The Threshold for Fault-Tolerant Quantum Computation

Daniel Gottesman
Perimeter Institute

Basics of Fault-Tolerance

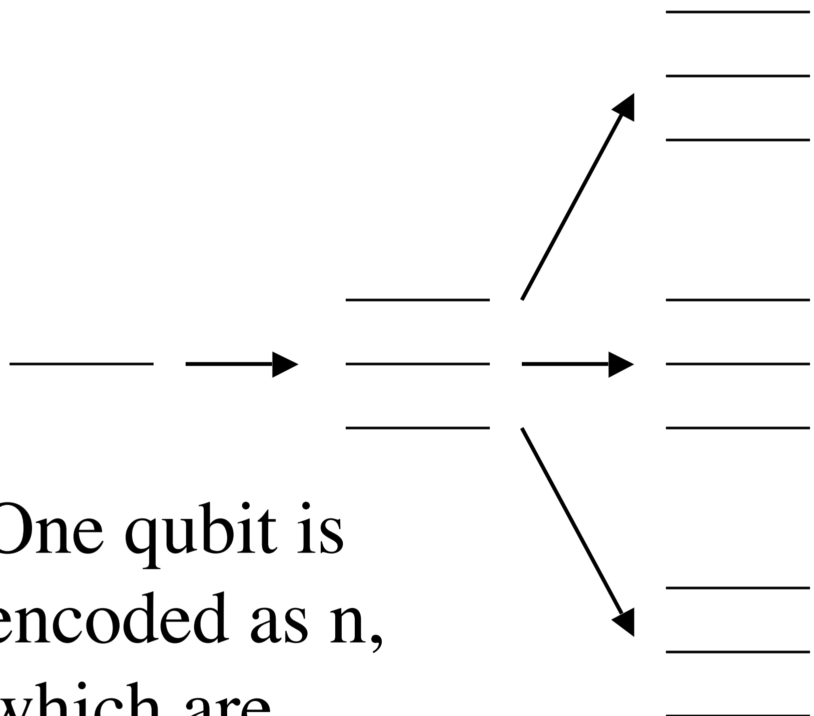
• The purpose of fault-tolerance is to enable reliable quantum computations when the computer's basic components are unreliable.

• To achieve this, the qubits in the computer are encoded in blocks of a quantum error-correcting code, which allows us to correct the state even when some qubits are wrong.

• A fault-tolerant protocol prevents catastrophic error propagation by ensuring that a single faulty gate or time step produces only a single error in each block of the quantum error-correcting code.

Concatenated Codes

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



Error correction is performed more frequently at lower levels of concatenation.

One qubit is encoded as n, which are encoded as n^2, …

Effective error rate

$$p \rightarrow Cp^2$$

Threshold for Fault-Tolerance

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

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

$$p_k < p_t (p/p_t)^{2^k}$$

and for a computation of length T, we need only log (log T) levels of concatention, requiring polylog (T) extra qubits, for sufficient accuracy.

Determining the Threshold Value

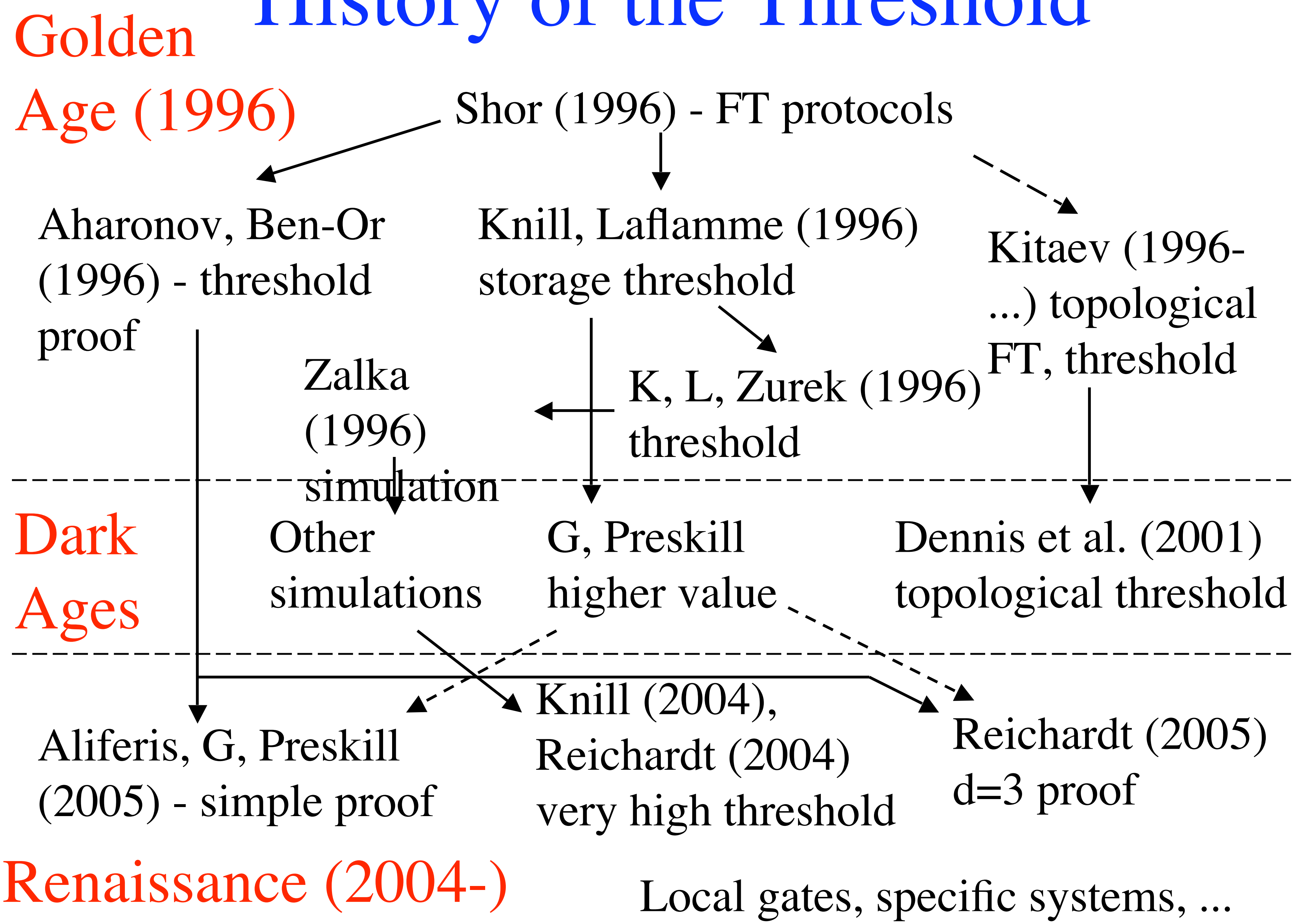
There are three basic methodologies used to determine the value of the threshold:

- Numerical simulation: Randomly choose errors on a computer, see how often they cause a problem. Tends to give high threshold value, but maybe this is an overestimate; only applies to simple error models.

- Rigorous proof: Prove a certain circuit is fault-tolerant for some error rate. Gives the lowest threshold value, but everything is included (up to proof's assumptions).

- Analytic estimate: Guess certain effects are negligible and calculate the threshold based on that. Gives intermediate threshold values.

History of the Threshold



Requirements for Fault-Tolerance

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

Additional Desiderata

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

Threshold Values

Best proofs of the threshold give $p_T \geq 2.7 \times 10^{-5}$ (Aliferis, Gottesman, Preskill, quant-ph/0504218; also Reichardt, quant-ph/0509203), assuming all desiderata.

Best methods trade extra ancilla qubits for error rate: Ancilla factories create complex ancilla states to substitute for most gates on the data. Errors on ancillas are less serious, since bad ancillas can be discarded safely.

Taking this idea to an extreme (e.g., overhead of 10^6 or more physical qubits per logical qubit), simulations show a threshold of 1% or higher (Knill, quant-ph/0404104, Reichardt, quant-ph/0406025.) Proofs have not fully included these techniques yet.

Local Gates

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

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

- Storage threshold using topological codes ≥ 10^{-4} in 2D (Dennis, Kitaev, Landahl, Preskill, quant-ph/0110143)
- With concatenation, in 2D, lose factor of 2-3 in threshold (Svore, Terhal, DiVincenzo, quant-ph/0410047, quant-ph/0604090)
- In (almost) 1D, lose factor of 10-100 in threshold (Szkopek et al., quant-ph/0411111)

Non-Markovian Errors

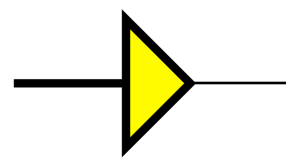
What happens when the environment has a memory?

- Questioning fault-tolerance for non-Markovian environments: Alicki, Horodecki^3 (quant-ph/0105115), Alicki, Lidar, Zanardi (quant-ph/0506201)
- Proof of fault-tolerant threshold with single-qubit errors and separate environments for separate qubits: Terhal, Burkhard (quant-ph/0402104)
- Proof of fault-tolerant threshold with shared environment: Aliferis, Gottesman, Preskill (quant-ph/0504218)
- With 2-qubit errors: Aharonov, Kitaev, Preskill (quant-ph/0510231)
- Unbounded Hamiltonians (spin boson model)? See Terhal, Burkhard and Klesse, Frank (quant-ph/0505153)

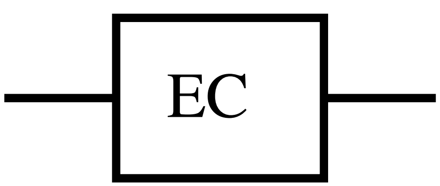
Distance 3 Proof

If a block of a QECC has errors, how do we define the state of the encoded data? How do we define when a state has errors?

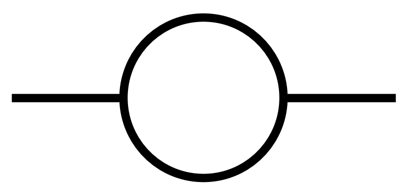
Solution: Use a syntactic notion of correctness, not a semantic one. States are not correct or incorrect, only operations.

Define encoded state using ideal decoder: 

Conventions:



FT error correction



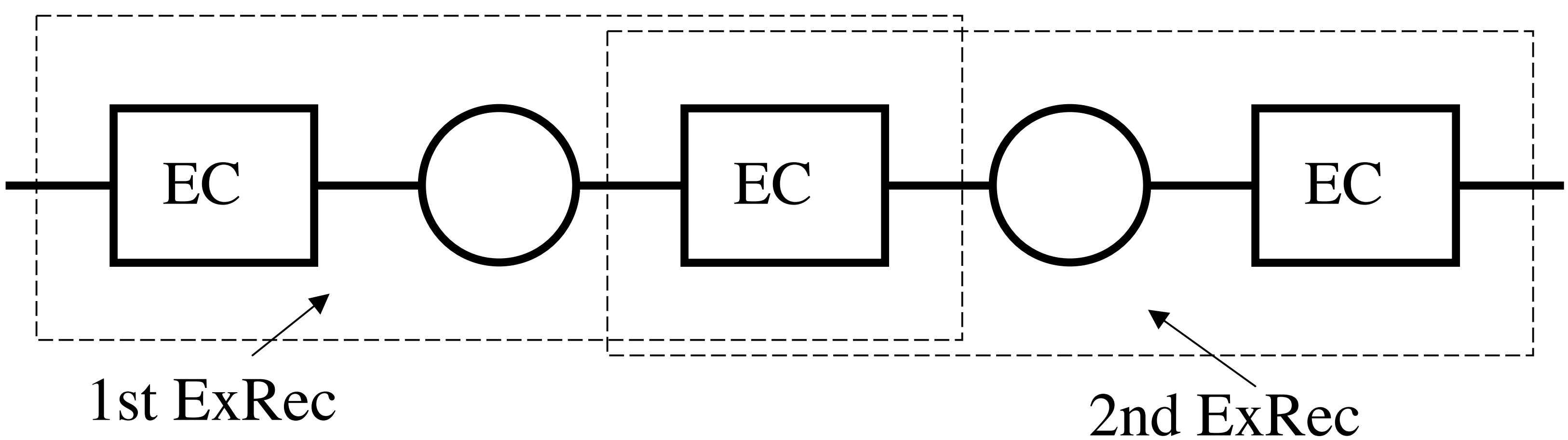
FT encoded gates

Extended Rectangles

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

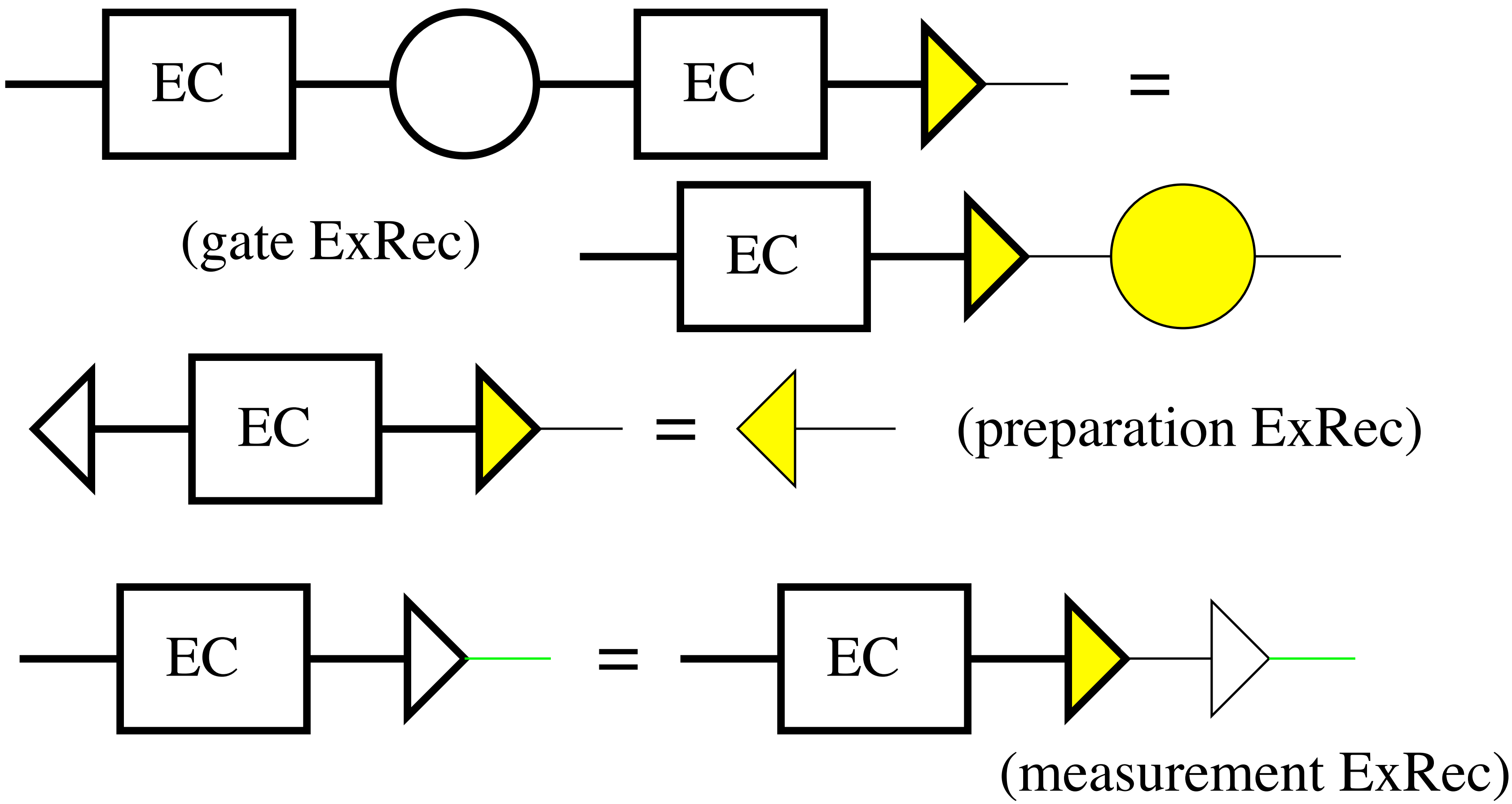
Definition: An ExRec is "good" if it contains at most one fault (roughly speaking). A fault is a bad lower-level rectangle or gate.

Note: Extended rectangles overlap with each other.



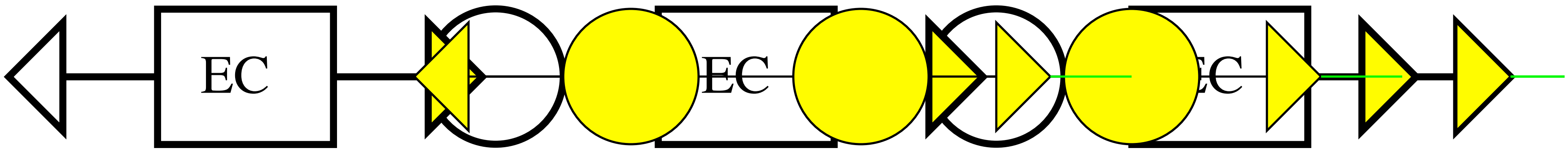
Good Circuits are Correct

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



Correct Means What It Ought To

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



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

The Future of Fault-Tolerance