Recent schemes for increasing the fault-tolerance threshold for fault-tolerant computation.

Schemes for fault-tolerant QC.

Let me just make clear we agree what the problem is. We start with a quantum circuit to solve some question. It is made up of basic gates like the CNOT gate. The thing of course is that gates in a real circuit won't be perfect; there'll be errors from some mode, or non-unit fidelity.

We want to expand or compile this ideal circuit into a physical circuit which tolerates the physical noise and still gives the right answer. We want to minimize the amount of noise which can be tolerated, while keeping the expansion relatively efficient.

There are four or five ideas which have been used recently to increase the tolerable noise threshold.

- postselection
- bottom-up decoding
- magic state distillation
- teleportation
- stabilizer purification

Let me start by describing a fairly standard scheme. This has been optimized for both efficiency and threshold by Steane, and applies to give a ≈ 50% threshold against independent depolarizing errors.

Steane ≈ 50%.

The idea is fairly straightforward: we simply encode each logical bit into any 4 physical bits using a quantum error correcting code like the Hadamard/Steane code.

Each time step you apply one logical operation, possibly involving multiple logical qubits, then do error correction.

How do we do error correction?
First, to correct $X$ errors, or bit-flip errors, we prepare an encoded $X$ gate.

Then we do a transverse, logical CNOT from the data to the ancilla, and measure the ancilla in the $Z$ eigenbasis:

$$X = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$X$ errors are copied from the data to the ancilla, but no entanglement is created. So, measuring the $Z$ syndromes tells us where the $X$ errors are, and they can be corrected.

Similarly, to correct phase-flip $Z$ errors, we do all the dual operations. There are a number of details to play with here, for example, how to prepare and verify good-encoded ancillas. Also, perhaps you should extract several syndromes to be sure of the error location before applying any correction, etc.

To get a threshold result, you concentrate this scheme.

So for example, say this is the circuit for preparing the encoded ancilla. Basically, start with $|0\rangle$ and $|\psi\rangle$, and apply a number of CNOT gates.

```
107 - 1 - 107
117 1
```

(In omitting the ancilla verification here.) To prepare the encoded ancilla with one level of correction, simply start with encoded $|0\rangle$ and $|\psi\rangle$, and apply encoded, transverse CNOTs, each one followed by error correction.

```
107 - 1 - 107
117 1
107 - 1 - 107
```

With error correction, it takes $\frac{d}{n}$ physical errors within a block before $\frac{1}{d}$ error correction corrects the block in the wrong direction. A simple modification of this scheme uses error rejection instead of error correction. If any bit error is detected, discard away the entire level of ancilla, and start over. Then it takes $\frac{d}{n}$ instead of $\frac{d}{n}$ errors to find error detection.

There is some overhead, of course, because all the blocks we just threw away, but simulations suggest that this modification increases the threshold by about a factor of three.
Kane has recently taken the idea of postselection/feedback-testing much further, and added some new ideas. Here's a theorem of his:

**Theorem.** Kane's CQI: Threshold for accurate error is $1/4$ for Bell measurements.

The accurate error model is one in which you are told when an error has occurred, you know which unitary was compromised. (So for example an error correcting code can correct 11 errors instead of just 8.)

A Bell measurement is a two-qubit measurement in the Bell basis, which can be implemented as

\[
\begin{pmatrix}
1 & 1 \\
1 & -1
\end{pmatrix}
\]

(Measure the first bit in the $X$ eigenbasis, the second in the $Z$ eigenbasis. Only Bell measurement ensures walter, not measures in many other operations, as well as.

This is an interesting theorem because it introduces the idea of teleportation for fault-tolerant computing, because it is a real theorem with a proof of a high threshold (not just simulations), and because it suggests directions for providing high thresholds in more general error models.

Let me just sketch Kane's proof.

**Teleportation**

\[
\begin{pmatrix}
1 & 1 \\
1 & -1
\end{pmatrix}
\]

**Computation**

\[
\begin{pmatrix}
1 & 1 \\
1 & -1
\end{pmatrix}
\]

<table>
<thead>
<tr>
<th>Correlation (CNOT)</th>
</tr>
</thead>
</table>

In general, everyone's familiar with teleportation, and the circuit. We do a Bell measurement on the state we wish to teleport and half a Bell state. What comes out is some Pauli matrix $-iX, iY, iZ = -i(ket)$. Now let's add in computation. Say we want to apply unitary $U$, to our data. Instead of doing it directly, we apply it to the second half of a Bell state that we then teleport into. We get back $(U|\Psi^+\rangle)$ for some Pauli $P$. This probably isn't what we want; we need to apply a correction $C_{U,P}$. We don't apply this correction directly to the data; instead we'll apply it in the next teleportation step. This also works with multiple pauli gates, and it is therefore important that the gate set conjugates the pauli group nicely. We can already...
see why the error threshold depends on Bell measurement errors. If an error occurs in one of these prepared states, we just throw it out and try again.

Next we add fault-tolerance.

\[ \text{Fault-tolerance} \]

\[ \rho \rightarrow \rho_1 \]

\[ \rho \rightarrow \rho_2 \]

\[ \rho \rightarrow \rho_3 \]

\[ \rho \rightarrow \rho_4 \]

We simply encode all the states we get out \( \rho_1, \rho_2 \), where \( \rho \) is the tensor product of Pauli matrices.

Now what happens if there are errors? Measuring the first half in the \( X \) basis lets us see the \( X \) part of any stabilizers, while measuring the second half in the \( Z \) basis lets us read off the \( Z \) part of any stabilizers. So while any one of the measurements is random, altogether they are correlated according to the syndrome.

This works for CSS codes, and also general stabilizer codes.

To check this just uses a little stabilizer algebra. For example if a code is stabilized by \( X^2 \) and logical \( X = X_1 \) and logical \( Z = X_1 Z \), then

\[ \rho_1 \rightarrow \rho_2 \rightarrow \rho_3 \rightarrow \rho_4 \]

Stabilizers are split into two all their \( X \) components in the first half, and all their \( Z \) components in the second half.

To finish the theorem, then, we just need to know that there exist stabilizer codes (random codes) which correct up to \( Z \) fraction error, asymptotically. So just choose one of these codes so that the probability of failure is

\[ \leq \frac{1}{2} \]

\[ \rho_1 \rightarrow \rho_2 \rightarrow \rho_3 \rightarrow \rho_4 \]

This just about proves the theorem. Note that unlike in a standard scheme there has been no code concatenation. Actually, to prevent exponential overhead, in the number of encoded Bell states which are rejected because of an erasure, you need exactly one level of concatenation.

This scheme works because logical \( Z \) does not touch the data, so can be verified to have worked. Also X and Z error correction are combined into a single operation using just one known error \( \text{CNOT, instead of two.} \)

This suggests that the scheme might also work well, perhaps provably, for error models besides the erasure error model.
In fact, there are stabilizer codes known which asymptotically correct up to 15% depolarizing error. So if we can prepare the teleportation state to have independent errors at a rate so that the total error rate from the previous correction, from the Bell measurement, and from the teleportation step is less than 15%, then we could again get a provable high-threshold result.

The question is how to prepare the teleportation state with intrinsic independent errors.

\[
(5 \oplus \frac{1}{2}) (10 \oplus \frac{1}{2})
\]

Knill has a method for doing this which appears to give a threshold at 2.2%, the highest threshold known, albeit with substantial overhead.

Now I'd like to describe Knill's method.

First, we can basically assume that \( \mathbb{C} \) is a Clifford group operation, so we are trying to prepare a stabilizer state. Why is this? Well, Bravyi and Kitaev have introduced the idea of magic states distillation. The ability to prepare the Hadamard gate \( +1 \) eigenstate, together with Clifford group operators, gives universality. And in fact, even if it can only be prepared with 15% error, the state can be purified to the Hadamard \( +1 \) eigenstate with Clifford operations, again giving universality.

\[
\begin{array}{c}
\text{Bravyi-Kitaev magic states distillation} \\
\text{Bravyi and Kitaev have introduced the idea of magic states distillation.}
\end{array}
\]

Here's what we do: prepare an encoded Bell state. Decode the first half, starting over if any errors are detected. Teleport in the dummy Hadamard eigenstate to get an encoded Hadamard eigenstate, with a legal error rate equal to the original eigenstate error rate, plus the Bell measurement error rate, plus the decoding error rate. As long as this total is less than 15%, we can purify the state using logical Clifford operations.