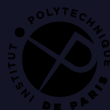




LE HUB QUANTIQUE

Fault-tolerant quantum computation: from resource estimations to experimental re- alization

Paul Hilaire • Inria / Télécom Paris • October 22nd 2026



Inria



Paul Hilaire

QURIOSITY (Inria / Télécom Paris)

Chaire de Professeur Junior

- Experimental PhD in Quantum Optics (2019)
- Postdoc in MBQC and quantum repeaters (2019-2022)
- Lead Quantum Information Scientist at Quandela (2022-2025)
- **Quantum error correction**
- **Fault-tolerant architectures**
- **Photonic quantum computing**

Teach QEC and Quantum Hardware at M2 QMI
(master in mathematical aspects of quantum information)

Ambitions of this talk

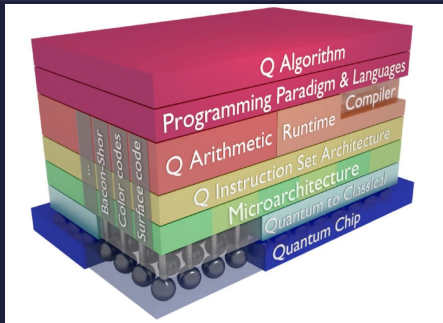


Figure: Stacks of a FT Quantum Computer

- Motivate fault-tolerance in quantum computing
- Introduce quantum error correction
- Challenges of building a fault-tolerant quantum computer
- When will fault-tolerance be achieved?
 - Experimental progress
 - Resource estimation

Ambitions of this talk

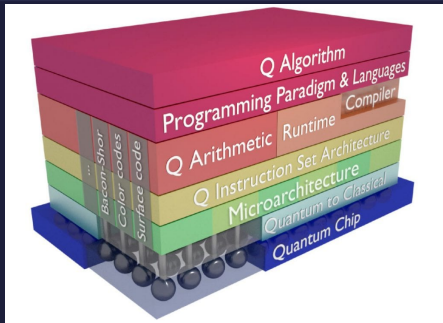


Figure: Stacks of a FT Quantum Computer

- Global overview of where FTQC is at
- Not too technical
- But not too high level as well
- Scientifically honest
- Not a company roadmap

Table of Contents



1 Why NISQ saturates

2 QEC foundations

3 Surface-code baseline

4 Resource estimation

5 Recent progress: Experiments

6 Recent progress: Theory

7 Conclusion

SECTION

1

Why NISQ saturates

Quantum Computing objectives

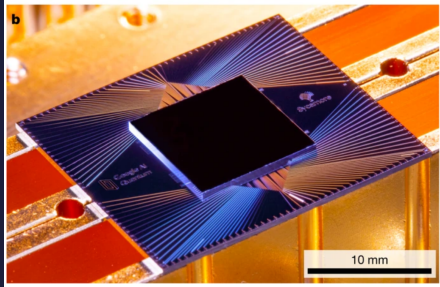


Figure: Google's Sycamore chip: Quantum advantage experiment

- Quantum advantage (supremacy) ...
- ... for useful applications.
- The challenge is noise.
- Objective: Building Noisy-Intermediate Scale Quantum (NISQ) devices for near-term applications.

A closer look at errors

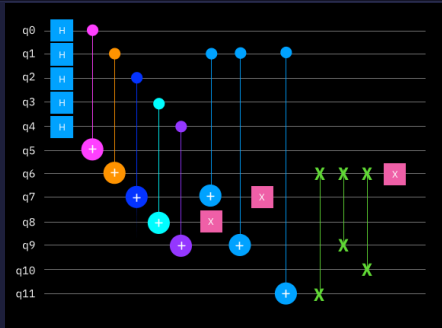


Figure: Quantum circuit model

- Q. Algo: sequence of operations on qubits.
- n_G noisy operations, error rate p each.
- **Intuition:** Algo. success probability $\propto P_{\text{no error}}$.

Back of the envelope calculation



$$P_{\text{no error}} \approx (1 - p)^{n_G}$$

$$p \ll 1$$

$$P_{\text{no error}} \approx 1 - n_G p$$

Algorithm success requires

$$n_G = O(1/p)$$

A closer look at errors

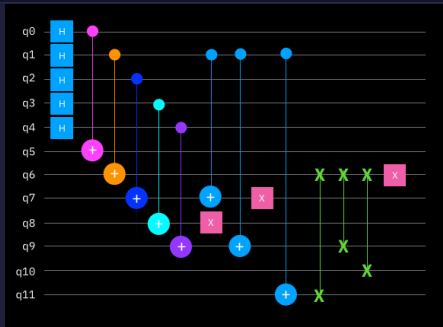


Figure: Quantum circuit model

- Q. Algo: sequence of operations on qubits.
- n_G noisy operations, error rate p each.
- **Intuition:** Algo. success requires $n_G = O(1/p)$
- **Objectives:**
 - Reduce n_G (NISQ)
 - Reduce p (hardware)
 - Replace p with p_L (FTQC)

NISQ solution

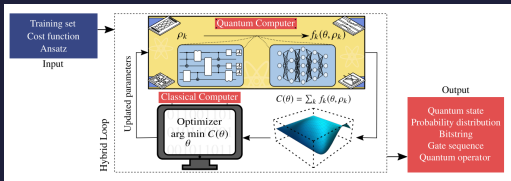


Figure: Variational Quantum Algorithm (VQA) for NISQ devices

- Main challenge: finding short-depth circuits for useful problems.
- Second challenge: no complexity proofs of advantage.
- Candidates: variational algorithms, quantum simulation.
- So far no success.
- Error handling: Hardware improvements and error mitigation.

Applications of quantum computing



Application	p_L	Reference
Shor's algorithm (RSA-2048)	$\sim 10^{-12}$	Gidney and Ekerå (2021), <i>Quantum</i>
FeMoco	10^{-14} – 10^{-17}	Otten et al. (2023), QREChem
Other molecular chemistry	10^{-9} – 10^{-17}	Otten et al. (2023), QREChem
Fermi–Hubbard model	10^{-8} – 10^{-9}	Kan and Symons (2025), <i>npj Quantum Information</i>
High-T superconductors	$\sim 2 \times 10^{-9}$	Kan and Symons (2025), <i>npj Quantum Information</i>

Applications with provable advantage (and their target error rates)

NISQ: useful in niches. Scalable advantage needs QEC.

How to reduce the error rate? Hardware improvements

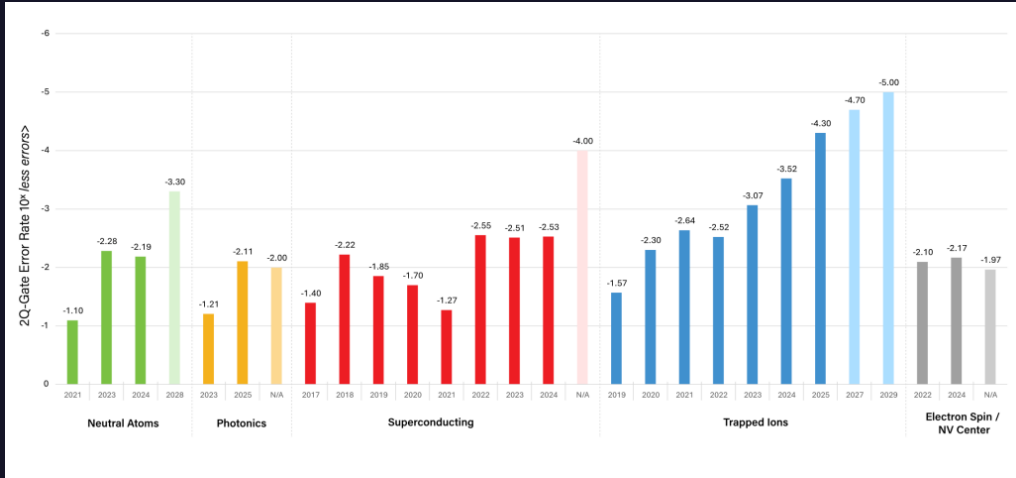


Figure: Improvement of 2Q-gate fidelity over the years, MIT Benchmarking Survey

Fault-tolerant quantum computation

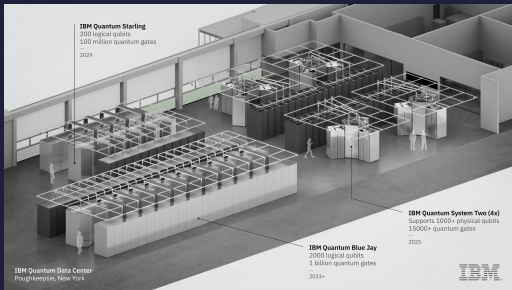


Figure: IBM's Fault-Tolerant Quantum Computer roadmap

- Physical errors cannot be reduced to arbitrarily low values.
- Other strategies to reduce errors?
- Classical computing strategies? Error correction
- FTQC idea: exploit error correction on quantum information.

How hard is it?

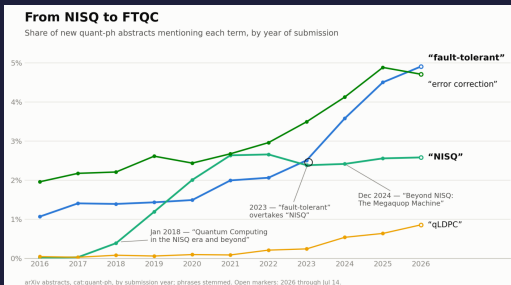


Figure: keywords in arXiv preprint submissions

- **2018:** Because of the **very hefty overhead cost** of quantum error correction, the era of fault tolerant quantum computing (FQTC) may still be rather distant. No one really knows how long it will take to get there. — John Preskill, Quantum Computing in the NISQ era and beyond
- **2026:** Preskill, founder of Oratomic, a neutral-atom FTQC start up.
- Talk's objectives:
 - What makes it hard?
 - Easier with recent developments.

Following outline

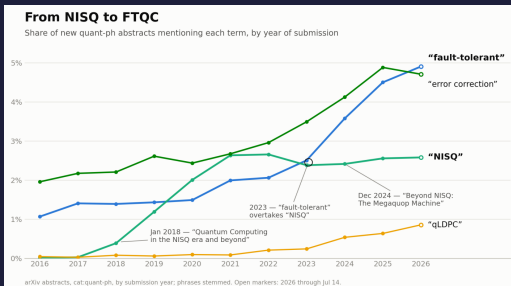


Figure: keywords in arXiv preprint submissions

- High-level view on FTQC
 - High-level view of QEC.
 - The surface code and qLDPC codes.
 - Resource estimates for FTQC
 - Experimental progress
 - Resource estimation
- Technical talk on classical and quantum error correction.

SECTION

2

QEC foundations

Classical error correction setting

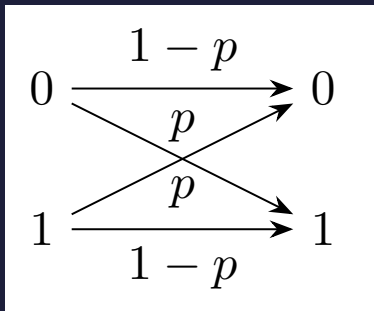


Figure: Bitflip (Binary symmetric channel)

- Information encoded in bits:
 $0 \oplus 0 = 0, \quad 1 \oplus 0 = 0 \oplus 1 = 1, \quad 1 \oplus 1 = 0$
- Bitstrings:
 $1010 \oplus 1110 = 0100$
- Noisy channel (environment).
- Bits can be corrupted by noise $0 \mapsto 1$
- Challenge: protect information from noise?

Classical coding: repetition

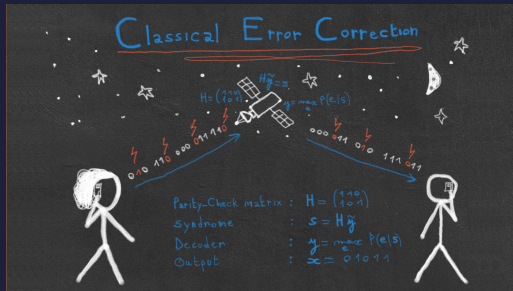


Figure: Sketch of classical error correction

- Encode: $0 \mapsto 000$
- Encode: $1 \mapsto 111$
- Majority vote
- Correct up to one error

General idea: Error correction

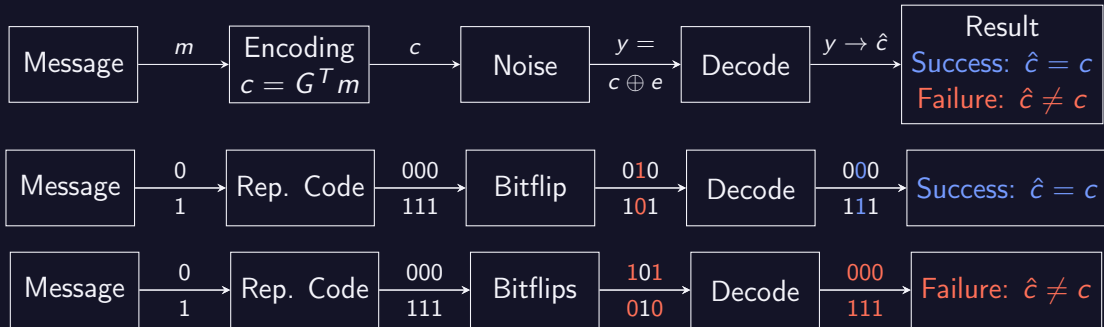


Figure: Error correction

Dissecting the repetition code

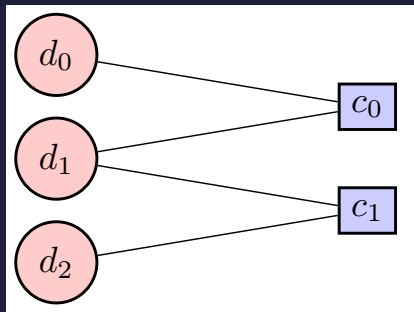


Figure: Tanner graph for 3-bit repetition code

- Parity check matrix: Error detection

$$H = \begin{array}{ccc|c} & d_0 & d_1 & d_2 \\ \hline c_0 & 1 & 1 & 0 \\ c_1 & 0 & 1 & 1 \end{array}$$

- Check c_0 : $d_1 \oplus d_2 = 0$
- Check c_1 : $d_2 \oplus d_3 = 0$
- Syndrome:

$$s = H(c \oplus e) = Hc \oplus He = He$$

- Error $\rightarrow$ nonzero syndrome

More general codes

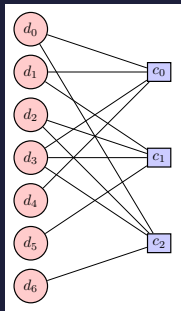


Figure: Tanner graph for the Hamming code

- $[n, k, d]$ code
 - n physical bits
 - k logical bits
 - d distance
 - Correct up to $t = \lfloor (d - 1)/2 \rfloor$ bitflips
- Hamming code: $[7, 4, 3]$ code.

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

More general codes

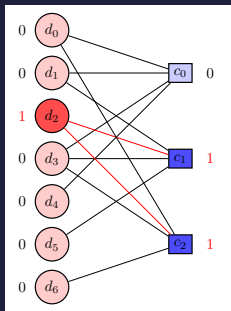


Figure: Tanner graph for the Hamming code

- $[n, k, d]$ code
 - n physical bits
 - k logical bits
 - d distance
 - Correct up to $t = \lfloor (d - 1)/2 \rfloor$ bitflips
- Hamming code: $[7, 4, 3]$ code.

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$



$$[n, k, d]$$

- k/n encoding rate
- d/n relative distance
- “Good codes”: $k/n, d/n = \Omega(1)$
- Easy construction: Random LDPC codes (sparse H)
- Easy decoding: belief propagation

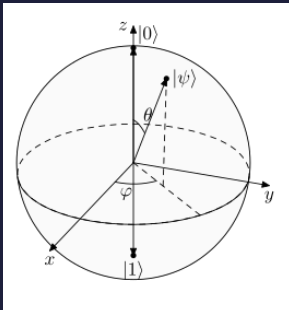


Figure: Bloch sphere representation of a qubit

- Qubit:

$$\alpha |0\rangle + \beta |1\rangle$$

- $\alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1$

- More generally

$$|\psi\rangle = \sum_{c \text{ bitstring}} \alpha_c |c\rangle$$

- Entanglement

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \neq |\psi_1\rangle \otimes |\psi_2\rangle$$

Measurement in Quantum Mechanics

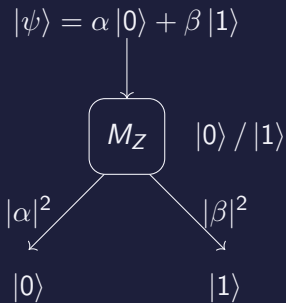


Figure: Meas. in the computational basis

- Measurement: projective, probabilistic.
- In arbitrary basis (importance of phase!):

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

Measurement in Quantum Mechanics



$$|\psi\rangle = \frac{(\alpha+\beta)}{\sqrt{2}} |+\rangle + \frac{(\alpha-\beta)}{\sqrt{2}} |-\rangle$$

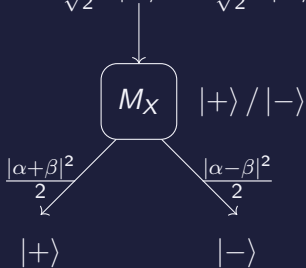


Figure: Meas. in the X basis

- Measurement: projective, probabilistic.
- In arbitrary basis (importance of phase!):

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

No-cloning theorem



Statement

- Impossible to copy an arbitrary quantum state $|\psi\rangle$.
- No unitary U such that

$$\forall |\psi\rangle, \quad U|\psi\rangle \otimes |0\rangle = |\psi\rangle \otimes |\psi\rangle$$

Implications

- Cannot copy quantum information
- Information when measured is lost
- Challenge for measuring syndromes.

Wootters & Zurek, Nature 299, 802–803 (1982)

Quantum errors: two axes

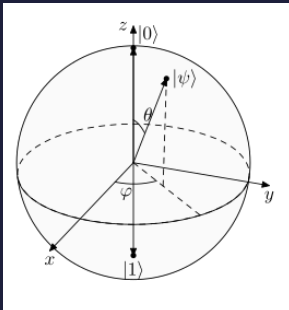


Figure: Bloch sphere representation of a qubit

- Bit flip: X :

$$X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle,$$

- Phase flip: Z

$$Z|+\rangle = |-\rangle, \quad Z|-\rangle = |+\rangle$$

- Combined: $Y = iXZ$

- Continuous errors:

$$U = u_I I + u_X X + u_Y Y + u_Z Z$$

- Error correction should protect against bit and phase flips.

Challenges for QEC

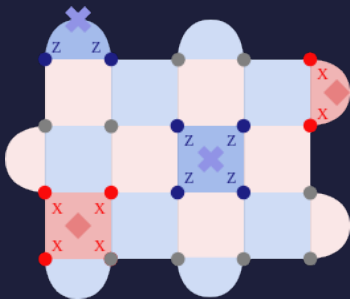


Figure: The Surface code

- Protect against bit and phase flips
- Finding codes: much harder than for classical.
- Checks are now multi-qubit Pauli operators:

$$K |\psi_L\rangle = |\psi_L\rangle$$

with $|\psi_L\rangle$ any quantum states

- Simplest example: the surface code
- $[[n = d^2, k = 1, d]]$ code.

QEC codes: Take Home Messages

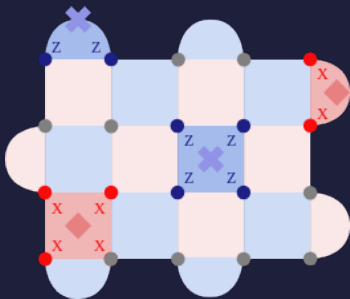


Figure: The Surface code

- Bit and phase flip error protection
- Restrictions on how to process quantum information
- More challenging than classical codes
- Existing code constructions: CSS codes
 - two classical codes H_Z and H_X parity check matrices
 - H_Z for bit flips
 - H_X for phase flips
 - CSS conditions: $H_Z H_X^T = 0$ (commuting checks)

What do you need to build a FT quantum computer?

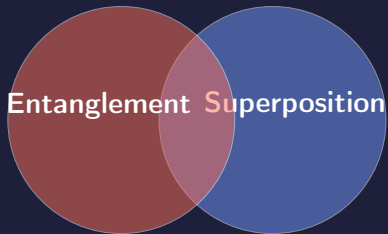


Figure: “Ingredient” for QC

- Low noise qubits / operations
- Perform logic
- Universal set of gates
- Clifford gates:
 - X bitflip: $X |0/1\rangle = |1/0\rangle$
 - Z phase-flip: $Z |\pm\rangle = \pm |\pm\rangle$
 - Hadamard: $H |0\rangle = |+\rangle$
 - Phase: $S |+\rangle = |+i\rangle$
 - CNOT $|+, 0\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$

Gottesman-Knill no-go theorem



Statement

- Clifford circuits:
 - $|0\rangle$ state preparation
 - Clifford gates
 - Pauli measurements
- Classically efficient simulation

Implications

- Clifford circuits are not universal
- No quantum advantage of Clifford circuits
- Need non-Clifford resources (“Magic”)
- Example $T = \sqrt{S} = \sqrt[4]{Z}$ gate
- Canonical extension: Clifford + T universality

Gottesman, ArXiv 9807006 (1998)

What do you need to build a FT quantum computer?

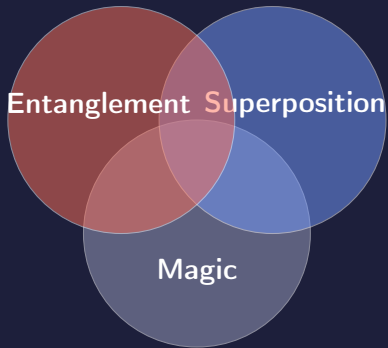


Figure: “Ingredient” for QC

- Low noise qubits / operations
- Perform logic
- Universal set of gates
- Clifford gates
- Gottesman-Knill no-go theorem.
- Clifford + T Universal

Eastin-Knill no-go theorem



Statement

- Transversal gates for a QEC code:

$$U_L = \prod_i U_i$$

- Universal gate sets (e.g. Clifford + T).
- Transversal gate set $\neq$ Universal gate set.

Implications

- Transversal gates: “Easy way” of doing gates
- Need other ways to perform universal gates on logical qubits
- Hard to compute on logical qubits
- Additional resource overhead

Eastin & Knill, Phys. Rev. Lett. 102, 110502 (2009)

Performing gates: Ways around

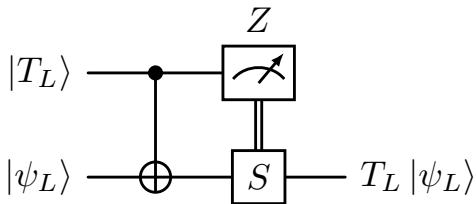


Figure: Magic state injection.

- Magic state distillation
- Lattice surgery
- Code switching
- Lead to additional resource overhead.

Performing gates: Ways around

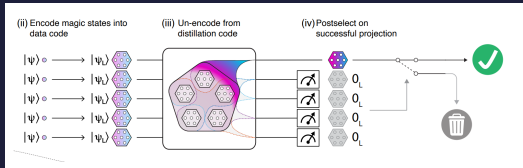


Figure: Magic state distillation.

- Magic state distillation
- Lattice surgery
- Code switching
- Lead to additional resource overhead.

Performing gates: Ways around

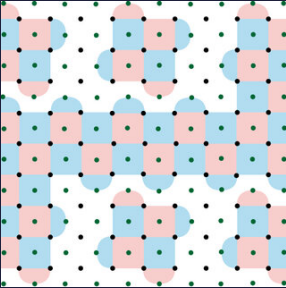


Figure: Lattice surgery.

- Magic state distillation
- Lattice surgery
- Code switching
- Lead to additional resource overhead.

Performing gates: Ways around

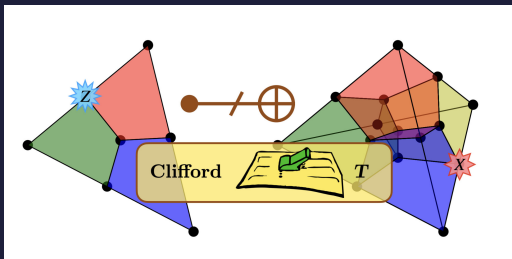


Figure: Code switching.

- Magic state distillation
- Lattice surgery
- Code switching
- Lead to additional resource overhead.

SECTION

3

Surface-code baseline

Why the surface code?

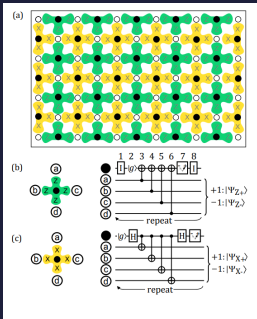


Figure: The Surface code

- Local checks
- 2D nearest-neighbor layout
- Highest circuit-level threshold ($\approx 1\%$)
- Mature decoding tools
- We know how to do FTQC with it

Code distance costs qubits

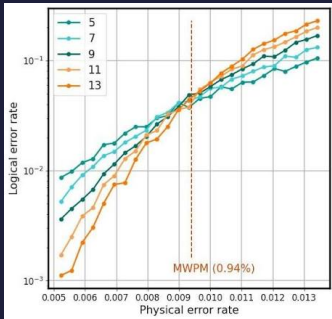


Figure: Threshold of the surface code

- Larger d : lower p_L

$$p_L \propto (p/p_{\text{th}})^{(d+1)/2}$$

- Code block: $O(d^2)$ qubits
- Teraquop regime ($p_L \lesssim 10^{-12}$), $d \approx 30$
- $d^2 \approx 1000$ qubits per logical qubit. . .
- FTQC leads to huge resource overhead

Surface code FTQC

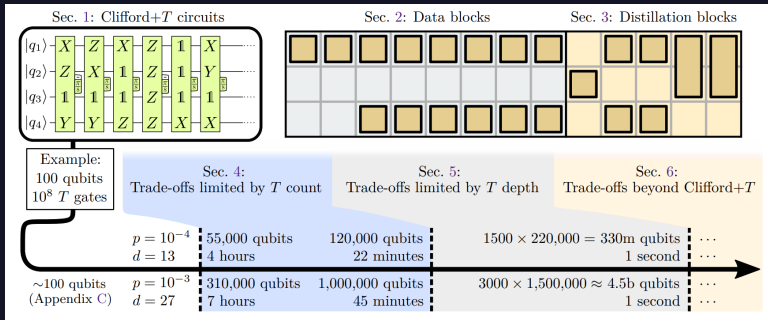


Figure: Surface code quantum computing, Litinski, Quantum 3, 128 (2019)

- Pauli-product measurements
- 2D lattice
- T -count limited
- Magic state distillation overheads

FTQC with surface code

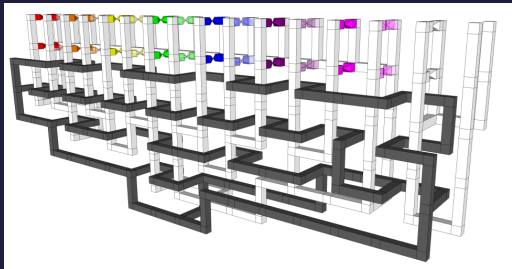


Figure: Space-time footprint of surface code QC

- Fault-tolerant quantum computing is possible
- Compatible with planar connectivity
- Huge overheads. . .
 - k logical qubits to $n \gg k$ physical qubits
 - FT quantum processing
 - Magic-state factories
- Resource estimation of FTQC!

SECTION

4

Resource estimation

Why resource estimation?

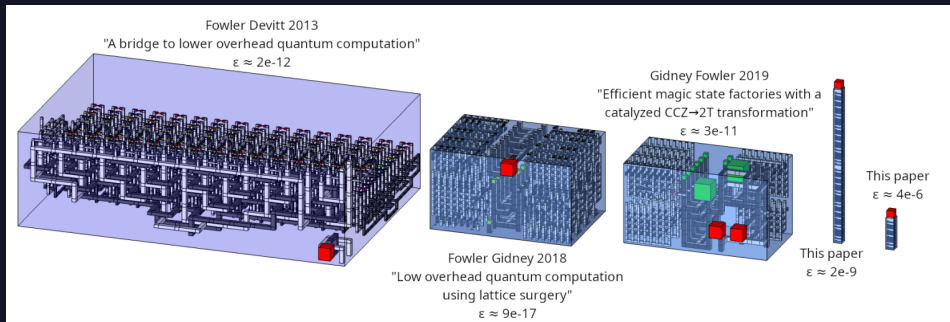


Figure: Architecture improvements, Gidney et al., ArXiv 2409.17595 (2024)

- Resources: qubits, time
- Motivate experimental developments
- Anticipate bottlenecks and trade-offs
- Baseline for architecture improvements

Resource estimation assumptions: Objectives

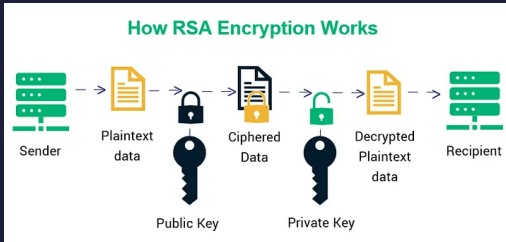


Figure: RSA-2048

- Useful
- Industry relevant
- Complexity theory guarantees
- A quantum circuit implementation
- Cryptographically relevant problems (RSA-2028, ECC-256)

Resource estimation assumptions: Other assumptions

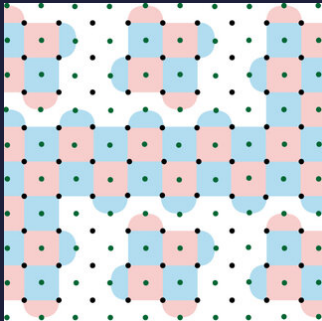


Figure: 2D layout

- Qubit connectivity
 - Arbitrary connectivity
 - 2D nearest-neighbor grid
- Clock-cycle time and reaction time
 - Hardware-dependent
 - Decoder / code dependent
- FT Gate implementations

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

- 2D planar connectivity
- 10^{-3} gate error
- $1 \mu\text{s}$ cycle; $10 \mu\text{s}$ reaction
- Surface code
- Lattice surgery
- Magic state distillation

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

- Full-stack simulation
- Single-chip architecture
- Resource overhead: is it realistic?
- A landmark point
- Benchmark improvements
- Architecture co-design

Example: Gidney, Ekerå(2021)



How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney¹ and Martin Ekerå^{2,3}

Figure: Gidney, Ekerå, Quantum 5, 433 (2021)

Assumptions

- Realistic?
- Correct?
- What can be improved?
- Surface code really the best?
- Noise model?
- 2D grid?

Can we improve on it?

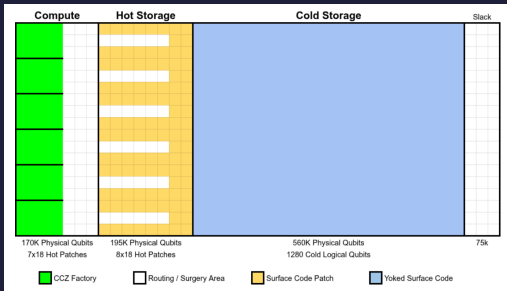


Figure: Architecture improvements

- Same assumptions
- Improvements on quantum algorithms
- Other code: “Yoked surface code” for storage
- Magic-state cultivation over distillation
- $< 1M$ qubits, < 1 week
- Guidance for experimental realizations
- Can we do much better?

Can we improve on it?

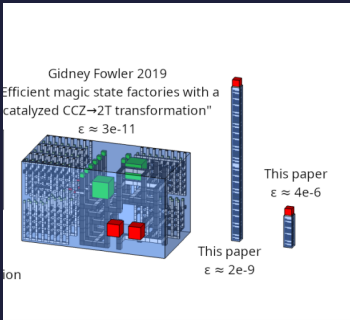


Figure: Magic state cultivation

- Same assumptions
- Improvements on quantum algorithms
- Other code: “Yoked surface code” for storage
- Magic-state cultivation over distillation
- $< 1M$ qubits, < 1 week
- Guidance for experimental realizations
- Can we do much better?

Can we improve on it?



Figure: Resource estimate (qubit count over the years)

- Same assumptions
- Improvements on quantum algorithms
- Other code: “Yoked surface code” for storage
- Magic-state cultivation over distillation
- $< 1M$ qubits, < 1 week
- Guidance for experimental realizations
- Can we do much better?

SECTION

5

Recent progress: Experiments

Experimental realizations

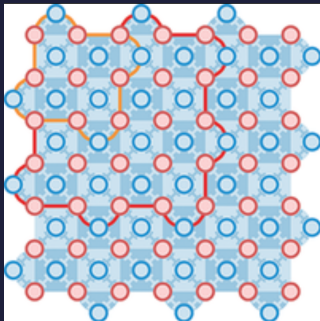


Figure: Surface code

- Clear path from resource estimations
- Target error rate
- Quantum error correction demonstration
- Logical gate primitives.

From break-even to below-threshold QEC

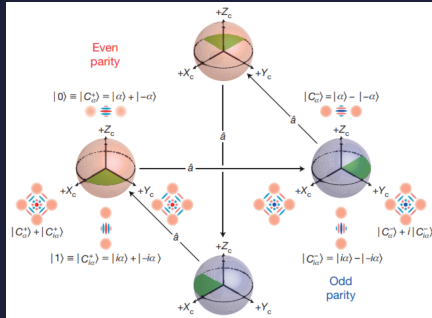


Figure: Cat qubit code beyond break-even

- Break-even demonstration $p_L < p$
- Below-threshold operation
- On different hardware platforms:
 - Superconducting qubits, $\Lambda \approx 2$, real-time decoding
 - Trapped ions, $\Lambda \approx 12$

From break-even to below-threshold QEC

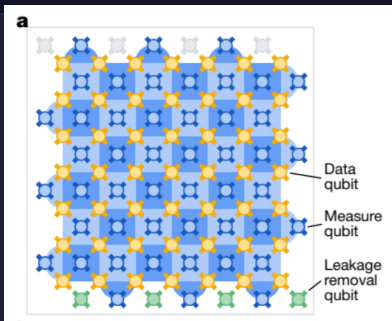


Figure: Below-threshold QEC with surface codes and superconducting platforms

- Break-even demonstration $p_L < p$
- Below-threshold operation
- On different hardware platforms:
 - Superconducting qubits, $\Lambda \approx 2$, real-time decoding
 - Trapped ions, $\Lambda \approx 12$

From break-even to below-threshold QEC

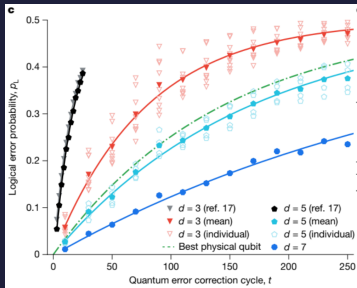


Figure: Below-threshold QEC with surface codes and superconducting platforms

- Break-even demonstration $p_L < p$
- Below-threshold operation
- On different hardware platforms:
 - Superconducting qubits, $\Lambda \approx 2$, real-time decoding
 - Trapped ions, $\Lambda \approx 12$

From break-even to below-threshold QEC

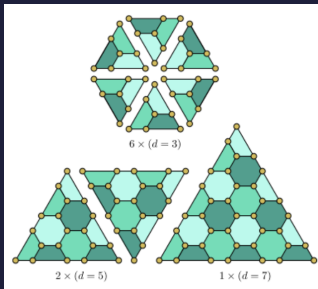


Figure: Below-threshold QEC with color codes and trapped ions

- Break-even demonstration $p_L < p$
- Below-threshold operation
- On different hardware platforms:
 - Superconducting qubits, $\Lambda \approx 2$, real-time decoding
 - Trapped ions, $\Lambda \approx 12$

Experimental logical gates

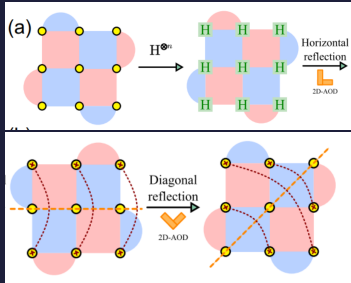


Figure: Transversal gates on neutral-atom qubits

- Logical memories
- Demonstrations of logical gates
 - Transversal gates
 - Lattice surgery
 - Magic state distillation
 - Magic state cultivation
 - Code switching

Experimental logical gates



Figure: Lattice surgery with trapped ions

- Logical memories
- Demonstrations of logical gates
 - Transversal gates
 - Lattice surgery
 - Magic state distillation
 - Magic state cultivation
 - Code switching

Experimental logical gates

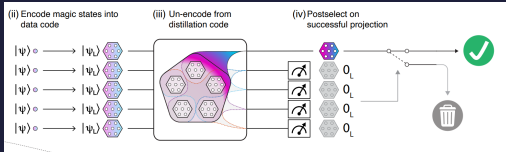


Figure: Magic-state distillation with cold atoms

- Logical memories
- Demonstrations of logical gates
 - Transversal gates
 - Lattice surgery
 - Magic state distillation
 - Magic state cultivation
 - Code switching

Experimental logical gates

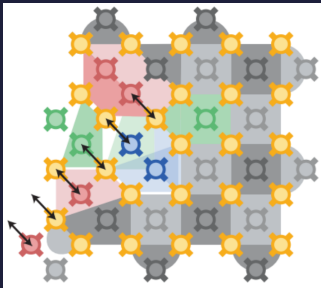


Figure: Magic-state cultivation

- Logical memories
- Demonstrations of logical gates
 - Transversal gates
 - Lattice surgery
 - Magic state distillation
 - Magic state cultivation
 - Code switching

Experimental logical gates

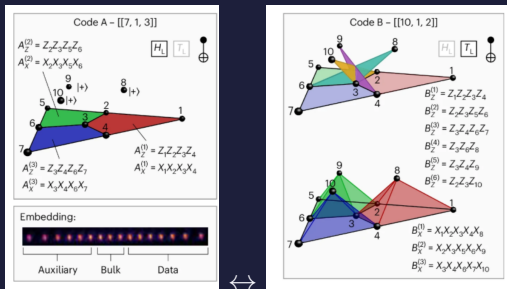


Figure: $[[7, 1, 3]] \leftrightarrow [[10, 1, 2]]$ Code switching

- Logical memories
- Demonstrations of logical gates
 - Transversal gates
 - Lattice surgery
 - Magic state distillation
 - Magic state cultivation
 - Code switching

Entering the “Early-FTQC” era

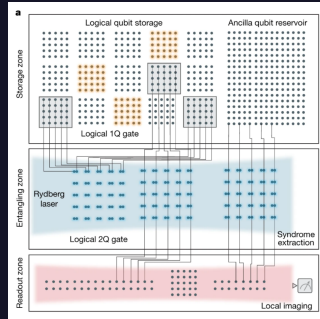


Figure: Early FTQC platforms, Bluvstein et al., Nature 626, 58–65 (2024)

- 48 logical qubits ($[[8, 3, 2]]$)
- transversal CZ, CCZ gates ...
- Milestone demonstrations
- not yet application scale

SECTION

6

Recent progress: Theory

Theoretical advances in QEC

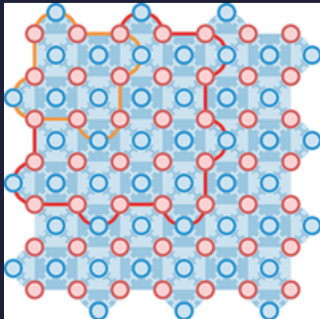


Figure: Surface code

- Goal reducing FTQC overheads
- Ex: Magic-state distillation $\rightarrow$ Cultivation
- Surface code $[[d^2, 1, d]]$ scaling issue. . .

$$\frac{k}{n} = \frac{1}{d^2} \xrightarrow{d \rightarrow \infty} 0$$

- Reducing errors $\implies$ reducing the rate
- Classical LDPC codes $[n, \Theta(n), \Theta(n)]$
- Existence of good qLDPC codes?

The Quest for “Good” qLDPC codes

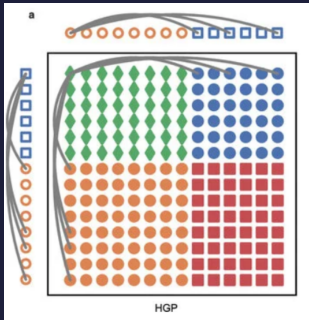


Figure: Hypergraph product codes

- 2013: Hypergraph product codes $[[n, \Theta(n), \Theta(n^{\frac{1}{2}})]]$
- 2021: Balanced-product codes $[[n, \Theta(n^{\frac{4}{5}}), \Omega(n^{\frac{3}{5}})]]$
- 2021-2022: Lifted-product codes $[[n, \Theta(n), \Theta(n)]]$
- 2022: Quantum Tanner codes $[[n, \Theta(n), \Theta(n)]]$
- Asymptotically good qLDPC codes exists. Explicit constructions?

Panteleev & Kalachev, Arxiv 2111.03654 (2021)

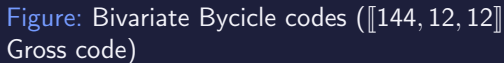
Other efficient qLDPC code constructions



$$H_X = \begin{pmatrix} F_0 & F_1 & F_2 & F_3 & F_4 & F_5 & G_0 & G_1 & G_2 & G_3 & G_4 & G_5 \\ F_5 & F_0 & F_1 & F_2 & F_3 & F_4 & G_5 & G_0 & G_1 & G_2 & G_3 & G_4 \\ F_4 & F_5 & F_0 & F_1 & F_2 & F_3 & G_4 & G_5 & G_0 & G_1 & G_2 & G_3 \end{pmatrix}$$
$$H_Z = \begin{pmatrix} G_0^T & G_5^T & G_4^T & G_3^T & G_2^T & G_1^T & F_0^T & F_5^T & F_4^T & F_3^T & F_2^T & F_1^T \\ G_1^T & G_0^T & G_5^T & G_4^T & G_3^T & G_2^T & F_1^T & F_0^T & F_5^T & F_4^T & F_3^T & F_2^T \\ G_2^T & G_1^T & G_0^T & G_5^T & G_4^T & G_3^T & F_2^T & F_1^T & F_0^T & F_5^T & F_4^T & F_3^T \end{pmatrix}$$

Figure: Affine Permutation Matrix (APM) codes

- Bivariate Bycycle codes:
 - Ex: $[[144, 12, 12]]$
 - High threshold code family
- Recent APM codes
 - Ultra-high encoding rates
 - $k/n \geq 1/2$
 - Ex: $[[9216, 4612, \leq 48]]$
- Excellent memories, how to do gates?



- Promise
 - Rate
 - Asymptotic scaling
- Blocker
 - Decoding complexity
 - Gate toolbox
 - Hardware mapping
- Challenge: Integrate in FTQC architectures

Beyond lattice surgery

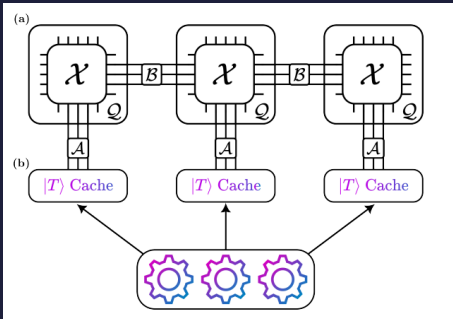


Figure: Extractor architecture for qLDPC codes

- “qLDPC” surgery challenge
 - $k \geq 1$ logical qubits per block
 - Arbitrary Pauli measurements?
- Generalizations of lattice surgery
 - 2022: HGP Surgery,
Cohen et al., Science Advances, 8, 20 (2022)
 - 2023: CSS/qLDPC surgery
Cowtan & Burton, Quantum 8, 1344 (2024)
 - 2026: Parallel surgery
Cowtan et al., PRX Quantum 7, 020325 (2026)
 - 2026: Clifford-measurement surgery
Chang et al., ArXiv 2610.03920 (2026)

Recent resource estimations

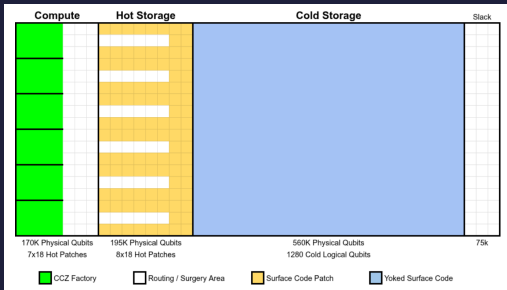


Figure: Superconducting architecture

- Gidney et al., (2025) for superconducting qubits
- Assumptions:
 - 2D connectivity
 - Surface code
 - $1 \mu\text{s}$ cycle
 - Lattice surgery

Recent resource estimations

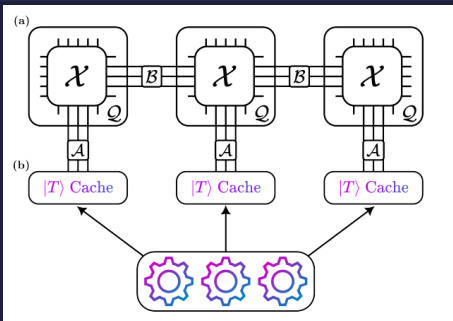


Figure: Modular qLDPC architecture

- Atomic platforms
- New Assumptions:
 - 2D connectivity
 - Surface code
 - $1\ \mu\text{s}$ cycle
 - Lattice surgery

Recent resource estimations

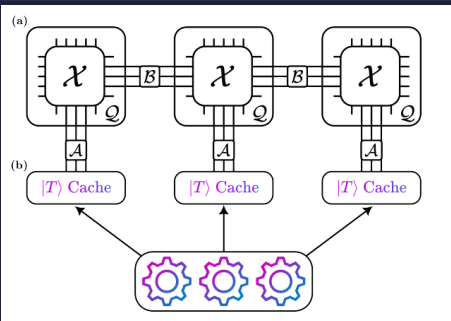


Figure: Modular qLDPC architecture

- Atomic platforms
- New Assumptions:
 - 2D-connectivity, Arbitrary connectivity
 - Surface code, qLDPC codes
 - $1\mu s$ -cycle, 1 ms cycle
 - Lattice surgery, qLDPC surgery

The Pinnacle Architecture

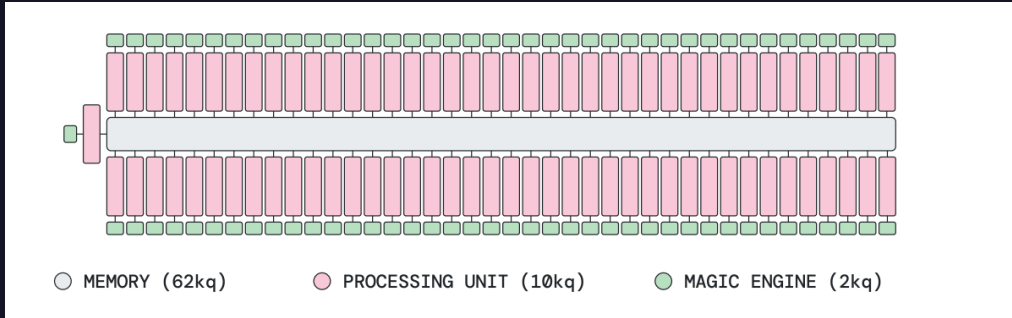


Figure: The Pinnacle architecture for qLDPC codes, Webster et al., ArXiv 2602.11457 (2026)

- Modular architecture
- Less than 100k physical qubits
- 3 code families
- Space-time trade-offs

The Pinnacle Architecture

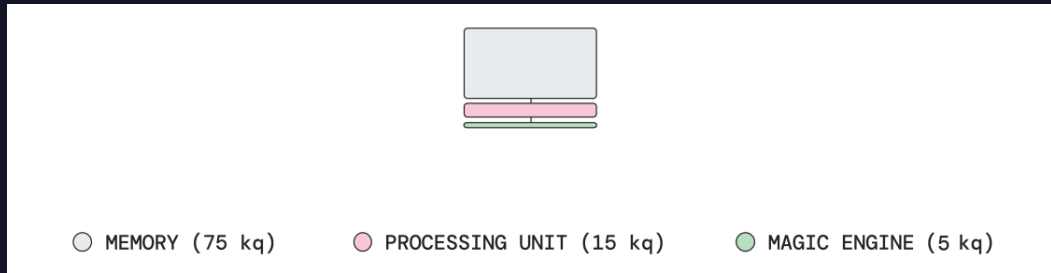


Figure: The Pinnacle architecture for qLDPC codes, Webster et al., ArXiv 2602.11457 (2026)

- Modular architecture
- 3 code families
- Less than 100k physical qubits
- Space-time trade-offs

Oratomic's architecture

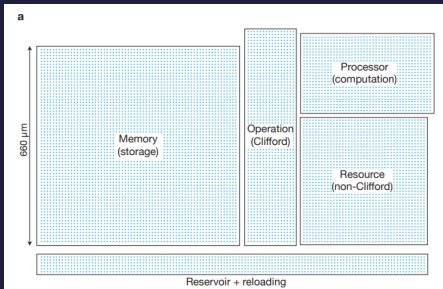


Figure: Oratomic's architecture for qLDPC codes

- Break RSA-2048
- qLDPC codes (LP) ($[[278, 1480, \leq 24]]$)
- With less than 10k physical qubits!
- ... in >100 years
- Space overhead seems feasible, but time overhead is huge
- Are these assumptions realistic?

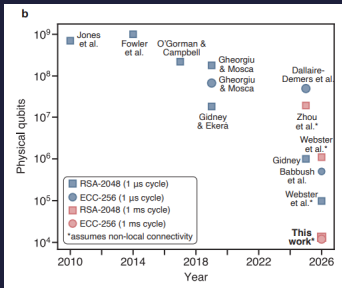


Figure: Oratomic's architecture for qLDPC codes

- Break RSA-2048
- qLDPC codes (LP) ($\llbracket 278, 1480, \leq 24 \rrbracket$)
- With less than 10k physical qubits!
- ... in >100 years
- Space overhead seems feasible, but time overhead is huge
- Are these assumptions realistic?

Other (more realistic) architectures

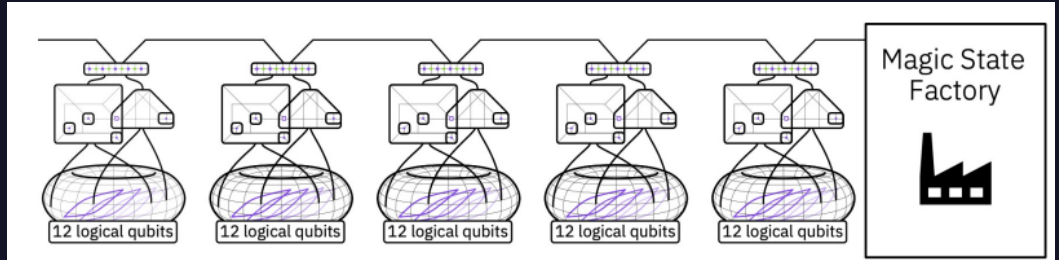


Figure: IBM's architecture with $[[288, 12, 18]]$ codes

Other (more realistic) architectures

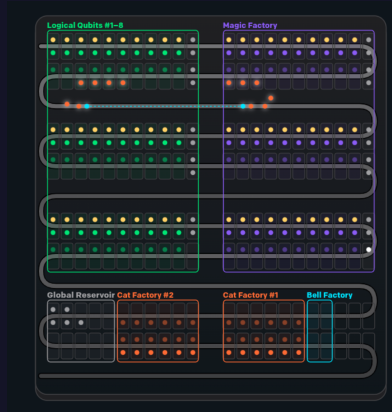


Figure: Ion Q's walking cat architecture with $[[90, 8, 10]]$ codes

SECTION

7

Conclusion

Main takeaways

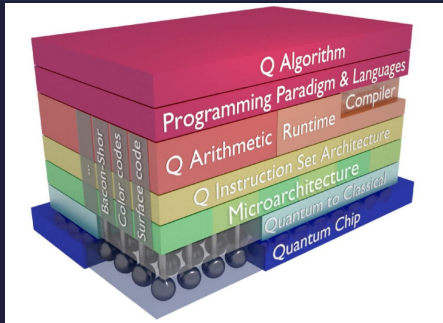


Figure: Stacks of a FT Quantum Computer

- NISQ and FTQC are very different
- QEC leads to huge overhead
- No-go theorems make it non-trivial
- QEC is now an experimental reality
- Many approaches are being explored
- FTQC resource estimations



What is now established

- Logical error suppression: demonstrated
- Core FT primitives: demonstrated
- FTQC engineering: no longer hypothetical

What is still hard

- Full-stack overheads: still large
- Non-Clifford resources: dominant bottleneck
- Decoder-control-hardware integration

Why am I optimistic about FTQC?

- Multiple pathways progressing in parallel
- Resource estimates improving quickly
- But what for?

FTQC: No longer a question of principle.
It is a system engineering problem.



Qurio

INRIA • TELECOM PARIS • IP PARIS

Thank you for your attention – Questions are welcome