Magic-State Model and Resource Estimates

Fault tolerance, distillation, and hardware tradeoffs

Paula Ulrich

Advanced Seminar

July 17, 2026

Main question and roadmap

Why do magic states appear in fault-tolerant quantum computing, and why do they create such large resource overheads?

Core message

Fault tolerance gives reliable Clifford operations, but Clifford operations are not universal. Magic states supply the missing non-Clifford resource, and their production becomes a throughput, fidelity, and hardware-scaling problem.

Fault tolerance $\rightarrow$ magic states $\rightarrow$ distillation $\rightarrow$ factories and resource estimates

Fault tolerance: from physical to logical qubits

Physical qubits are noisy, but large algorithms require many gates and long storage times.

A logical qubit stores one quantum state in many physical qubits:

$$|\psi_L\rangle = \alpha\,|0_L\rangle + \beta\,|1_L\rangle\,.$$

Quantum error correction

Syndrome measurements detect small error patterns without measuring the logical amplitudes α and β.

Fault tolerance

The computation must also be protected while gates, measurements, routing, and feedforward are performed.

Why error propagation matters

A gate can spread a single physical error into several errors.

For a CNOT,

$$\boxed{\mathrm{CNOT}(X \otimes I) = (X \otimes X)\mathrm{CNOT}.}$$

Using the repetition-code intuition,

$$|0_L\rangle = |000\rangle\,, \qquad |1_L\rangle = |111\rangle\,,$$

one bit flip is correctable:

$$|000\rangle \xrightarrow{X_1} |100\rangle\,.$$

But if a bad operation spreads it,

$$|100\rangle \to |110\rangle \to |111\rangle\,,$$

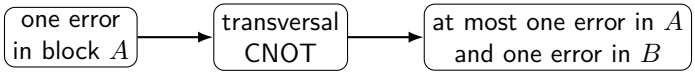
then one physical fault has become a logical error.

Transversal gates

A transversal gate acts bitwise across code blocks.

For a logical CNOT between blocks A and B:

$$\mathrm{CNOT}_L = \mathrm{CNOT}_{A_1B_1}\mathrm{CNOT}_{A_2B_2}\cdots\mathrm{CNOT}_{A_nB_n}.$$



Key point

The error may spread between blocks, but it does not spread to many qubits within the same block.

Transversal CNOT example

Start with two encoded blocks:

$$|0_L\rangle_A |0_L\rangle_B = |000\rangle_A |000\rangle_B .$$

A single error occurs:

$$|000\rangle_A |000\rangle_B \to |100\rangle_A |000\rangle_B .$$

Apply transversal CNOT:

$$\mathrm{CNOT}_{A_1 B_1} \mathrm{CNOT}_{A_2 B_2} \mathrm{CNOT}_{A_3 B_3}.$$

Then

$$|100\rangle_A |000\rangle_B \to |100\rangle_A |100\rangle_B .$$

Still correctable

One error in block A, one error in block B.

Eastin–Knill obstruction

Ideally, we would like

$$\text{quantum code} + \text{universal transversal gate set.}$$

Eastin–Knill theorem

No finite-dimensional quantum error-correcting code has a universal set of transversal logical gates.

Consequence

At least one additional fault-tolerant method is needed for universality, for example code switching, gauge fixing, lattice surgery, or magic-state injection.

Clifford gates are useful, but not enough

Typical "easy" fault-tolerant gates are

$$H, \qquad S, \qquad \mathrm{CNOT}.$$

These are Clifford gates.

Gottesman–Knill theorem

Clifford circuits with standard measurements can be efficiently simulated classically.

For universality, we need a non-Clifford resource. A standard choice is

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}, \qquad \{H, S, \mathrm{CNOT}, T\} \text{ is universal.}$$

Magic-state idea

Instead of directly applying the hard non-Clifford gate, prepare and consume a special state.

The standard T-type magic state is

$$|A\rangle = T\,|+\rangle = \frac{|0\rangle + e^{i\pi/4}\,|1\rangle}{\sqrt{2}}.$$

Idea

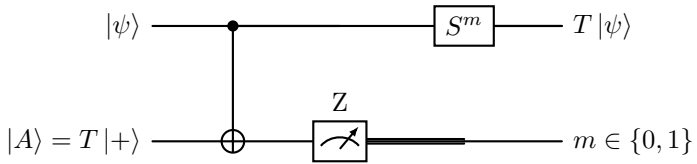
Replace a hard non-Clifford gate by a special non-stabilizer resource state.

One high-fidelity magic state gives one high-fidelity logical T gate.

Magic-state injection

Goal:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \quad \longmapsto \quad T\,|\psi\rangle\,.$$



$$m = 0:\ T\,|\psi\rangle\,, \qquad m = 1:\ T^\dagger\,|\psi\rangle \xrightarrow{S} ST^\dagger\,|\psi\rangle = T\,|\psi\rangle\,.$$

Key idea

The circuit uses CNOT, measurement, and a conditional Clifford correction. The magic state supplies the non-Clifford resource.

Magic-state injection: mathematical action

Let

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \qquad |A\rangle = \frac{|0\rangle + e^{i\pi/4}\,|1\rangle}{\sqrt{2}}.$$

After the CNOT,

$$|\Psi\rangle = \alpha\,|0\rangle\,|A\rangle + \beta\,|1\rangle\,X\,|A\rangle\,.$$

Measuring the ancilla in the computational basis gives

$$m = 0: \quad T\,|\psi\rangle\,,$$

or

$$m = 1: \quad T^\dagger\,|\psi\rangle\,.$$

For $m = 1$, apply the Clifford correction S:

$$ST^\dagger\,|\psi\rangle = T\,|\psi\rangle\,.$$

Result

Both measurement outcomes produce $T\,|\psi\rangle$, up to a conditional S correction.

Why distillation is needed

Physical preparation of magic states is noisy:

$$|A\rangle\langle A| \quad \longrightarrow \quad \rho_A.$$

If we inject ρ_A instead of the ideal $|A\rangle$, the implemented T gate is also noisy.

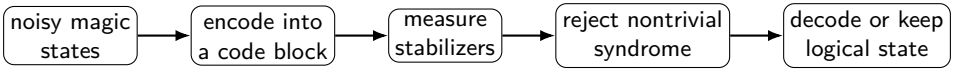
Goal

Use many imperfect magic states to produce fewer states with higher fidelity to $|A\rangle$.

$$\rho_A^{\otimes n} \quad \longrightarrow \quad \rho_{\text{out}} \approx |A\rangle\langle A|.$$

Distillation as quantum error detection

Magic-state distillation uses a quantum error-correcting code as a filter.



Connection to QEC

The syndrome detects whether the state has left the code space, without measuring the encoded logical state.

Difference from ordinary QEC

In distillation, detected errors are usually rejected instead of corrected.

Simple intuition: a stabilizer parity check

Consider the two-qubit stabilizer

$$S = Z_1 Z_2, \qquad P_+ = \frac{I + Z_1 Z_2}{2}.$$

It accepts the even-parity subspace:

$$P_+ = |00\rangle\langle 00| + |11\rangle\langle 11|.$$

For a good state,

$$|\psi_{\text{good}}\rangle = \alpha\,|00\rangle + \beta\,|11\rangle, \qquad P_+\,|\psi_{\text{good}}\rangle = |\psi_{\text{good}}\rangle.$$

For an X_1 error,

$$X_1\,|\psi_{\text{good}}\rangle = \alpha\,|10\rangle + \beta\,|01\rangle, \qquad P_+ X_1\,|\psi_{\text{good}}\rangle = 0.$$

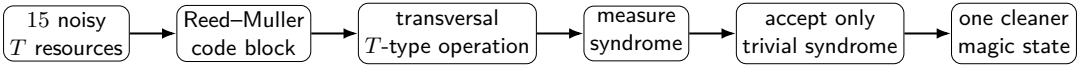
Analogy

A stabilizer check is a quantum parity check.

15-to-1 distillation: the high-level protocol

A standard T-state protocol uses the $[[15, 1, 3]]$ Reed–Muller code:

$$15 \text{ noisy } |A\rangle \quad \longrightarrow \quad 1 \text{ cleaner } |A\rangle\,.$$



Why this code?

The code has distance 3 and admits a transversal T-type logical operation. Therefore physical T resources can be checked by stabilizer measurements.

Why the error suppression is cubic

For small input error probability ϵ, the 15-to-1 protocol gives

$$\epsilon_{\text{out}} = 35\epsilon^3 + O(\epsilon^4).$$

- Distance 3 means all weight-1 and weight-2 error patterns are detected.
- The first undetected logical errors have weight 3.
- In the standard idealized stochastic model, there are 35 such leading triple-fault patterns.

$$\boxed{15 \text{ inputs of quality } \epsilon \quad \longmapsto \quad 1 \text{ output of quality } \approx 35\epsilon^3.}$$

Slogan

Magic-state distillation sacrifices quantity for quality.

Logical output and lattice surgery

The accepted state can be viewed as a logical magic state:

$$|A_L\rangle = \frac{|0_L\rangle + e^{i\pi/4}\,|1_L\rangle}{\sqrt{2}}.$$

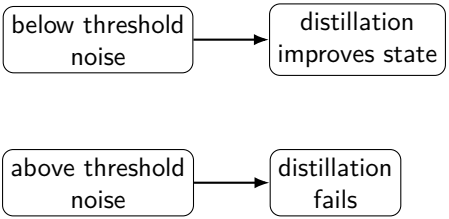
Two viewpoints

- Decode the accepted block to obtain a cleaner physical-level $|A\rangle$ state.
- Keep the state encoded and consume it directly as a logical resource $|A_L\rangle$.

In surface-code architectures, logical magic states can be routed, merged, and consumed through joint logical measurements, for example using **lattice surgery**.

Concatenation and threshold behavior

Distillation only works if the input states are already good enough.



Concatenation

The output of one distillation level can be used as the input to the next level, similar in spirit to building larger error-correction structures.

Using only the leading-order map, $\epsilon = 10^{-3}$ becomes approximately $35 \cdot 10^{-9} = 3.5 \cdot 10^{-8}$ after one round.

Why magic states create overhead

Each logical T gate consumes one distilled magic state:

$$1\ T \text{ gate} \quad \Longrightarrow \quad 1 \text{ distilled magic state.}$$

Therefore an algorithm with N_T logical T gates needs roughly N_T high-fidelity magic states.

Production problem

The question becomes: how fast can the architecture distill, route, store, and inject enough states at the required fidelity?

$$N_{\text{factories}} \approx \frac{N_T}{R_{\text{fac}}\tau}.$$

Here R_{fac} is the accepted output rate of one factory and τ is the allowed runtime.

Tradeoffs in magic-state production

Choice	Consequence
More factories	higher throughput, more physical qubits
Larger code distance	lower logical error, larger and slower patches
More distillation levels	cleaner states, lower throughput
Longer runtime	fewer factories, more storage exposure
Faster classical control	better decoding and feedforward throughput

Main point

Resource estimation is a space-time-fidelity tradeoff, not a single qubit-count calculation.

Fowler estimate: assumptions behind the numbers

The estimate describes one particular surface-code architecture.

Assumption	Effect on the estimate
Measurement time $t_M = 100\,\mathrm{ns}$	sets the duration of the sequential Toffoli circuit
Surface-code cycle $\approx 200\,\mathrm{ns}$	sets the production rate of a magic-state factory
Physical error rate $p = 10^{-3}$	requires code distances $d_1 = 17$ and $d_2 = 34$
Injection error $p_I = 5 \times 10^{-3}$	requires two levels of magic-state distillation

Important

The resulting runtime and qubit count are consequences of these assumptions, not universal constants.

Fowler estimate: runtime and demand

For $N = 2000$, the modular-exponentiation circuit requires

$$40N^3 \approx 3.2 \times 10^{11}$$

sequential Toffoli gates.
Each Toffoli takes approximately three measurement times:

$$t_{\text{Toffoli}} \approx 3t_M.$$

With $t_M = 100\,\text{ns}$,

$$t_{\text{total}} = 40N^3 \cdot 3t_M \approx 26.7 \text{ hours}.$$

Each Toffoli consumes seven logical magic states:

$$N_A = 7 \cdot 40N^3 \approx 2.2 \times 10^{12}.$$

Measurement time sets the runtime; the Toffoli count sets the demand.

Fowler estimate: factory footprint

One factory produces approximately

$$2 \ |A_L\rangle \text{ states per } 100\,\mu\text{s} = 2 \times 10^4 \text{ states/s}.$$

During 26.7 hours, one factory produces approximately

$$N_{A,\text{fac}} \approx 2 \times 10^9$$

states.
Therefore,

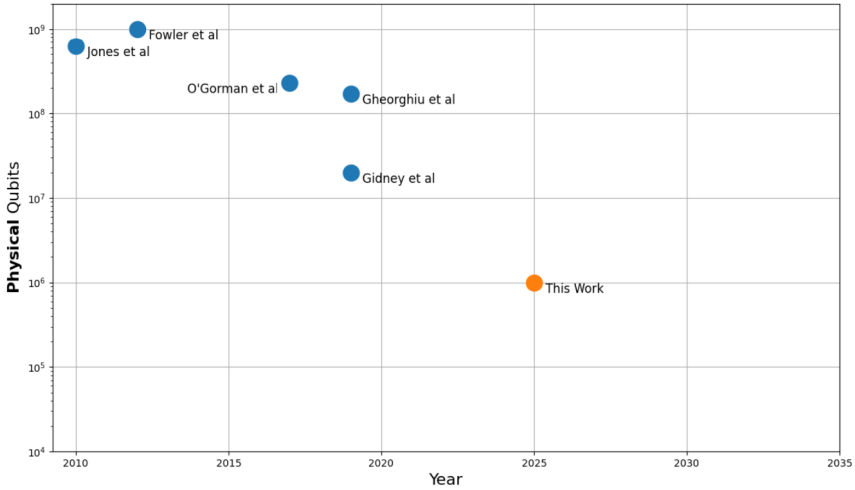
$$N_{\text{fac}} \approx \frac{2.2 \times 10^{12}}{2 \times 10^9} \approx 1200.$$

With approximately 8×10^5 physical qubits per factory,

$$N_{\text{phys}} \approx 1200 \cdot 8 \times 10^5 \approx 10^9.$$

Higher throughput requires more parallel factories and more qubits.

Resource estimates have improved



Source: C. Gidney, *How to factor 2048 bit RSA integers with less than a million noisy qubits*, arXiv:2505.15917, Fig. 1.

Final message

Fault tolerance $\rightarrow$ reliable Clifford operations

Eastin–Knill $\rightarrow$ need non-Clifford resources

T gates $\rightarrow$ magic states

distillation $\rightarrow$ postselected error detection with QEC codes

many magic states $\rightarrow$ factories, routing, storage, and control overhead

Conclusion

Magic-state production is a major throughput and space-time constraint. Depending on the architecture and algorithm, it can dominate the physical footprint, but this is not universal.

Thank you

References


M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, 10th Anniversary Edition, 2010.


S. Bravyi and A. Kitaev, *Universal quantum computation with ideal Clifford gates and noisy ancillas*, Phys. Rev. A 71, 022316 (2005), arXiv:quant-ph/0403025.


E. T. Campbell, B. M. Terhal, and C. Vuillot, *Roads towards fault-tolerant universal quantum computation*, Nature 549, 172–179 (2017).


A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Surface codes: Towards practical large-scale quantum computation*, Phys. Rev. A 86, 032324 (2012), arXiv:1208.0928.


D. Litinski, *A Game of Surface Codes: Large-Scale Quantum Computing with Lattice Surgery*, Quantum 3, 128 (2019), arXiv:1808.02892.


C. Gidney and M. Ekerå, *How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits*, Quantum 5, 433 (2021), arXiv:1905.09749.


C. Gidney, *How to factor 2048 bit RSA integers with less than a million noisy qubits*, arXiv:2505.15917 (2025).