Fault-tolerant Quantum Computing

Bryan Eastin

Northrop Grumman Corporation
Aurora, CO

December 2014

What do we mean by *quantum computer*?

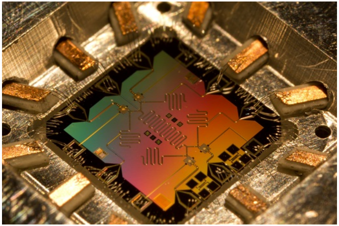
NORTHROP GRUMMAN

Quantum computer properties (in theory)

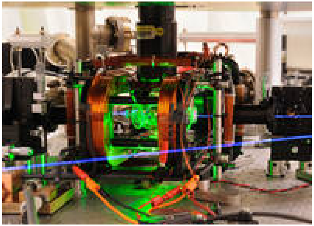
1. General purpose - Not limited to a single class of problems. Universal.
2. Accurate - The probability of an error on the output can be made arbitrarily small.
3. Scalable - Resource requirements do not grow exponentially in the size or target error probability of the computation.

The goal of fault-tolerant quantum computing is to achieve these properties in an imperfect device.

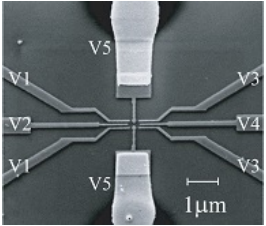
Lucero



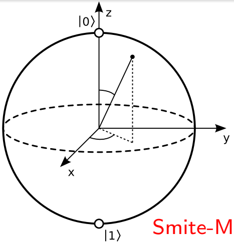
Colombe



Chang



NORTHROP GRUMMAN



Smite-Meister

Qubit Quantum bit, i.e., a two-state quantum system.

$$\alpha|0\rangle + \beta|1\rangle \qquad \text{where } |\alpha|^2 + |\beta|^2 = 1$$

Gate Discrete operator, typically unitary, e.g.

- The Pauli operators

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad Y = \begin{bmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- Other single-qubit rotations

$$H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad Z\left(\frac{\pi}{2}\right) \cong S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \qquad Z\left(\frac{\pi}{4}\right) \cong T = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{\mathrm{i}\pi}{4}} \end{bmatrix}$$

- Multi-qubit unitary operators

$$^{C}X = \begin{bmatrix} I & 0 \\ 0 & X \end{bmatrix} \qquad ^{CC}X = \mathtt{TOFFOLI} = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & X \end{bmatrix}$$

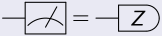
- Measurement

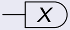
M_Z = Measure in Z eigenbasis $\qquad M_X$ = Measure in X eigenbasis

Quantum circuit diagrams

Circuit diagrams used in this talk

- Measurement

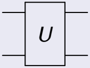
  $M_Z =$ 

  $M_X =$ 

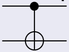
- Single-qubit unitaries

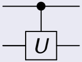
  $U =$ 

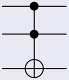
- Multi-qubit unitaries

  $U =$ 

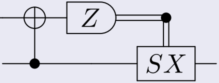
- More multi-qubit unitaries

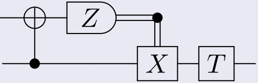
  ${}^{C}X =$  (controlled-NOT)

  ${}^{C}U =$ 

  ${}^{CC}X =$  (TOFFOLI)

Example quantum circuit identity

$|\pi/4\rangle$ 

$=$
$|+\rangle$ 

Pauli and Clifford groups

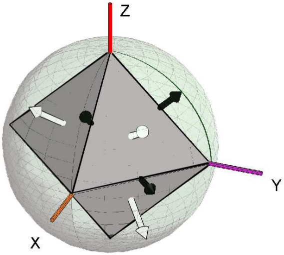
Pauli product A tensor product of Pauli operators, e.g.,
$X \otimes Y \otimes Z \otimes I$ or $XYZI$ or $X_1 Y_2 Z_3 I_4$.

Pauli group The group of all Pauli products of a given length augmented by $\{\pm 1, \pm \mathrm{i}\}$.

Clifford group The group of unitary gates that preserves the Pauli group under conjugation.
Includes X, Y, Z, H, S, and ${}^{C}X$.

Clifford gate A gate that can be decomposed into unitary gates from the Clifford group along with measurement and preparation in the fiducial basis.

Stabilizer state A state constructible using only probabilistic Clifford gates.
A.K.A. Clifford state.



Dam 0907.3189

Clifford gates are classically simulable

Gottesman-Knill Theorem Gottesman quant-ph/9705052

Any quantum computation composed exclusively of Clifford gates can be efficiently simulated using a classical computer.

Sketch: The computer is always in the $+1$ eigenstate of a complete set of commuting Pauli products, so the Clifford gates act simply in the Heisenberg picture.

Clifford gates can generate arbitrary amounts of entanglement but are computationally weak.

Additional quantum operations are needed to enable quantum speedups.

Universality

Universal Capable of implementing any operation allowed by quantum mechanics with arbitrarily high precision.

H, T, and ${}^{C}X$ make up a universal set of unitary gates

- Any unitary operator can be decomposed into single-qubit unitaries and ${}^{C}X$ gates.
- H and T can be used to generate irrational rotations about two axes of the bloch sphere.
- Any single-qubit unitary can be approximated using these irrational rotations (efficiently, see Solovay-Kitaev)

Augmenting the Clifford gates by any non-Clifford unitary gate allows for efficient universal quantum computing.

The Toffoli and Fredkin gates and T, the $\pi/4$ Z rotation, are not Clifford gates.

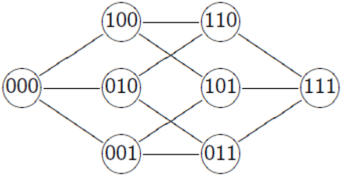
Quantum error correction

Classical repetition code, $\mathscr{R}_3$:

$$000, 111$$

Quantum repetition code, $\mathscr{R}_3$:

$$\alpha|000\rangle + \beta|111\rangle$$



Quantum data cannot be directly inspected for error.

$$\alpha|001\rangle + \beta|110\rangle \xrightarrow{M_{Z_1}} |001\rangle \text{ or } |110\rangle$$

Errors are continuous.

$$(\sqrt{1-\delta^2}I + i\delta X_1)|000\rangle = \sqrt{1-\delta^2}|000\rangle + i\delta|100\rangle$$

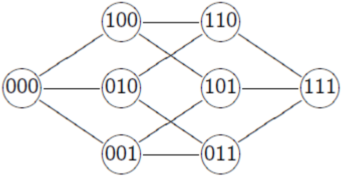
Quantum error correction

Classical repetition code, $\mathscr{R}_3$:

$$000, 111$$

Quantum repetition code, $\mathscr{R}_3$:

$$\alpha|000\rangle + \beta|111\rangle$$



Quantum data cannot be directly inspected for error.

$$\alpha|001\rangle + \beta|110\rangle \xrightarrow{M_{Z_1}} |001\rangle \text{ or } |110\rangle$$

Measure non-local check operators: $Z_1 Z_2 \to 1$, $Z_2 Z_3 \to -1$.

Errors are continuous.

$$(\sqrt{1-\delta^2} I + i\delta X_1)|000\rangle = \sqrt{1-\delta^2}|000\rangle + i\delta|100\rangle$$

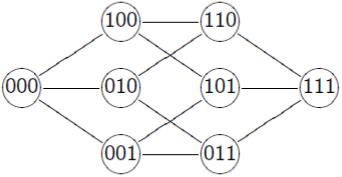
Quantum error correction

Classical repetition code, $\mathscr{R}_3$:

$$000, 111$$

Quantum repetition code, $\mathscr{R}_3$:

$$\alpha|000\rangle + \beta|111\rangle$$



Quantum data cannot be directly inspected for error.

$$\alpha|001\rangle + \beta|110\rangle \xrightarrow{M_{Z_1}} |001\rangle \text{ or } |110\rangle$$

Measure non-local check operators: $Z_1 Z_2 \to 1$, $Z_2 Z_3 \to -1$.

Syndrome Measurement outcomes for a set of check operators.

Syndrome decoding Inferring the location of the errors from the syndrome.

Errors are continuous.

$$(\sqrt{1-\delta^2} I + i\delta X_1)|000\rangle = \sqrt{1-\delta^2}|000\rangle + i\delta|100\rangle$$

Quantum error correction

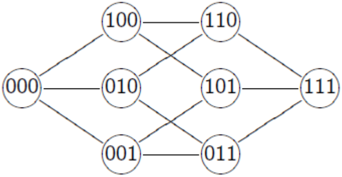
NORTHROP GRUMMAN

Classical repetition code, $\mathscr{R}_3$:

$$000, 111$$

Quantum repetition code, $\mathscr{R}_3$:

$$\alpha|000\rangle + \beta|111\rangle$$



Quantum data cannot be directly inspected for error.

$$\alpha|001\rangle + \beta|110\rangle \xrightarrow{M_{Z_1}} |001\rangle \text{ or } |110\rangle$$

Measure non-local check operators: $Z_1Z_2 \to 1$, $Z_2Z_3 \to -1$.

Syndrome Measurement outcomes for a set of check operators.

Syndrome decoding Inferring the location of the errors from the syndrome.

Errors are continuous.

$$(\sqrt{1-\delta^2}I + i\delta X_1)|000\rangle = \sqrt{1-\delta^2}|000\rangle + i\delta|100\rangle$$

Use linearity of quantum mechanics, correct a basis, e.g. X, Y, and Z.

Stabilizer codes

Stabilizer Commuting group of Pauli products each of which square to the identity, e.g., II, XX, $-YY$, and ZZ

Stabilizer state $+1$ eigenstate of some stabilizer or a mixture thereof

Stabilizer generator Set of Pauli products that generate a stabilizer under multiplication, e.g., XX and ZZ

Stabilizer code Code whose check operators can be chosen to be a stabilizer generator

If A stabilizes $|\Psi\rangle$, $\langle\Psi|E^\dagger AE|\Psi\rangle = -1$ for any error E s.t. $AE = -EA$.

Four-qubit error-detecting code

stabilizer generator $= \begin{bmatrix} X \otimes X \otimes X \otimes X \\ Z \otimes Z \otimes Z \otimes Z \end{bmatrix}$

$$\bar{X}_1 = X \otimes X \otimes I \otimes I$$
$$\bar{Z}_1 = Z \otimes I \otimes I \otimes Z$$
$$\bar{X}_2 = X \otimes I \otimes I \otimes X$$
$$\bar{Z}_2 = Z \otimes Z \otimes I \otimes I$$

Minimum distance The minimum size (in number of qubits affected) of an undetectable (nontrivial) error, denoted d.

CSS (Calderbank-Shor-Steane) codes

CSS code Code where the stabilizer generators can be chosen as either X-type or Z-type Pauli products

Symmetric CSS code CSS code which is symmetric under exchange of X and Z

CSS codes can be constructed from certain pairs of classical codes.

For symmetric CSS codes, qubit-wise application of X, Y, Z, H, ${}^{C}X$, M_X, and M_Z are encoded operations.

Seven-qubit Steane error-correcting code

X-type stabilizer generator
$$= \begin{bmatrix} X\,I\,X\,I\,X\,I\,X \\ I\,X\,X\,I\,I\,X\,X \\ I\,I\,I\,X\,X\,X\,X \end{bmatrix}$$

$$\bar{X} = XXXXXXX$$
$$\bar{Z} = ZZZZZZZ$$

A code with minimum distance d can correct errors on any $\left\lfloor \frac{(d-1)}{2} \right\rfloor$ qubits.

If errors E and F are indistinguishable, $E^\dagger A_i E = F^\dagger A_i F$ for all stabilizers A_i which implies $EF^\dagger$ is an undetectable error.

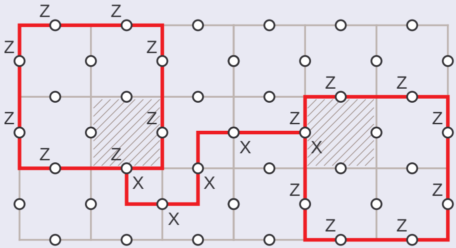
Additional types of quantum codes

Subsystem code Quantum code that encode more logical qubits than used

LDPC code Quantum code with low-weight stabilizer generators

Topological code Quantum code associated with a topology such that logical operators correspond to non-trivial topological features and stabilizer generators have local support

Kitaev's surface code Dennis quant-ph/0110143 Fowler 0803.0272

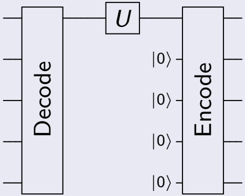


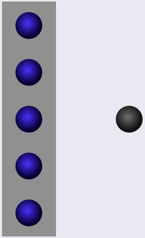
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $U S_i U^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

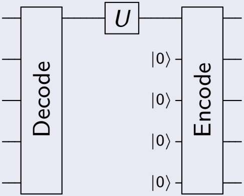
Generally, qubits in an encoded block should not interact.

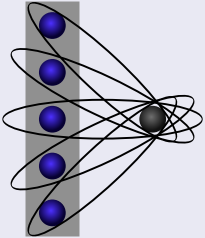
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $US_iU^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

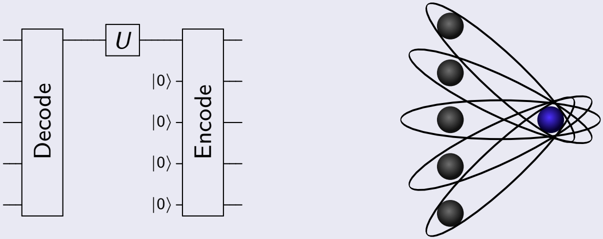
Generally, qubits in an encoded block should not interact.

Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $U S_i U^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate



Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

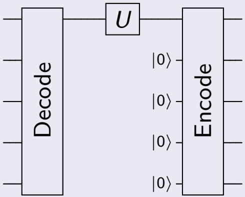
Generally, qubits in an encoded block should not interact.

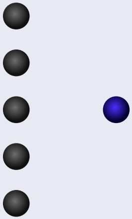
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $US_iU^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

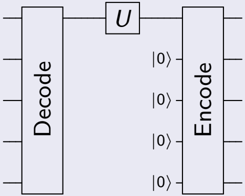
Generally, qubits in an encoded block should not interact.

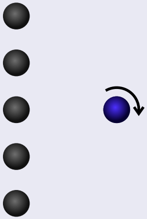
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $U S_i U^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

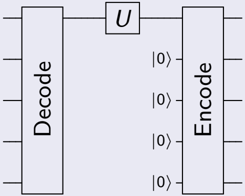
Generally, qubits in an encoded block should not interact.

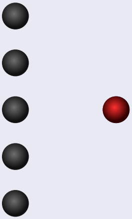
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $U S_i U^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

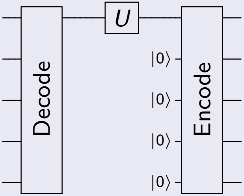
Generally, qubits in an encoded block should not interact.

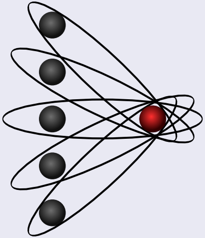
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $US_iU^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

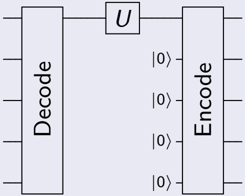
Generally, qubits in an encoded block should not interact.

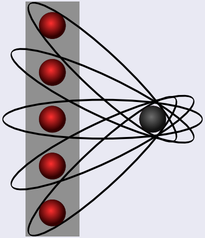
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $US_iU^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

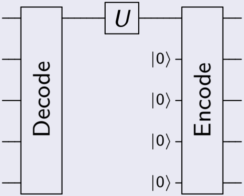
Generally, qubits in an encoded block should not interact.

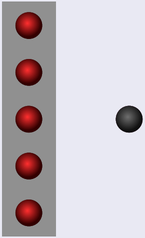
Encoded gates and fault tolerance

A unitary gate U is a valid encoded gate if $U \sum_i S_i U^\dagger = \sum_i S_i$, e.g., for any stabilizer S_i, $US_iU^\dagger$ is a stabilizer.

For unitary Clifford gates checking this and how the logical Pauli operators transform is easy.

Bad method of applying an encoded gate





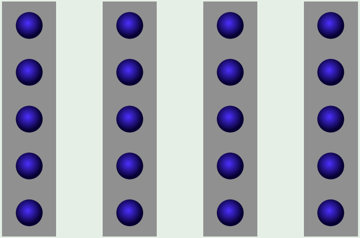
Code block A group of qubits that are error corrected as a unit

Fault tolerance A circuit is fault tolerant against t failures if failures in t elements results in at most t errors per code block.

Generally, qubits in an encoded block should not interact.

Transversal encoded gates

Code with four code blocks



Transversal gates are fault tolerant because each code block is corrected independently.

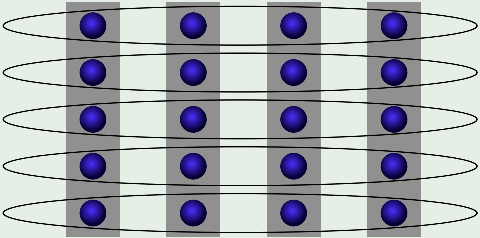
Eastin-Knill Theorem Eastin 0811.4262 (See also Zeng 0706.1382.)

No code capable of detecting single-qubit errors has a universal, transversal encoded unitary gate set.

Sketch: An infinitesimal, transversal logical unitary gate looks like a superposition of single-qubit errors.

Transversal encoded gates

Transversal partition of four code blocks



Transversal gates are fault tolerant because each code block is corrected independently.

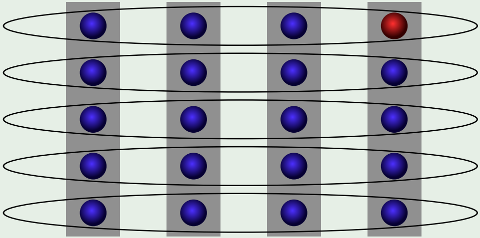
Eastin-Knill Theorem Eastin 0811.4262 (See also Zeng 0706.1382.)

No code capable of detecting single-qubit errors has a universal, transversal encoded unitary gate set.

Sketch: An infinitesimal, transversal logical unitary gate looks like a superposition of single-qubit errors.

Transversal encoded gates

Transversal partition of four code blocks



Transversal gates are fault tolerant because each code block is corrected independently.

Eastin-Knill Theorem Eastin 0811.4262 (See also Zeng 0706.1382.)

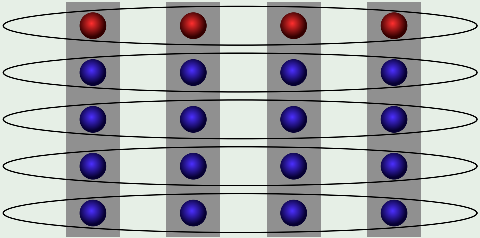
No code capable of detecting single-qubit errors has a universal, transversal encoded unitary gate set.

Sketch: An infinitesimal, transversal logical unitary gate looks like a superposition of single-qubit errors.

Transversal encoded gates

NORTHROP GRUMMAN

Transversal partition of four code blocks



Transversal gates are fault tolerant because each code block is corrected independently.

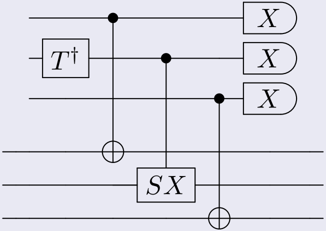
Eastin-Knill Theorem Eastin 0811.4262 (See also Zeng 0706.1382.)

No code capable of detecting single-qubit errors has a universal, transversal encoded unitary gate set.

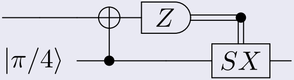
Sketch: An infinitesimal, transversal logical unitary gate looks like a superposition of single-qubit errors.

Techniques for achieving fault tolerance

Transversal gates

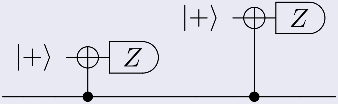


Ancillary states

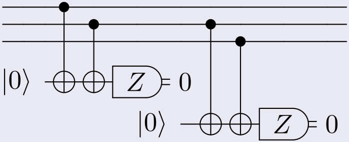


$$|\pi/4\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\pi/4}|1\rangle)$$

Repetitive measurement



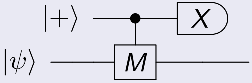
Discard



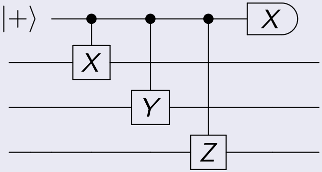
Measurement circuit

How do you perform coherent measurement of multiqubit observables?

Measuring M where $M^2 = 1$



Measuring $X_1 Y_2 Z_3$



Algebra

$$
\begin{aligned}
{}^{C}M_{12}|+\rangle|\psi\rangle &= {}^{C}M_{12}\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle|\psi\rangle + |1\rangle M_2|\psi\rangle) \\
&= \frac{1}{2}((|+\rangle + |-\rangle)|\psi\rangle + (|+\rangle - |-\rangle)M_2|\psi\rangle) \\
&= |+\rangle\frac{(I_2 + M_2)}{2}|\psi\rangle + |-\rangle\frac{(I_2 - M_2)}{2}|\psi\rangle
\end{aligned}
$$

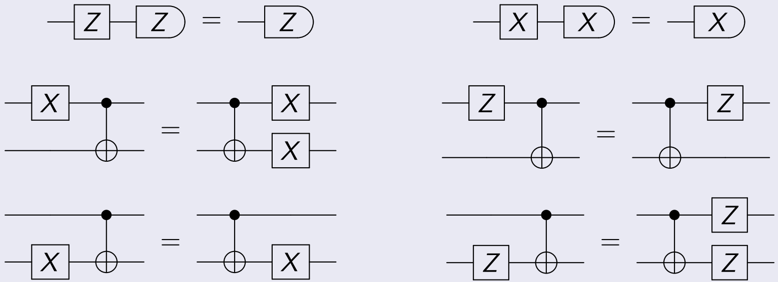
Frequently, measuring things in this way is not a good idea.

Circuit identities for quantum error correction

Error propagation is a valuable tool for understanding quantum error correction

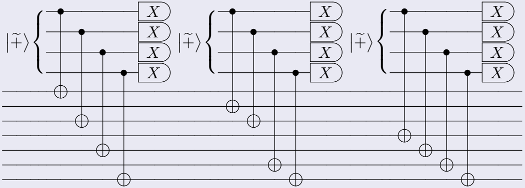
- Fault-tolerant error correction typically requires only Clifford gates
- Errors can be expanded in terms of Pauli products (and $Y = iZX$)
- Pauli products can be propagated through Clifford gates
- Logical errors correspond to certain Pauli products

Circuit identities used in this talk

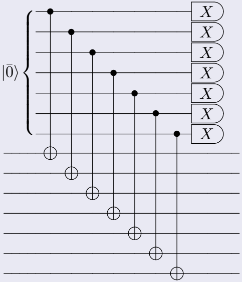


Approaches to fault-tolerant error correction

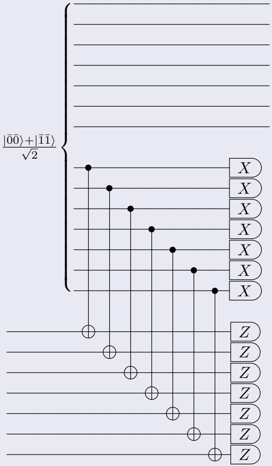
Shor Z-error correction (partial)



Steane Z-error correction



Knill X- & Z- error correction

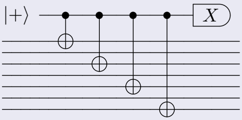


Shor-style error correction

Shor error correction Shor quant-ph/9605011

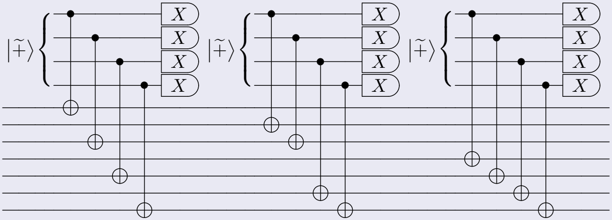
- Simple measurement of check operators
- Requires cat states
- Typically, FT procedures require between $t+1 = \left\lceil \frac{d}{2} \right\rceil$ and d repetitions
- Time per repetition scales like max number of check operators per qubit

Non-FT $X_1X_3X_5X_7$ measurement



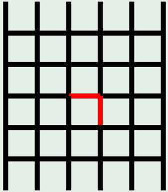
$$|+\rangle = \left(|0\rangle + |1\rangle\right)/\sqrt{2}$$

Shor Z-error correction



$$|\tilde{+}\rangle = \left(|0000\rangle + |1111\rangle\right)/\sqrt{2}$$

Exception: Surface code

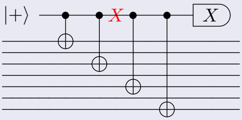


Shor-style error correction

Shor error correction Shor quant-ph/9605011

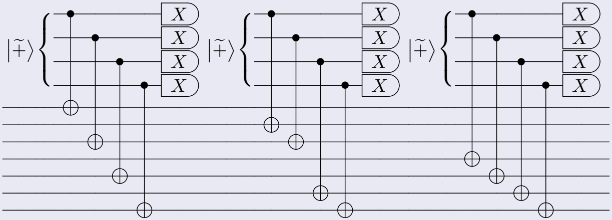
- Simple measurement of check operators
- Requires cat states
- Typically, FT procedures require between $t+1=\left\lceil \frac{d}{2} \right\rceil$ and d repetitions
- Time per repetition scales like max number of check operators per qubit

Non-FT $X_1X_3X_5X_7$ measurement



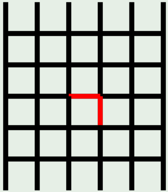
$$|+\rangle = \left(|0\rangle + |1\rangle\right)/\sqrt{2}$$

Shor Z-error correction



$$|\tilde{+}\rangle = \left(|0000\rangle + |1111\rangle\right)/\sqrt{2}$$

Exception: Surface code

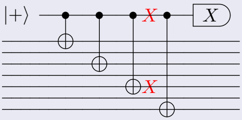


Shor-style error correction

Shor error correction Shor quant-ph/9605011

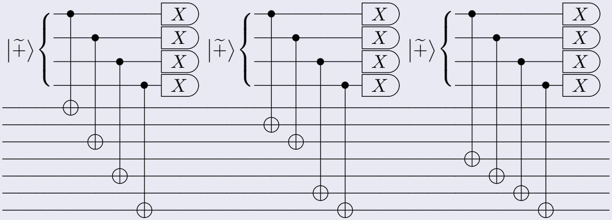
- Simple measurement of check operators
- Requires cat states
- Typically, FT procedures require between $t+1 = \left\lceil \frac{d}{2} \right\rceil$ and d repetitions
- Time per repetition scales like max number of check operators per qubit

Non-FT $X_1X_3X_5X_7$ measurement



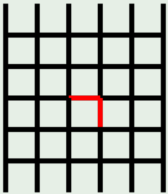
$$|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$$

Shor Z-error correction



$$|\tilde{+}\rangle = (|0000\rangle + |1111\rangle)/\sqrt{2}$$

Exception: Surface code

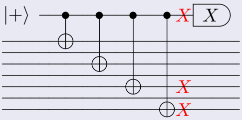


Shor-style error correction

Shor error correction Shor quant-ph/9605011

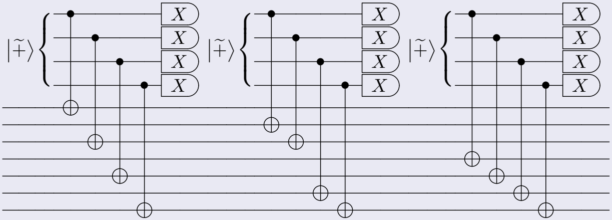
- Simple measurement of check operators
- Requires cat states
- Typically, FT procedures require between $t+1=\left\lceil \frac{d}{2} \right\rceil$ and d repetitions
- Time per repetition scales like max number of check operators per qubit

Non-FT $X_1X_3X_5X_7$ measurement



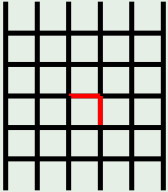
$$|+\rangle = \left(|0\rangle + |1\rangle\right)/\sqrt{2}$$

Shor Z-error correction



$$|\tilde{+}\rangle = \left(|0000\rangle + |1111\rangle\right)/\sqrt{2}$$

Exception: Surface code

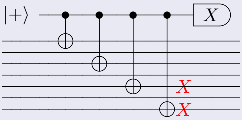


Shor-style error correction

Shor error correction Shor quant-ph/9605011

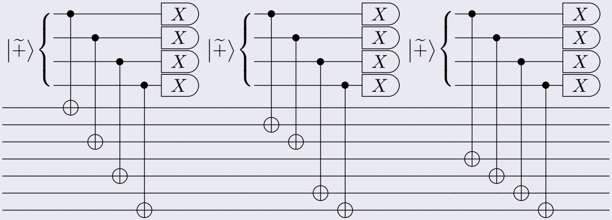
- Simple measurement of check operators
- Requires cat states
- Typically, FT procedures require between $t+1 = \left\lceil \frac{d}{2} \right\rceil$ and d repetitions
- Time per repetition scales like max number of check operators per qubit

Non-FT $X_1 X_3 X_5 X_7$ measurement



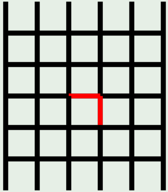
$$|+\rangle = \left(|0\rangle + |1\rangle\right)/\sqrt{2}$$

Shor Z-error correction



$$|\tilde{+}\rangle = \left(|0000\rangle + |1111\rangle\right)/\sqrt{2}$$

Exception: Surface code

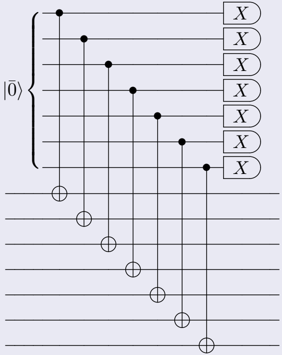


Steane-style error correction

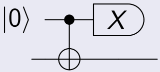
Steane error correction Steane quant-ph/9708021

- Trivial logical circuit
- Requires encoded $|0\rangle$ and $|+\rangle$ states
- Can be used with ancillae verified against one or both kinds of error
- For every X/Z correction
  - At least $t+1$ repetitions are required for partially verified ancillae
  - 1 coupling is sufficient for fully verified ancillae

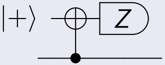
Steane Z-error correction



Logical circuit for Steane Z EC



Logical circuit for Steane X EC

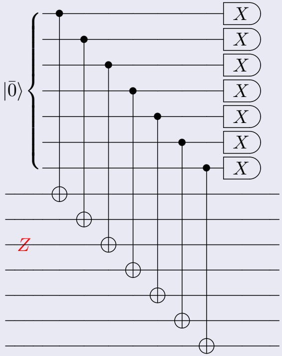


Steane-style error correction

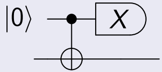
Steane error correction Steane quant-ph/9708021

- Trivial logical circuit
- Requires encoded $|0\rangle$ and $|+\rangle$ states
- Can be used with ancillae verified against one or both kinds of error
- For every X/Z correction
  - At least $t+1$ repetitions are required for partially verified ancillae
  - 1 coupling is sufficient for fully verified ancillae

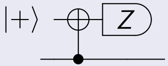
Steane Z-error correction



Logical circuit for Steane Z EC



Logical circuit for Steane X EC

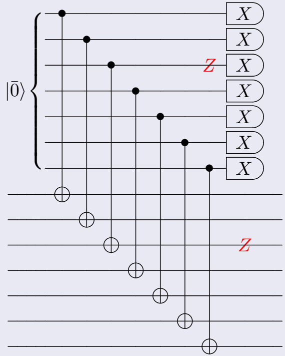


Steane-style error correction

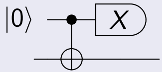
Steane error correction Steane quant-ph/9708021

- Trivial logical circuit
- Requires encoded $|0\rangle$ and $|+\rangle$ states
- Can be used with ancillae verified against one or both kinds of error
- For every X/Z correction
  - At least $t+1$ repetitions are required for partially verified ancillae
  - 1 coupling is sufficient for fully verified ancillae

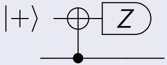
Steane Z-error correction



Logical circuit for Steane Z EC



Logical circuit for Steane X EC

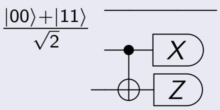


Knill-style error correction

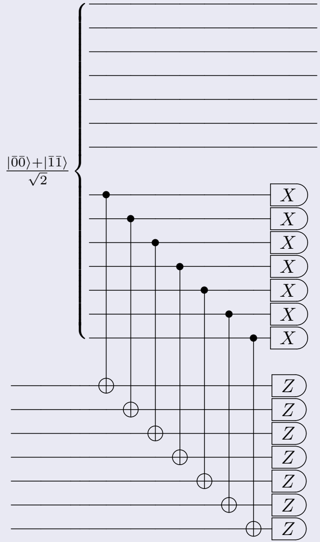
Knill error correction Knill quant-ph/0410199

- Logical circuit is teleportation
- Requires encoded $(|00\rangle + |11\rangle)/\sqrt{2}$ states
- One coupling for both X and Z error correction
- Physical errors cannot propagate through
- Teleportation eliminates leakage

Logical circuit for Knill EC

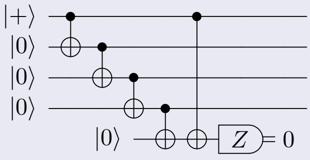


Knill X- & Z- error correction

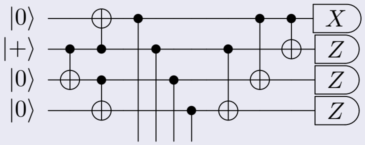


Ancilla construction approaches

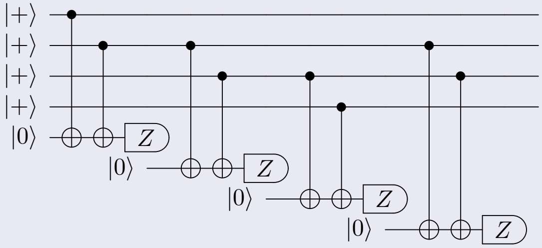
Make-and-measure



Make-and-measure-later



Measure-to-make

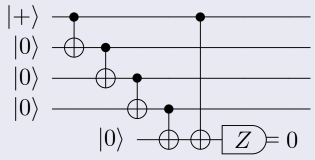


Make-and-measure ancilla construction

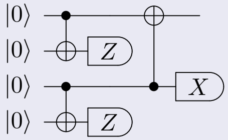
Make-and-measure ancilla construction Shor quant-ph/9605011

- $O(n)$ time to construct an arbitrary Clifford state
- Preparation can convert low- to high-weight errors
- States must be verified against artificially high-weight errors
- Error checks can take many forms
- Generically, a hierarchy of $\approx \log(d/2)$ transversal verification rounds (as shown for $d = 3$) using $\approx d^2/4$ states adequately suppresses errors
- Carefully chosen preparation circuits can decrease needed verification
  Paetznick 1106.2190

Make-and-measure 4-cat



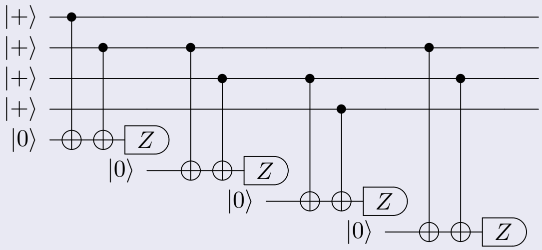
Logical circuit for d=3 verification



Measure-to-make ancilla construction Dennis quant-ph/0110143

- Starts in a product state, e.g., $|+\rangle^{\otimes n}$
- Uses Shor-style measurement of check operators to project into the code space
- Often used for surface codes

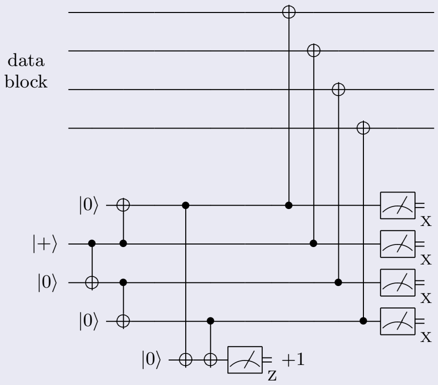
Measure-to-make



Make-and-measure-later ancilla construction DiVincenzo quant-ph/0607047

- Ancillae checked for errors after use
- Technique works for most operations on the Steane code.
- Circuits can be challenging to find for larger codes
- $O(m)$ time to de/construct an arbitrary m-qubit Clifford state
- Good for slow measurements

Make-and-measure

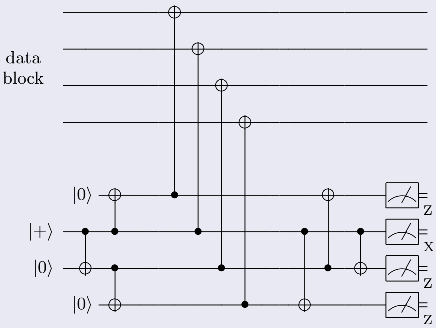


Make-and-measure-later ancilla construction

Make-and-measure-later ancilla construction DiVincenzo quant-ph/0607047

- Ancillae checked for errors after use
- Technique works for most operations on the Steane code.
- Circuits can be challenging to find for larger codes
- $O(m)$ time to de/construct an arbitrary m-qubit Clifford state
- Good for slow measurements

Make-and-measure-later

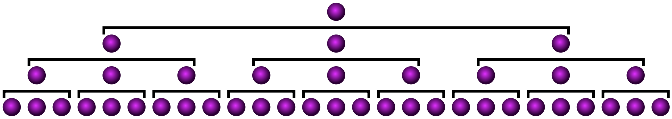


Computing forever

The encoded error rate cannot be made arbitrarily small with a finite code.

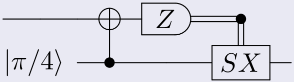
Approaches to increasing the error suppression of a code

- Switching to a larger instance of the code family
  - Often $d \propto \sqrt{n}$ or even n
  - Preparation of logical basis states can be challenging
  - Syndrome decoding can be challenging
  - Well suited to surface and other LDPC codes
- Concatenation
  - Iterates the encoding map, so each level of encoding decreases the effective error rate
  - Simple recursive ancilla preparation
  - Concatenated syndrome decoding gives $\lceil d/2 \rceil^l$ order suppression, $\lceil d^l/2 \rceil$ requires a multi-level decoder, e.g., message passing

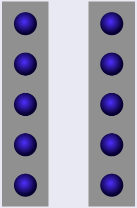


Achieving universality

Universality through teleportation



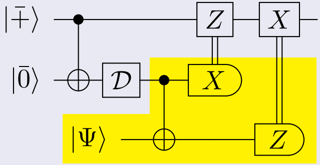
$$|\pi/4\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \exp^{i\pi/4}|1\rangle\right)$$



Recipe for achieving universality:

1. Prepare a computationally useful logical state (using, e.g., state injection)
2. Purify it or otherwise check for error
3. Use it to apply a gate through teleportation

State injection



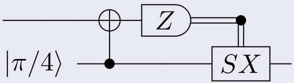
Dennis quant-ph/9905027

Knill quant-ph/0402171

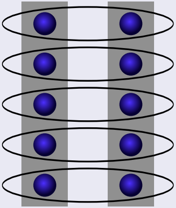
Bravyi quant-ph/0403025

Achieving universality

Universality through teleportation



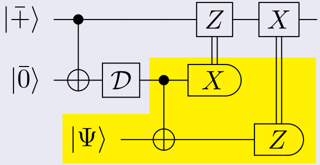
$$|\pi/4\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \exp^{i\pi/4}|1\rangle\right)$$



Recipe for achieving universality:

1. Prepare a computationally useful logical state (using, e.g., state injection)
2. Purify it or otherwise check for error
3. Use it to apply a gate through teleportation

State injection



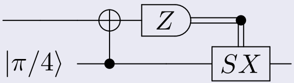
Dennis quant-ph/9905027

Knill quant-ph/0402171

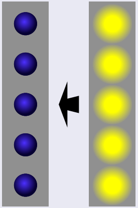
Bravyi quant-ph/0403025

Achieving universality

Universality through teleportation



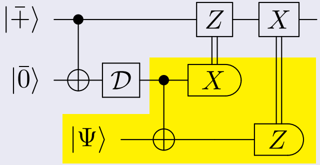
$$|\pi/4\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \exp^{i\pi/4}|1\rangle\right)$$



Recipe for achieving universality:

1. Prepare a computationally useful logical state (using, e.g., state injection)
2. Purify it or otherwise check for error
3. Use it to apply a gate through teleportation

State injection



Dennis quant-ph/9905027

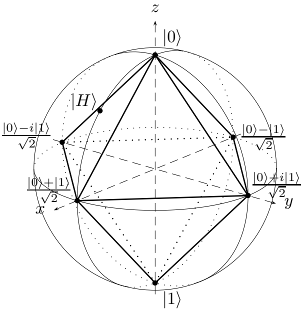
Knill quant-ph/0402171

Bravyi quant-ph/0403025

Magic-state distillation

State distillation The conversion of multiple faulty copies of a state into fewer copies of higher fidelity.

Magic state distillation The distillation of certain non-Clifford states using perfect Clifford gates.



Reichardt quant-ph/0608085

Procedure for magic state distillation:

1. Input imperfect magic states and perfect basis states
2. Measure some stabilizers of a code $\mathscr{S}$
3. Correct to $+1$ eigenspace of measured operators
4. Measure the remaining stabilizers of $\mathscr{S}$
5. On successful projection into $\mathscr{S}$, decode the resulting magic state

Twirling

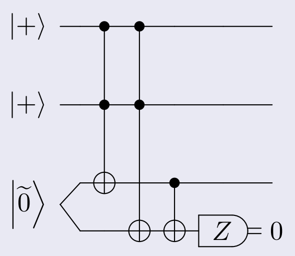
$$\mathcal{T}(\rho_A) = \sum_i T_i \rho_A T_i^\dagger \qquad \text{where } T_i|A\rangle = |A\rangle$$

Magic-state distillation

Alternative procedure for magic state distillation:

1. Prepare a perfect Clifford state encoded in a code $\mathscr{S}$
2. Fault-tolerantly implement a logical non-Clifford gate using non-Clifford states
3. Measure the stabilizers of $\mathscr{S}$
4. On successful projection into $\mathscr{S}$, decode the resulting magic state

Toffoli state distillation



Routines exist for multi-qubit states, multiple outputs, and qudits Aliferis quant-ph/0703230 Meier 1204.4221 Campbell 1205.3104

Efficiency of magic-state distillation

- $\xi = \log_{\{\text{order of error suppression}\}} \left(\{\text{\# input magic states}\} / \{\text{\# output magic states}\} \right)$
- $\xi \geq 1$ conjectured Bravyi 1209.2426
- $\xi \to 1$ in existing protocols Jones 1210.3388

Many techniques for avoiding distillation

Shor quant-ph/9605011 Knill quant-ph/9610011 Paetznick 1304.3709

Thresholds for quantum computation

Encoding does not always help. Error correction with unreliable components can make things worse.

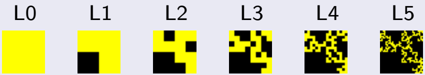
QC threshold The physical error probability below which an arbitrary quantum computation can be performed efficiently

Pseudothreshold The physical error rate such that

$$\left\{\begin{array}{c}\text{Encoded}\\ \text{failure probability}\end{array}\right\} < \left\{\begin{array}{c}\text{Unencoded}\\ \text{failure probability}\end{array}\right\}$$

Necessarily, below threshold the logical error probability can be made arbitrarily small

Worst-case good qubits



Pseudothresholds are difficult to define rigorously

- Error probability does not fully characterize the error model
- Picking a starting logical state is tricky
- "Good" physical qubits are better than "good" logical qubits

Rigorous threshold bounds using the AGP method

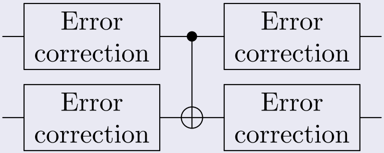
The AGP method rigorously defines recursion so that the ExRec pseudothreshold bounds the threshold

Aliferis quant-ph/0504218

AGP answers to pseudothreshold issues

- Issue: Error model freedom
  - Answer: Adversarial error model
- Issue: Starting state
  - Answer: ExRecs
- Issue: "Good" logical qubits are less good
  - Answer: Define "good" using ideal decoder

ExRec (Extended Rectangle)



Highest rigorous threshold lower bounds: 1.3×10^{-3}

Paetznick 1106.2190

Aliferis 0809.5063

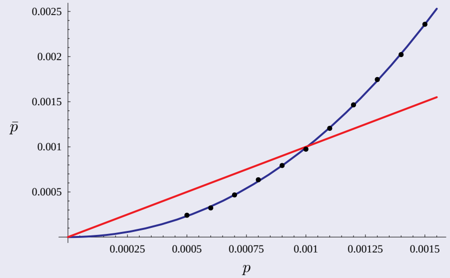
Numerically estimating the threshold

NORTHROP GRUMMAN

Numerical estimates of the threshold are generally obtained using Monte-Carlo routines

- Pauli errors are generated probabilistically
- Errors collected using error propagation
- Failure declared if an ideal decoder would miscorrect the Pauli errors
- Threshold approximated with pseudothreshold

Pseudothreshold crossover



Disadvantages of Monte-Carlo routines:

- Require significant time and computational power
- Effectiveness decreases as event rate goes down
- Error model must be fixed in advance

Highest threshold estimates: $.5 - 3\%$ Knill quant-ph/0410199 Fowler 0803.0272

Conclusion

NORTHROP GRUMMAN

There's much much more...

Topics to explore on your own

- Resource overhead for quantum computing
- The effect of coherent, correlated, and leakage errors on quantum error correction
- The construction of quantum from classical codes
- Subsystem, LDPC, and topological codes
- Decoherence free subspaces/subsystems
- Self-correction and quantum feedback
- Upper bounds on the threshold
- Gate decompositions
- Quantum coding bounds
- Randomized benchmarking and tomography