

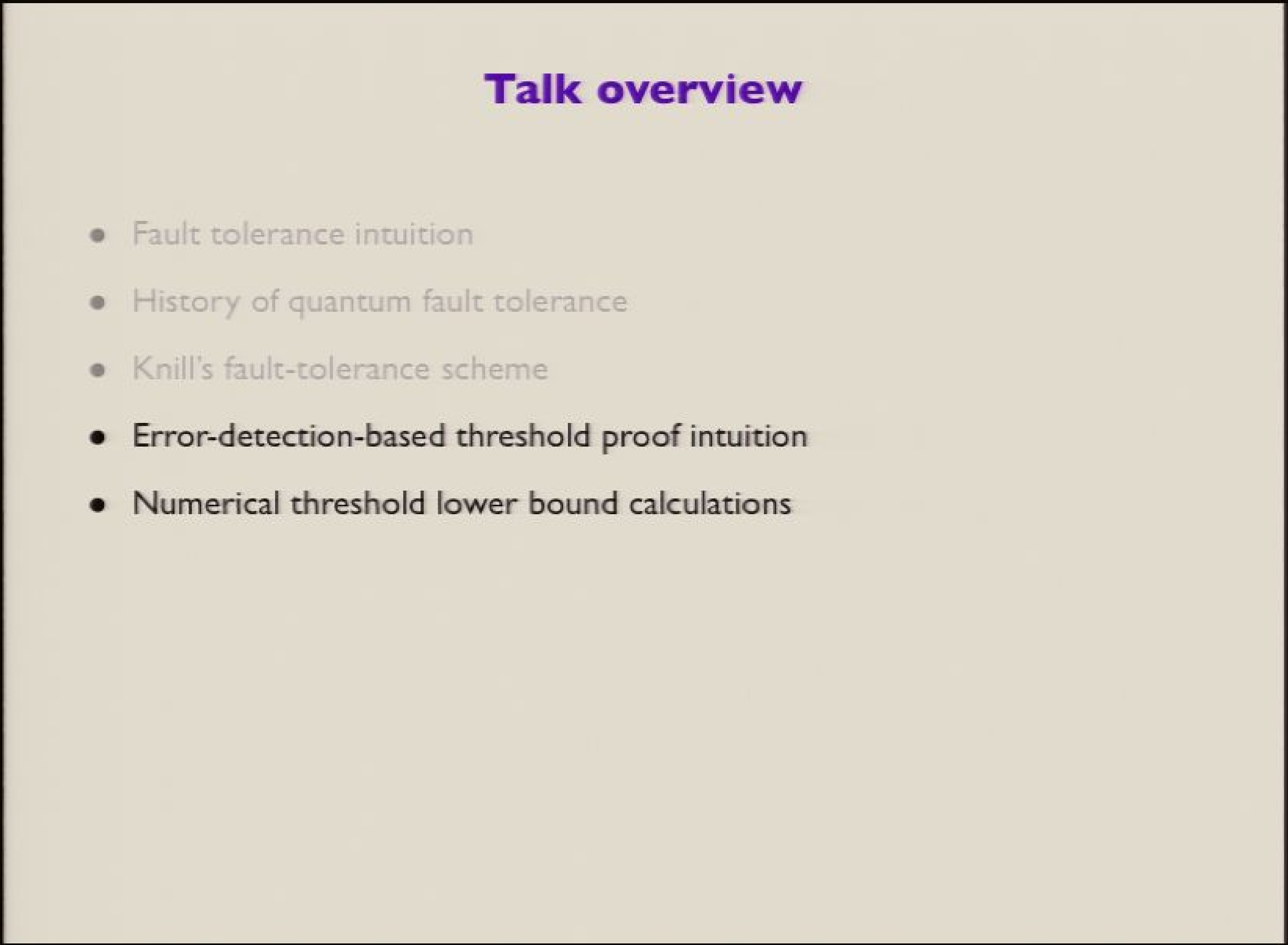


Renormalization frustrates previous proofs

Most of the time, errors are detected — but (counterintuitively) survival probability for uncontrolled portion could be much *higher*

Proofs based on controlling events most of the time, with occasional failures

Uncontrolled fraction of probability mass increases exponentially after renormalization!

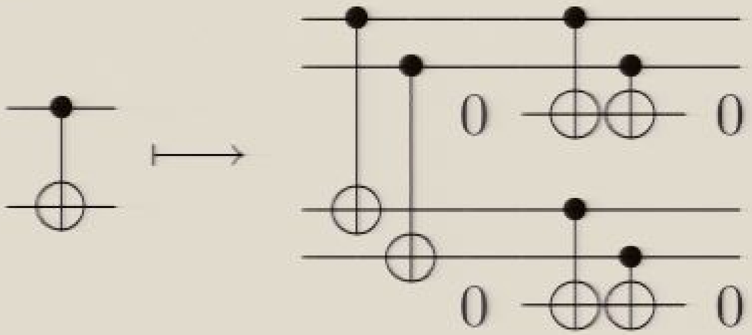


Talk overview

- Fault tolerance intuition
- History of quantum fault tolerance
- Knill's fault-tolerance scheme
- Error-detection-based threshold proof intuition
- Numerical threshold lower bound calculations

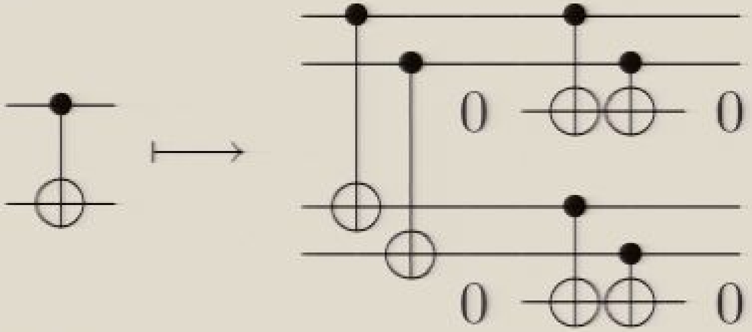
Fault-tolerance based on error *detection*

CNOT gate

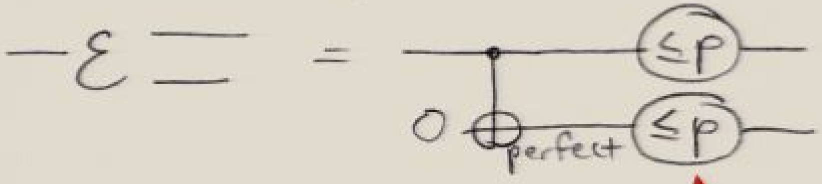


Proof intuition

CNOT gate



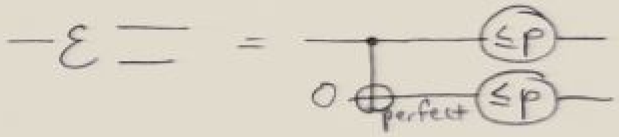
Notation: Noisy encoder

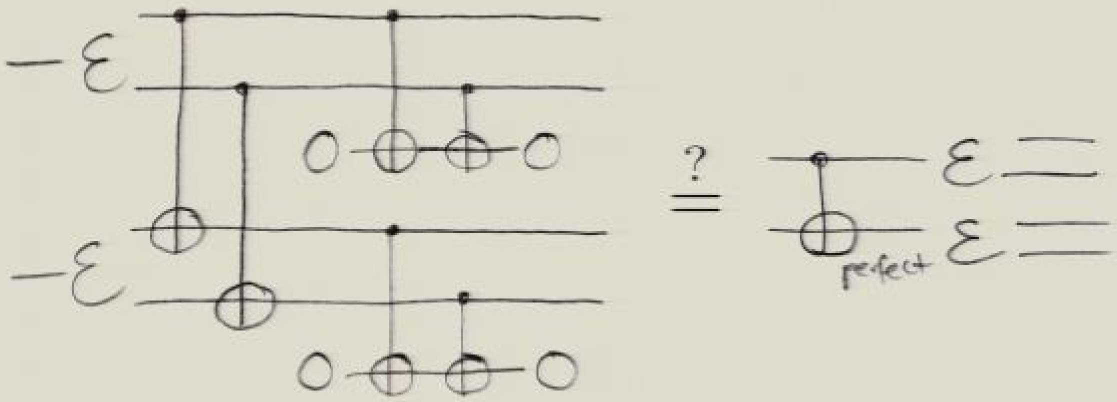


Remark: Distribution here can be arbitrary

Proof intuition

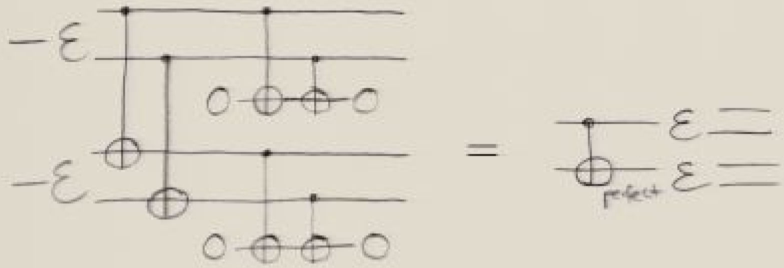
Notation



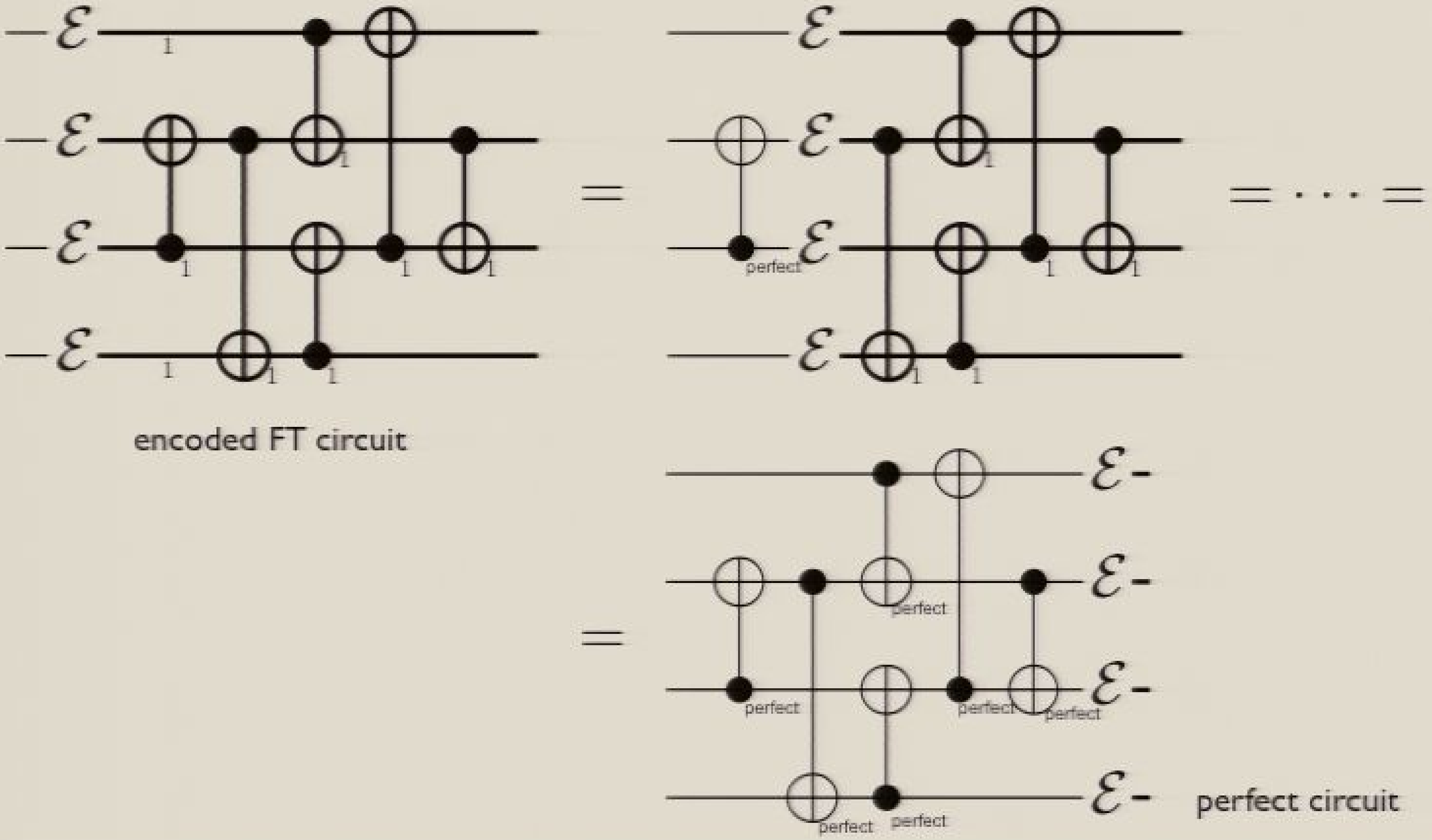


bitwise-independent errors
preceding encoded CNOT gate

bitwise-independent errors
following encoded CNOT gate



would be nice, since

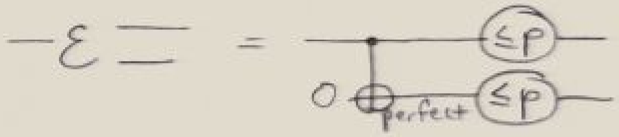


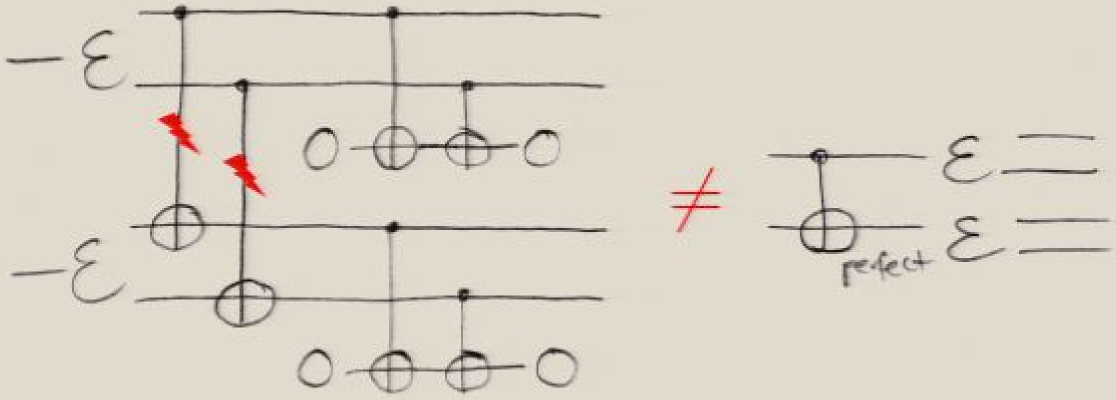
encoded FT circuit

perfect circuit

Proof intuition

Notation

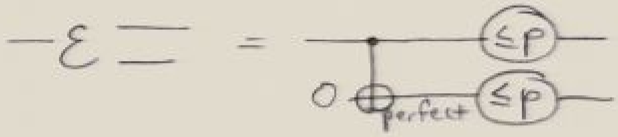


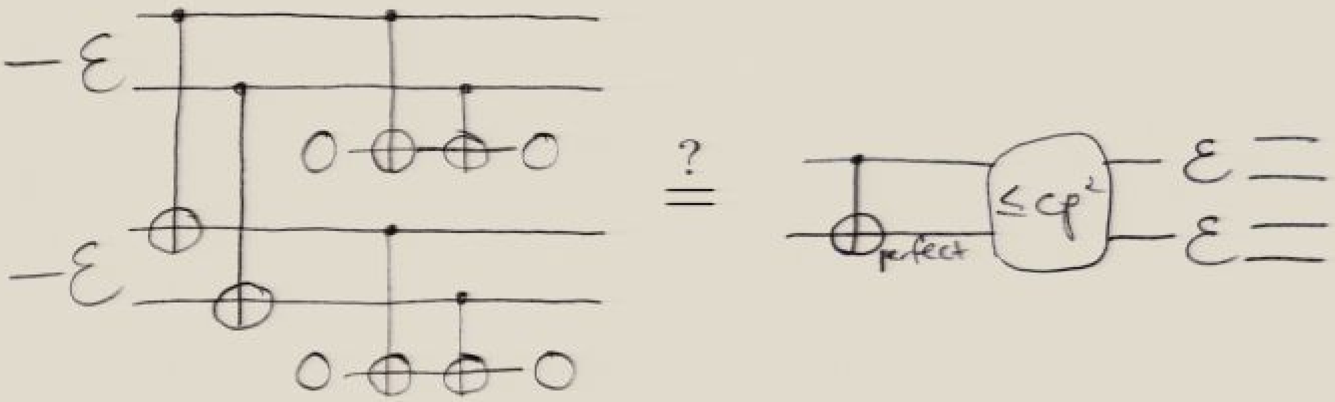


P[XXXX] ~ p^2 P[XXXX] = p^4

Proof intuition

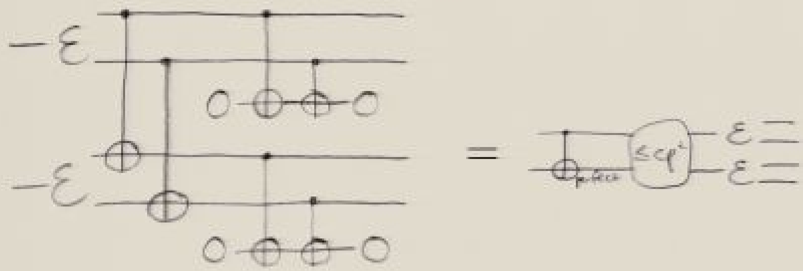
Notation



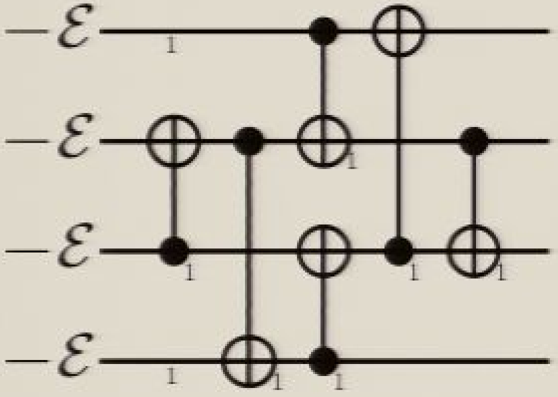


bitwise-independent errors
preceding encoded CNOT gate

bitwise-independent errors
following encoded CNOT gate,
plus quadratically suppressed
independent logical errors



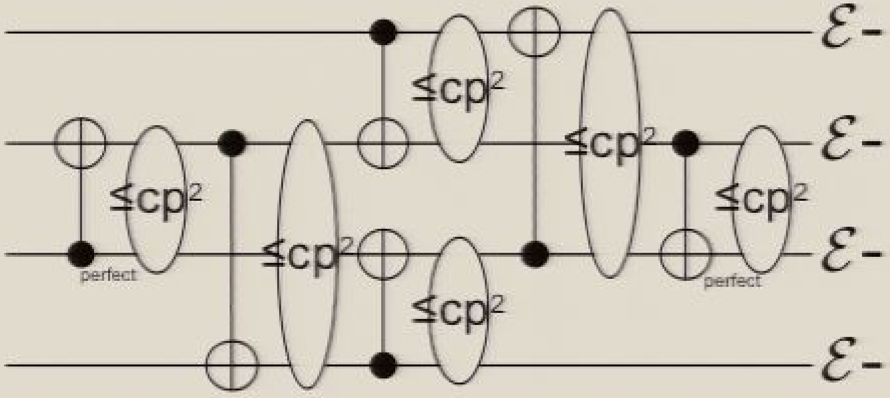
would be nice, since



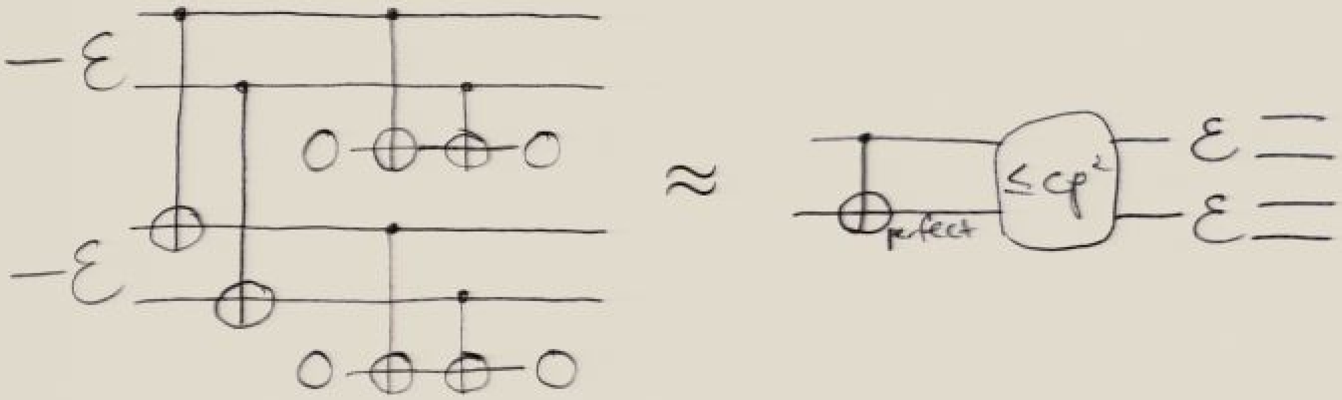
encoded FT circuit

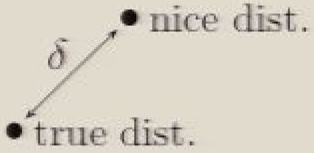
$= \cdots =$

$=$



Proof intuition

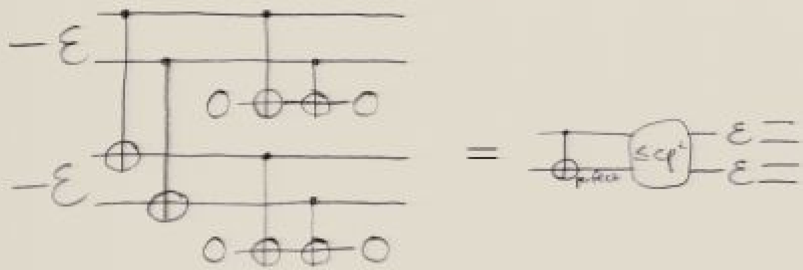




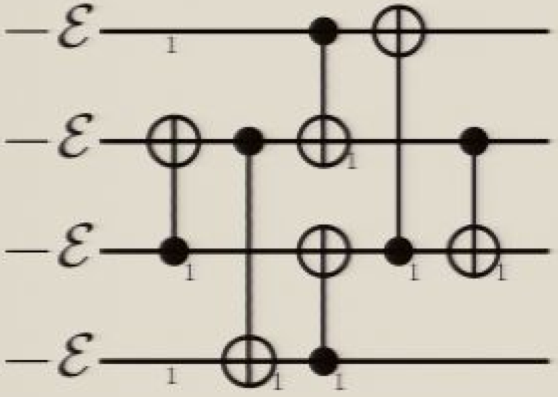
nice dist.

δ

true dist.



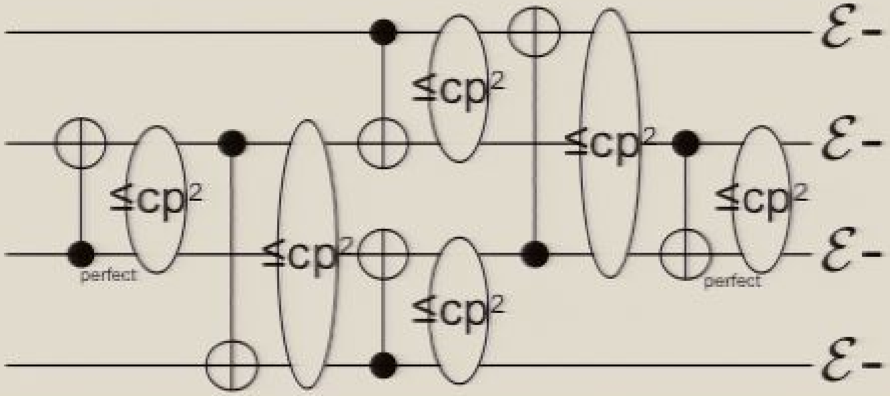
would be nice, since



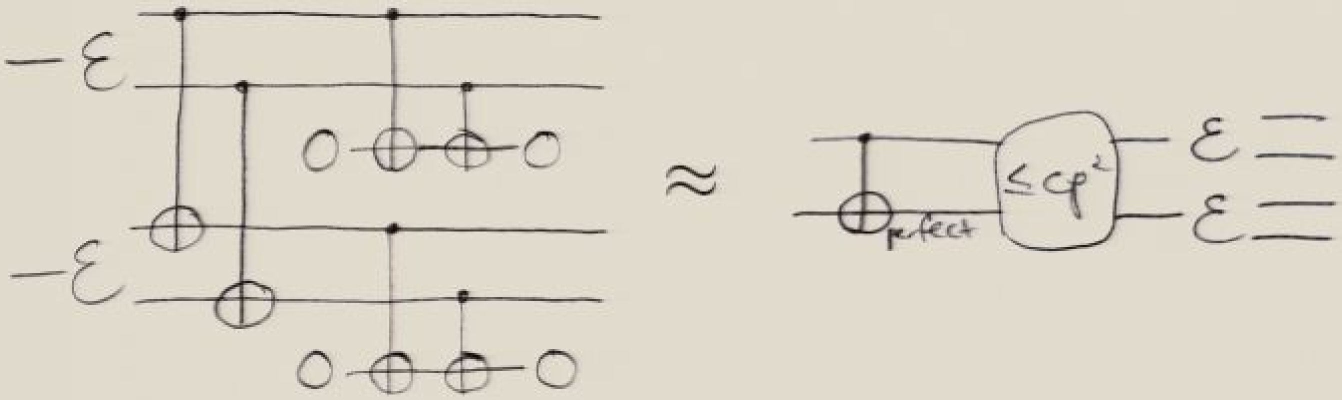
encoded FT circuit

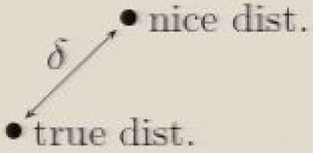
$= \cdots =$

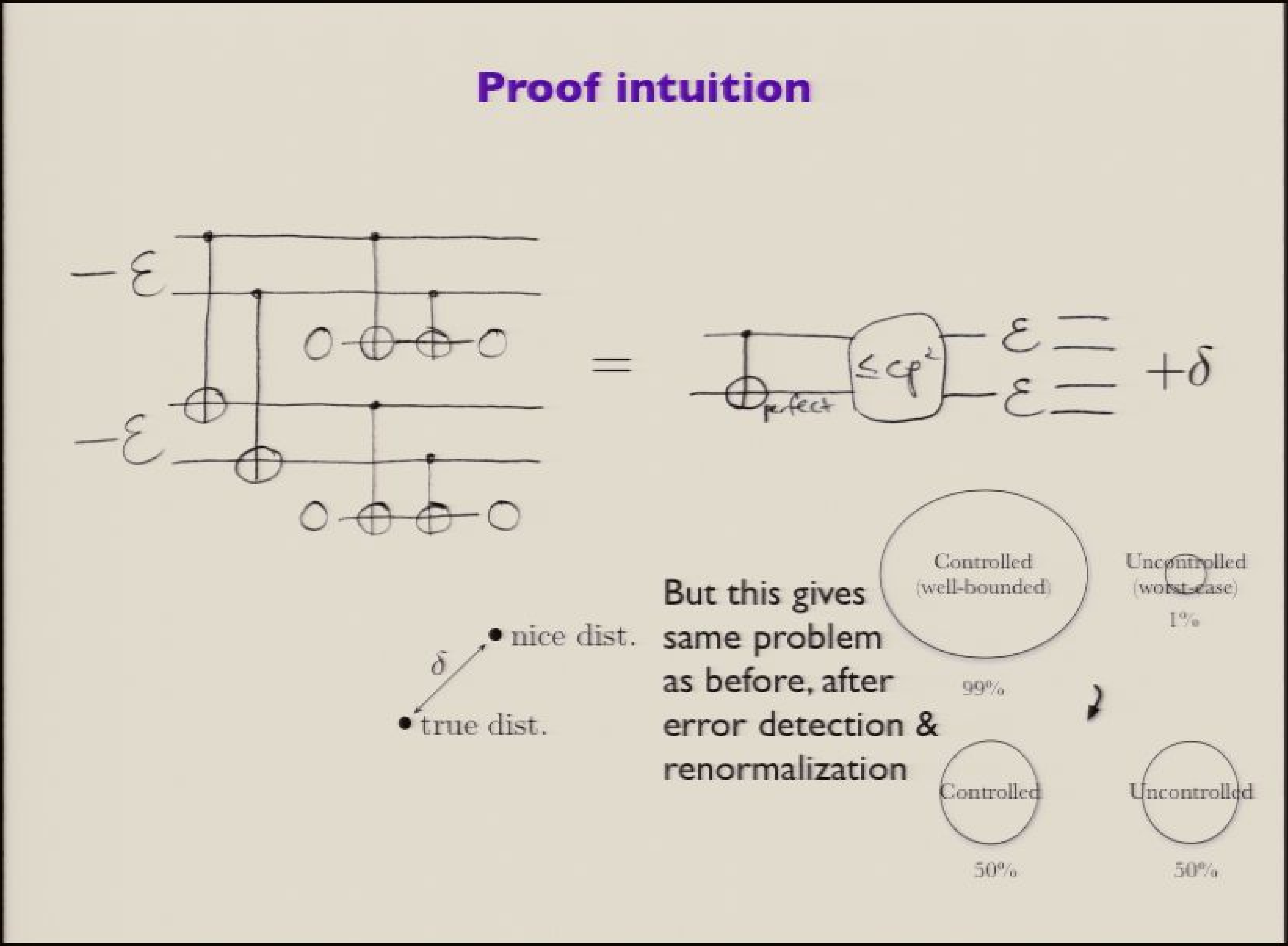
$=$



Proof intuition







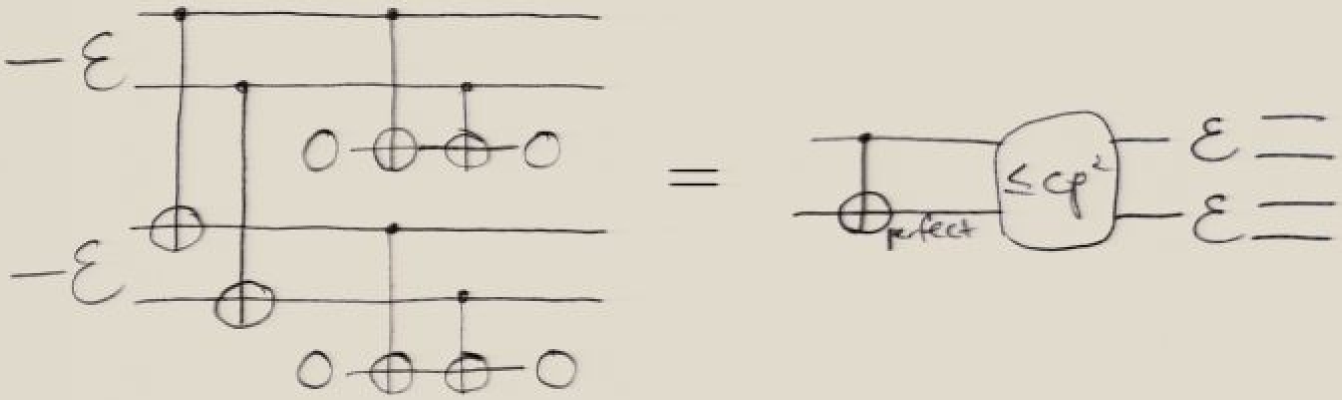
Proof intuition

• nice dist.

• true dist.

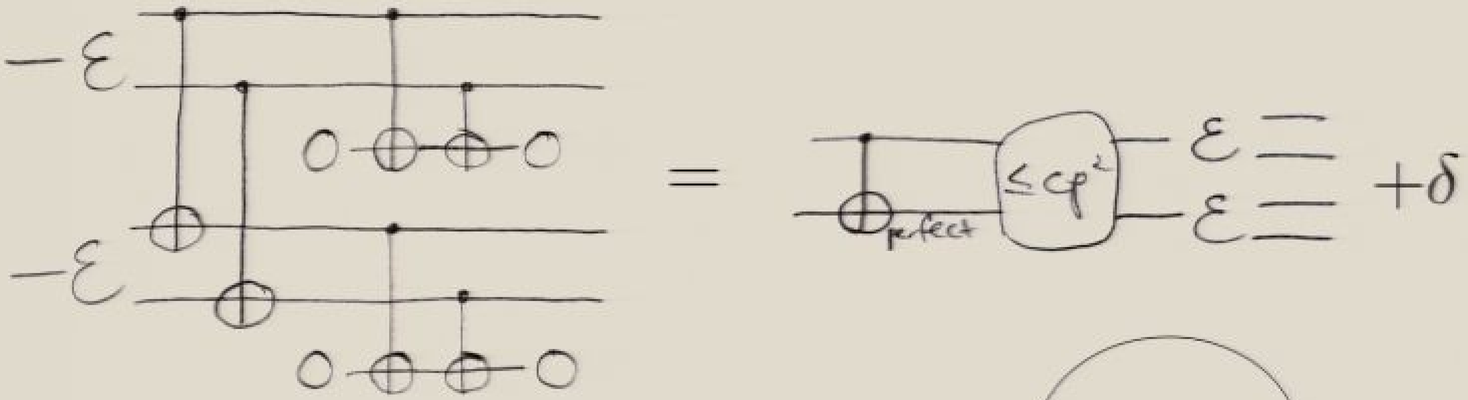
But this gives same problem as before, after error detection & renormalization

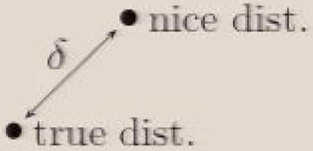
Known error model



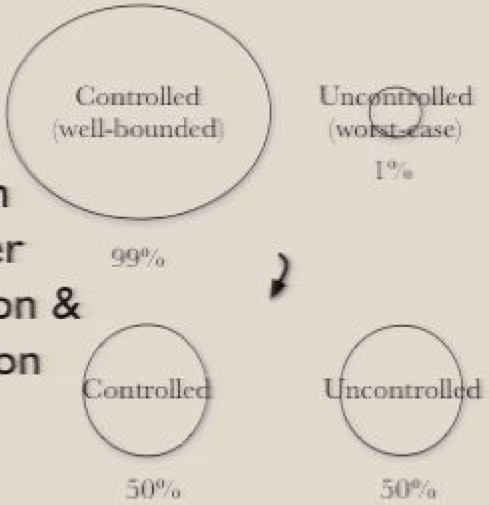
- If error distribution is known exactly, then can deliberately introduce errors to force equality
    - Pauli errors can be (effectively) introduced by changing the Pauli frame
- Results:
    - 0.7% depolarization per CNOT (other operations similar noise rates), by permutation- and Hadamard-symmetrizing (2^{8-2}=64) dimensions down to 11
    - 1.1% symmetrizing to 17-dimensions (permutations only)

Proof intuition

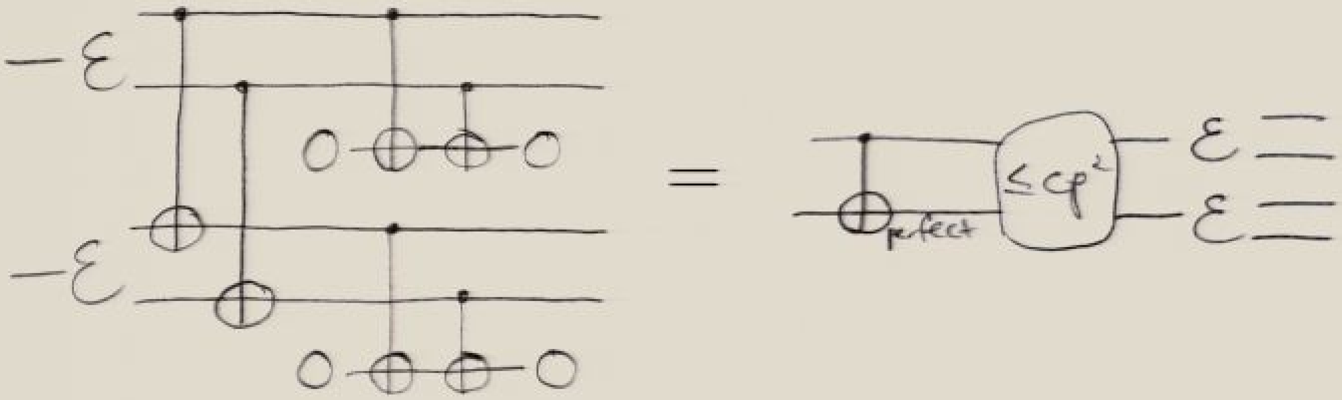




But this gives same problem as before, after error detection & renormalization

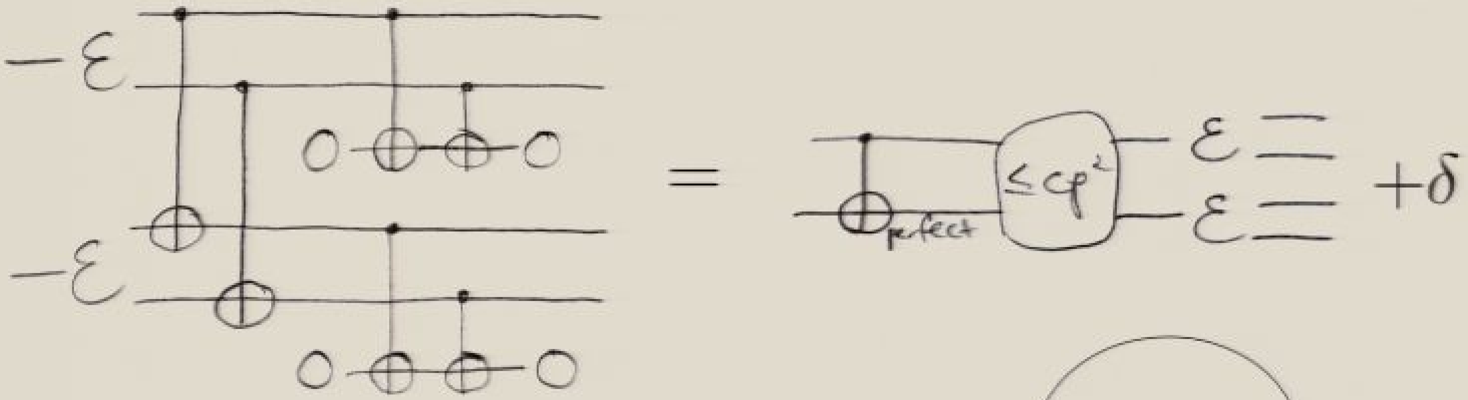


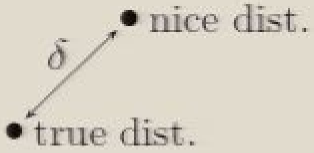
Known error model



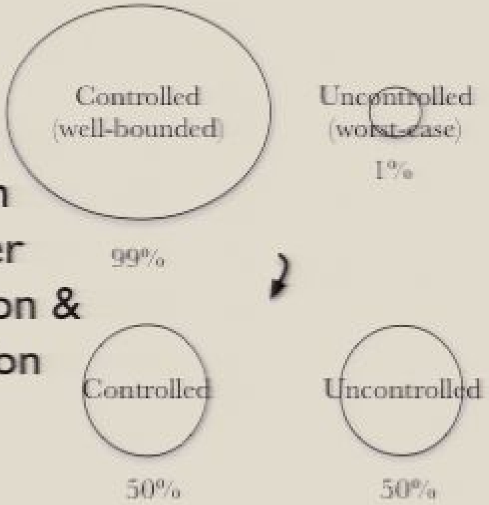
- If error distribution is known exactly, then can deliberately introduce errors to force equality
  - Pauli errors can be (effectively) introduced by changing the Pauli frame
- Results:
  - 0.7% depolarization per CNOT (other operations similar noise rates), by permutation- and Hadamard-symmetrizing (2^{8-2}=64) dimensions down to 11
  - 1.1% symmetrizing to 17-dimensions (permutations only)

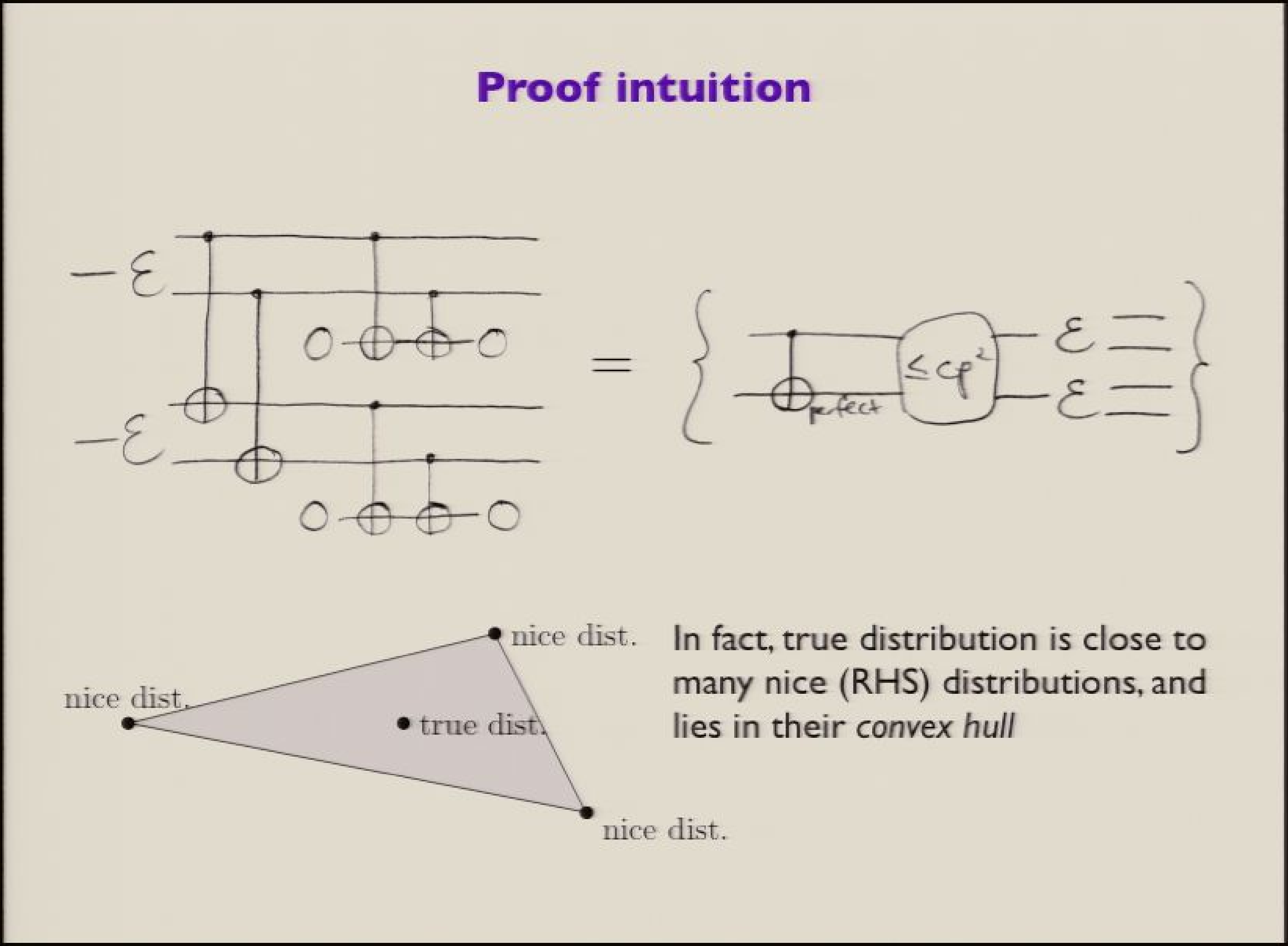
Proof intuition





But this gives same problem as before, after error detection & renormalization

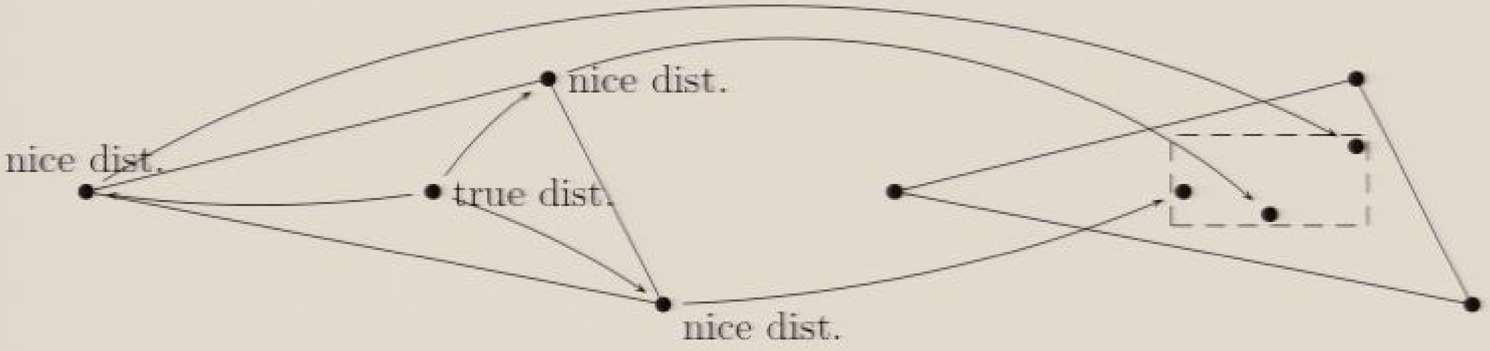




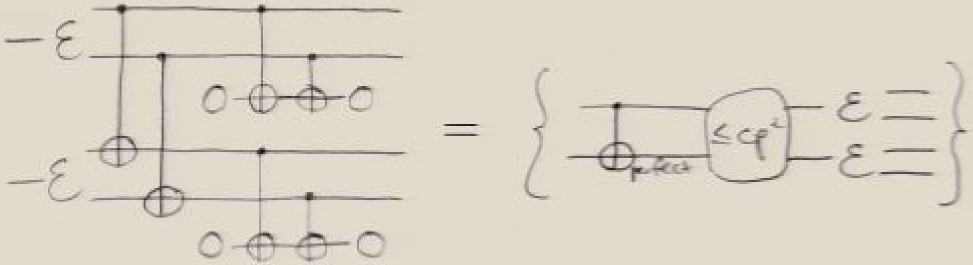
Proof intuition

In fact, true distribution is close to many nice (RHS) distributions, and lies in their *convex hull*

Induction step

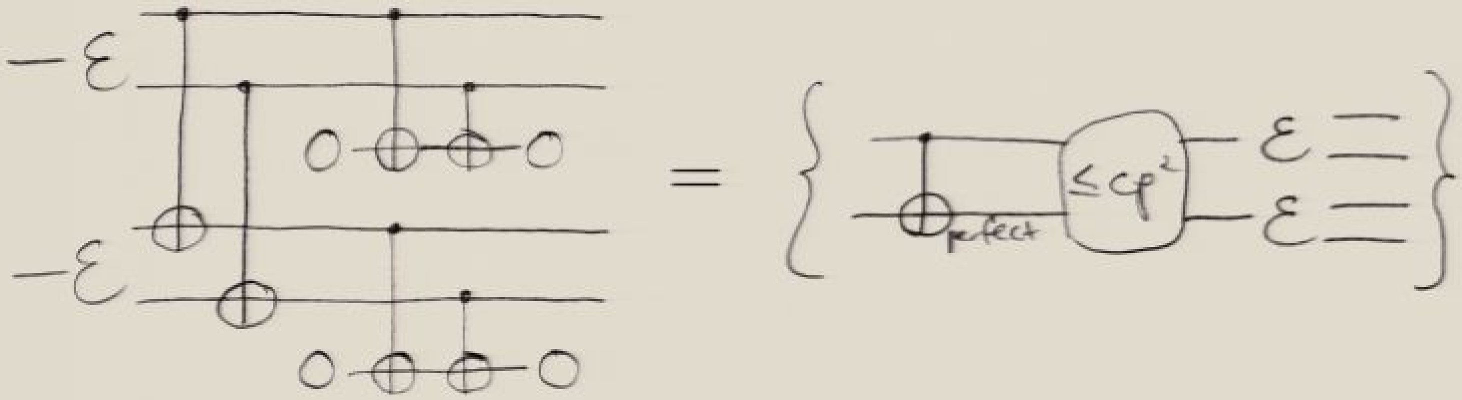


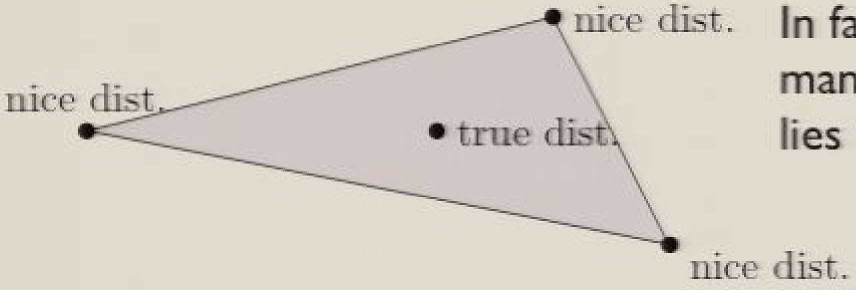
Analysis of the next encoded CNOT gate proceeds by *picking* one of the vertices — a nice distribution — then applying the CNOT mixing lemma:



Each output distibution can again be rewritten as mixture of nice distributions, etc.

Proof intuition

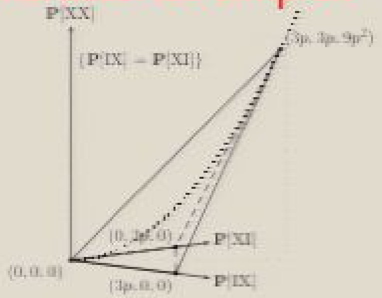




In fact, true distribution is close to many nice (RHS) distributions, and lies in their *convex hull*

Proving that mixing works

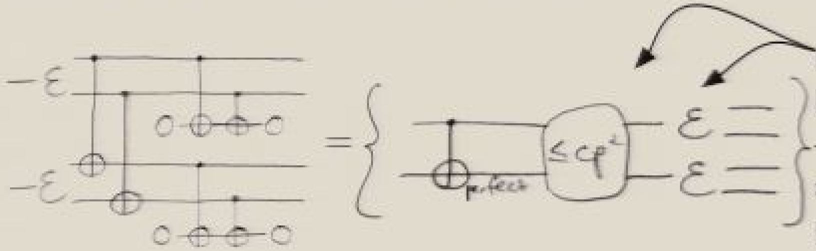
- Existence argument (for threshold existence proofs):
  - characterize simplex convex hull of dit-wise independent distributions



Mixing Lemma

  - "pull back" actual distribution onto distn. on dits

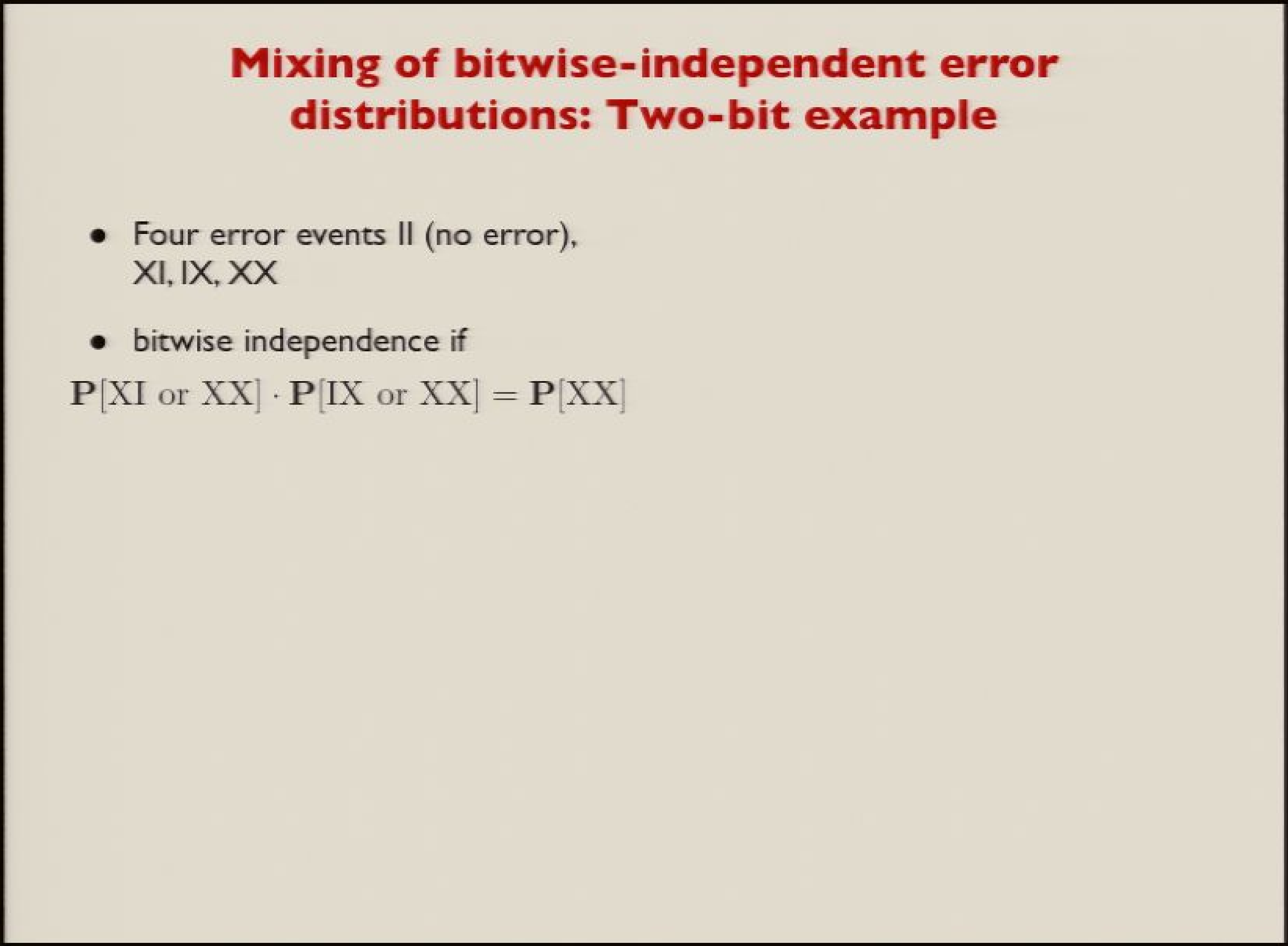
Two-bit case is simple because every error event has distinct effect (convex hull of n points in n-1 dimensions)



Now, different events can lead to same error — convex hull no longer a simplex

E.g., convex hull of 64 points in 15 dimensions

- Numerical approach (for numerical threshold lower bounds)…



Mixing of bitwise-independent error distributions: Two-bit example

- Four error events II (no error), XI, IX, XX
- bitwise independence if

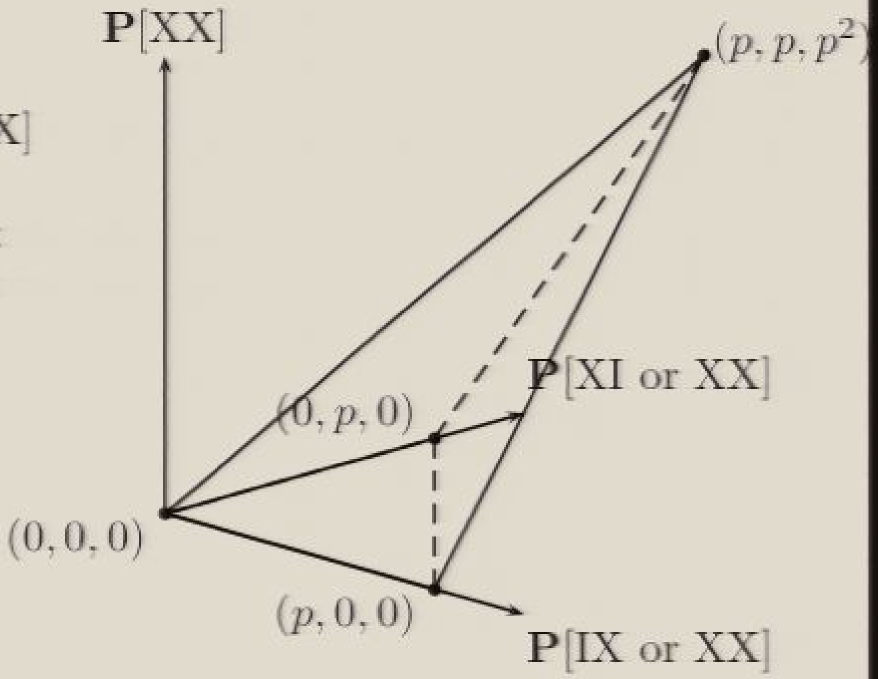
$$\mathbf{P}[\mathrm{XI\ or\ XX}] \cdot \mathbf{P}[\mathrm{IX\ or\ XX}] = \mathbf{P}[\mathrm{XX}]$$

Mixing of bitwise-independent error distributions: Two-bit example

- Four error events II (no error), XI, IX, XX
- bitwise independence if

$$\mathbf{P}[\text{XI or XX}] \cdot \mathbf{P}[\text{IX or XX}] = \mathbf{P}[\text{XX}]$$

- Claim: Convex hull of all product distributions with bit error rates ≤ p is:

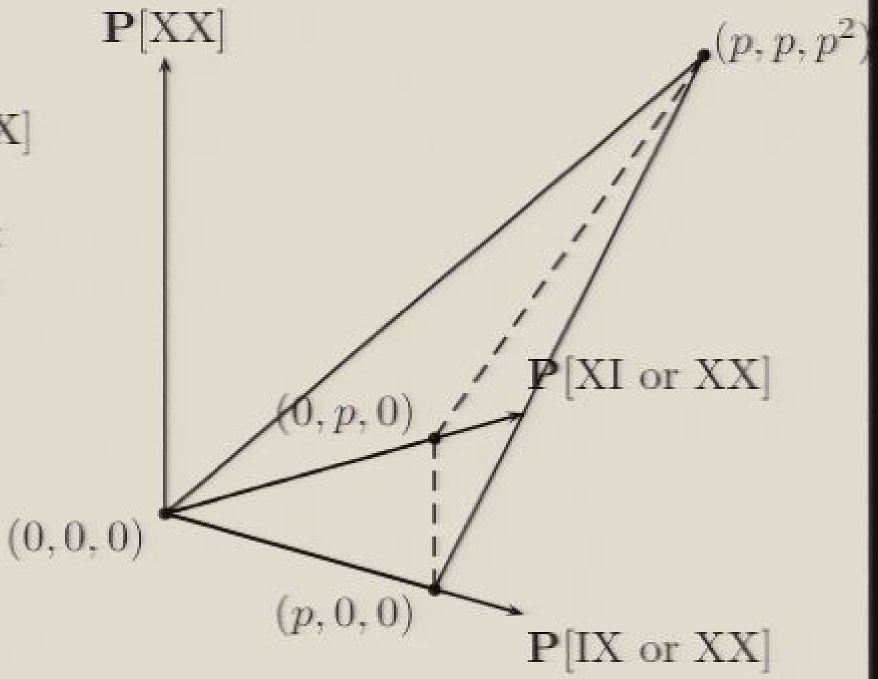


Mixing of bitwise-independent error distributions: Two-bit example

- Four error events II (no error), XI, IX, XX
- bitwise independence if

$$\mathbf{P}[\mathrm{XI\ or\ XX}] \cdot \mathbf{P}[\mathrm{IX\ or\ XX}] = \mathbf{P}[\mathrm{XX}]$$

- Claim: Convex hull of all product distributions with bit error rates ≤ p is:


$\mathbf{P}[\mathrm{XX}]$
(p, p, p^2)
$\mathbf{P}[\mathrm{XI\ or\ XX}]$
$(0, p, 0)$
$(0, 0, 0)$
$(p, 0, 0)$
$\mathbf{P}[\mathrm{IX\ or\ XX}]$

Remarks:

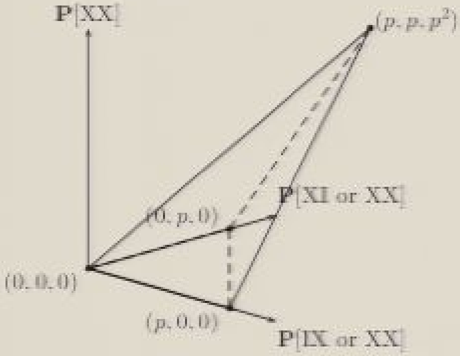
1. Natural lattice coordinates
2. 3=4-1 dimensions
3. 4=2^2 extremal distributions (each bit can be noisy or not) → simplex

Product distribution mixing: General case

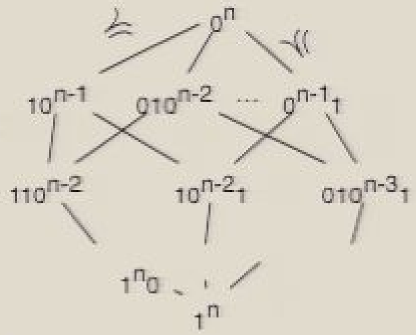
- Four error events II (no error), XI, IX, XX
- bitwise independence if

$$\mathbf{P}[\mathrm{XI\ or\ XX}] \cdot \mathbf{P}[\mathrm{IX\ or\ XX}] = \mathbf{P}[\mathrm{XX}]$$

- Claim: Convex hull of all product distributions with bit error rates ≤ p is:


$\mathbf{P}[\mathrm{XX}]$
(p,p,p^2)
$\mathbf{P}[\mathrm{XI\ or\ XX}]$
$(0,p,0)$
$(0,0,0)$
$(p,0,0)$
$\mathbf{P}[\mathrm{IX\ or\ XX}]$

- Lattice:



- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is $\{\mathbf{P}[\bullet]\}$ s.t.:

$$\forall x \in \{0,1\}^n$$

$$\sum_{y \preccurlyeq x} (-1)^{|y|-|x|} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p(\{z \preccurlyeq y\})} \geq 0$$

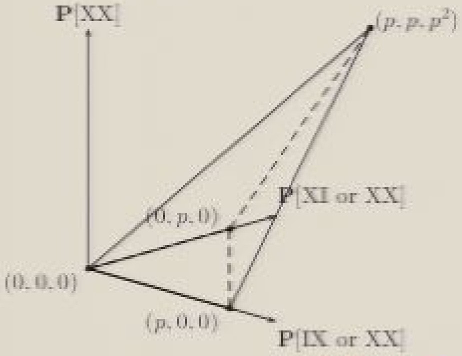
where $p(\{z \preccurlyeq y\}) \equiv \prod_i p_i^{y_i}$

Product distribution mixing: General case

- Four error events II (no error), XI, IX, XX
- bitwise independence if

$$\mathbf{P}[\mathrm{XI\ or\ XX}] \cdot \mathbf{P}[\mathrm{IX\ or\ XX}] = \mathbf{P}[\mathrm{XX}]$$

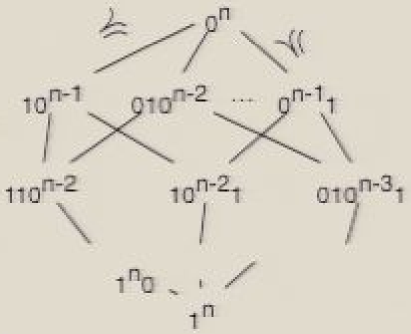
- Claim: Convex hull of all product distributions with bit error rates ≤ p is:



$\mathbf{P}[\mathrm{XX}]$
(p, p, p^2)
$\mathbf{P}[\mathrm{XI\ or\ XX}]$
$(0, p, 0)$
$(0, 0, 0)$
$(p, 0, 0)$
$\mathbf{P}[\mathrm{IX\ or\ XX}]$

1. Natural lattice coordinates
2. 3=4-1 dimensions
3. Simplex of 4=2² extremal distributions (each bit can be noisy or not)

- Lattice:



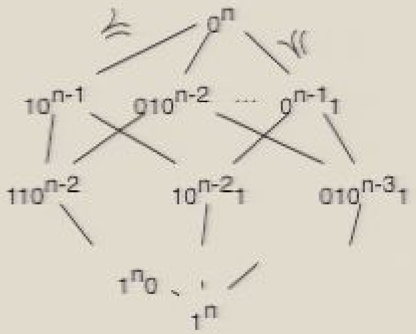
- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is {**P[•]**} s.t.:

$$\forall x \in \{0,1\}^n$$

$$\sum_{y \preccurlyeq x} (-1)^{|y|-|x|} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p(\{z \preccurlyeq y\}} \geq 0$$

where $p(\{z \preccurlyeq y\}) \equiv \prod_i p_i^{y_i}$

Product distribution mixing: General case



- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is {**P**[•]} s.t.:

$$\forall x \in \{0,1\}^n$$
$$\sum_{y \preccurlyeq x} (-1)^{|y|-|x|} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p(\{z \preccurlyeq y\})} \geq 0$$

where $p(\{z \preccurlyeq y\}) \equiv \prod_i p_i^{y_i}$

E.g., $p_i = p$:

$$\frac{\mathbf{P}[\{z \preccurlyeq x\}]}{p^{|x|}} - \sum_{y: y \preccurlyeq x, |y|=|x|+1} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+1}} + \sum_{y: y \preccurlyeq x, |y|=|x|+2} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+2}} - \cdots \geq 0$$

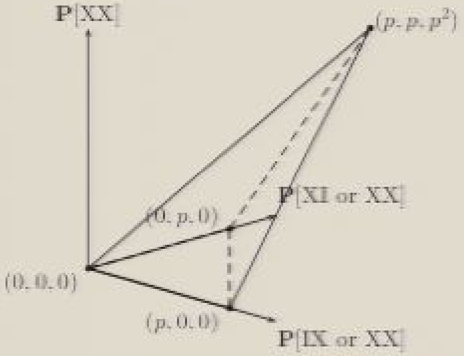
- Corollary: If **P**[{y≤x}]=$\Theta(p^{|x|})$ for all x in $\{0,1\}^n$, then **P**[•] lies in convex hull of product distributions with bit error rates O(p).

Product distribution mixing: General case

- Four error events II (no error), XI, IX, XX
- bitwise independence if

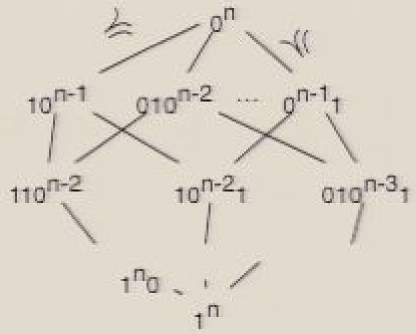
$$\mathbf{P}[\mathrm{XI\ or\ XX}]\cdot\mathbf{P}[\mathrm{IX\ or\ XX}]=\mathbf{P}[\mathrm{XX}]$$

- Claim: Convex hull of all product distributions with bit error rates ≤ p is:


$\mathbf{P}[\mathrm{XX}]$
(p,p,p^2)
$\mathbf{P}[\mathrm{XI\ or\ XX}]$
$(0,p,0)$
$(0,0,0)$
$(p,0,0)$
$\mathbf{P}[\mathrm{IX\ or\ XX}]$

1. Natural lattice coordinates
2. 3=4-1 dimensions
3. Simplex of $4=2^2$ extremal distributions (each bit can be noisy or not)

- Lattice:



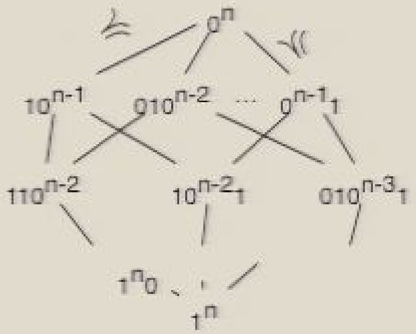
- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is $\{\mathbf{P}[\bullet]\}$ s.t.:

$$\forall x\in\{0,1\}^n$$

$$\sum_{y\preccurlyeq x}(-1)^{|y|-|x|}\frac{\mathbf{P}[\{z\preccurlyeq y\}]}{p(\{z\preccurlyeq y\})}\geq 0$$

where $p(\{z\preccurlyeq y\})\equiv\prod_i p_i^{y_i}$

Product distribution mixing: General case



- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is $\{\mathbf{P}[\cdot]\}$ s.t.:

$$\forall x \in \{0,1\}^n$$
$$\sum_{y \preccurlyeq x} (-1)^{|y|-|x|} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p(\{z \preccurlyeq y\})} \geq 0$$

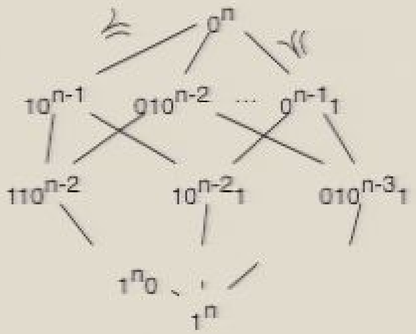
where $p(\{z \preccurlyeq y\}) \equiv \prod_i p_i^{y_i}$

E.g., $p_i = p$:

$$\frac{\mathbf{P}[\{z \preccurlyeq x\}]}{p^{|x|}} - \sum_{y:y \preccurlyeq x, |y|=|x|+1} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+1}} + \sum_{y:y \preccurlyeq x, |y|=|x|+2} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+2}} - \cdots \geq 0$$

- Corollary: If $\mathbf{P}[\{y \leq x\}] = \Theta(p^{|x|})$ for all x in $\{0,1\}^n$, then $\mathbf{P}[\cdot]$ lies in convex hull of product distributions with bit error rates O(p).

Product distribution mixing: General case



- Mixing Lemma: Convex hull of all product distributions with ith bit error rate ≤ p_i, is {**P**[•]} s.t.:

$\forall x \in \{0$ 0

Generalizes further to
{0,1,...,m}^{n}, e.g.,
{0,1,2,3}^{n} = {I,X,Y,Z}^{n}

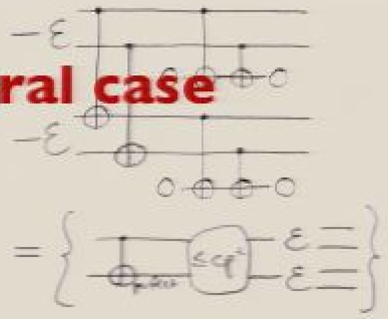
where $p(\{z \preccurlyeq y\}) \equiv \prod_i p_i^{y_i}$

E.g., $p_i = p$:

$$\frac{\mathbf{P}[\{z \preccurlyeq x\}]}{p^{|x|}} - \sum_{y: y \preccurlyeq x, |y|=|x|+1} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+1}} + \sum_{y: y \preccurlyeq x, |y|=|x|+2} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{p^{|x|+2}} - \cdots \geq 0$$

- Corollary: If **P**[{y≤x}]=Θ($p^{|x|}$) for all x in {0,1}^{n}, then **P**[•] lies in convex hull of product distributions with bit error rates O(p).

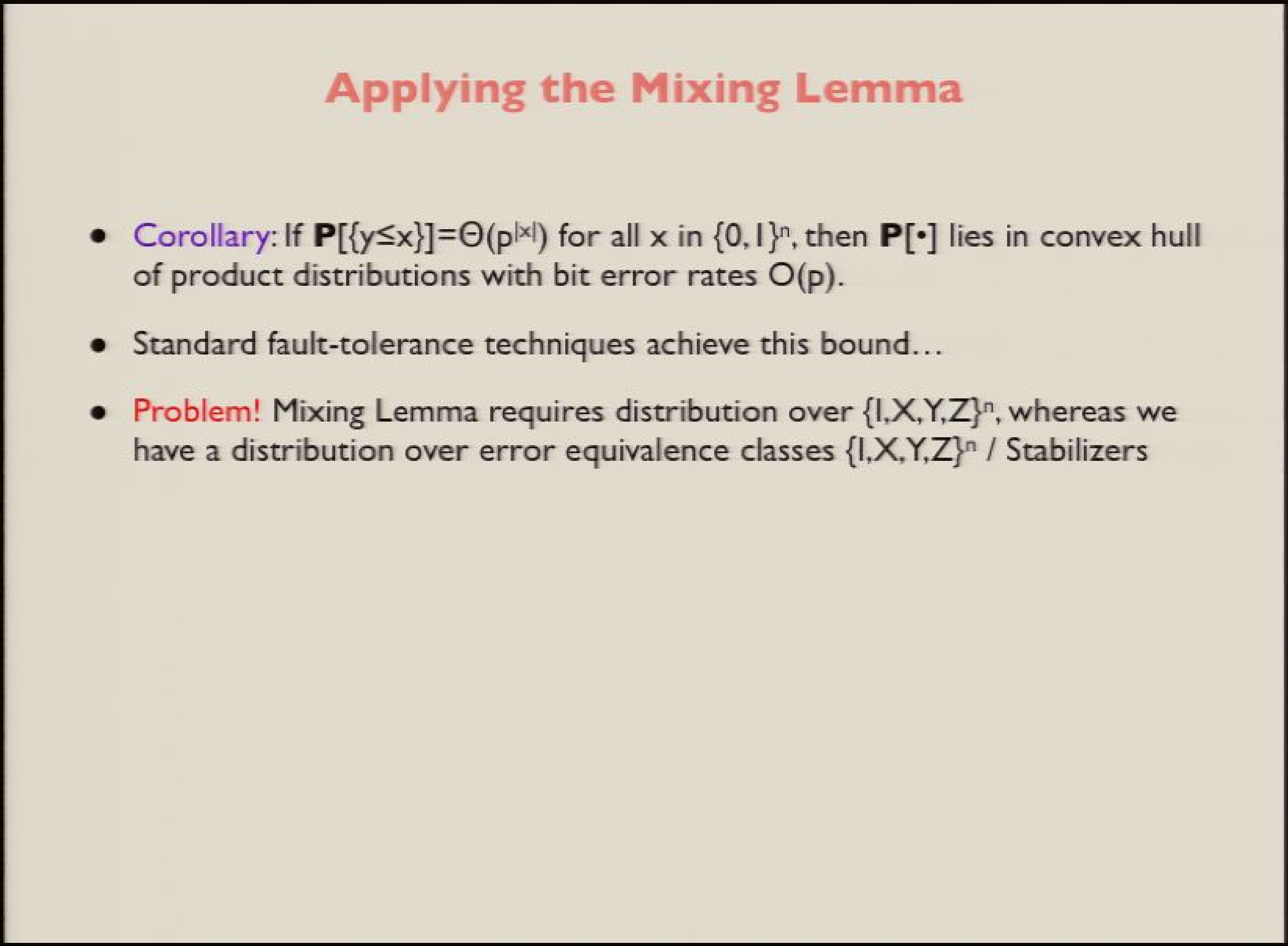
Product distribution mixing: General case



Theorem (Mixing Lemma)**.** *For $i \in \{1, \ldots, n\}$, fix probabilities p^i_j satisfying $\sum_{j=0}^m p^i_j = 1$. Then the convex hull $\mathcal{S}$ of all product distributions with dit error rates q^i_j such that $\sum_{j \neq 0} \frac{q^i_j}{p^i_j} \leq 1$, is given exactly by those $\mathbf{P}[\cdot]$ satisfying*

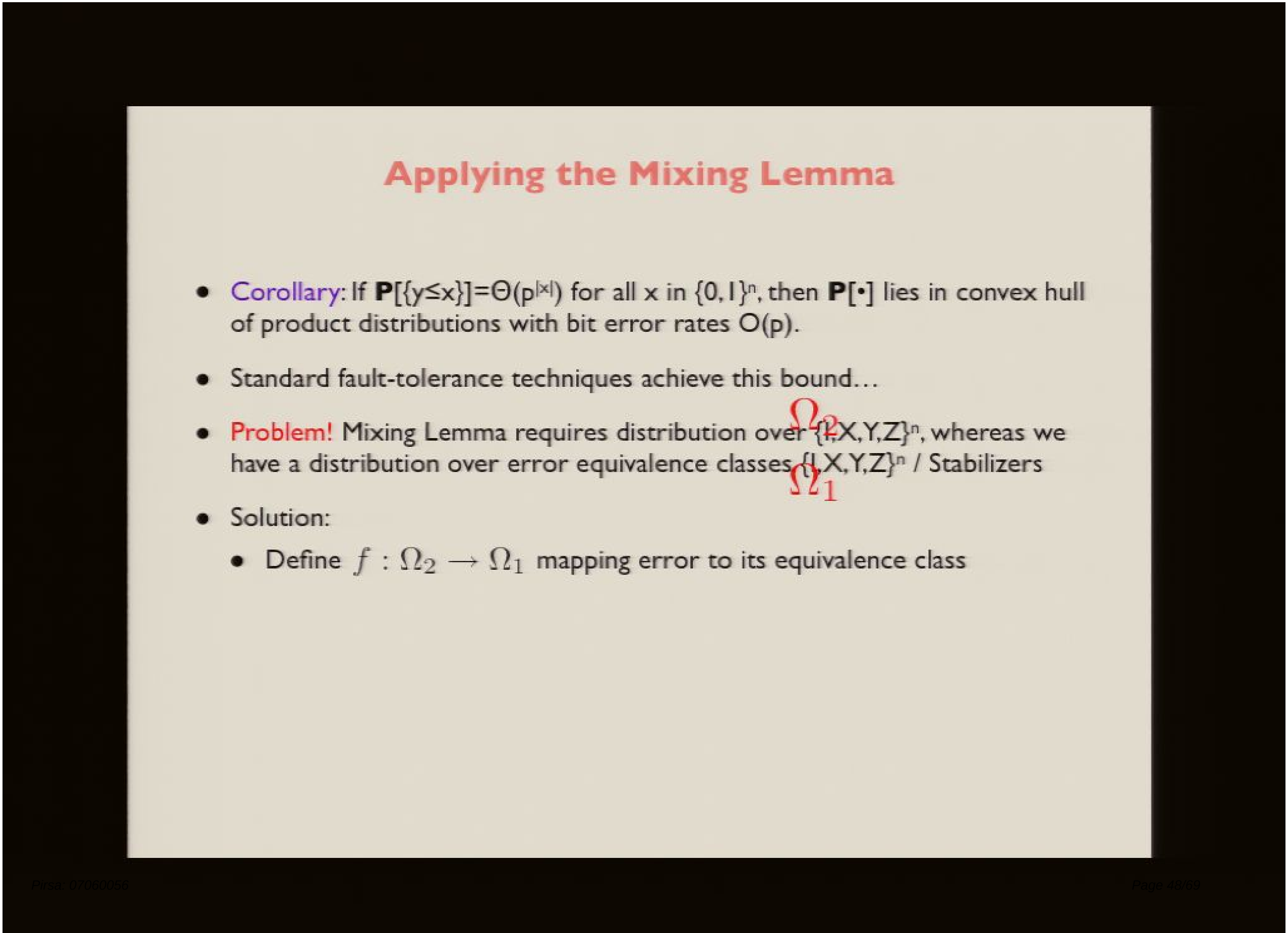
$$\forall x \in \{0, 1, \ldots, m\}^n : \qquad \sum_{y \preccurlyeq x} (-1)^{|y|-|x|} \frac{\mathbf{P}[\{z \preccurlyeq y\}]}{y^p(\{z \preccurlyeq y\})} \geq 0 \ , \tag{1}$$

where $y^p(\{z \preccurlyeq y\}) \equiv \prod_{i : y_i \neq 0} p^i_{y_i}$.



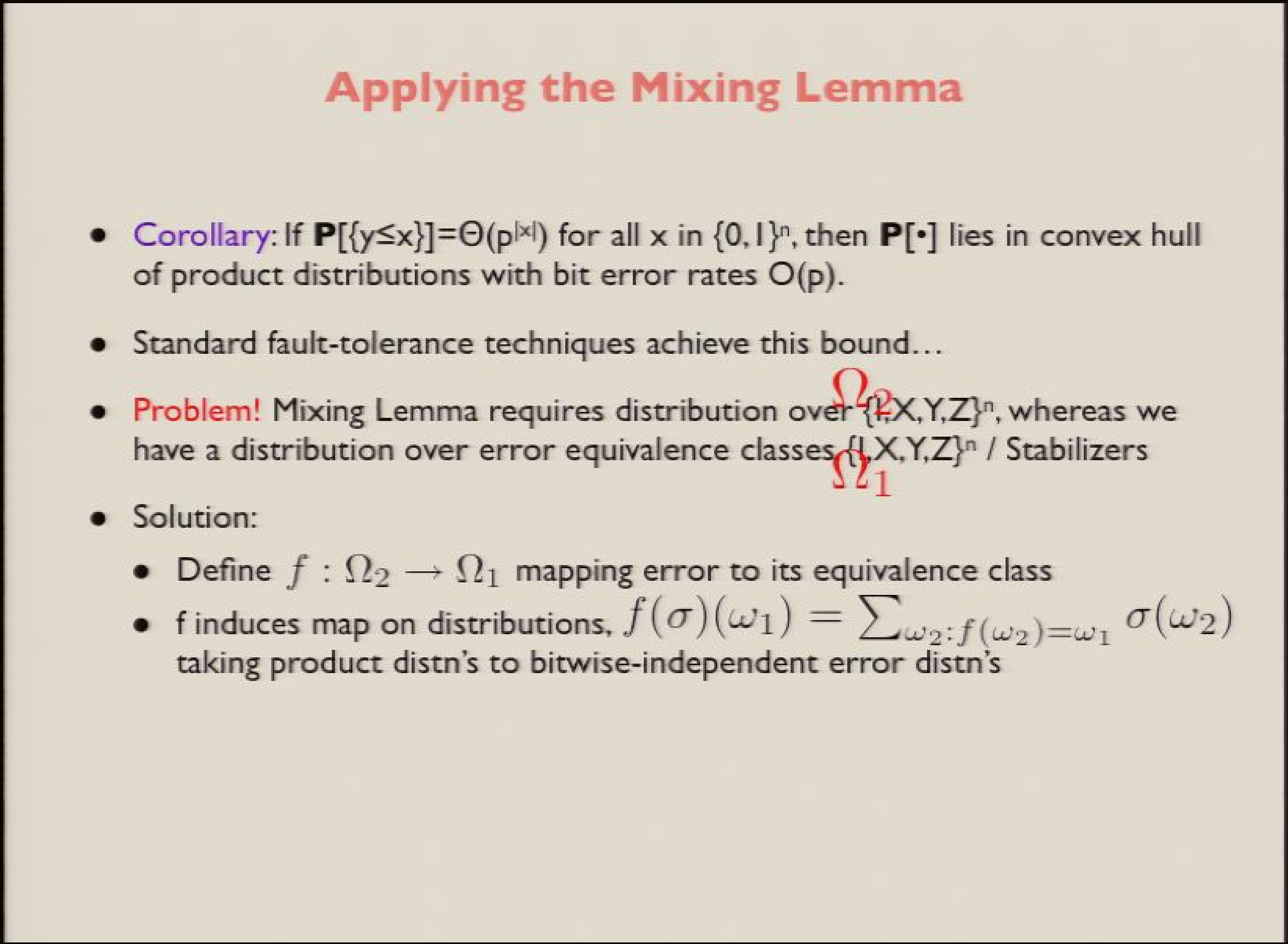
Applying the Mixing Lemma

- Corollary: If **P**[{y≤x}]=$\Theta(p^{|x|})$ for all x in $\{0,1\}^n$, then **P**[•] lies in convex hull of product distributions with bit error rates O(p).
- Standard fault-tolerance techniques achieve this bound…
- Problem! Mixing Lemma requires distribution over $\{I,X,Y,Z\}^n$, whereas we have a distribution over error equivalence classes $\{I,X,Y,Z\}^n$ / Stabilizers



Applying the Mixing Lemma

- Corollary: If $\mathbf{P}[\{y \leq x\}] = \Theta(p^{|x|})$ for all x in $\{0,1\}^n$, then $\mathbf{P}[\cdot]$ lies in convex hull of product distributions with bit error rates O(p).
- Standard fault-tolerance techniques achieve this bound…
- Problem! Mixing Lemma requires distribution over $\overset{\Omega_2}{\{I,X,Y,Z\}^n}$, whereas we have a distribution over error equivalence classes $\underset{\Omega_1}{\{I,X,Y,Z\}^n}$ / Stabilizers
- Solution:
  - Define $f : \Omega_2 \rightarrow \Omega_1$ mapping error to its equivalence class

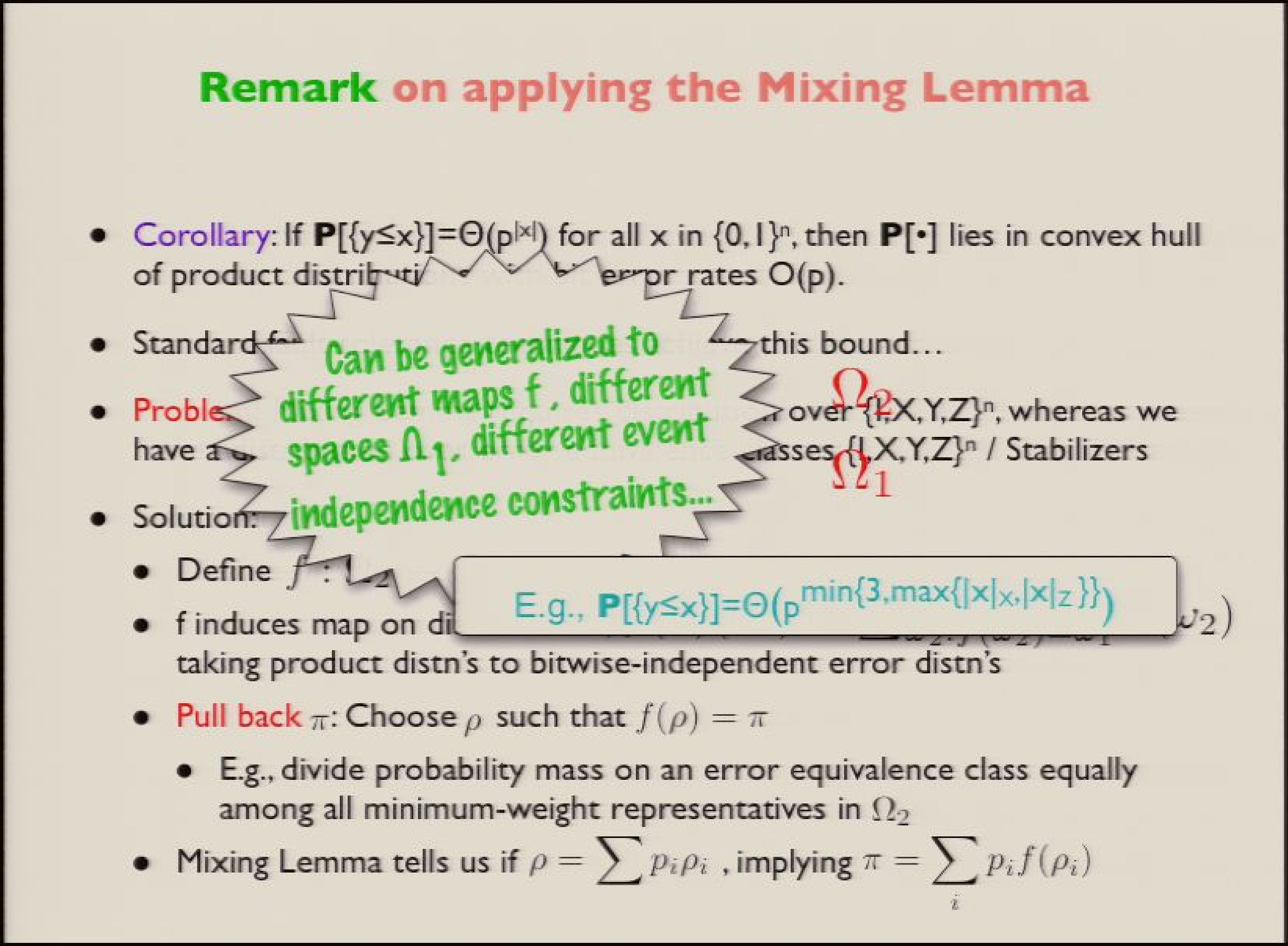


Applying the Mixing Lemma

- Corollary: If $\mathbf{P}[\{y \leq x\}] = \Theta(p^{|x|})$ for all x in $\{0,1\}^n$, then $\mathbf{P}[\bullet]$ lies in convex hull of product distributions with bit error rates O(p).
- Standard fault-tolerance techniques achieve this bound…
- Problem! Mixing Lemma requires distribution over $\overset{\Omega_2}{\{I,X,Y,Z\}^n}$, whereas we have a distribution over error equivalence classes $\underset{\Omega_1}{\{I,X,Y,Z\}^n}$ / Stabilizers
- Solution:
  - Define $f : \Omega_2 \rightarrow \Omega_1$ mapping error to its equivalence class
  - f induces map on distributions, $f(\sigma)(\omega_1) = \sum_{\omega_2 : f(\omega_2) = \omega_1} \sigma(\omega_2)$ taking product distn's to bitwise-independent error distn's

Applying the Mixing Lemma

- Corollary: If **P**[{y≤x}]=$\Theta(p^{|x|})$ for all x in {0,1}^{n}, then **P**[•] lies in convex hull of product distributions with bit error rates O(p).
- Standard fault-tolerance techniques achieve this bound…
- Problem! Mixing Lemma requires distribution over $\overset{\Omega_2}{\{I,X,Y,Z\}^n}$, whereas we have a distribution over error equivalence classes $\underset{\Omega_1}{\{I,X,Y,Z\}^n / \text{Stabilizers}}$
- Solution:
  - Define $f : \Omega_2 \longrightarrow \Omega_1$ mapping error to its equivalence class
  - f induces map on distributions, $f(\sigma)(\omega_1) = \sum_{\omega_2 : f(\omega_2) = \omega_1} \sigma(\omega_2)$ taking product distn's to bitwise-independent error distn's
  - Pull back π: Choose ρ such that $f(\rho) = \pi$
    - E.g., divide probability mass on an error equivalence class equally among all minimum-weight representatives in Ω_2
  - Mixing Lemma tells us if $\rho = \sum p_i \rho_i$, implying $\pi = \sum_i p_i f(\rho_i)$



Remark on applying the Mixing Lemma

- Corollary: If **P**[{y≤x}]=Θ($p^{|x|}$) for all x in {0,1}^{n}, then **P**[•] lies in convex hull of product distrib error rates O(p).
- Standard this bound…
- Proble over {I,X,Y,Z}^{n}, whereas we have a classes {I,X,Y,Z}^{n} / Stabilizers Ω_2 Ω_1
- Solution:

Can be generalized to different maps f, different spaces Ω_1, different event independence constraints...

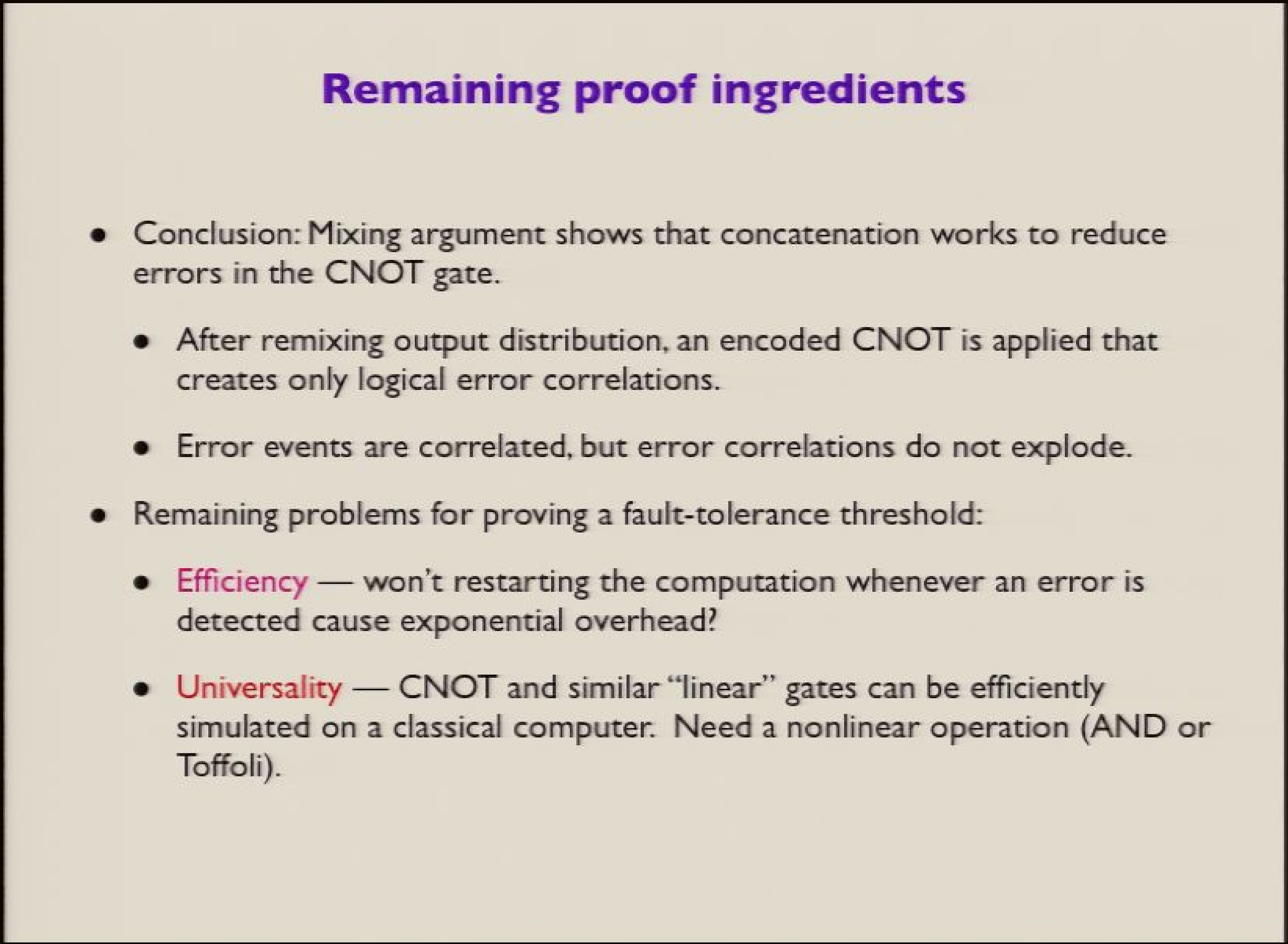
- Define f :
- f induces map on di $\omega_2)$ taking product distn's to bitwise-independent error distn's

E.g., **P**[{y≤x}]=Θ($p^{\min\{3,\max\{|x|_X,|x|_Z\}\}}$)

- Pull back π: Choose ρ such that $f(\rho) = \pi$
  - E.g., divide probability mass on an error equivalence class equally among all minimum-weight representatives in Ω_2
- Mixing Lemma tells us if $\rho = \sum p_i \rho_i$, implying $\pi = \sum_i p_i f(\rho_i)$

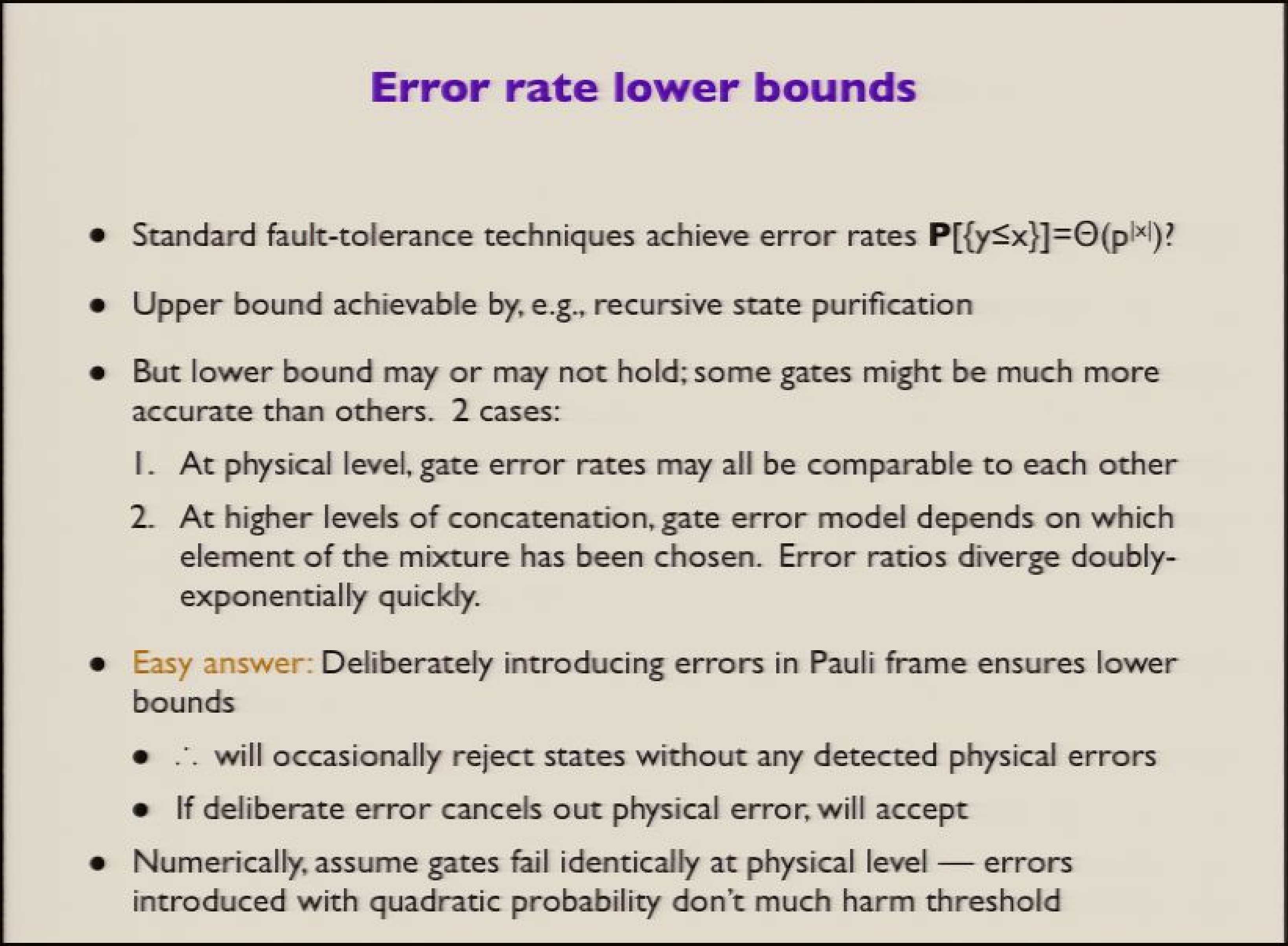
Error rate lower bounds

- Standard fault-tolerance techniques achieve error rates **P**$[\{y \leq x\}] = \Theta(p^{|x|})$?
- Upper bound achievable by, e.g., recursive state purification
- But lower bound may or may not hold; some gates might be much more accurate than others. 2 cases:
  1. At physical level, gate error rates may all be comparable to each other
  2. At higher levels of concatenation, gate error model depends on which element of the mixture has been chosen. Error ratios diverge doubly-exponentially quickly.
- Easy answer: Deliberately introducing errors in Pauli frame ensures lower bounds
  - $\therefore$ will occasionally reject states without any detected physical errors
  - If deliberate error cancels out physical error, will accept
- Numerically, assume gates fail identically at physical level — errors introduced with quadratic probability don't much harm threshold



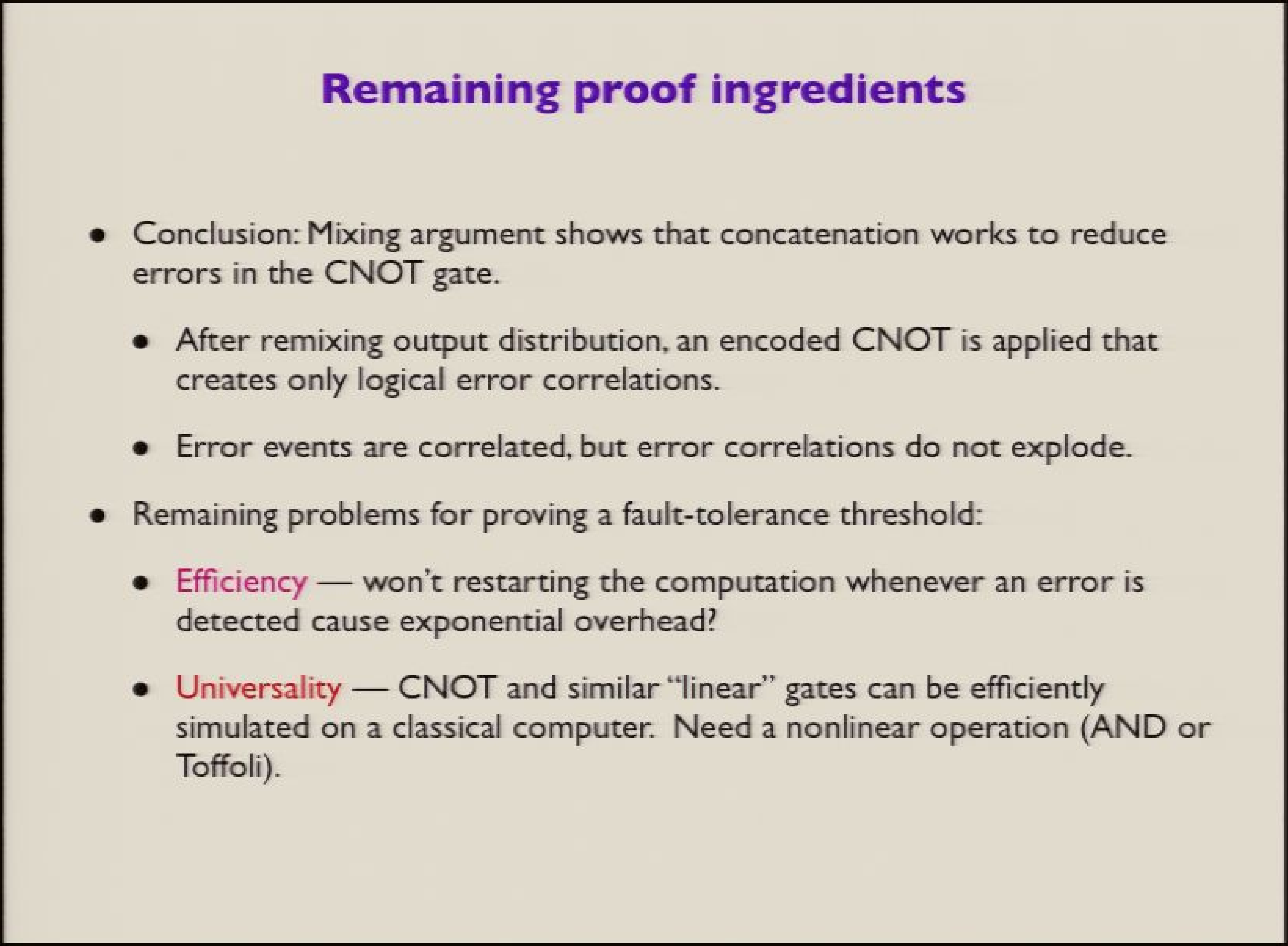
Remaining proof ingredients

- Conclusion: Mixing argument shows that concatenation works to reduce errors in the CNOT gate.
  - After remixing output distribution, an encoded CNOT is applied that creates only logical error correlations.
  - Error events are correlated, but error correlations do not explode.
- Remaining problems for proving a fault-tolerance threshold:
  - Efficiency — won't restarting the computation whenever an error is detected cause exponential overhead?
  - Universality — CNOT and similar "linear" gates can be efficiently simulated on a classical computer. Need a nonlinear operation (AND or Toffoli).



Error rate lower bounds

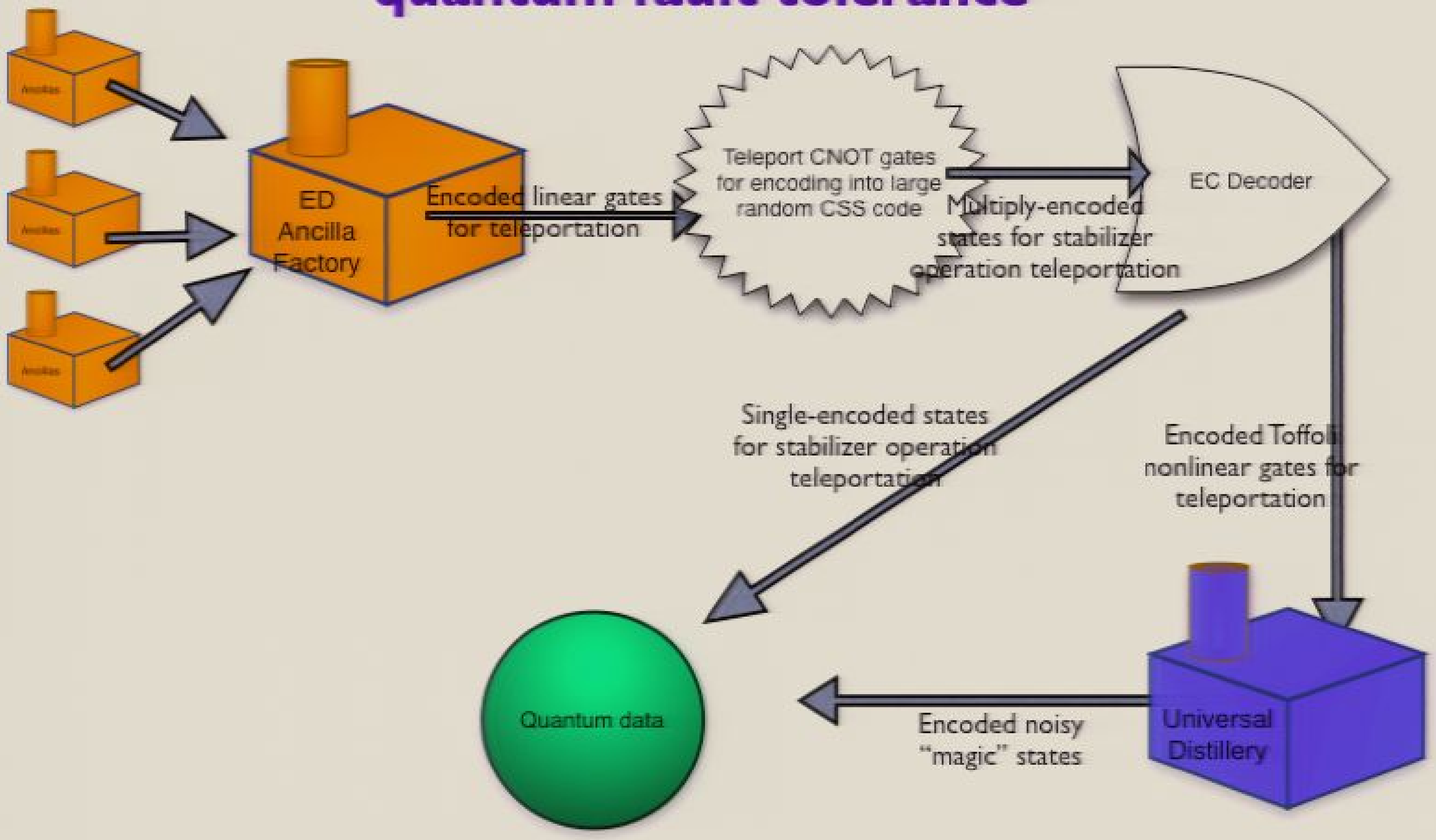
- Standard fault-tolerance techniques achieve error rates **P**[{y≤x}]=$\Theta(p^{|x|})$?
- Upper bound achievable by, e.g., recursive state purification
- But lower bound may or may not hold; some gates might be much more accurate than others. 2 cases:
  1. At physical level, gate error rates may all be comparable to each other
  2. At higher levels of concatenation, gate error model depends on which element of the mixture has been chosen. Error ratios diverge doubly-exponentially quickly.
- Easy answer: Deliberately introducing errors in Pauli frame ensures lower bounds
  - $\therefore$ will occasionally reject states without any detected physical errors
  - If deliberate error cancels out physical error, will accept
- Numerically, assume gates fail identically at physical level — errors introduced with quadratic probability don't much harm threshold



Remaining proof ingredients

- Conclusion: Mixing argument shows that concatenation works to reduce errors in the CNOT gate.
  - After remixing output distribution, an encoded CNOT is applied that creates only logical error correlations.
  - Error events are correlated, but error correlations do not explode.
- Remaining problems for proving a fault-tolerance threshold:
  - Efficiency — won't restarting the computation whenever an error is detected cause exponential overhead?
  - Universality — CNOT and similar "linear" gates can be efficiently simulated on a classical computer. Need a nonlinear operation (AND or Toffoli).

Error-detection/postselection-based quantum fault tolerance



ED Ancilla Factory

Encoded linear gates for teleportation

Teleport CNOT gates for encoding into large random CSS code

EC Decoder

Multiply-encoded states for stabilizer operation teleportation

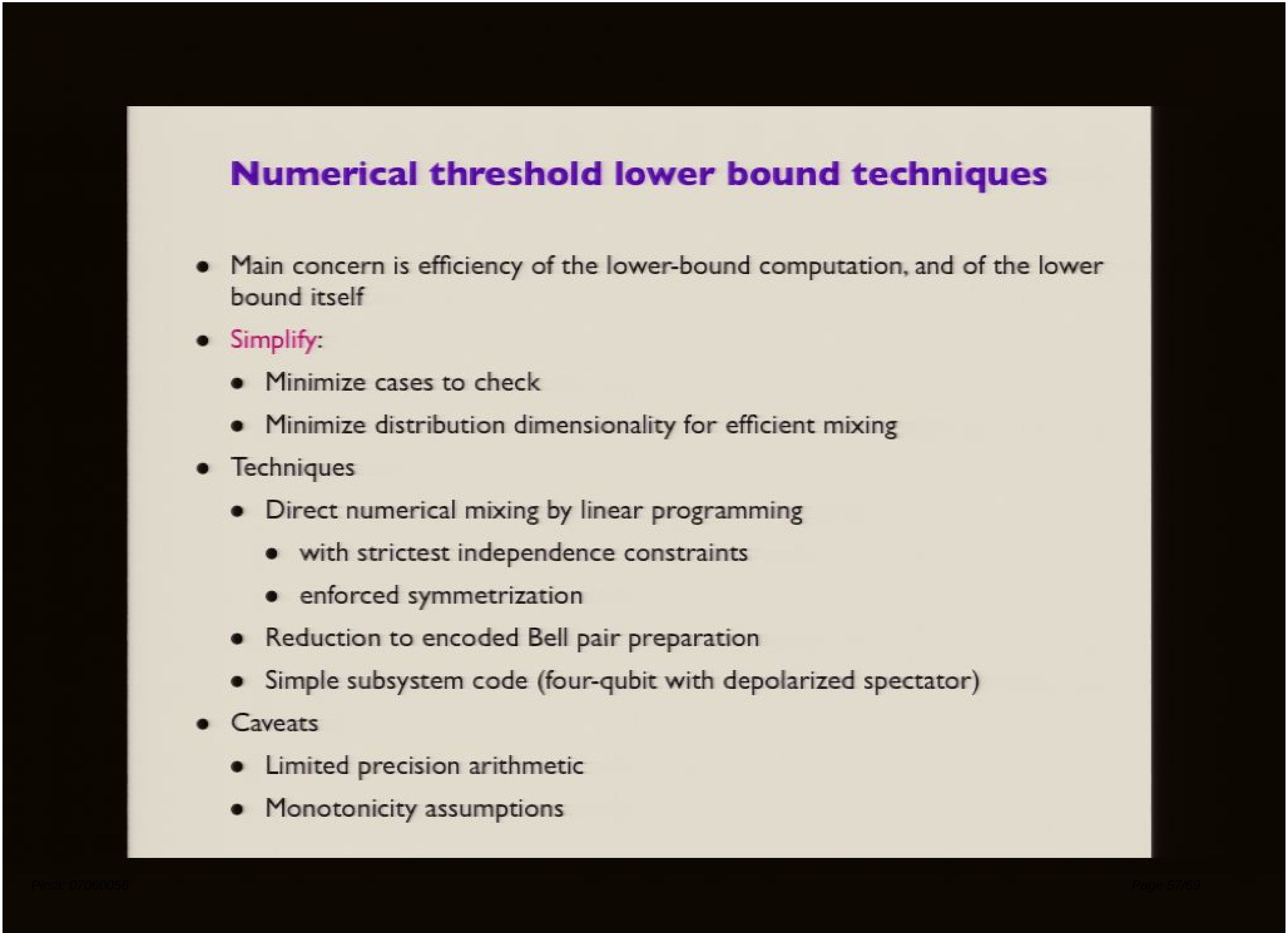
Single-encoded states for stabilizer operation teleportation

Encoded Toffoli nonlinear gates for teleportation

Universal Distillery

Encoded noisy “magic” states

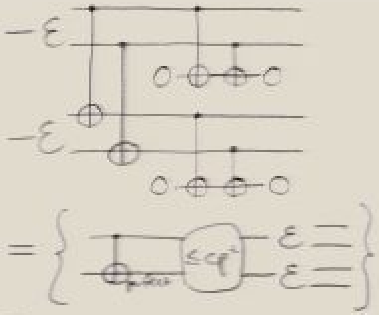
Quantum data



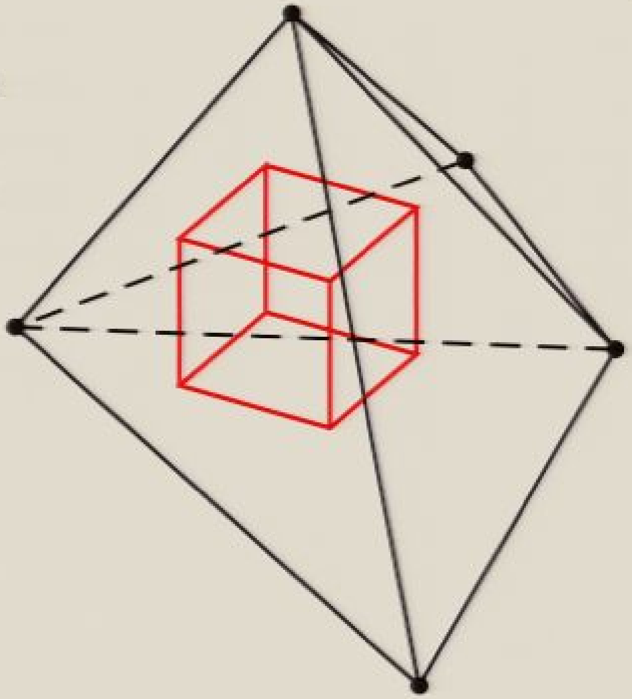
Numerical threshold lower bound techniques

- Main concern is efficiency of the lower-bound computation, and of the lower bound itself
- Simplify:
  - Minimize cases to check
  - Minimize distribution dimensionality for efficient mixing
- Techniques
  - Direct numerical mixing by linear programming
    - with strictest independence constraints
    - enforced symmetrization
  - Reduction to encoded Bell pair preparation
  - Simple subsystem code (four-qubit with depolarized spectator)
- Caveats
  - Limited precision arithmetic
  - Monotonicity assumptions

Direct numerical mixing

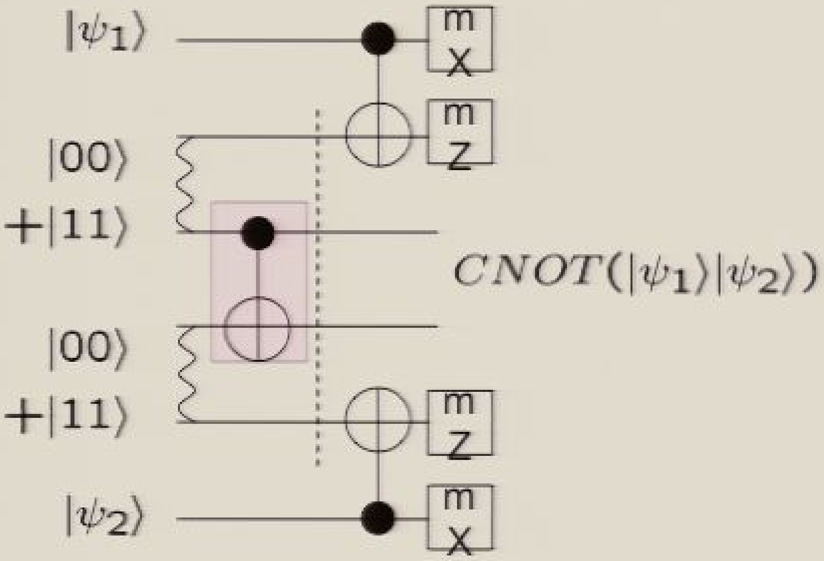


2. Numerically: We are given upper and lower bounds for each coordinate of the distribution… So use a linear program to check that each vertex of the hypercube lies in the convex hull of extremal "nice" distributions. (Computationally expensive in high dimensions.)



Reduction to encoded Bell pair preparation: Teleporting a CNOT gate

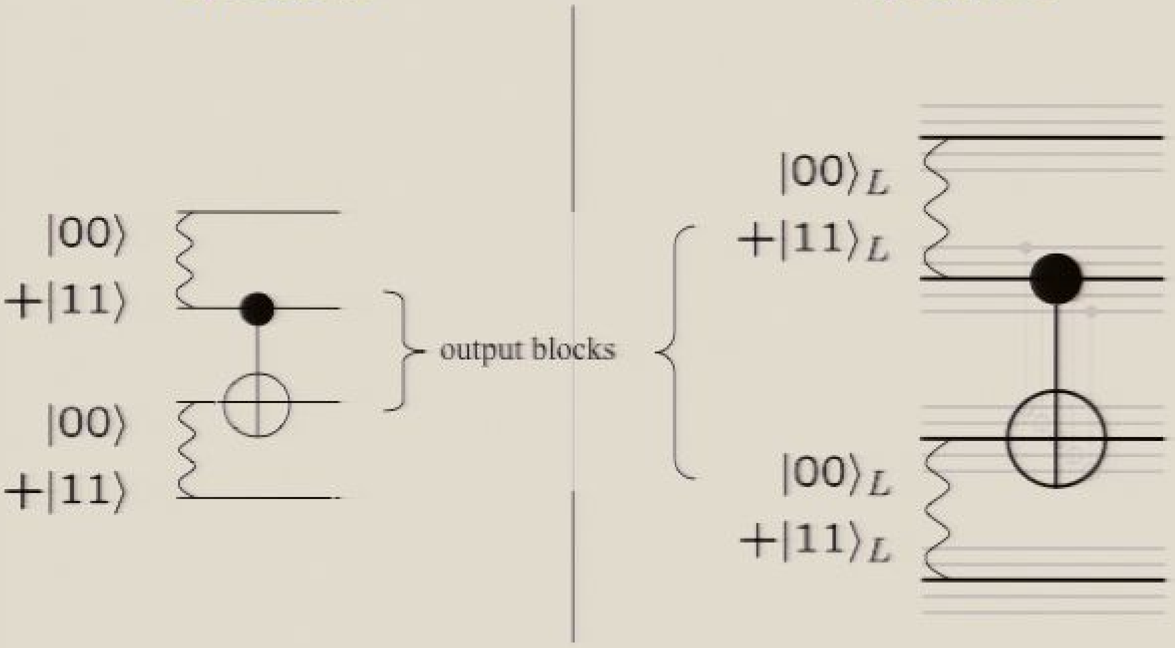
Logical operations


$|\psi_1\rangle$
$|00\rangle + |11\rangle$
$|00\rangle + |11\rangle$
$|\psi_2\rangle$
m X
m Z
m Z
m X
$CNOT(|\psi_1\rangle|\psi_2\rangle)$

Teleporting a CNOT gate

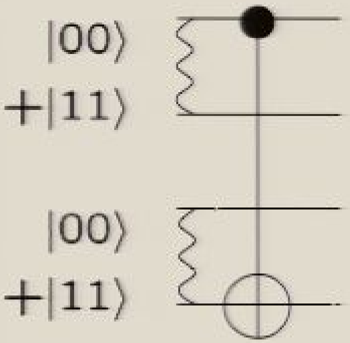
Logical operations

Physical operations

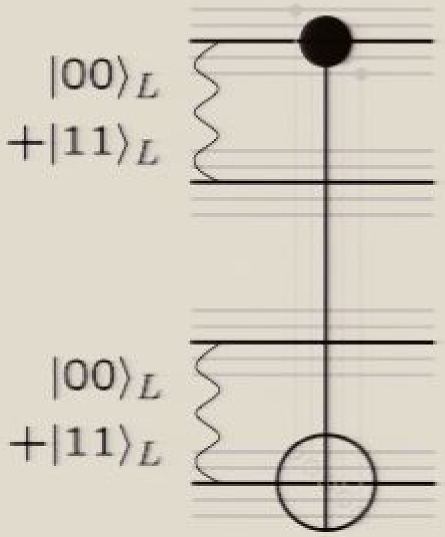


Teleporting a CNOT gate

Logical operations



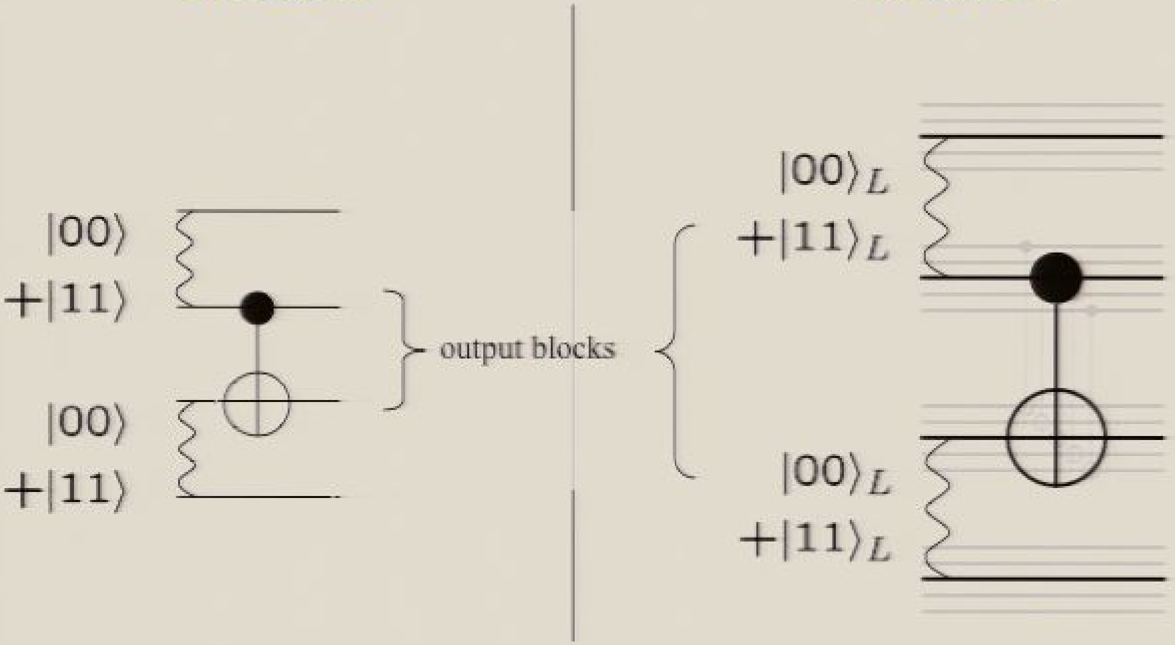
Physical operations



Teleporting a CNOT gate

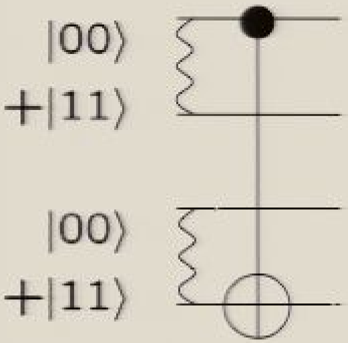
Logical operations

Physical operations

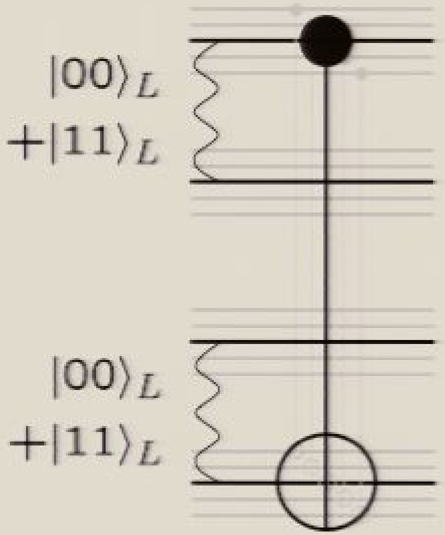


Teleporting a CNOT gate

Logical operations

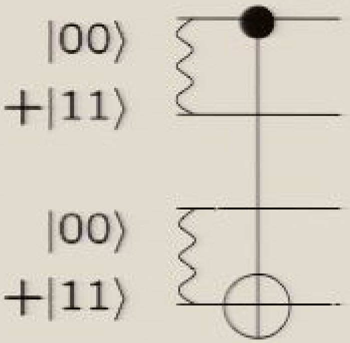


Physical operations

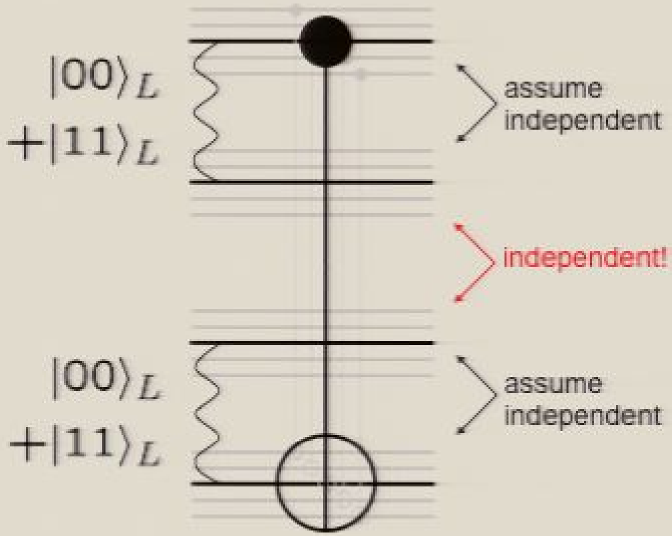


Teleporting a CNOT gate

Logical operations

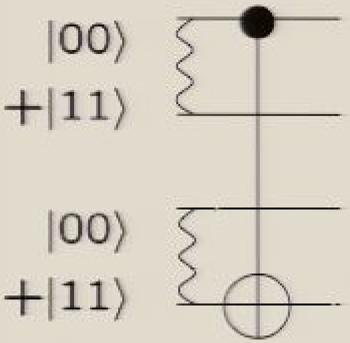


Physical operations

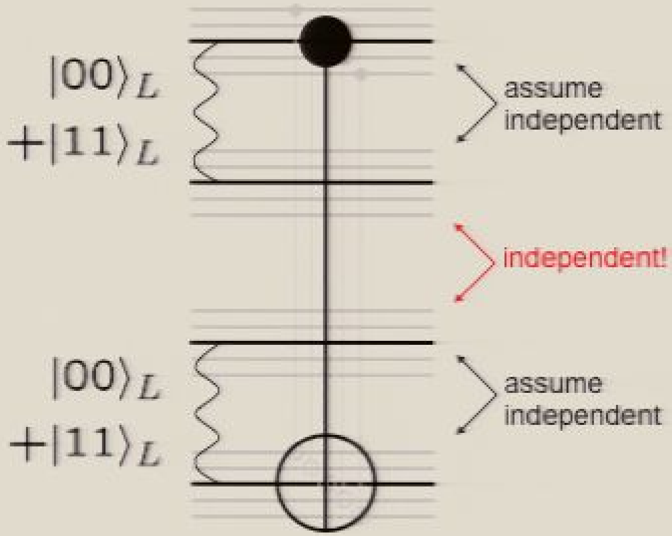


Teleporting a CNOT gate

Logical operations



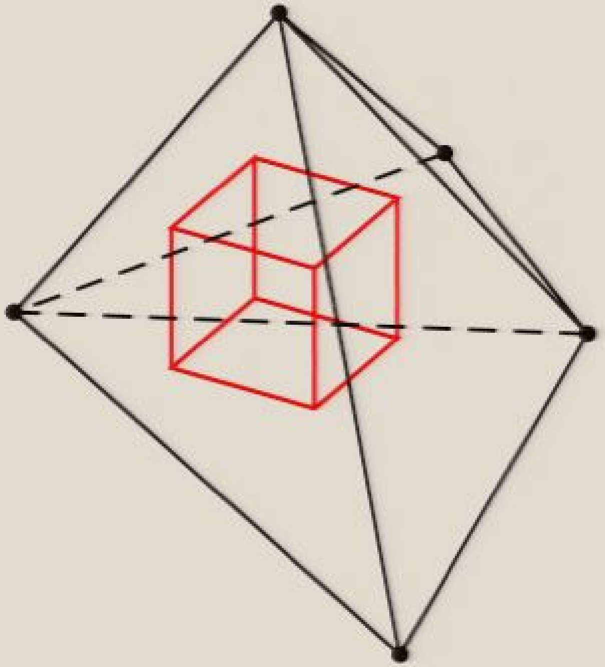
Physical operations

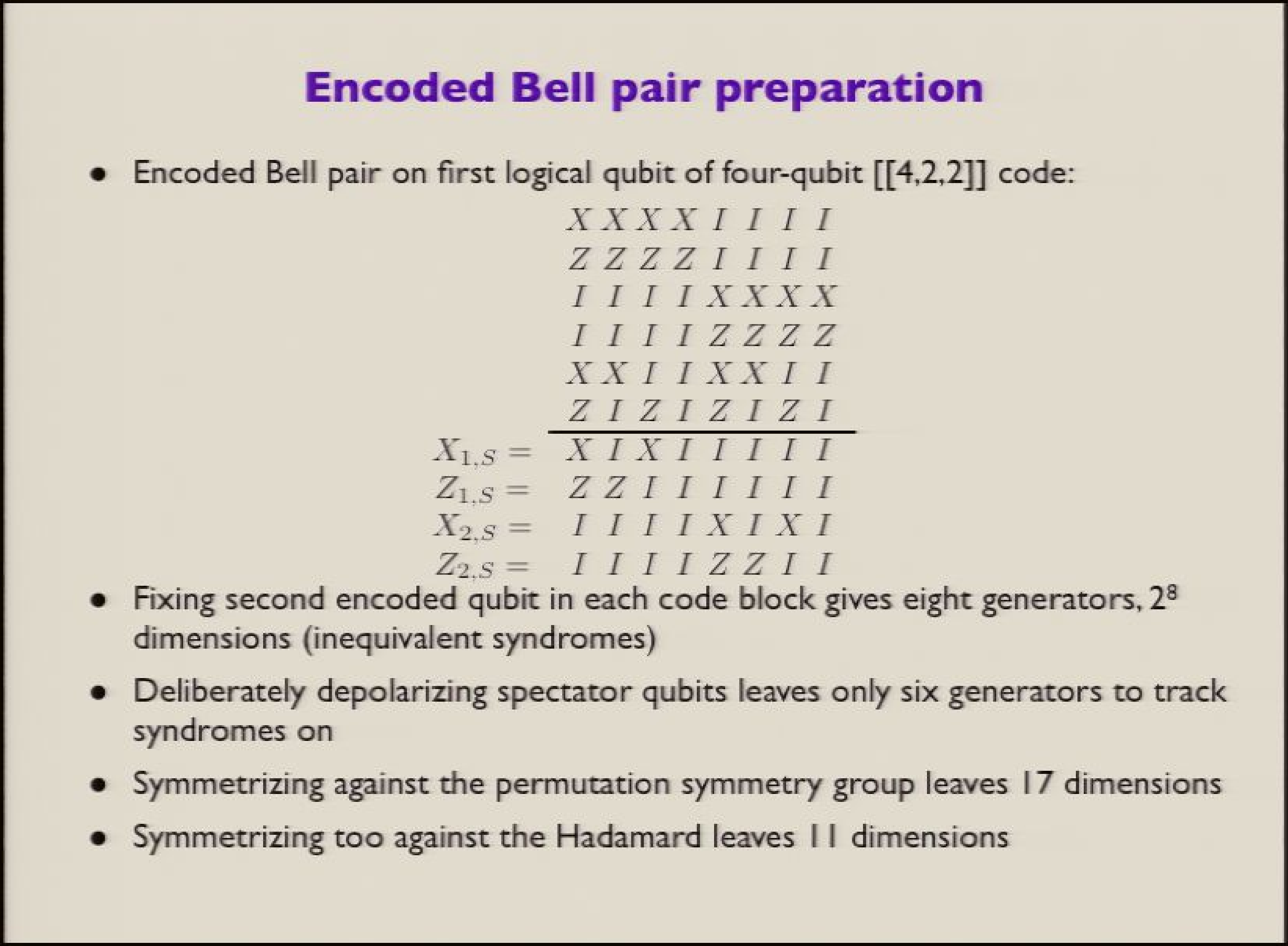


⇒ Achieving independent errors on CNOT output blocks reduces to preparing encoded Bell states with block-independent errors

Encoded Bell pair preparation in [[4,2,2]] code

- Encoded CNOT teleportation state has 4*4=16 qubits, 2^{16} dimensions
- Reducing to Bell pair preparation: 8 qubits, 2^{8} dimensions
  - Far too many for direct brute force numerical mixing: 2^{256} vertices to check!



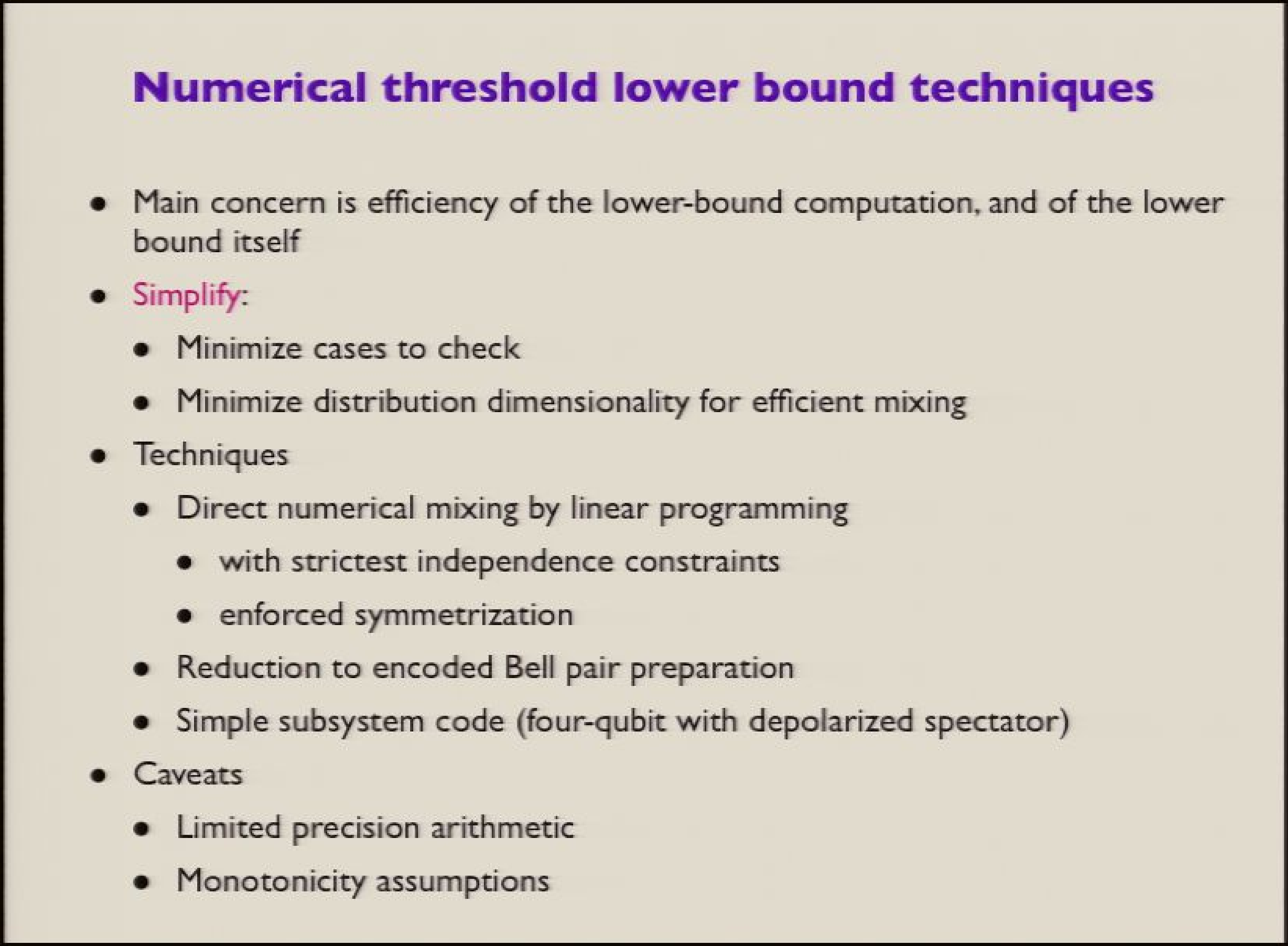


Encoded Bell pair preparation

- Encoded Bell pair on first logical qubit of four-qubit [[4,2,2]] code:

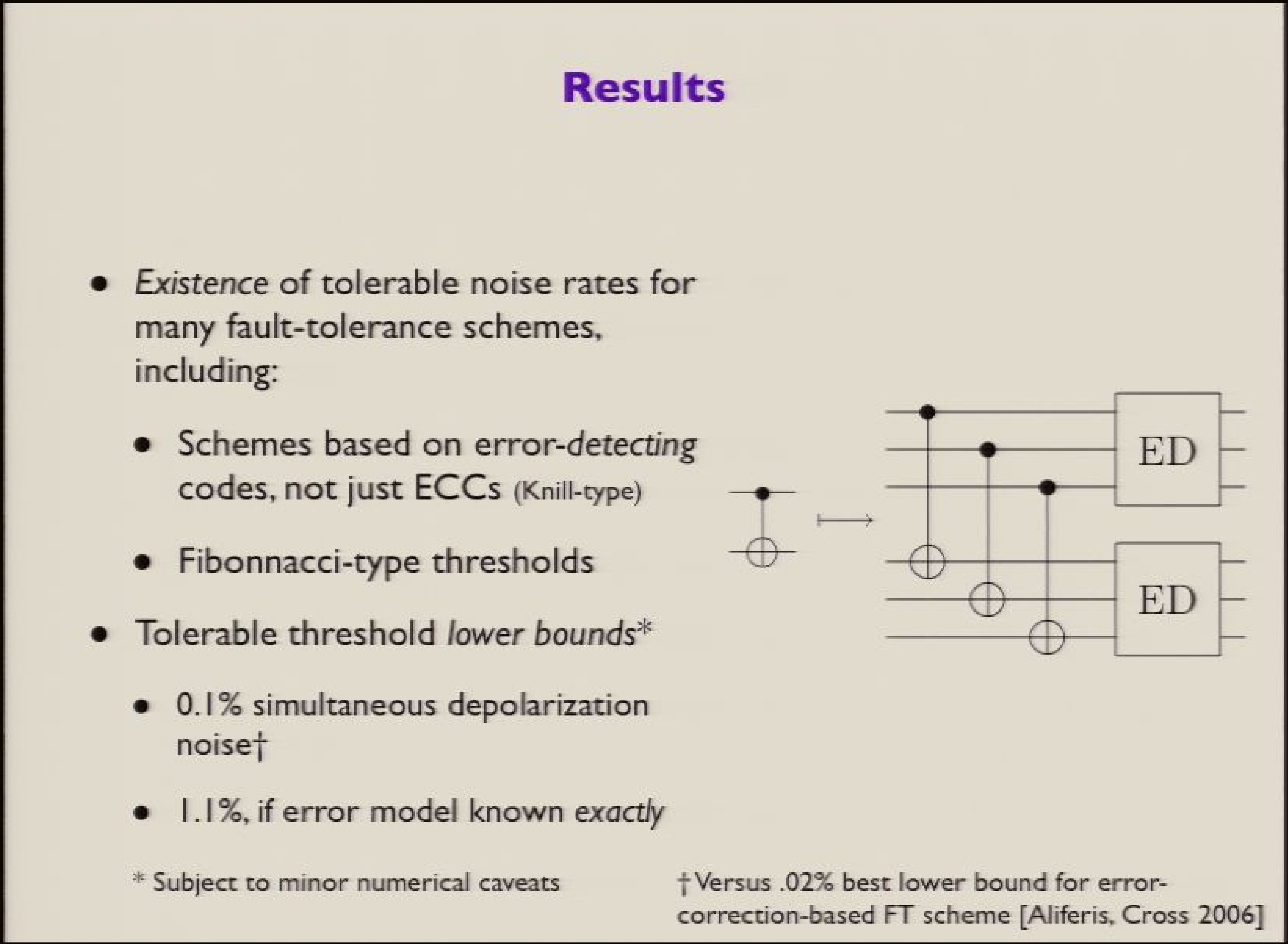
$$\begin{array}{rcccccccc}
 & X & X & X & X & I & I & I & I \\
 & Z & Z & Z & Z & I & I & I & I \\
 & I & I & I & I & X & X & X & X \\
 & I & I & I & I & Z & Z & Z & Z \\
 & X & X & I & I & X & X & I & I \\
 & Z & I & Z & I & Z & I & Z & I \\
\cline{2-9}
X_{1,S} = & X & I & X & I & I & I & I & I \\
Z_{1,S} = & Z & Z & I & I & I & I & I & I \\
X_{2,S} = & I & I & I & I & X & I & X & I \\
Z_{2,S} = & I & I & I & I & Z & Z & I & I
\end{array}$$

- Fixing second encoded qubit in each code block gives eight generators, 2^8 dimensions (inequivalent syndromes)
- Deliberately depolarizing spectator qubits leaves only six generators to track syndromes on
- Symmetrizing against the permutation symmetry group leaves 17 dimensions
- Symmetrizing too against the Hadamard leaves 11 dimensions



Numerical threshold lower bound techniques

- Main concern is efficiency of the lower-bound computation, and of the lower bound itself
- Simplify:
  - Minimize cases to check
  - Minimize distribution dimensionality for efficient mixing
- Techniques
  - Direct numerical mixing by linear programming
    - with strictest independence constraints
    - enforced symmetrization
  - Reduction to encoded Bell pair preparation
  - Simple subsystem code (four-qubit with depolarized spectator)
- Caveats
  - Limited precision arithmetic
  - Monotonicity assumptions



Results

- *Existence* of tolerable noise rates for many fault-tolerance schemes, including:
  - Schemes based on error-*detecting* codes, not just ECCs (Knill-type)
  - Fibonnacci-type thresholds
- Tolerable threshold *lower bounds**
  - 0.1% simultaneous depolarization noise†
  - 1.1%, if error model known *exactly*

* Subject to minor numerical caveats

† Versus .02% best lower bound for error-correction-based FT scheme [Aliferis, Cross 2006]