WE BUILD QUANTUM COMPUTERS

Jan Goetz

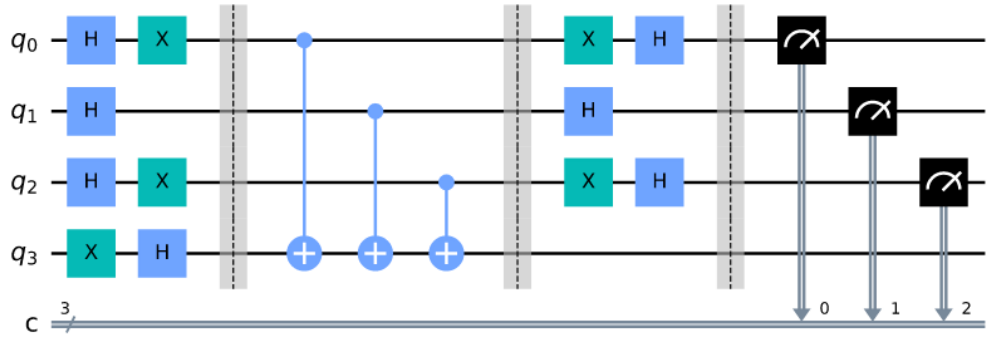
www.meetiqm.com

Lecture notes on PHYS-C0254 Quantum Circuits

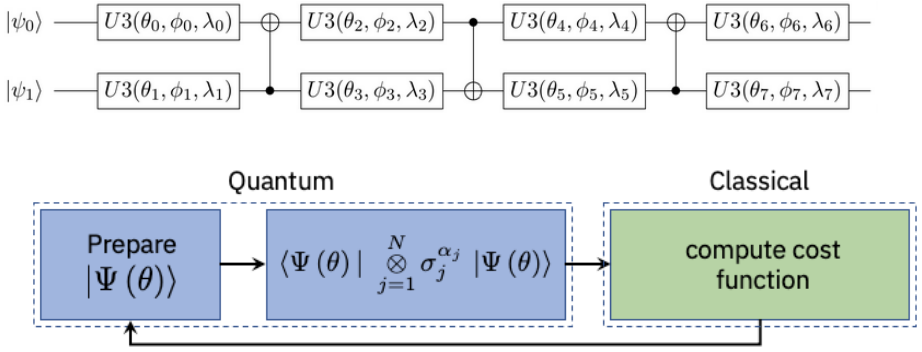
Short recap from last week

11. Quantum algorithms
    a. Deutsch-Josza Algorithm
    b. Parameterised circuits and VQE

a.



b.



IQM

Agenda for lectures 7-12

7. Quantization of electrical networks
    a. Harmonic oscillator: Lagrangian, eigenfrequency
    b. Transfer step: LC oscillator, Legendre transform to Hamiltonian
    d. Quantization of oscillators
8. Superconducting quantum circuits
    a. Qubits: Transmon qubit, Charge qubit, Flux qubit 1st DiVincenzo criteria
    b. Circuit-QED: Rabi model
    c. Rotating Wave approximation: Jaynes-Cummings model
9. Single-qubit operations:
    a. Initialization 2nd DiVincenzo criteria
    b. Readout 5th DiVincenzo criteria
    c. Control:T1, T2 measurements, Randomized benchmarking 3rd DiVincenzo criteria
10. Two-qubit operations: Architectures for 2-qubit gates 4th DiVincenzo criteria
    a. iSWAP
    b. cPhase
    c. cNot
11. Quantum algorithms
    a. Deutsch-Josza Algorithm
    b. Parameterised circuits and VQE
12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

General challenge: Scaling up

- Creating a single quantum circuit with high accuracy is simple. Scaling to millions is extremely challenging.

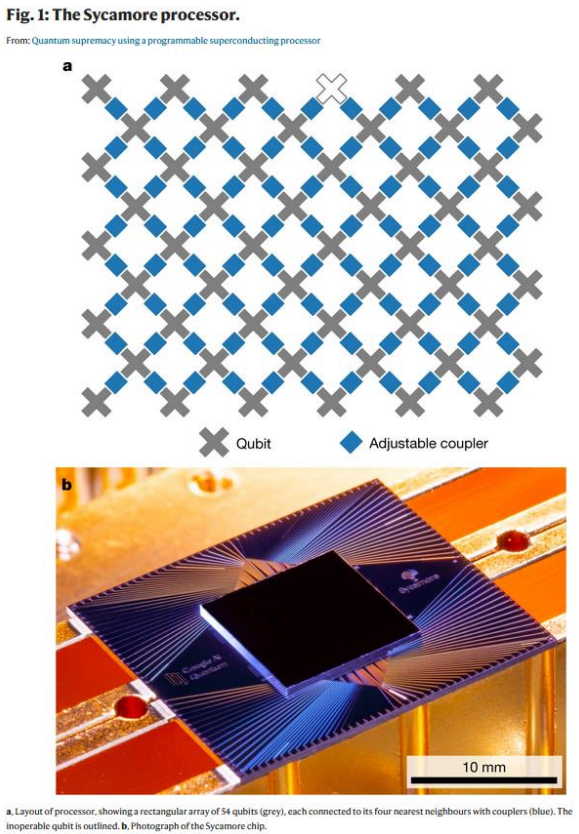
- To compensate for the errors of the system, one can detect and correct them. However, error correction creates a qubit overhead of 100 to 1000.

- Currently known algorithms require thousands – millions of qubits. Currently available processors contain less than 100 qubits. This is called the Software-Hardware gap.

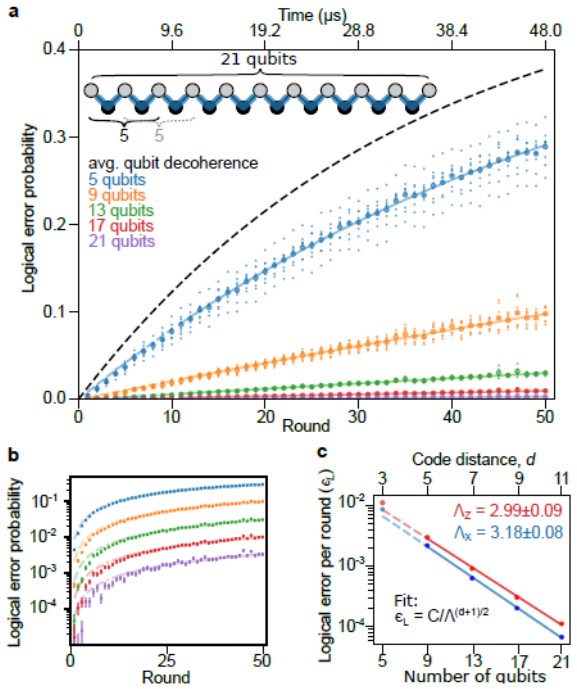
Agenda for today

12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

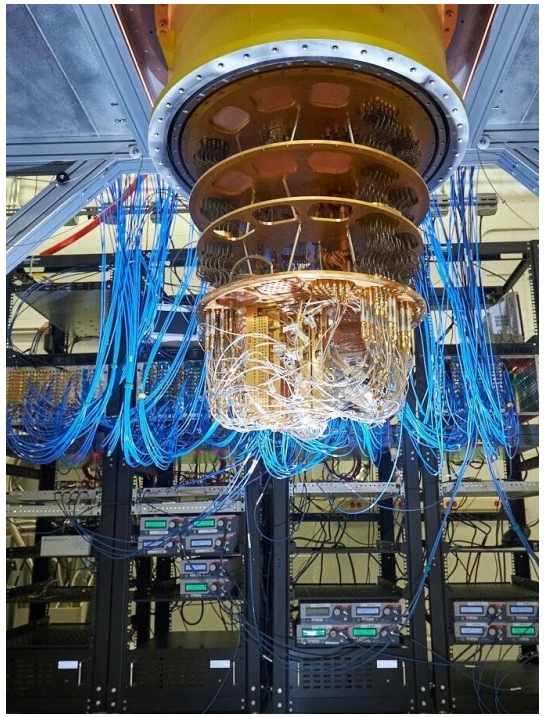
a.



b.



c.



IQM

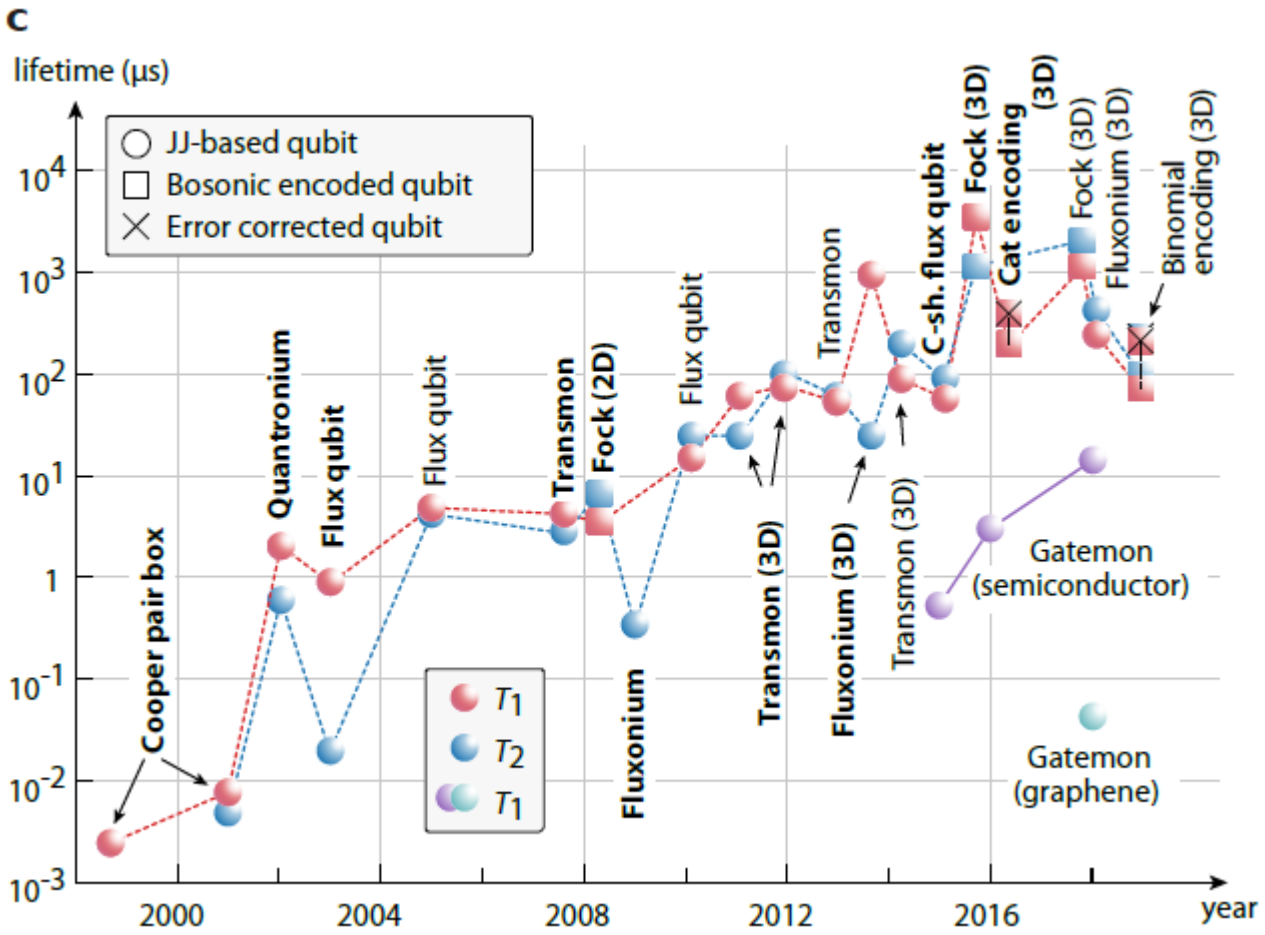
General note: SW- HW gap

- Even though qubit count and quality have been increasing over the years, we are still far away from closing the gap between SW requirements and HW accessibility.

- A big milestone was reaching quantum supremacy by Google. They created an arbitrary circuit that is hard to simulate classically.

- Bringing usefulness into the equation creates further overhead. Reaching quantum advantage requires probably thousands of qubits.

Qubit quality (solid state qubits)



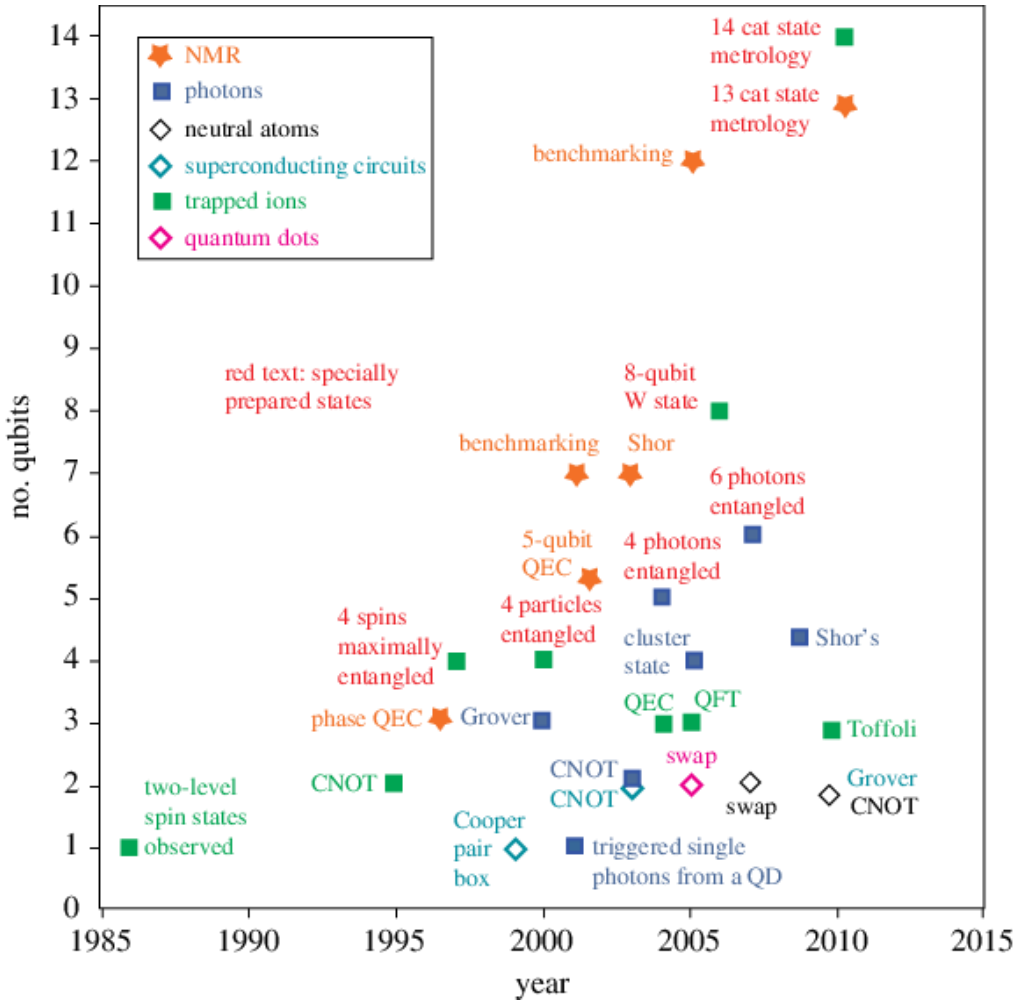
The qubit lifetime has been improving over the years.

However, in the current years, no real new breakthrough has been achieved.

This raises the question: “What fundamentally limits the lifetime of superconducting qubits?”

arXiv:1905.13641v3

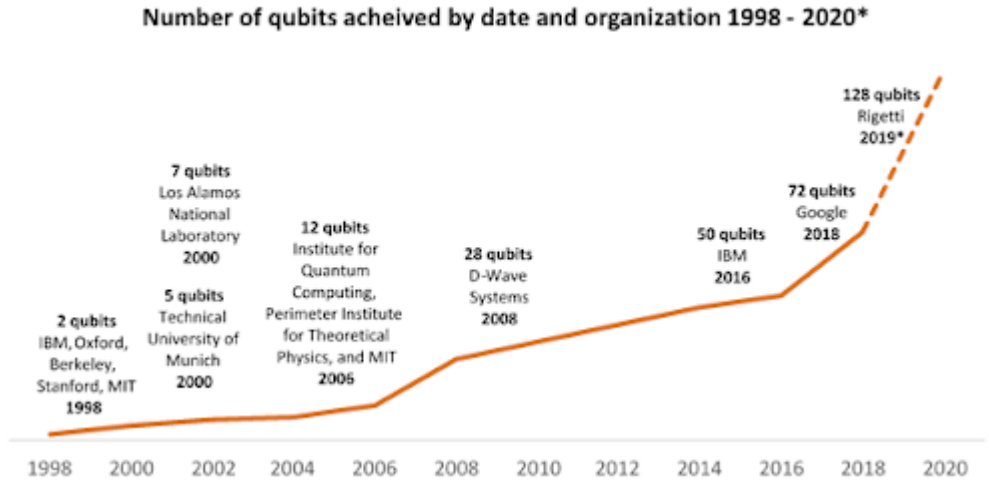
Qubit count



The qubit count has been increasing over the years.

However, certain announced milestones have not been reached.

Generally, only increasing the number of qubits does not help. One needs to be able to perform high quality gates as well.



IQM

10.1098/rsta.2011.0352

Gate quality (superconducting qubits)

Table 1 State of the art high-fidelity two-qubit gates in superconducting qubits

Acronym[a]	Layout[b]	First demonstration [Year]	Highest fidelity [Year]	Gate time
CZ (ad.)	T–T	DiCarlo et al. (72) [2009]	99.4%† Barends et al. (3) [2014]	40 ns
			99.7%† Kjaergaard et al. (73) [2020]	60 ns
$\sqrt{\text{iSWAP}}$	T–T	Neeley et al. (81)° [2010]	90%⋆ Dewes et al. (74) [2014]	31 ns
CR	F–F	Chow et al. (75) [2011]	99.1%† Sheldon et al. (5) [2016]	160 ns
$\sqrt{\text{bSWAP}}$	F–F	Poletto et al. (76) [2012]	86%⋆ ibid.	800 ns
MAP	F–F	Chow et al. (77) [2013]	87.2%⋆ ibid.	510 ns
CZ (ad.)	T–(T)–T	Chen et al. (56) [2014]	99.0%† ibid.	30 ns
RIP	3D F	Paik et al. (78) [2016]	98.5%† ibid.	413 ns
$\sqrt{\text{iSWAP}}$	F–(T)–F	McKay et al. (79) [2016]	98.2%† ibid.	183 ns
CZ (ad.)	T–F	Caldwell et al. (80) [2018]	99.2%† Hong et al. (6) [2019]	176 ns
CNOT_L	BEQ-BEQ	Rosenblum et al. (13) [2018]	∼99%□ ibid.	190 ns
CNOT_{T-L}	BEQ-BEQ	Chou et al. (82) [2018]	79%⋆ ibid.	4.6 µs

Gates ordered by year of first demonstration. Gate time is for the highest fidelity gate.
[a]Full names: CZ (ad.): Adiabatic controlled phase, $\sqrt{\text{iSWAP}}$: square-root of the iSWAP, CR: Cross-resonance, $\sqrt{\text{bSWAP}}$: Square-root of the Bell-Rabi SWAP, MAP: Microwave activated phase, RIP: Resonator induced phase gate, CNOT_L: Logical CNOT, CNOT_{T-L}: Teleported logical CNOT.
[b]F is short 'fixed frequency', T is short for 'tunable'. For all non-bosonic encoded qubit gates, the qubits were of the transmon variety (except for the first demonstration of $\sqrt{\text{iSWAP}}$, using phase qubits, and first demonstration of CR which used capacitively shunted flux qubits). Terms in parenthesis is a coupling element. '3D F' is short for a fixed frequency transmon qubit in a three-dimensional cavity. 'BEQ' is short for bosonic encoded qubit (see Sec. 2.4).
°Implemented with phase qubits.
†Determined by interleaved randomized Clifford benchmarking (70).
□Determined by repeated application of the gate to various input states and observing state fidelity decay as function of applied gates. See (13) for details.
⋆Determined by quantum process tomography.
Gates implemented on flux-tunable qubits.
All-microwave gates.
Combination of tunable and fixed frequency components.
Gates on bosonic encoded qubits.

The gate quality has been increasing over the years.

However, to reach error-correction limits at scale, further improvement is needed.

Generally, finding an efficient and high-quality implementation of CNOT gates is challenging.

arXiv:1905.13641v3

Reaching sufficient error rates

Logic gates at the surface code threshold: Superconducting qubits poised for fault-tolerant quantum computing

R. Barends,[1,*] J. Kelly,[1,*] A. Megrant,[1] A. Veitia,[2] D. Sank,[1] E. Jeffrey,[1] T. C. White,[1] J. Mutus,[1] A. G. Fowler,[1,3] B. Campbell,[1] Y. Chen,[1] Z. Chen,[1] B. Chiaro,[1] A. Dunsworth,[1] C. Neill,[1] P. O'Malley,[1] P. Roushan,[1] A. Vainsencher,[1] J. Wenner,[1] A. N. Korotkov,[2] A. N. Cleland,[1] and John M. Martinis[1]

[1]*Department of Physics, University of California, Santa Barbara, CA 93106, USA*
[2]*Department of Electrical Engineering, University of California, Riverside, CA 92521, USA*
[3]*Centre for Quantum Computation and Communication Technology, School of Physics, The University of Melbourne, Victoria 3010, Australia*

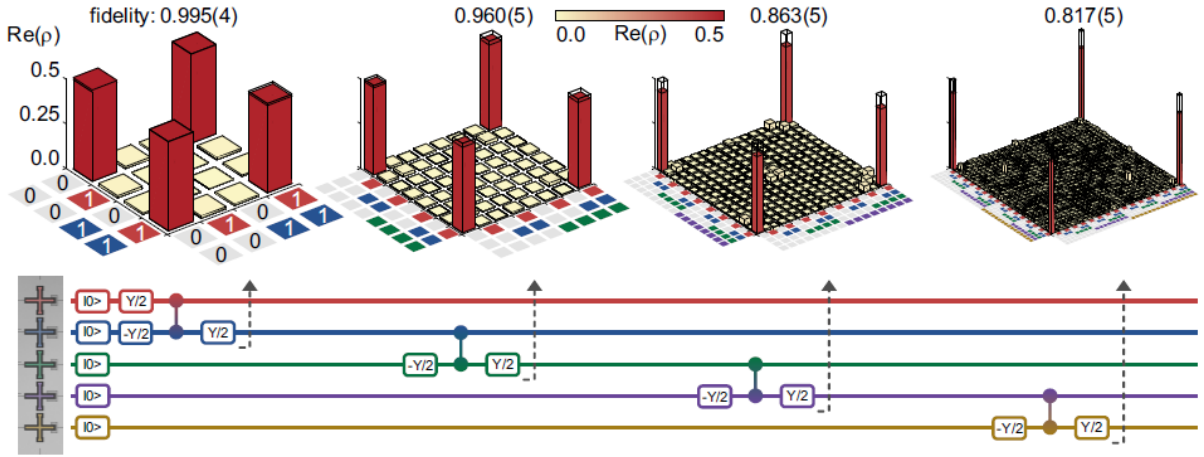


FIG. 4: **Quantum state tomography and generation of the GHZ state.** Top row: Real part of the density matrix ρ for the $N = 2$ Bell state and the $N = 3$, 4 and 5 GHZ states, measured by quantum state tomography. Ideal density matrix elements are transparent, with value 0.5 at the four corners. Bottom row: Algorithm used to construct the states. See Supplementary Information for $\mathrm{Im}(\rho)$, the Pauli operator representation, and the full gate sequence, which includes Hahn spin-echo pulses.

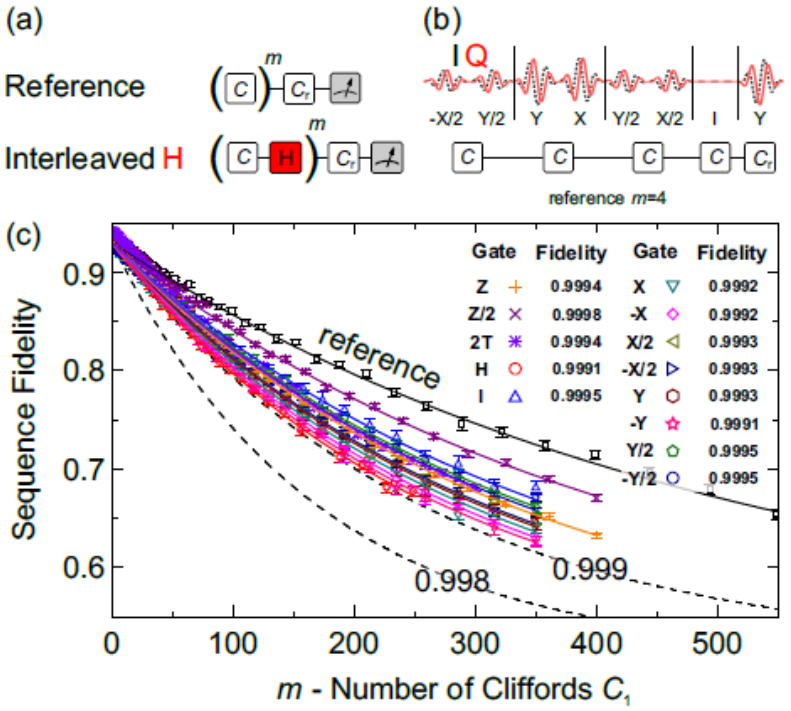


FIG. 2: **Single qubit randomised benchmarking.** (a) A reference experiment is performed by generating a sequence of m random Cliffords, which are inverted by the recovery Clifford C_r. A specific gate (H) is tested using a sequence that interleaves H with m random Cliffords. The difference between interleaved and reference decay gives the gate fidelity. (b) Representative pulse sequence for a set of four Cliffords and their recovery, generated with π and $\pi/2$ rotations about X and Y, displaying both the real (I) and imaginary (Q) microwave pulse envelopes before up-conversion by quadrature mixing to the qubit frequency. (c) Randomised benchmarking measurement for the set of single-qubit gates for qubit Q_2, plotting reference and gate fidelities as a function of the sequence length m; the fidelity for each value of m was measured for $k = 40$ different sequences. The fit to the reference set yields an average error per Clifford of $r_{\mathrm{ref}} = 0.0011$, consistent with an average gate fidelity of $1 - r_{\mathrm{ref}}/1.875 = 0.9994$ (Supplementary Information). The dashed lines indicate the thresholds for exceeding gate fidelities of 0.998 and 0.999. The fidelities for each of the single-qubit gates are tabulated in the legend, we find that all gates have fidelities greater than 0.999. Standard deviations are typically $5 \cdot 10^{-5}$.

Quantum supremacy

Article | Published: 23 October 2019

Quantum supremacy using a programmable superconducting processor

Frank Arute, Kunal Arya, [...] John M. Martinis

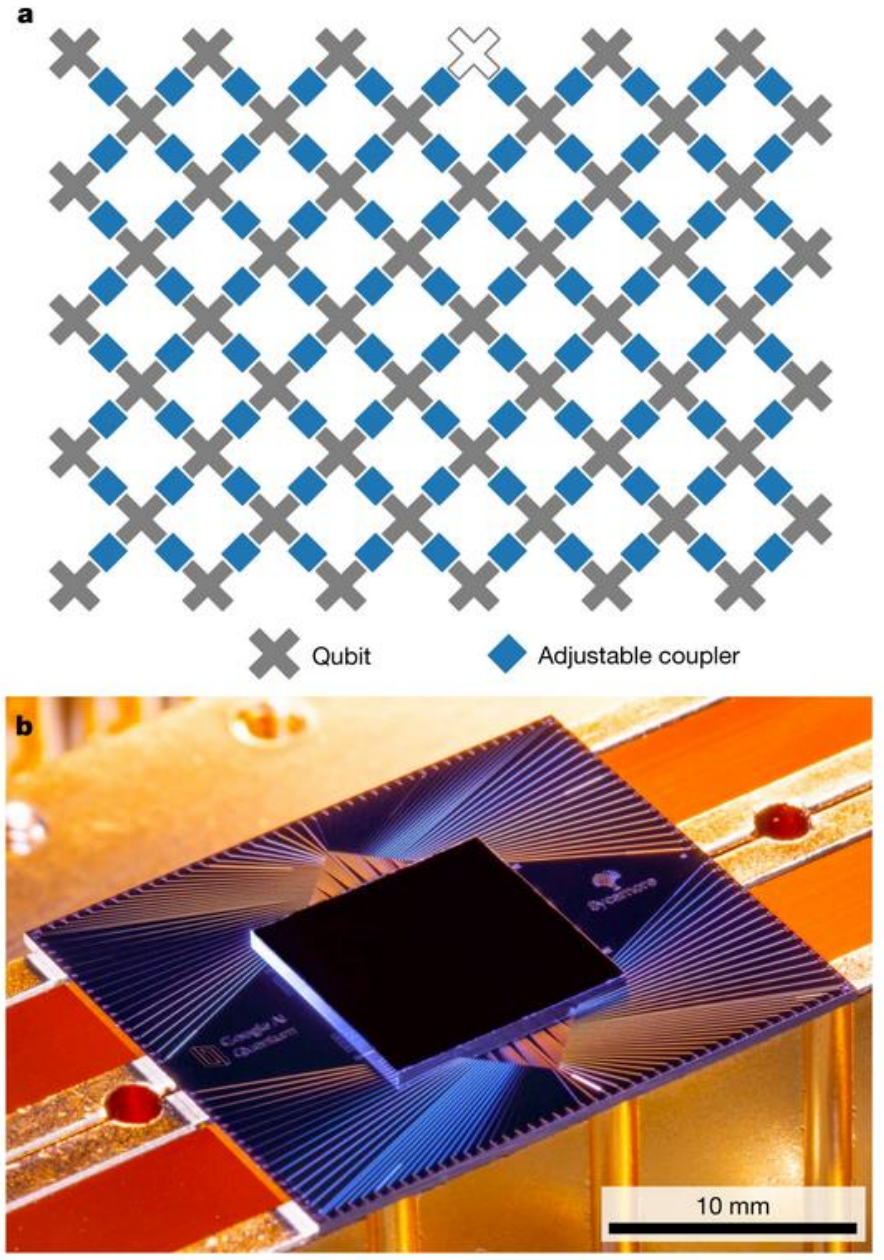
Nature **574**, 505–510(2019) | Cite this article

Abstract

The promise of quantum computers is that certain computational tasks might be executed exponentially faster on a quantum processor than on a classical processor[1]. A fundamental challenge is to build a high-fidelity processor capable of running quantum algorithms in an exponentially large computational space. Here we report the use of a processor with programmable superconducting qubits[2,3,4,5,6,7] to create quantum states on 53 qubits, corresponding to a computational state-space of dimension 2^{53} (about 10^{16}). Measurements from repeated experiments sample the resulting probability distribution, which we verify using classical simulations. Our Sycamore processor takes about 200 seconds to sample one instance of a quantum circuit a million times—our benchmarks currently indicate that the equivalent task for a state-of-the-art classical supercomputer would take approximately 10,000 years. This dramatic increase in speed compared to all known classical algorithms is an experimental realization of quantum supremacy[8,9,10,11,12,13,14] for this specific computational task, heralding a much-anticipated computing paradigm.

Fig. 1: The Sycamore processor.

From: Quantum supremacy using a programmable superconducting processor



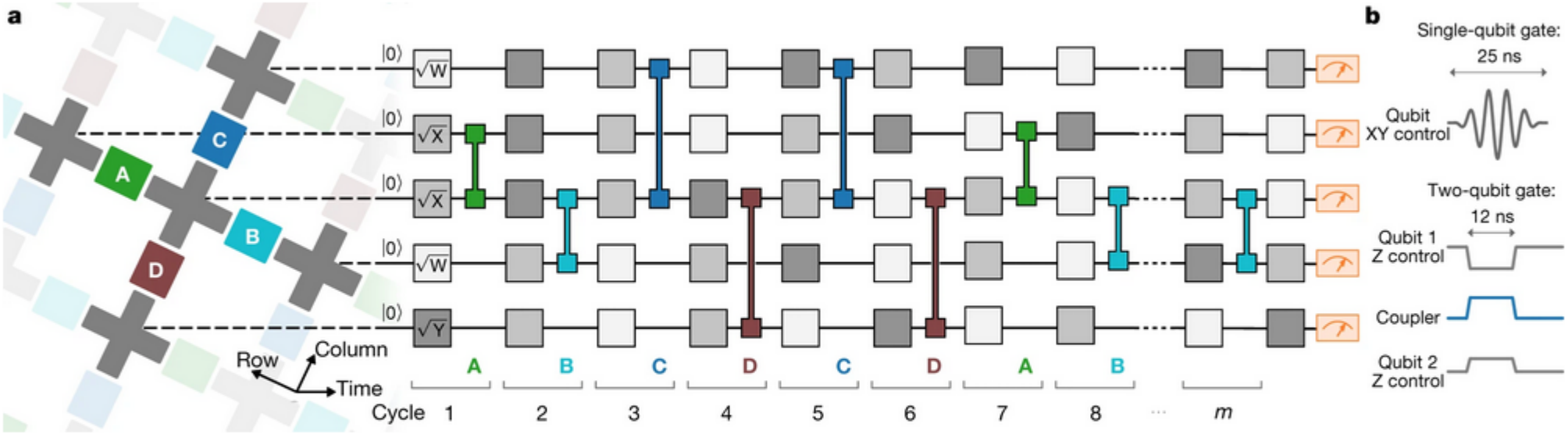
a, Layout of processor, showing a rectangular array of 54 qubits (grey), each connected to its four nearest neighbours with couplers (blue). The inoperable qubit is outlined. **b**, Photograph of the Sycamore chip.

IQM

Quantum supremacy

Fig. 3: Control operations for the quantum supremacy circuits.

From: Quantum supremacy using a programmable superconducting processor



a, Example quantum circuit instance used in our experiment. Every cycle includes a layer each of single- and two-qubit gates. The single-qubit gates are chosen randomly from $\{\sqrt{X}, \sqrt{Y}, \sqrt{W}\}$, where $W = (X + Y)/\sqrt{2}$ and gates do not repeat sequentially. The sequence of two-qubit gates is chosen according to a tiling pattern, coupling each qubit sequentially to its four nearest-neighbour qubits. The couplers are divided into four subsets (ABCD), each of which is executed simultaneously across the entire array corresponding to shaded colours. Here we show an intractable sequence (repeat ABCDCDAB); we also use different coupler subsets along with a simplifiable sequence (repeat EFGHEFGH, not shown) that can be simulated on a classical computer. **b**, Waveform of control signals for single- and two-qubit gates.

IQM

Quantum advantage?

A Threshold for Quantum Advantage in Derivative Pricing

Shouvanik Chakrabarti,[1,2] Rajiv Krishnakumar,[1] Guglielmo Mazzola,[3]
Nikitas Stamatopoulos,[1] Stefan Woerner,[3] and William J. Zeng[1]

[1] *Goldman, Sachs & Co., New York, NY*
[2] *University of Maryland, College Park, MD*
[3] *IBM Quantum, IBM Research – Zurich*
(Dated: December 17, 2020)

We give an upper bound on the resources required for valuable quantum advantage in pricing derivatives. To do so, we give the first complete resource estimates for useful quantum derivative pricing, using autocallable and Target Accrual Redemption Forward (TARF) derivatives as benchmark use cases. We uncover blocking challenges in known approaches and introduce a new method for quantum derivative pricing - the *re-parameterization method* - that avoids them. This method combines pre-trained variational circuits with fault-tolerant quantum computing to dramatically reduce resource requirements. We find that the benchmark use cases we examine require 7.5k logical qubits and a T-depth of 46 million. We estimate that quantum advantage would require a executing this program at the order of a second. While the resource requirements given here are out of reach of current systems, we hope they will provide a roadmap for further improvements in algorithms, implementations, and planned hardware architectures.

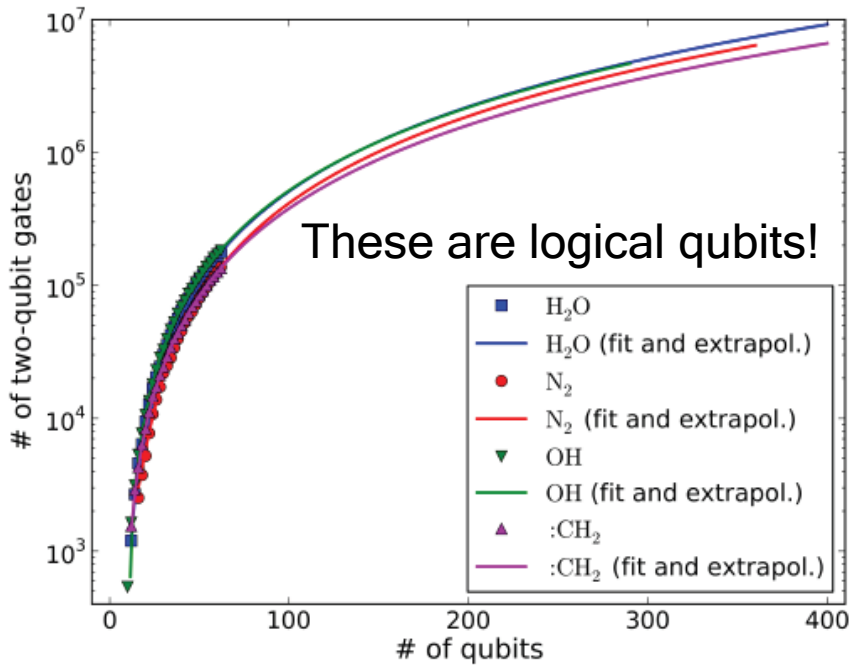


Figure 7. **Estimated number of required two-qubit gates for the UCCSD-VQE treatment of H_2O, N_2, OH and $:CH_2$.** We exploit gate cancellations as well as T_2-amplitude pre-screening based on MP2 to reduce gate counts. The number of required two-qubit gates is evaluated for up to 64 qubits applying the cc-pV5Z basis set. A quadratic polynomial is fitted to the calculated data and then used for extrapolation to a larger number of qubits, see also Section III B 3. For further information see Figures 8-11.

https://arxiv.org/abs/1812.06814

Quantum advantage?

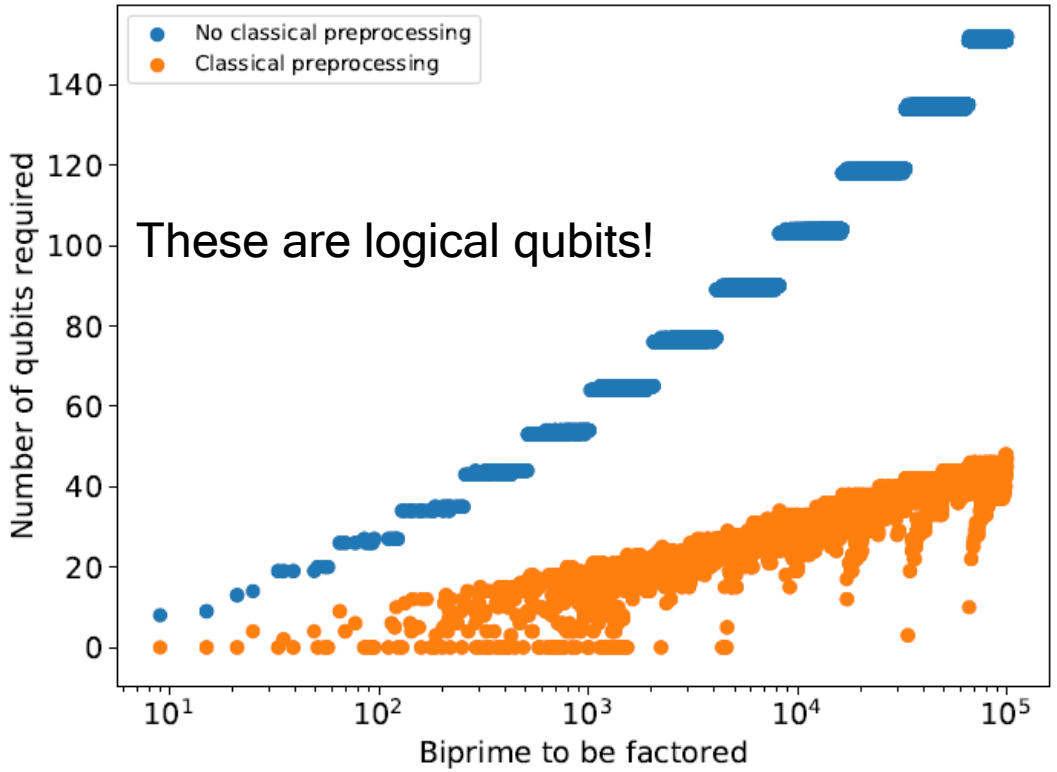
Variational Quantum Factoring

Eric R. Anschuetz,* Jonathan P. Olson,† Alán Aspuru-Guzik,‡ and Yudong Cao§

Zapata Computing Inc., 501 Massachusetts Avenue, Cambridge MA 02138

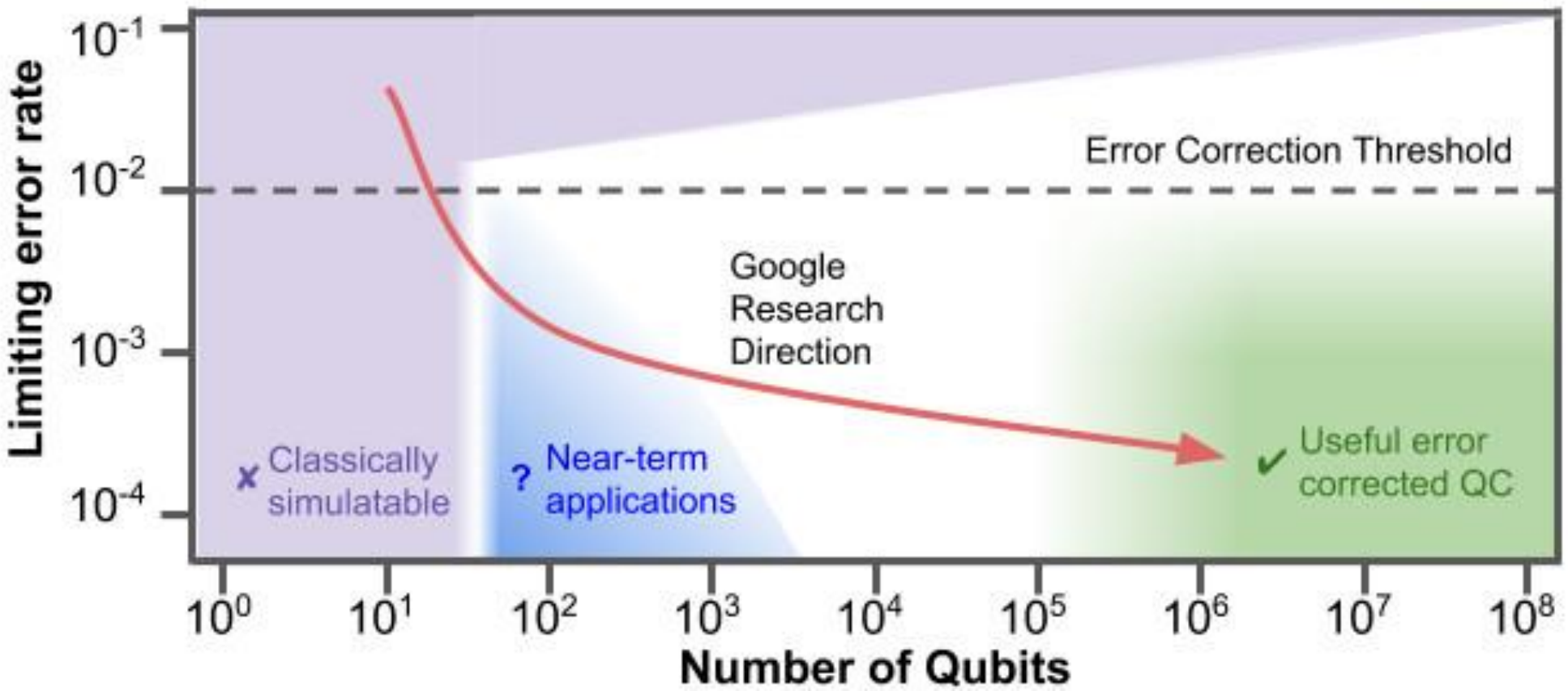
Abstract

Integer factorization has been one of the cornerstone applications of the field of quantum computing since the discovery of an efficient algorithm for factoring by Peter Shor. Unfortunately, factoring via Shor's algorithm is well beyond the capabilities of today's noisy intermediate-scale quantum (NISQ) devices. In this work, we revisit the problem of factoring, developing an alternative to Shor's algorithm, which employs established techniques to map the factoring problem to the ground state of an Ising Hamiltonian. The proposed variational quantum factoring (VQF) algorithm starts by simplifying equations over Boolean variables in a preprocessing step to reduce the number of qubits needed for the Hamiltonian. Then, it seeks an approximate ground state of the resulting Ising Hamiltonian by training variational circuits using the quantum approximate optimization algorithm (QAOA). We benchmark the VQF algorithm on various instances of factoring and present numerical results on its performance.



IQM

Envisioned roadmap



IQM

SW-HW gap

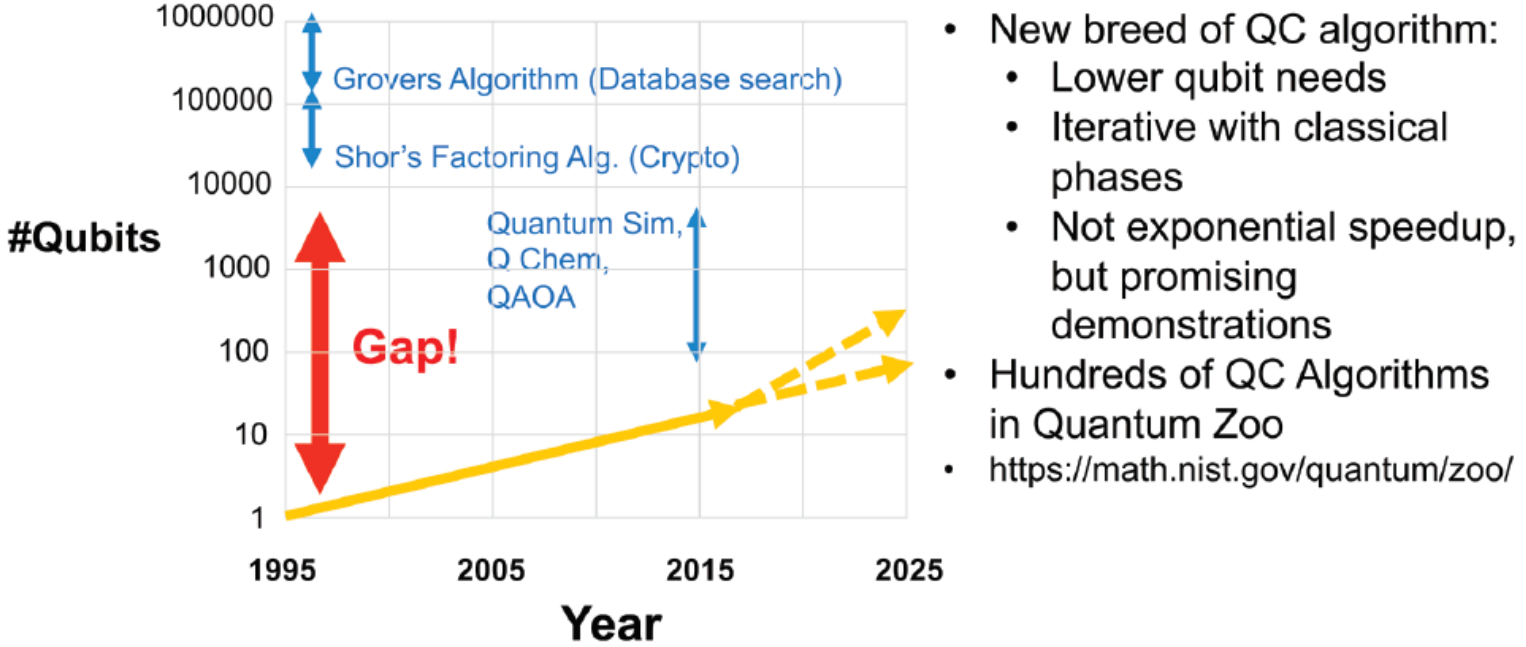


Figure 1: The Algorithms-to-Machines gap illustrates how well-known QC algorithms (such as Shor's and Grover's) have resource requirements that far exceed the qubit counts (shown in yellow) of systems we are able to build.

IQM

General note: SW- HW gap

- Even though qubit count and quality have been increasing over the years, we are still far away from closing the gap between SW requirements and HW accessibility.

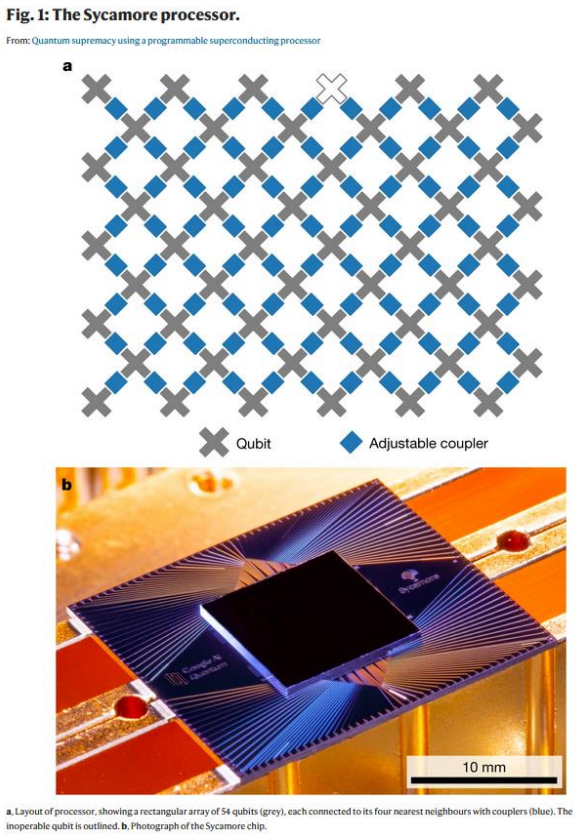
- A big milestone was reaching quantum supremacy by Google. They created an arbitrary circuit that is hard to simulate classically.

- Bringing usefulness into the equation creates further overhead. Reaching quantum advantage requires probably thousands of qubits.

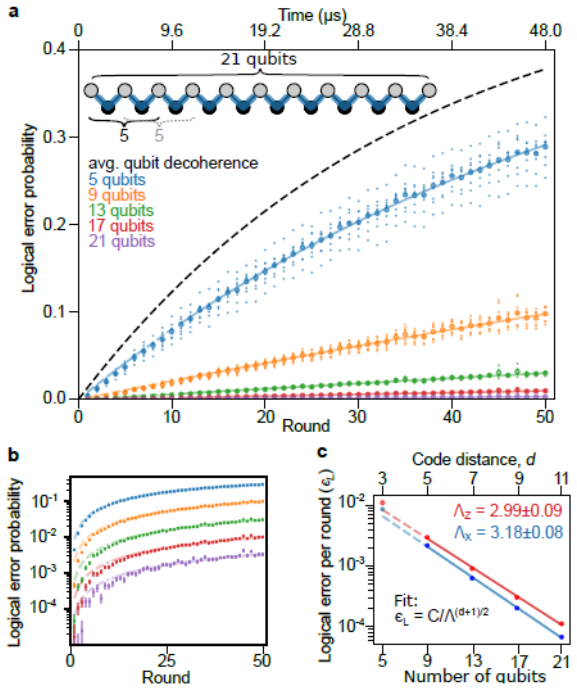
Agenda for today

12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

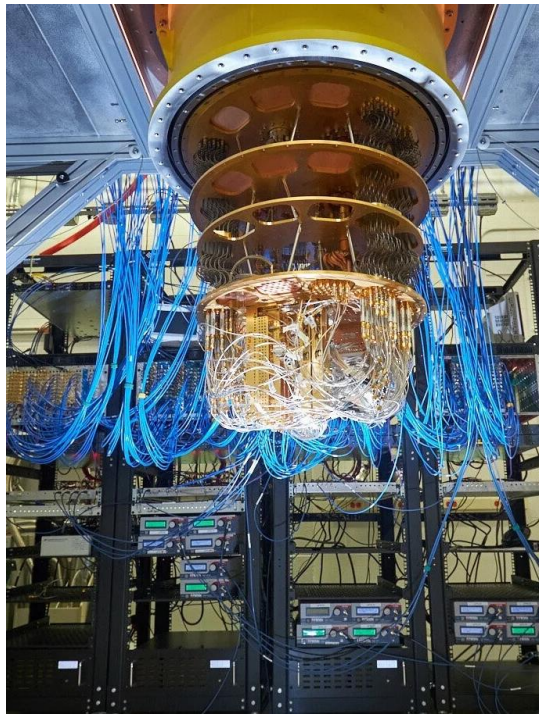
a.



b.



c.



IQM

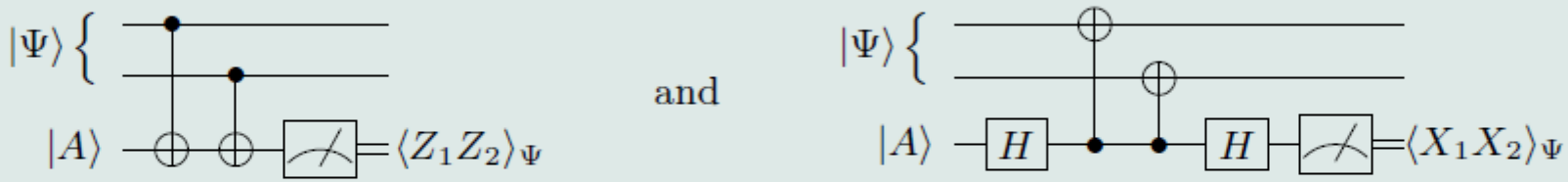
General approach: Error correction

- Quantum systems will never be perfect, hence errors will always occur.
- If the error rate is small enough, it is possible to detect the errors and correct for them.
- Detecting the errors in a non-invasive way creates a strong overhead in the required qubit number.

Error detection

Parity measurements - a workhorse in quantum error detection and correction

Many quantum error correction schemes rely on parity measurements. In the left circuit below, the ancilla qubit $|A\rangle$ is used to infer the bit parity (via information about $\langle Z_1 Z_2 \rangle$) of the two data qubits in state $|\Psi\rangle$, and in the right circuit qubit $|A\rangle$ infers the phase parity (via $\langle X_1 X_2 \rangle$),



In the absence of errors on the ancilla qubit, the eigenvalue of $|A\rangle$ will contain information reflecting whether the two-qubit state $|\Psi\rangle$ is an eigenstate of $Z_1 Z_2$ (or $X_1 X_2$) with eigenvalue $+1$ or -1 without collapsing the state of the individual qubits in $|\Psi\rangle$. Since the operators $Z_1 Z_2$ and $X_1 X_2$ (and even multiples of more Z and X operators) commute, combinations of parity measurements across a larger grid of qubits can therefore be used to infer if and where a bit- or phase-flip error ocurred, without collapsing the underlying quantum data. The collection of ancilla qubit measurements is typically referred to as the syndrome of the error, and inferring the underlying error is known as decoding.

arXiv:1905.13641v3

Error correction: Bit flip example

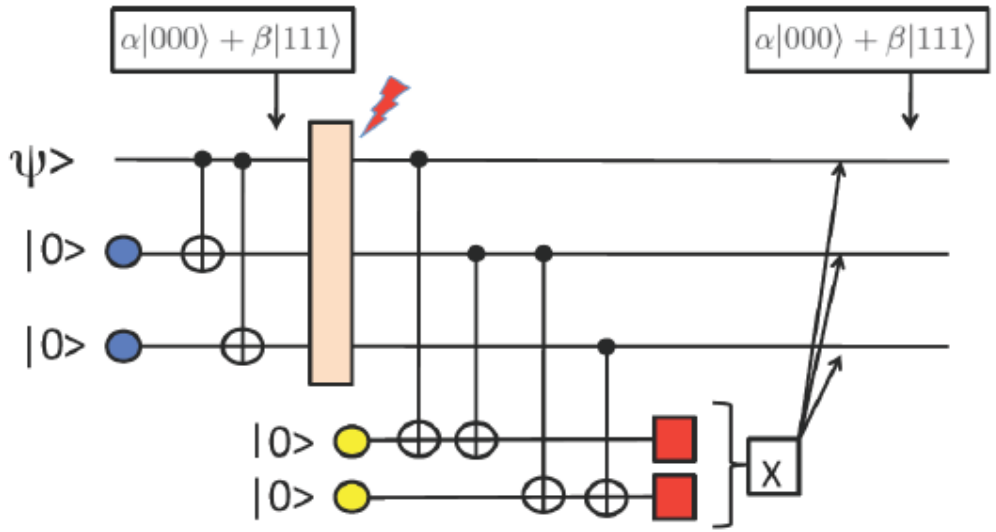


Figure 22: Standard bit-flip QEC with two ancillas for detecting and storing the error syndrome, and correcting the error. In this way the code words are not perturbed by measurement and recoding is not needed. The figure illustrates explicit error detection followed by feedforward control using e.g. FPGA electronics. The correction can also be implemented via Toffoli gates. In the case of phase flips one uses Hadamard gates to transform phase flips into bit flips, checks the parity, and then transforms back.

Example $\alpha = 1$: $\Psi = |000\rangle$

Zero bit-flip error:
On measuring (00), do nothing.

Bit-flip error in qubit 1:
On measuring (10), X-flip qubit 1.

Bit-flip error in qubit 2:
On measuring (11), X-flip qubit 2.

Bit-flip error in qubit 3:
On measuring (01), X-flip qubit 3.

IQM arXiv:1905.13641v3

Error correction codes

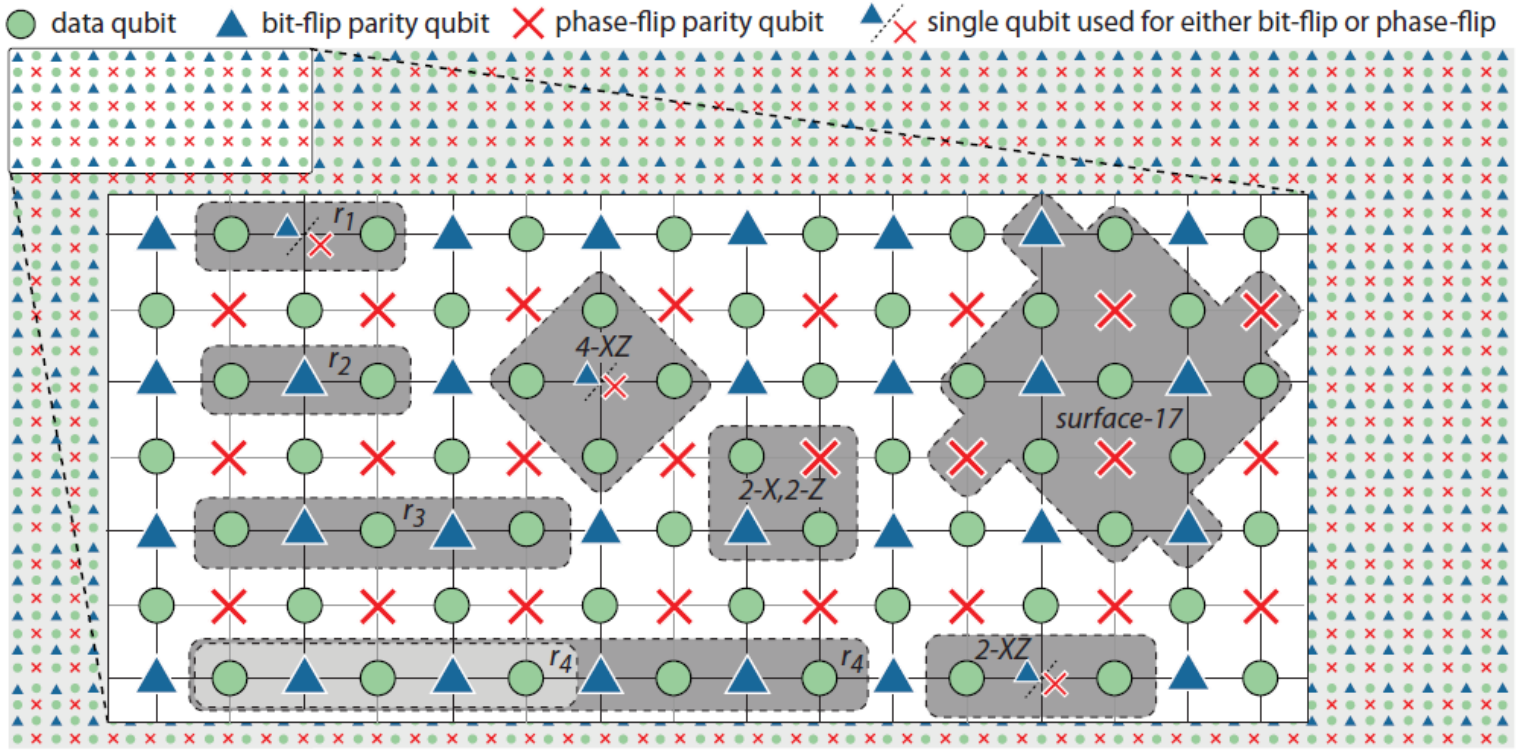


Figure 4

A section of the qubit layout of the surface code, with 40 × 20 data qubits (shown as circles), and associated bit-flip and phase-flip parity qubits are shown as triangles and crosses, respectively. Inset shows a subsection, in which shaded areas indicate parity experiments that have been reported, except 'surface-17' which is currently being pursued in multiple laboratories (see text for details). Experiment r_1 by Reed et al. (188), r_2 by Chow et al. (189), r_3 by Risté et al (193), r_4 by Kelly et al. (14), 2-X,2-Z by Córcoles et al. (194), 2-XZ by Andersen et al. (195) and Bultink et al. (196), 4-XZ by Takita et al. (197).

Since superconducting qubits operate in 2D grids, error correction codes are called surface codes. They correct errors on a rectangular qubit grid.

In this grid, qubits are assigned different tasks. Data qubits store the information and are being actively corrected. Bit-flip and phase-flip qubits detect the errors and trigger the correction.

Depending on the number of qubits involved, the codes get names like "surface-17".

 arXiv:1905.13641v3

Repetitive error correction

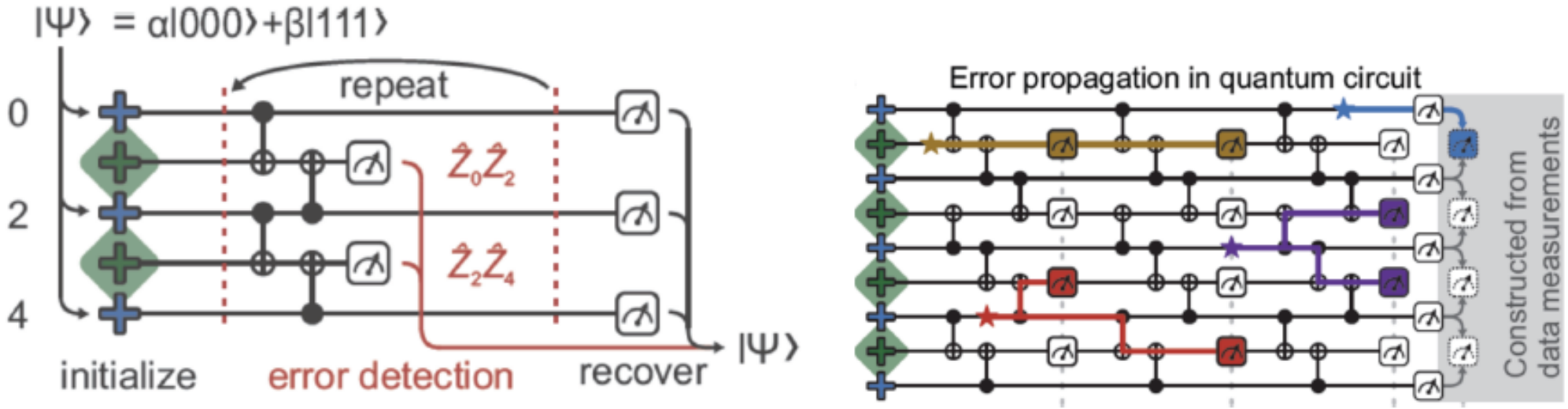


Figure 24: (a) Left: 3q repetition code $\alpha|000\rangle + \beta|111\rangle$: algorithm (same circuit as in Fig. 22. (b) Right: 5q repetition code $\alpha|00000\rangle + \beta|11111\rangle$: error propagation and identification. Adapted from [35].

arXiv:1905.13641v3

Repetitive error correction

Exponential suppression of bit or phase flip errors with repetitive error correction

Google Quantum AI*

(Dated: February 12, 2021)

Realizing the potential of quantum computing will require achieving sufficiently low logical error rates [1]. Many applications call for error rates in the 10^{-15} regime [2–9], but state-of-the-art quantum platforms typically have physical error rates near 10^{-3} [10–14]. Quantum error correction (QEC) [15–17] promises to bridge this divide by distributing quantum logical information across many physical qubits so that errors can be detected and corrected. Logical errors are then exponentially suppressed as the number of physical qubits grows, provided that the physical error rates are below a certain threshold. QEC also requires that the errors are local and that performance is maintained over many rounds of error correction, two major outstanding experimental challenges. Here, we implement 1D repetition codes embedded in a 2D grid of superconducting qubits which demonstrate exponential suppression of bit or phase-flip errors, reducing logical error per round by more than 100× when increasing the number of qubits from 5 to 21. Crucially, this error suppression is stable over 50 rounds of error correction. We also introduce a method for analyzing error correlations with high precision, and characterize the locality of errors in a device performing QEC for the first time. Finally, we perform error detection using a small 2D surface code logical qubit on the same device [18, 19], and show that the results from both 1D and 2D codes agree with numerical simulations using a simple depolarizing error model. These findings demonstrate that superconducting qubits are on a viable path towards fault tolerant quantum computing.

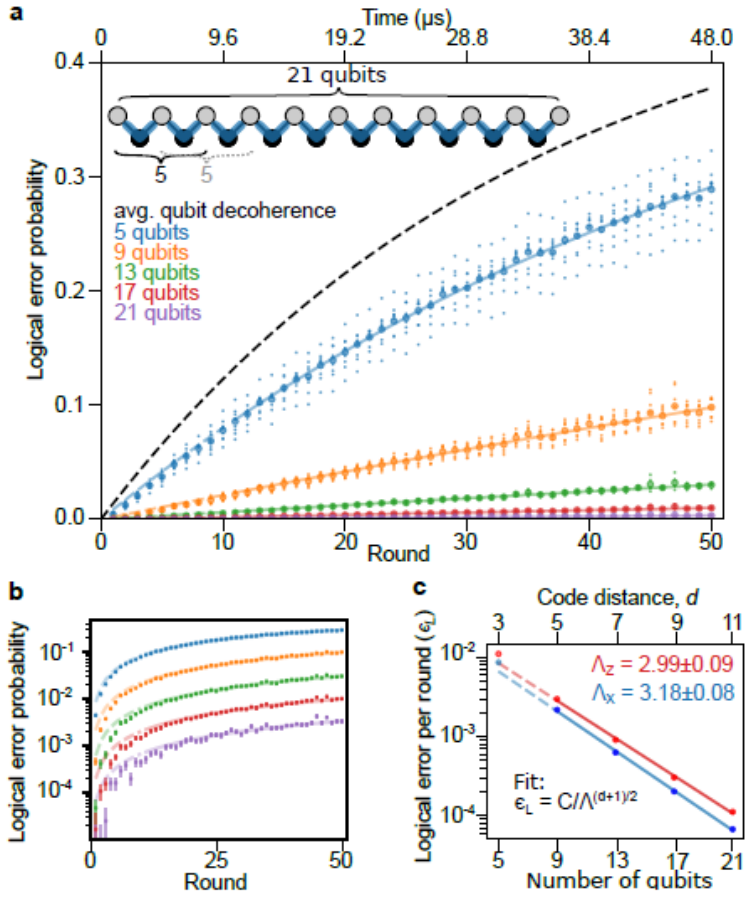


FIG. 3. **Logical errors in the repetition code. a,** Logical error probability versus number of detection rounds and number of qubits for the phase flip code. Smaller code sizes are subsampled from the 21 qubit code as shown in the inset; small dots are data from subsamples and large dots are averages. **b,** Semilog plot of the averages from **a** showing even spacing in log(error probability) between the code sizes. Error bars are estimated standard error from binomial sampling. The lines are exponential fits to data for rounds greater than 10. **c,** Logical error per round (ϵ_L) vs. number of qubits, showing exponential suppression of error rate for both bit and phase flip, with extracted Λ factors. The fit excludes $n_{\text{qubits}} = 3$ to reduce the influence of spatial boundary effects [35].

Fault-tolerant quantum computers

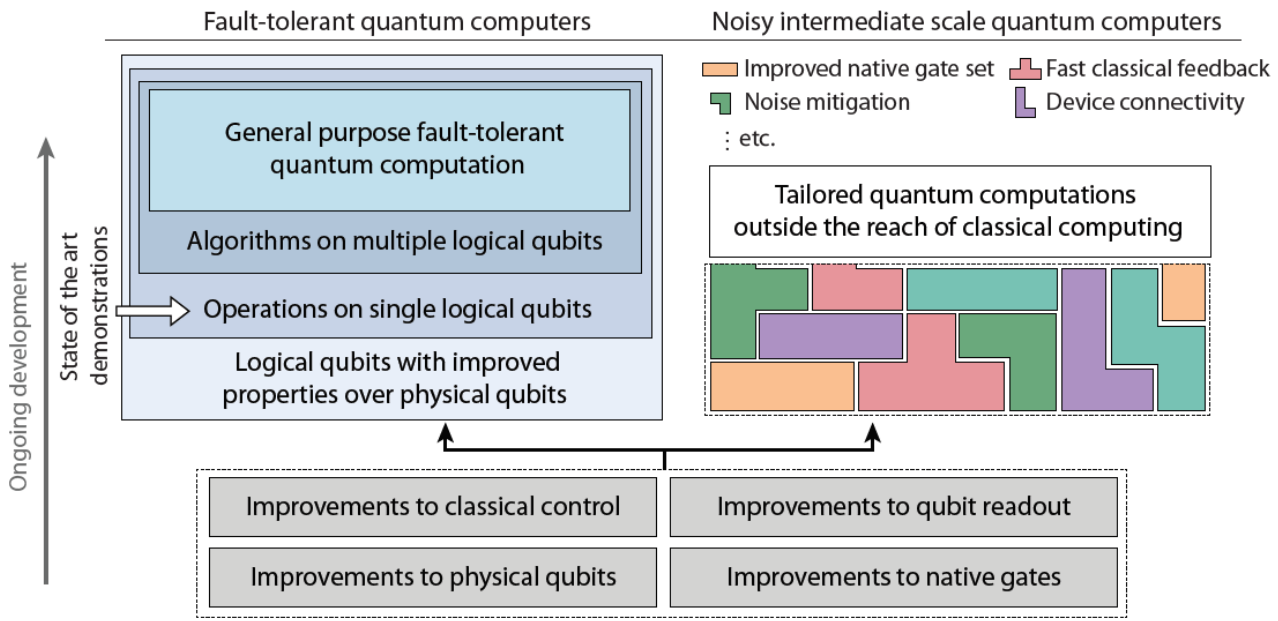


Figure 1

Path towards fault-tolerant quantum error-corrected quantum computers (left) as well as noisy intermediate scale quantum computing (right) using superconducting qubits. The left track follows the path towards quantum computers capable of performing arbitrarily long programs to arbitrary precision, based on logical (i.e. encoded and error-corrected) qubits. The right track is the 'NISQ' approach (see Ref. (7)), where highly optimized quantum algorithms and quantum simulations, which typically take into account details of the quantum processor, can be executed without generalized quantum error correction procedures. The two tracks are pursued in parallel in many academic, government, and industrial laboratories.

Building an error-corrected quantum computer requires more than "just" many qubits. Improvements are needed on several fronts.

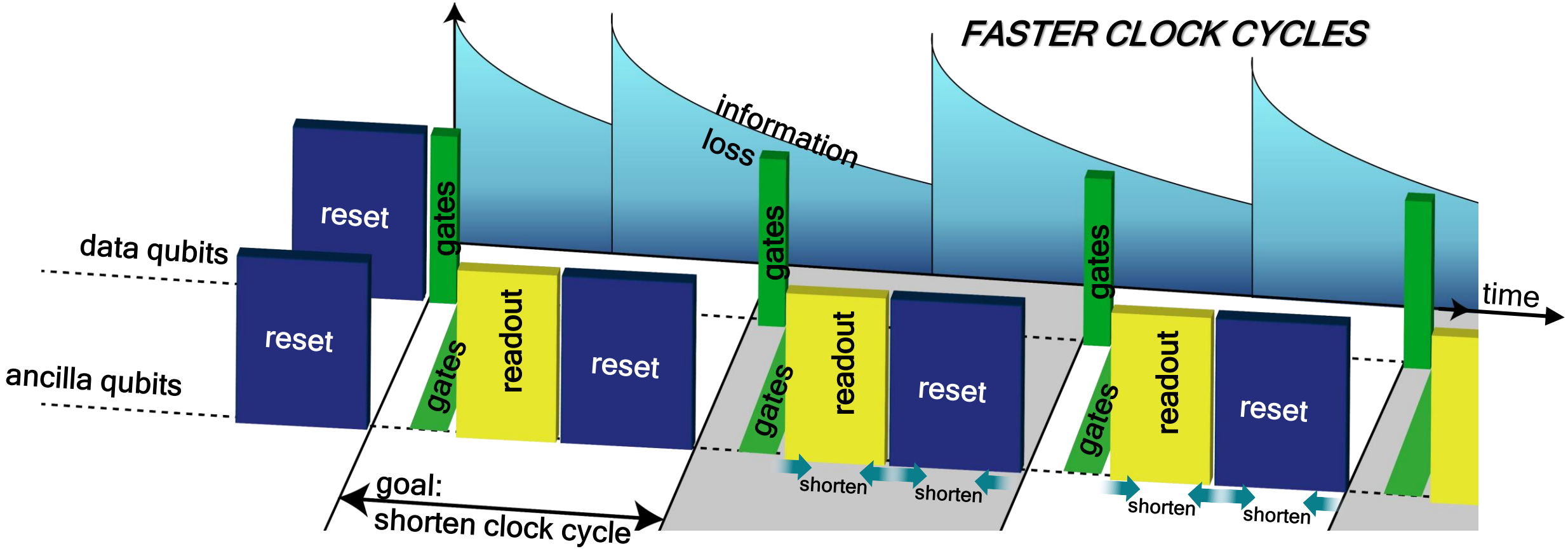
On the computer architecture side, developments in real-time feedback, connectivity, compilers, etc need to be further developed.

On the hardware side, readout, reset and gates need to be further improved.

 arXiv:1905.13641v3

IQM's way to address error correction:

- Loss of quantum information is the biggest challenge!
- Reset and readout take the longest time
- The faster you make them, the less information you lose



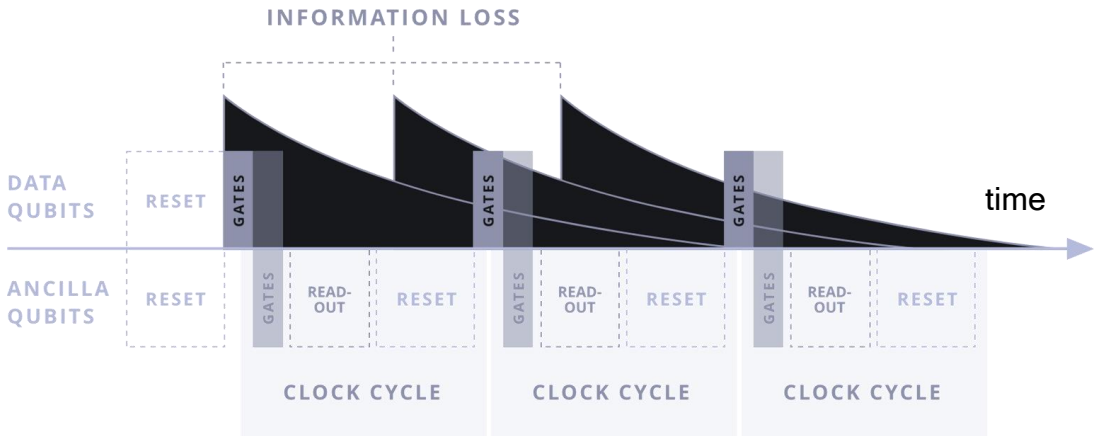
Accelerating clock cycles in IQM

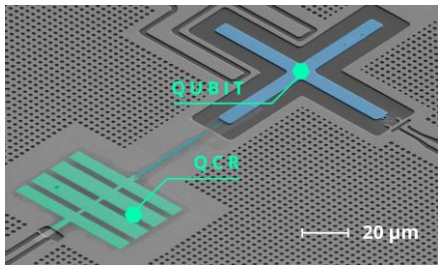
PROBLEM

Quantum operations too error prone for **quantum advantage** and too slow for **error correction**.

SOLUTION

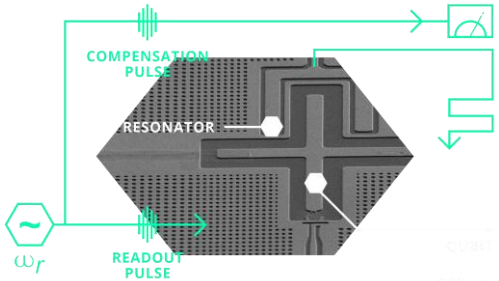
IQM builds the most precise and fastest quantum processors by improving the three basic quantum operations: reset, gates, readout.





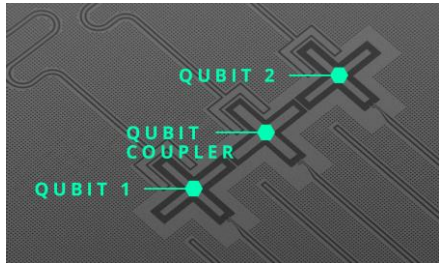
FASTER RESET: QCR WITHOUT FEEDBACK

Unconditional reset
10x speedup to current alternatives
Resets also higher quantum states (leakage errors)
No RF fields required (quasi-DC technology)
Suitable for active on-chip cooling
Protected technology suitable for licensing



FASTER READOUT: MULTICHANNEL INSTEAD OF SINGLE DRIVE

Based on a multichannel driving scheme
No on-chip overhead
Protected technology suitable for licensing



FASTER GATES: TUNABLE COUPLER & *N*-JUNCTION QUBITS

Tunable coupler with ultrafast gate times
Utilizing higher nonlinearity of novel qubit types

IQM

Qubit overhead

Surface codes: Towards practical large-scale quantum computation

Austin G. Fowler
Centre for Quantum Computation and Communication Technology, School of Physics, The University of Melbourne, Victoria 3010, Australia

Matteo Mariantoni
Department of Physics, University of California, Santa Barbara, CA 93106-9530, USA and California Nanosystems Institute, University of California, Santa Barbara, CA 93106-9530, USA

John M. Martinis and Andrew N. Cleland
California Nanosystems Institute, University of California, Santa Barbara, CA 93106-9530, USA
(Dated: October 26, 2012)

This article provides an introduction to surface code quantum computing. We first estimate the size and speed of a surface code quantum computer. We then introduce the concept of the stabilizer, using two qubits, and extend this concept to stabilizers acting on a two-dimensional array of physical qubits, on which we implement the surface code. We next describe how logical qubits are formed in the surface code array and give numerical estimates of their fault-tolerance. We outline how logical qubits are physically moved on the array, how qubit braid transformations are constructed, and how a braid between two logical qubits is equivalent to a controlled-NOT. We then describe the single-qubit Hadamard, $\hat{S}$ and $\hat{T}$ operators, completing the set of required gates for a universal quantum computer. We conclude by briefly discussing physical implementations of the surface code. We include a number of appendices in which we provide supplementary information to the main text.

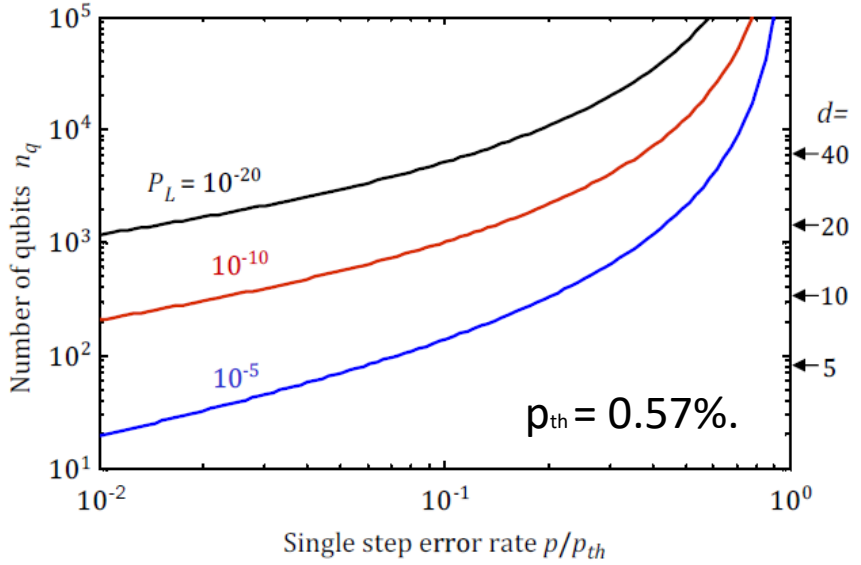


FIG. 6. (Color online) Estimated number of physical qubits n_q per logical qubit, versus the single-step error rate p, the latter normalized to the threshold error rate p_{th}, plotted for different target logical error rates P_L. Notation on the right axis corresponds to the array distance d for a single logical qubit.

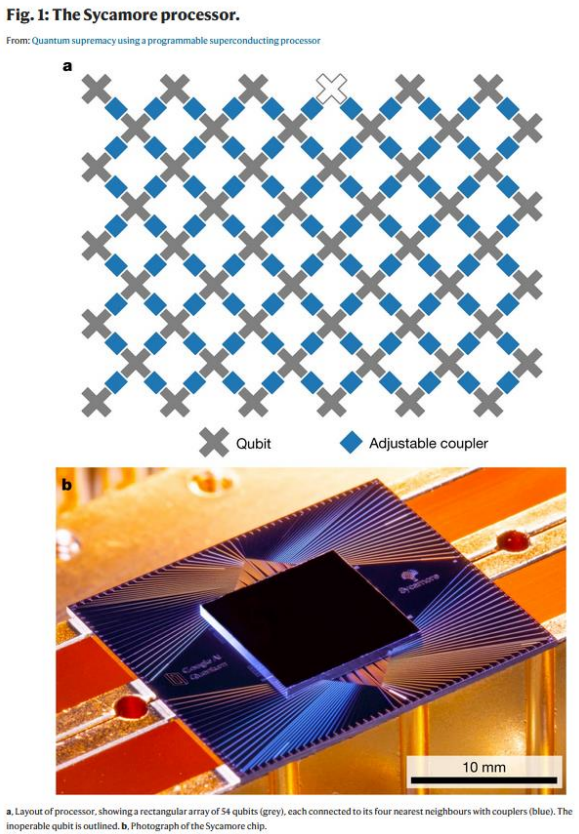
General approach: Error correction

- Quantum systems will never be perfect, hence errors will always occur.
- If the error rate is small enough, it is possible to detect the errors and correct for them.
- Detecting the errors in a non-invasive way creates a strong overhead in the required qubit number.

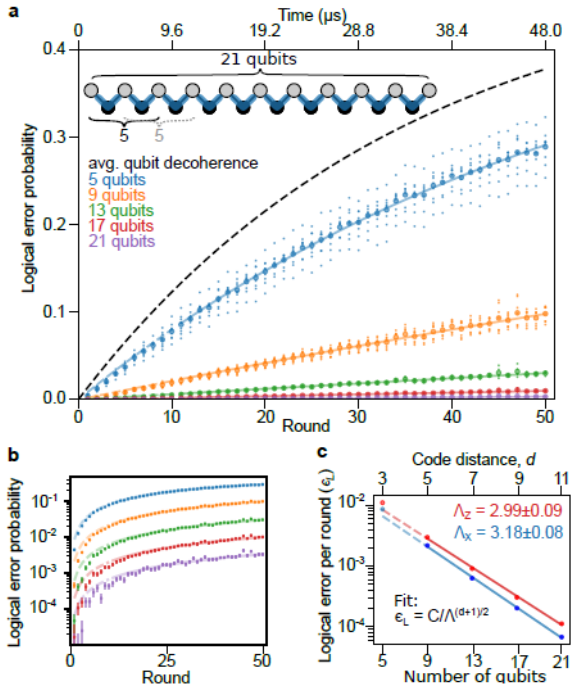
Agenda for today

12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

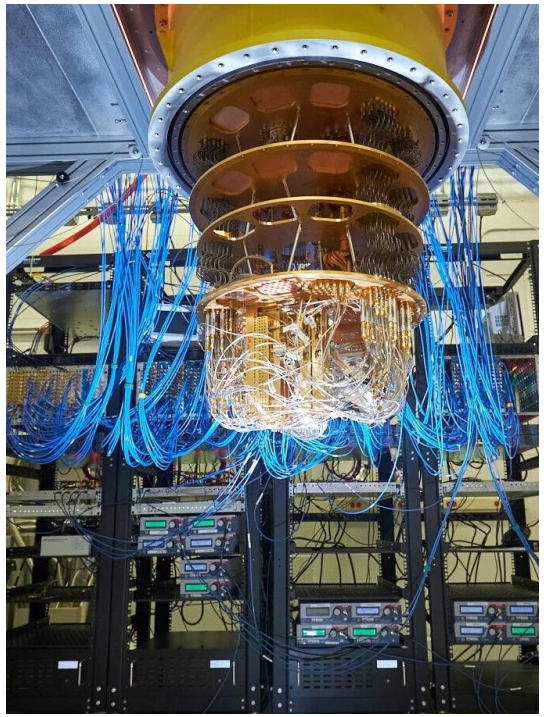
a.



b.



c.



IQM

Scaling challenges

NEAR-TERM CHALLENGES

- **Control and high coherence in medium-scale devices**: For medium- and large-scale devices, the individual qubit coherences are not necessarily the same as those in a simpler few-qubit devices. Maintaining high coherence and high-fidelity control across a large chip is a key challenge.
- **Scalable calibration techniques**: Advanced software strategies are also needed to calibrate medium-to-large scale quantum processors due to the large number of non-trivial cross-calibration terms while finding simultaneous optimal operating parameters.
- **Verification and validation**: As the number of qubits increases, efficiently determining the fidelity of quantum operations across the entire chip using e.g. Clifford randomized benchmarking (70) becomes infeasible and new techniques for validation and verification will be needed. Techniques such as 'cross entropy benchmarking' (108) and 'direct benchmarking' (216) have recently been proposed and implemented.
- **Improving qubit connectivity**: While impressive progress has been made in three-dimensional integration of superconducting circuits (e.g. Ref. (217)), non-planar connectivity of high-fidelity qubits has yet to be demonstrated.
- **Improved gate fidelity**: Continued improvements to gate fidelities will be an important step towards bringing down the overhead of physical qubits needed to encode a single logical qubit as well as important for demonstrating the efficacy of NISQ algorithms.
- **Robust & reproducible fabrication**: The fabrication of medium-to-large scale superconducting circuits will need to be consistent with continued improvements to qubit coherence and 3D integration techniques.

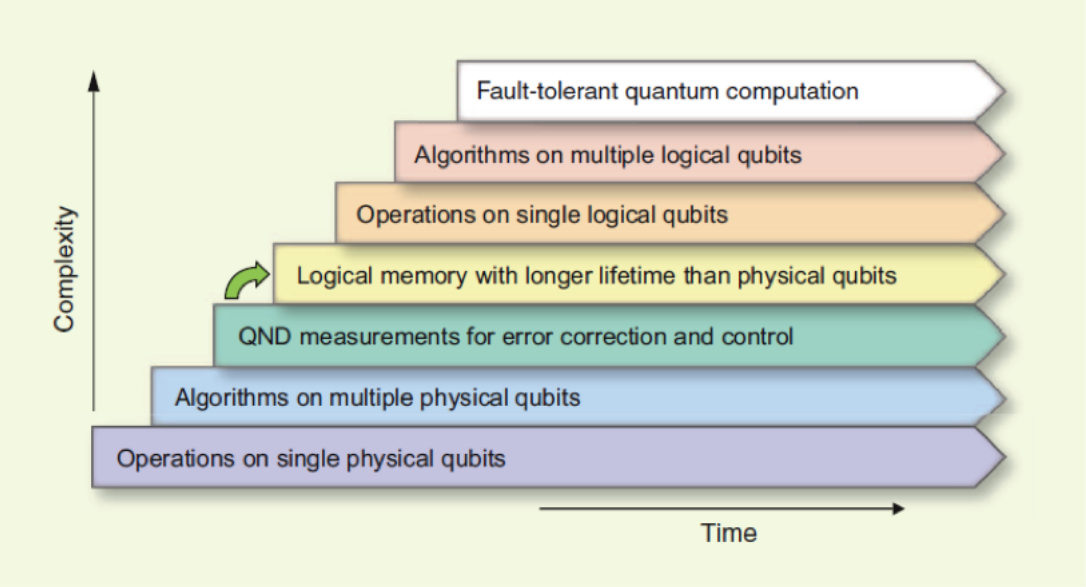
arXiv:1905.13641v3

Different hardware platforms have different challenges when scaling up the systems. For superconducting systems, a good summary is given on the left.

As a rule of thumb, researchers think that up to 1000 qubits, we can scale using existing technology like coaxial cables and the like. Beyond 1000 qubits, certain components and approaches need to be replaced.

Scaling up the hardware inevitably means that one also must scale the software performance. For example, tuning up a 50-qubit processor "by hand" is impossible.

Scaling towards fault-tolerance



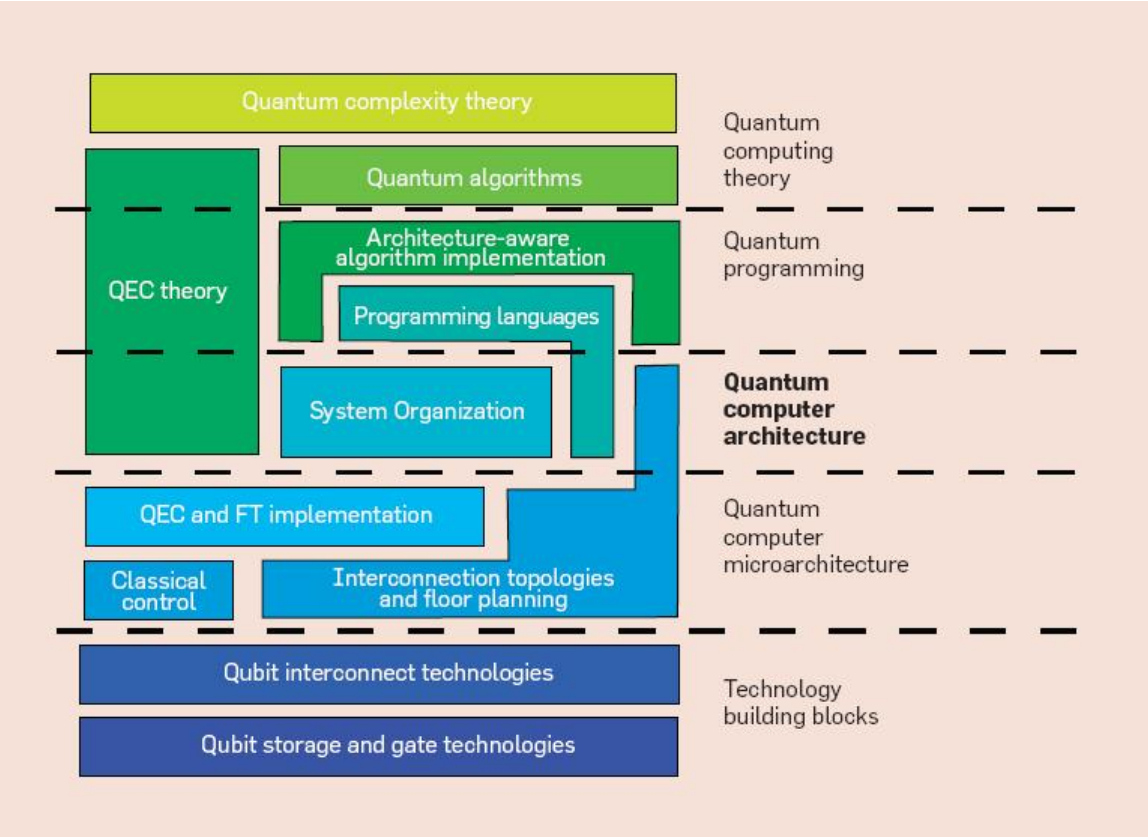
One main goal of the community is to reach fault-tolerant quantum computing using error-corrected system. This process requires several steps (see left picture).

To run algorithms on multiple logical qubits, the physical qubit count has to exceed several thousand.

In parallel, control electronics, calibration software, detection schemes and the like need to be improved.

http://qulab.eng.yale.edu/documents/papers/Superconducting%20Circuits%20for%20Quantum%20Information%20-%20An%20Outlook.pdf

Quantum computer architecture



A quantum computer is much more than just qubits on a chip and microwave electronics. It contains several layers of abstraction.

To build a full-stack scalable systems, all the different layers need to be integrated to each other. This requires the definition of interfaces and standards.

An additional challenge is that people working on the different layers have different background and don't necessarily speak the "same language". Fabrication engineers need to understand the challenges of software architects and vice versa.

https://quantumcomputing.stackexchange.com/questions/9067/how-is-a-quantum-computer-programmed

Runtime environment

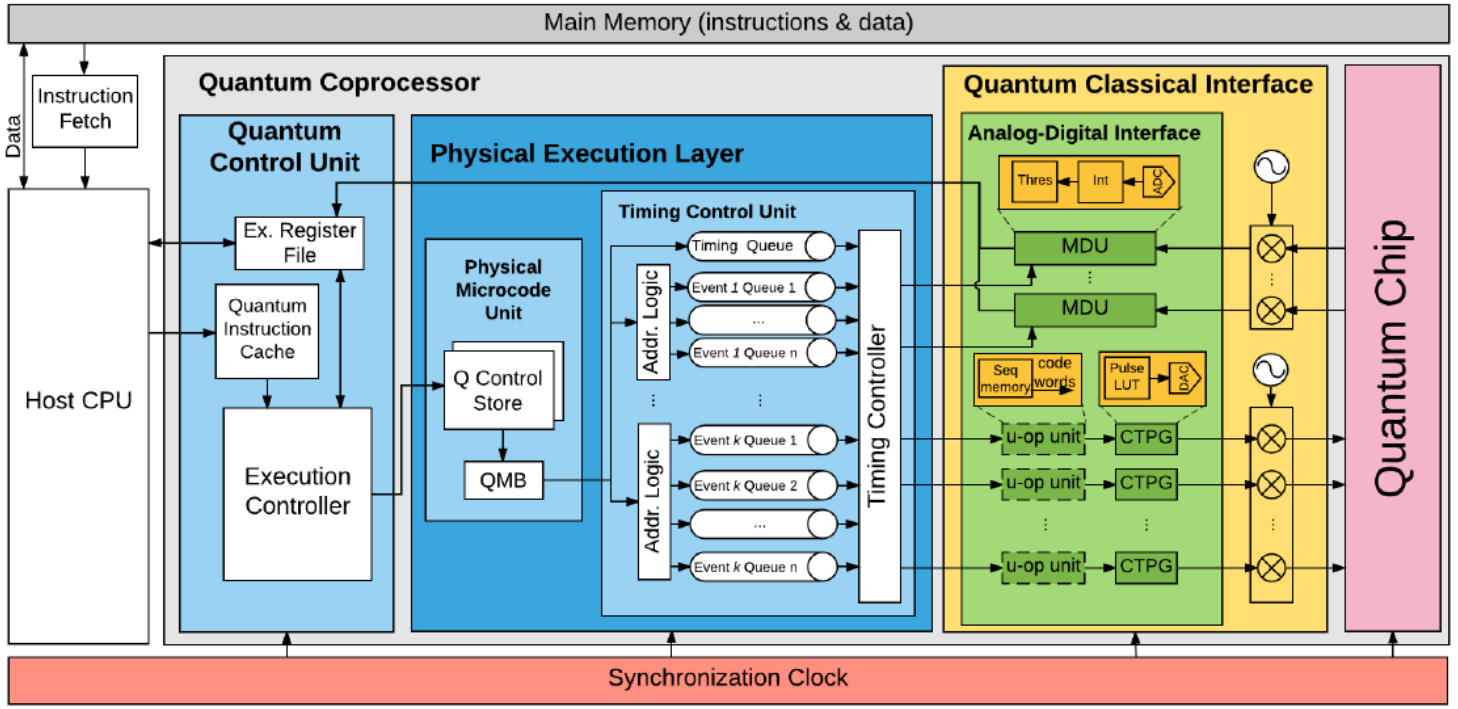


Figure 4: Overview of the Quantum MicroArchitecture (QuMA).

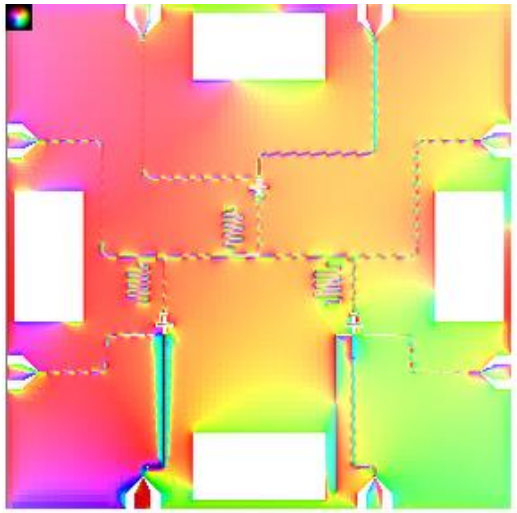
To operate a quantum computer, many operations need to be coordinated in real-time. The orchestration of the runtime environment is a challenging task.

The real-time execution is usually coordinated by FPGA cards, which are synchronized by a master clock.

The overall control of the system has a host CPU. Hence, quantum computers will never work as a standalone device. Probably they will be integrated into supercomputing centers.

https://arxiv.org/abs/1708.07677

Design challenge: EM Simulations



Experiment

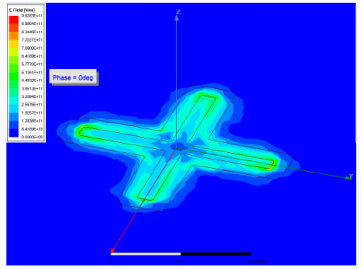
G13	QB1 flux	QB2 flux	QB3 flux
QB1 source	714	16	5
QB2 source	<1	505	60
QB3 source	30	25	1059

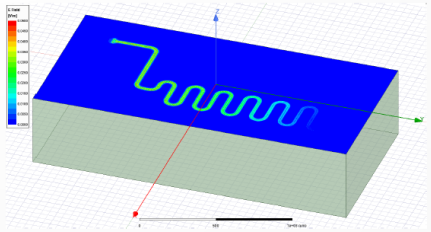
Simulations

Simulated crosstalk is following:

XF1	flux1	flux2	flux3
input1	714.703	-30.7141	13.8963
input2	-28.4896	442.939	97.097
input3	17.441	-46.7387	797.8

Finding a usable model to estimate quality factor of different geometries





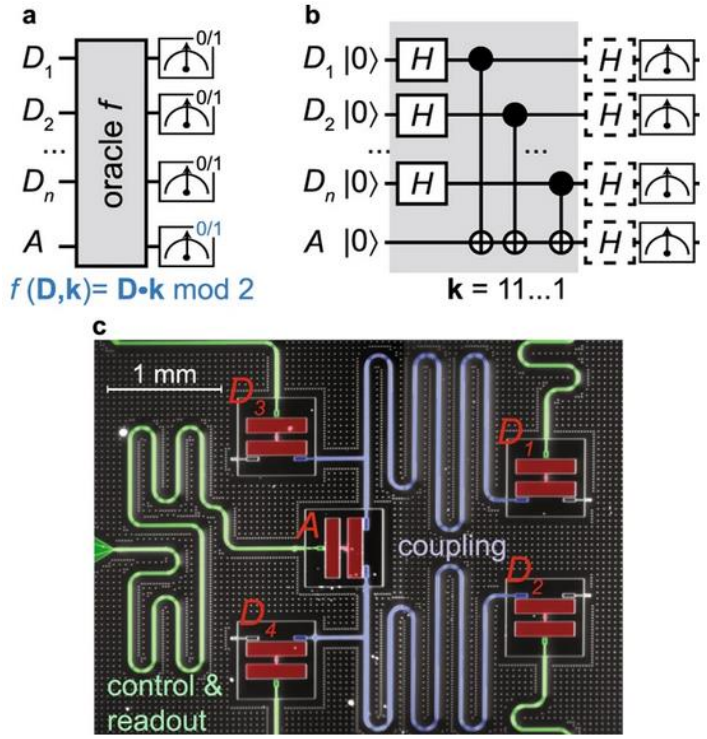
To create a large-scale quantum processor, operating frequencies, coupling strengths, etc need to be carefully designed. This is usually done in FEM simulations.

One task of these simulations is to reduce crosstalk between different on-chip elements because this crosstalk generates errors.

FEM simulations of large system is a challenging task by itself. Usually, superconducting processors have a very large aspect ratio between lateral dimensions and height. This makes the discretization of the geometries hard.

IQM

Fabrication challenges

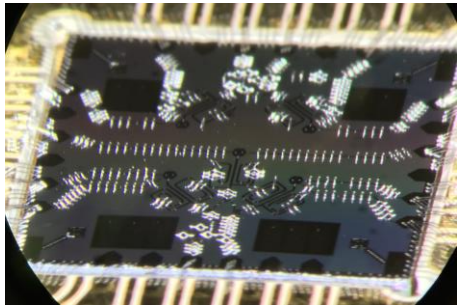


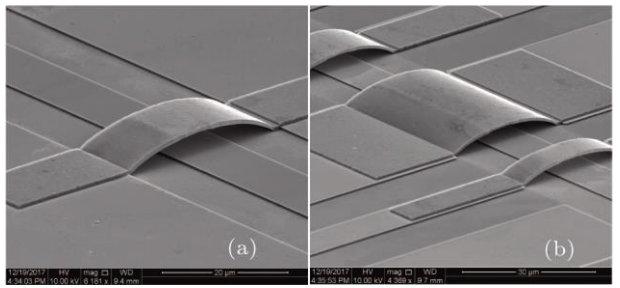
Implementation of a parity function in a superconducting circuit. **a** Conceptual diagram of parity learning. The (classical or quantum) oracle *f* ideally maps the parity of a subset of *n* data bits (or qubits), defined by the bit string ***k***, into bit *A*. Repeated queries of the oracle allow the reconstruction of ***k*** by reading the output register. **b** Gate sequence implementing a quantum parity oracle with ***k*** = 11...1. Random examples are generated by preparing the data qubits {$D_1,\ldots,D_n$} in a uniform superposition. *Vertical lines* indicate CNOT gates between each D_i (control) and the ancilla qubit *A* (target). Quantum learning differs from classical learning only by the addition of single-qubit gates (*dashed boxes*) applied before measurement (see also Supplementary Information). **c** Optical image of the superconducting quantum processor (*qubits in red*). *A* is coupled to each D_i by means of two bus resonators (*blue*). Each qubit is also coupled to a dedicated resonator for control and readout (*green*)[27]

Since superconducting processors are fabricated on silicon wafers, we usually have only 2D spheres available.

Bringing control and readout lines into the center of a larger qubit grid creates a very crowded situation with a lot of crosstalk.

For larger systems, crossing lines are unavoidable. Hence, we must use the third dimension to build scalable quantum processors.

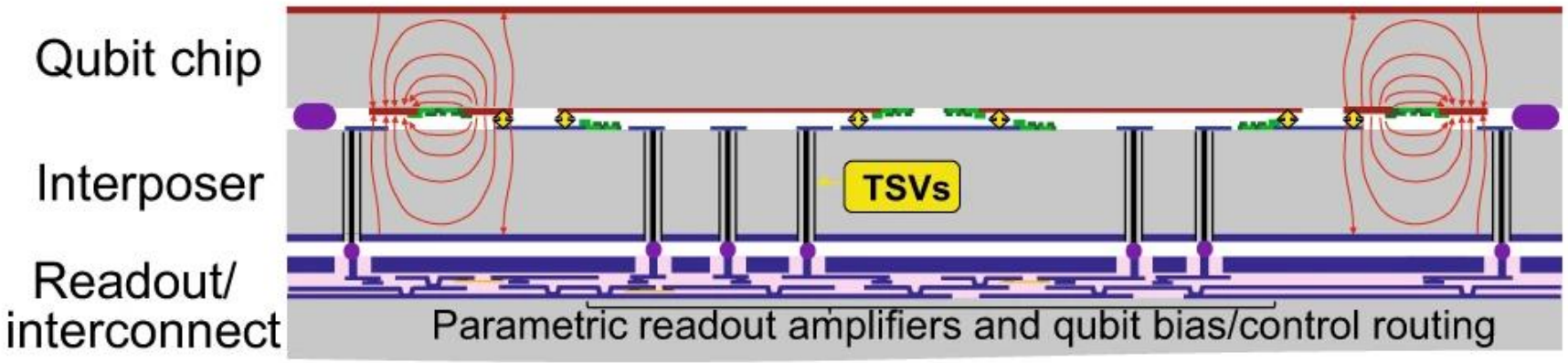




https://www.nature.com/articles/s41534-017-0044-0#Fig1

http://cpb.iphy.ac.cn/article/2018/1958/cpb_27_10_100310.html

Fabrication challenges



Envisioned scheme for control and readout of a large-scale, 3D integrated quantum processor. The qubit, interposer, and readout/interconnect chips are connected using indium bump bonds. The qubits are separated from the readout and control layer by an interposer chip with through-substrate vias that provide input/output (I/O) connectivity to/from the qubits. Because the chips are fabricated separately, each fabrication process can be optimized independently

https://www.nature.com/articles/s41534-017-0044-0#Fig1

Fabrication of Josephson junctions

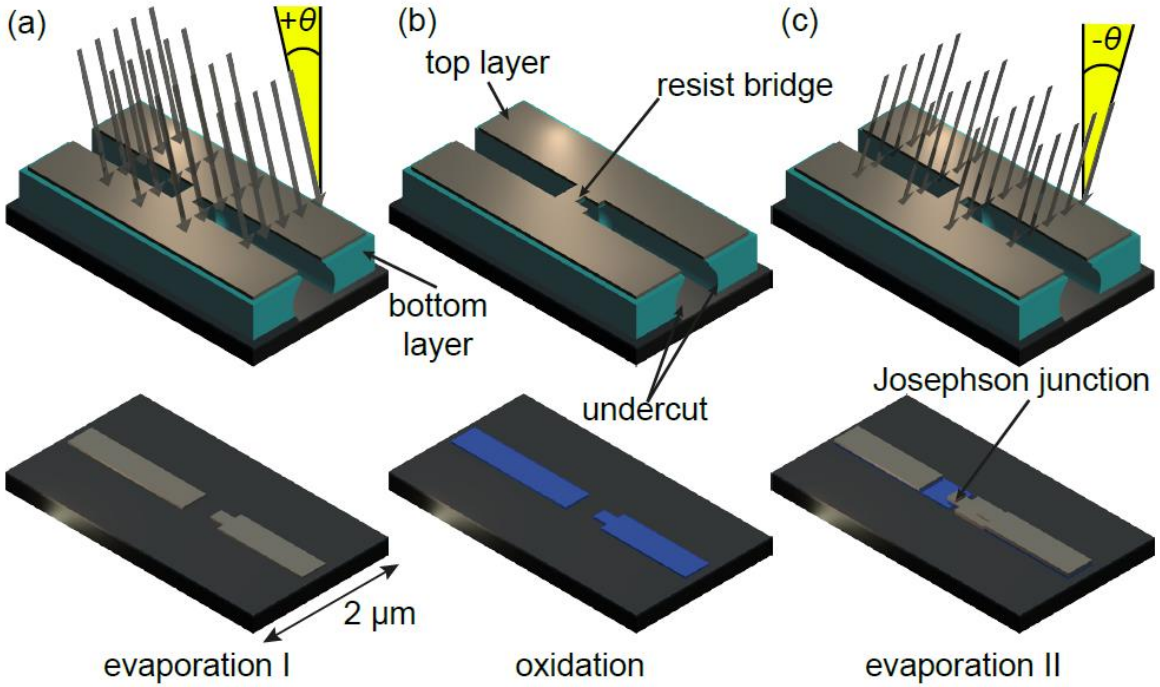


Figure 3.7: Different fabrication steps for Josephson junctions including both resist layers in the top row. These layers are omitted in the bottom row to show the resulting Al films. **(a)** First evaporation step at angle $+\theta$. **(b)** Oxidation of the evaporated Al-film. **(c)** Second evaporation step at angle $-\theta$. The overlap of the two Al films is determined by the evaporation angles, the width of the resits bridge, and the thickness of the lower resist layer. Drawings taken from Ref. 299.

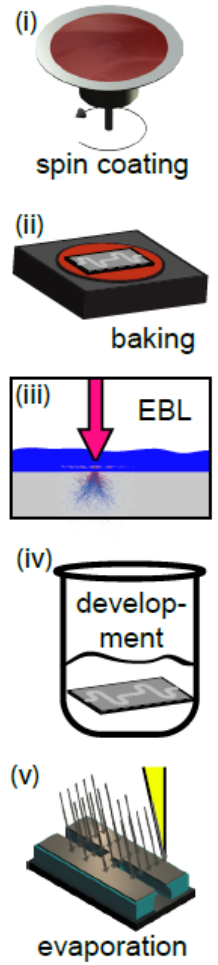
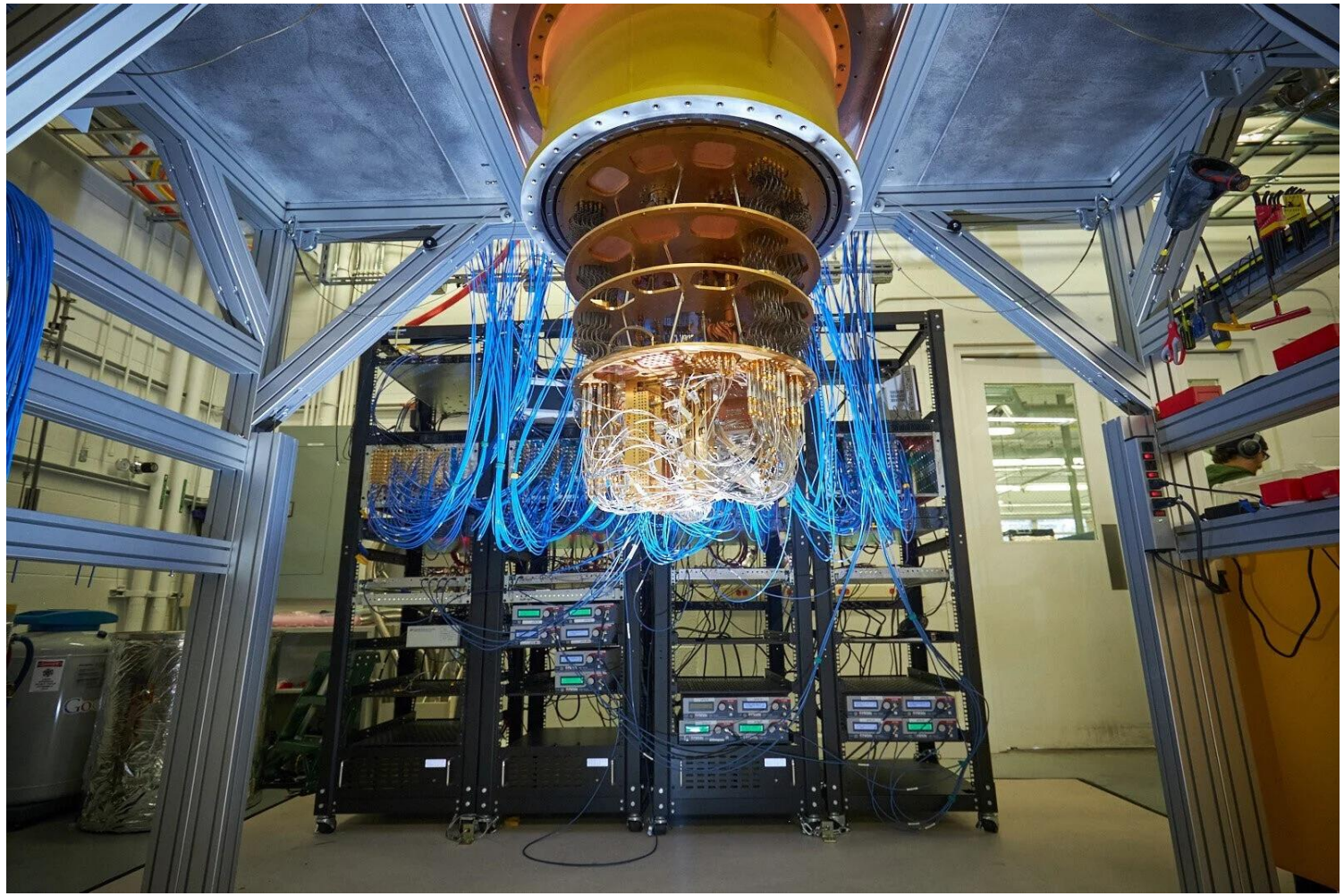


Figure 3.1: Main fabrication steps for Al based Josephson junctions.

IQM

Control challenges



Increasing the qubit counts creates an increased amount of control lines.

Microwave signals are carried in bulky coaxial cables (blue on the left).

For larger systems, developing scalable cabling systems is a must-have requirement.

https://www.scmp.com/news/china/science/article/3125539/chinese-scientists-challenge-googles-quantum-supremacy-claim-new

Cryogenic control: CMOS based

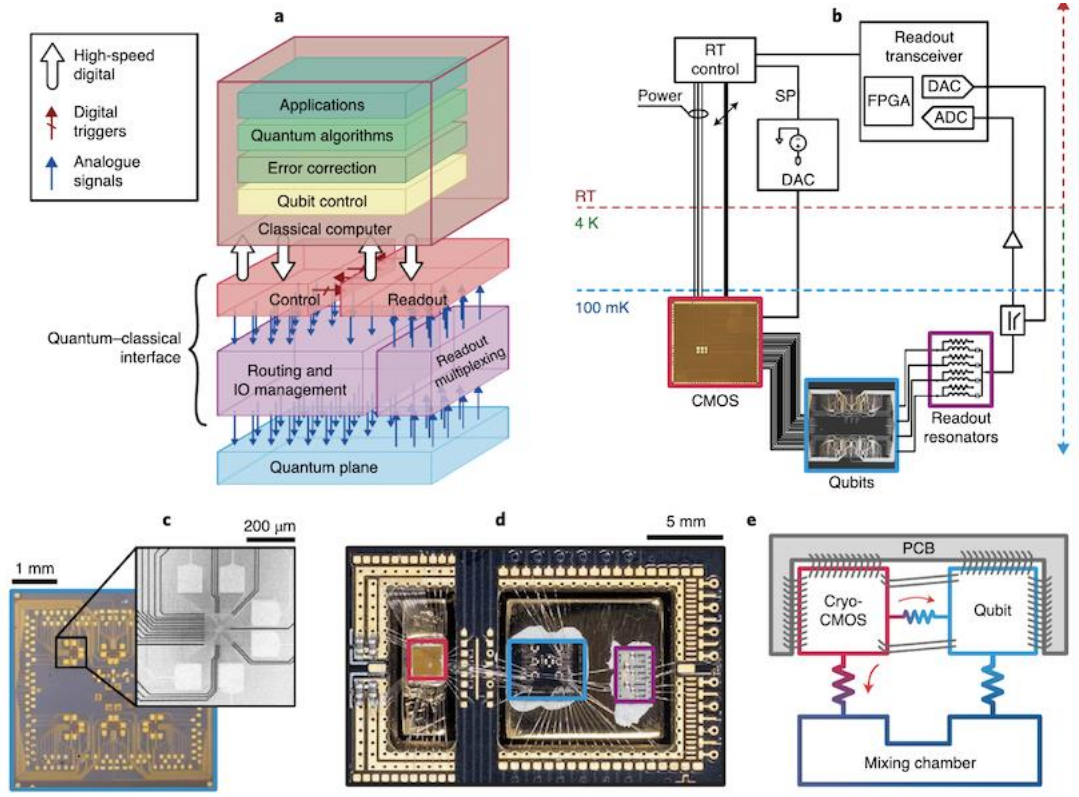


Fig. 1 | The quantum–classical interface of a quantum computer. a, The generic stack of elements needed for quantum computing. **b**, Control and readout sub-systems, distributed between room temperature (RT) and 100 mK. A readout transceiver with integrated field-programmable gate array (FPGA), digital-to-analogue converter (DAC) and analogue-to-digital converter (ADC) is used to readout multiple qubits at once. The cryo-CMOS (brown) chip addresses the input–output (IO) bottleneck for control signals. **c**, Photograph and electron micrograph of our qubit test platform based on GaAs QDs (see Methods for details). **d**, Photograph showing the cryo-CMOS chip (red box), qubit test chip (blue box) and resonator chip (purple box). Each chip is anchored onto a gold-plated copper thermalization pillar, with a separate pillar used for the CMOS chip. **e**, Simplified thermal conductance model of the setup. The intended use of the partially separate cooling pillars is to increase the thermal conductivity to the mixing chamber (big red arrow) while reducing the direct heat (little red arrow) flowing from the hot CMOS chip to the qubit devices.

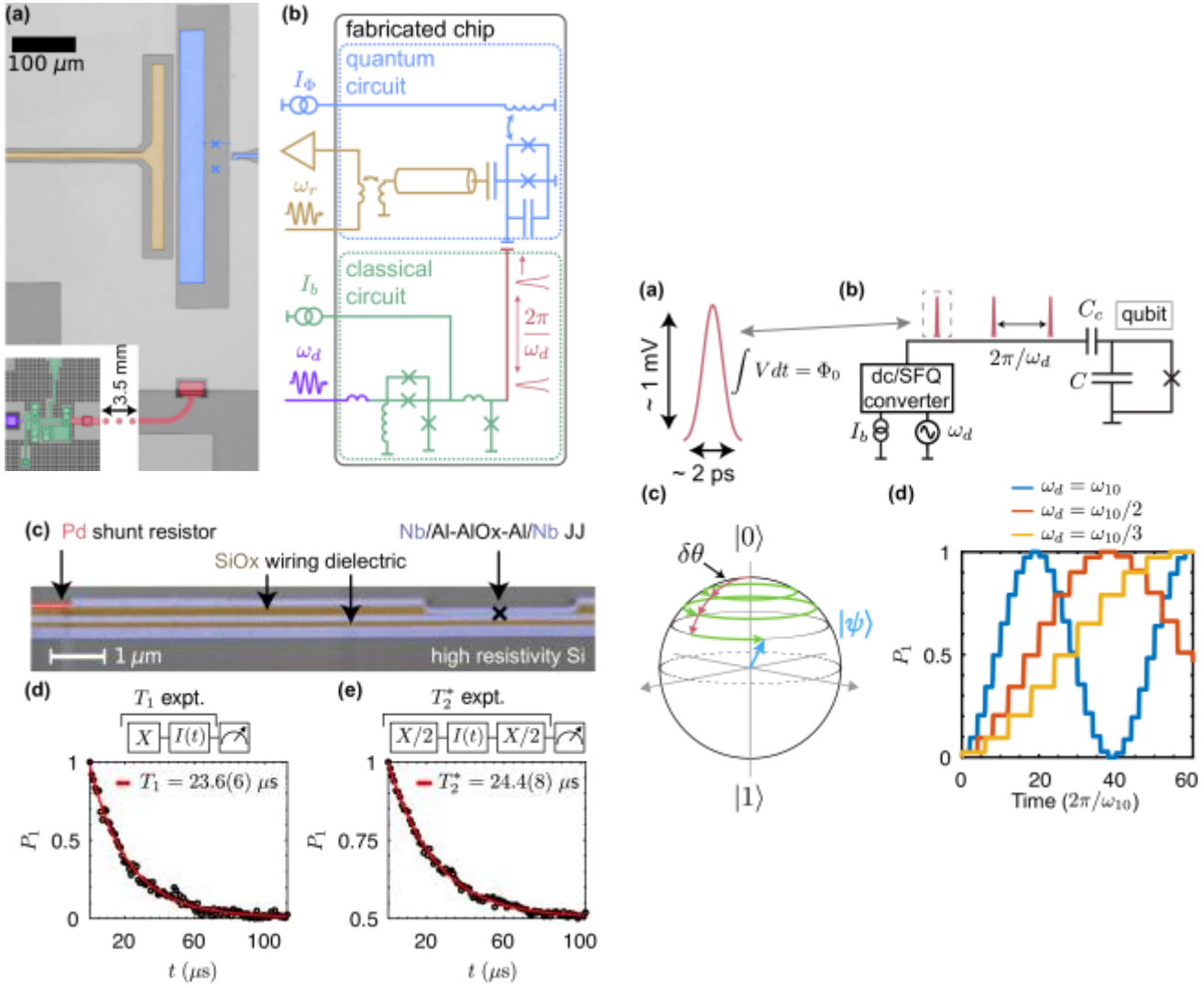
For more than 1000 qubits, feeding all control lines into a cryostat becomes hard. Hence, we need alternative solutions to create control signals.

One approach is to create the signals at cryogenic temperatures inside the cryostat using CMOS based circuits.

The disadvantage of this approach is the large heat generation at low temperatures.

https://www.hpcwire.com/2021/01/28/microsoft-develops-cryo-controller-chip-gooseberry-for-quantum-computing/

Cryogenic control: JJ based



For more than 1000 qubits, feeding all control lines into a cryostat becomes hard. Hence, we need alternative solutions to create control signals.

Another approach is to create the signals at cryogenic temperatures inside the cryostat using Josephson Junction based circuits.

The advantage of this approach is the low heat generation at low temperatures.

IQM https://www.groundai.com/project/digital-coherent-control-of-a-superconducting-qubit/1

Software challenges: Calibration

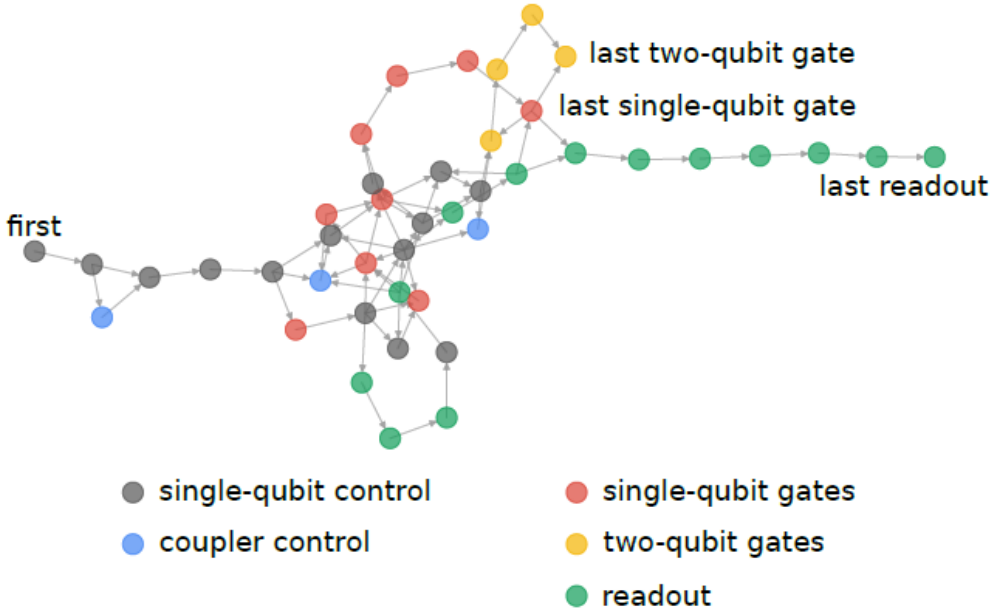


FIG. S10. **Optimus calibration graph for Sycamore.** Calibration of physical qubits is a bootstrapping procedure between different pulse sequences or "experiments" to extract control and system parameters. Initial experiments are coarse and have interplay between fundamental operations and elements such as single-qubit gates, readout, and the coupler. Final experiments involve precise metrology for each of the qubit operations: single-qubit gates, two-qubit gates, and readout.

Automated tune up of many qubits is a hard task by itself.

People use machine learning tools to create effective methods.

 arXiv:1910.11333v2

General challenge: Scaling up

- Creating a single quantum circuit with high accuracy is simple. Scaling to millions is extremely challenging.

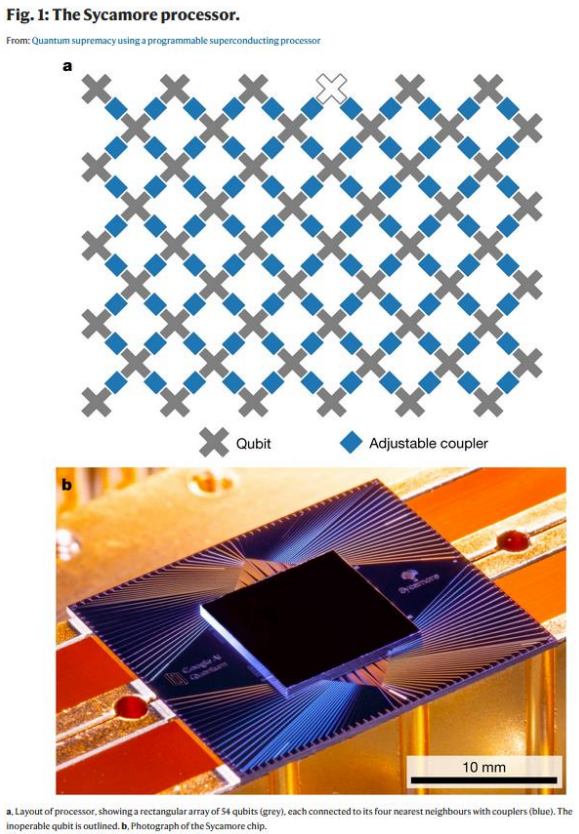
- To compensate for the errors of the system, one can detect and correct them. However, error correction creates a qubit overhead of 100 to 1000.

- Currently known algorithms require thousands – millions of qubits. Currently available processors contain less than 100 qubits. This is called the Software-Hardware gap.

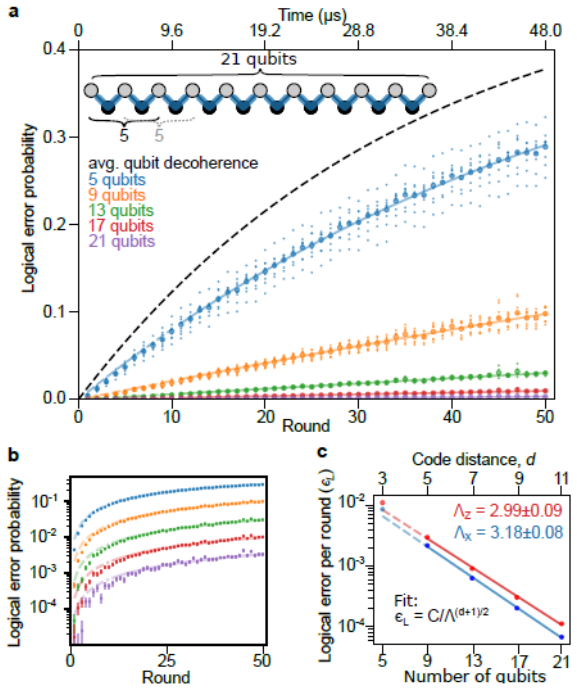
Agenda for today

12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

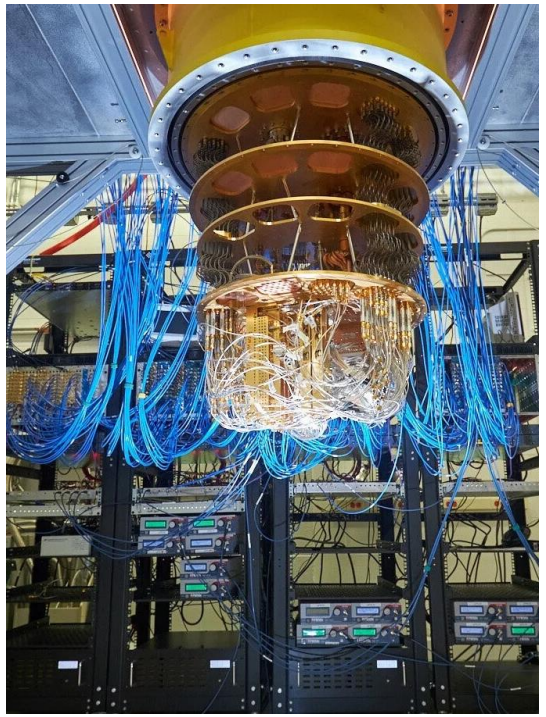
a.



b.



c.



IQM

Agenda for lectures 7-12

7. Quantization of electrical networks
    a. Harmonic oscillator: Lagrangian, eigenfrequency
    b. Transfer step: LC oscillator, Legendre transform to Hamiltonian
    d. Quantization of oscillators
8. Superconducting quantum circuits
    a. Qubits: Transmon qubit, Charge qubit, Flux qubit 1st DiVincenzo criteria
    b. Circuit-QED: Rabi model
    c. Rotating Wave approximation: Jaynes-Cummings model
9. Single-qubit operations:
    a. Initialization 2nd DiVincenzo criteria
    b. Readout 5th DiVincenzo criteria
    c. Control:T1, T2 measurements, Randomized benchmarking 3rd DiVincenzo criteria
10. Two-qubit operations: Architectures for 2-qubit gates 4th DiVincenzo criteria
    a. iSWAP
    b. cPhase
    c. cNot
11. Quantum algorithms
    a. Deutsch-Josza Algorithm
    b. Parameterised circuits and VQE
12. Challenges in quantum computing
    a. SW-HW gap (qubit quality & number, gate depth)
    b. Error-correction
    c. Scaling challenges

IQM