# Prediction: 2025 is the year quantum computing advances from physical qubits to logical qubits

Prediction: 2025 is the year quantum computing advances from physical qubits to logical qubits© Shutterstock / Phonlamai

Quantum computing has long been a subject of fascination and excitement, promising to solve complex problems far beyond the capabilities of classical computers. As we enter 2025, this transformative technology is poised to take a giant leap forward, progressing from physical qubits to logical qubits. This shift marks a pivotal moment in the quantum industry’s journey, one which sets the stage for exciting advancements across various industries and addresses the technical challenges that have, until now, constrained the potential of quantum computers.

# Predicting the leap from physical to logical qubits

In a similar way that classical computers use bits to store information, quantum computers are built on the use of physical qubits to store quantum information. Unfortunately, physical qubits are sensitive to environmental noise, making them error-prone and unsuitable for solving large computational problems. This limitation can be overcome by using quantum error correction which encodes information across multiple physical qubits to create more reliable, error-resistant units called logical qubits. This transition will allow quantum computers to tackle real-world problems, moving the technology from experimental to practical, large-scale applications.

To effectively create many logical qubits, quantum computing hardware needs to incorporate multiple advanced technologies and algorithms and provide sufficient reliable computational resources in a sustainable way. Recent technical advances across the quantum industry, high-profile industrial partnerships, and an increasing number of scientists and engineers working on quantum error correction has accelerated the timeline to creating logical qubits much sooner than expected.

# What the shift to logical qubits will enable

The transition to logical qubits in 2025 will dramatically enhance the capabilities of quantum computers, with far-reaching implications across multiple sectors.

Quantum chemistry is expected to be one of the first quantum computing applications to leverage logical qubits to simulate chemical reactions with much higher precision than classical computers. The first wave of studies will be highly scientific, but there will be a quick turning point to the exploration of real-world applications that will have tangible economic and societal value.

Another field which will benefit from the transition to logical qubits is renewable energy and battery development. By simulating physical quantum processes, such as the behavior of electrons in new materials, quantum computers will help accelerate the development of more efficient batteries and energy storage solutions. This could lead to breakthroughs in electric vehicles, renewable energy grids, and the quest for sustainable energy solutions.

The list of applications expands further as logical qubit counts and quality increase. For example, accelerated exploration of vast chemical spaces for potential drug identification for pharmaceutical applications, modeling of complex systems in the financial sector, optimizing interconnected supply chain problems for the manufacturing industry, modeling physical properties of new materials, and improving the performance of machine learning applications. All of these will be accelerated through the availability of logical qubits, allowing users to run deeper and more complex algorithms than before.

Aside from the growing interest in quantum computing applications, one key issue that has become increasingly prominent is the question on the sustainability of the quantum technologies themselves. As we have seen with AI advancements and data centers, the physical and ecological footprint of digital technologies can be drastic, and quantum computing will have to find its place in a much more environmentally friendly way. Sustainably-scalable modalities such as neutral-atom computing are gaining popularity in the quantum field due to its rapid advances in technical performance and its relatively small ecological footprint: a full-scale neutral-atom system fits inside a typical conference room and consumes less energy than a single data center rack.

# 2025: a quantum leap forward

As we approach 2025, the quantum computing industry is on the verge of a significant transformation. The move from physical to logical qubits will be a game-changer, addressing the challenges of error rates and scalability that have held back quantum computing for years. With forward-thinking companies leading the way, the next generation of quantum systems will be more stable, sustainable, and powerful than ever before.

This transition will open the door to a new era of quantum computing, one in which previously unsolvable problems are tackled head-on. By the end of 2025, we may witness quantum computing move from theoretical promise to practical reality, transforming industries and reshaping the future of technology.

# Humanity May Reach Singularity Within Just 6 Years, Trend Shows

- By one unique metric, we could approach technological singularity by the end of this decade, if not sooner.
- A translation company developed a metric, Time to Edit (TTE), to calculate the time it takes for professional human editors to fix AI-generated translations compared to human ones. This may help quantify the speed toward singularity.
- An AI that can translate speech as well as a human could change society.

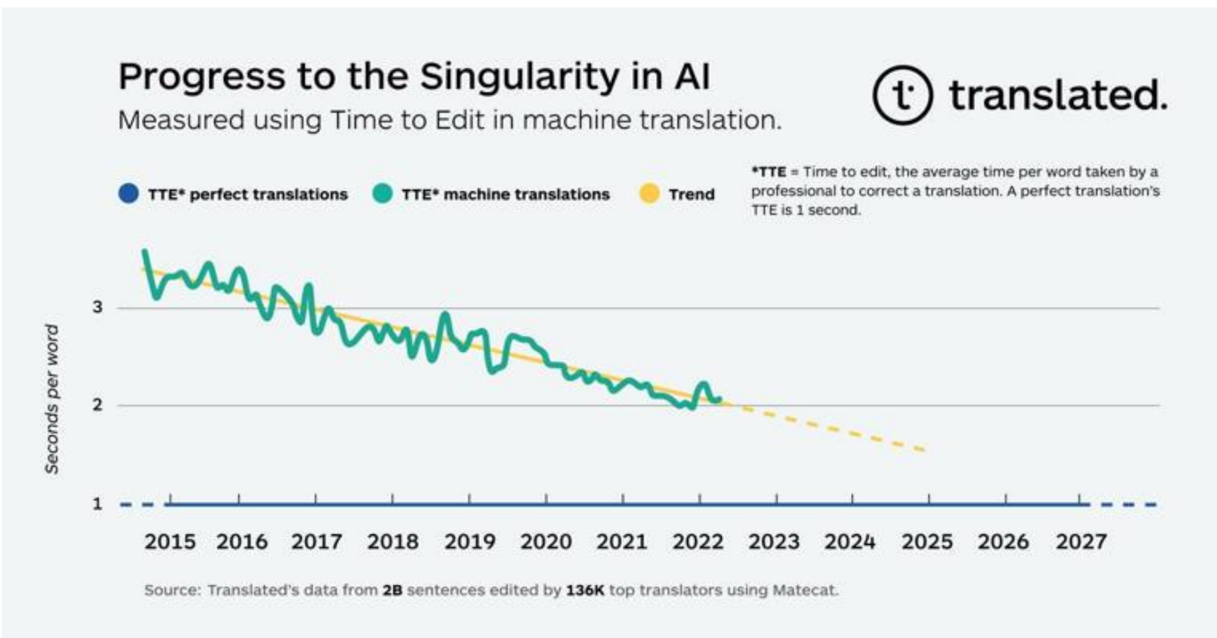
In the world of artificial intelligence, the idea of "singularity" looms large. This slippery concept describes the moment AI exceeds beyond human control and rapidly transforms society. The tricky thing about AI singularity (and why it borrows terminology from black hole physics) is that it's enormously difficult to predict where it begins and nearly impossible to know what's beyond this technological "event horizon."

However, some AI researchers are on the hunt for signs of reaching singularity measured by AI progress approaching the skills and ability comparable to a human.

One such metric, defined by Translated, a Rome-based translation company, is an AI's ability to translate speech at the accuracy of a human. Language is one of the most difficult AI challenges, but a computer that could close that gap could theoretically show signs of Artificial General Intelligence (AGI).

"That's because language is the most natural thing for humans," Translated CEO Marco Trombetti said at a conference in Orlando, Florida, in December 2022. "Nonetheless, the data Translated collected clearly shows that machines are not that far from closing the gap."

The company tracked its AI's performance from 2014 to 2022 using a metric called "Time to Edit," or TTE, which calculates the time it takes for professional human editors to fix AI-generated translations compared to human ones. Over that 8-year period and analyzing over 2 *billion* post-edits, Translated's AI showed a slow, but undeniable improvement as it slowly closed the gap toward human-level translation quality.




Data showing speed to singularity© Translated

On average, it takes a human translator roughly one second to edit each word of another human translator, according to Translated. In 2015, it took professional editors approximately 3.5 seconds per word to check a machine-translated (MT) suggestion—today, that number is just 2 seconds. If the trend continues, Translated's AI will be as good as human-produced translation by the end of the decade (or even sooner).

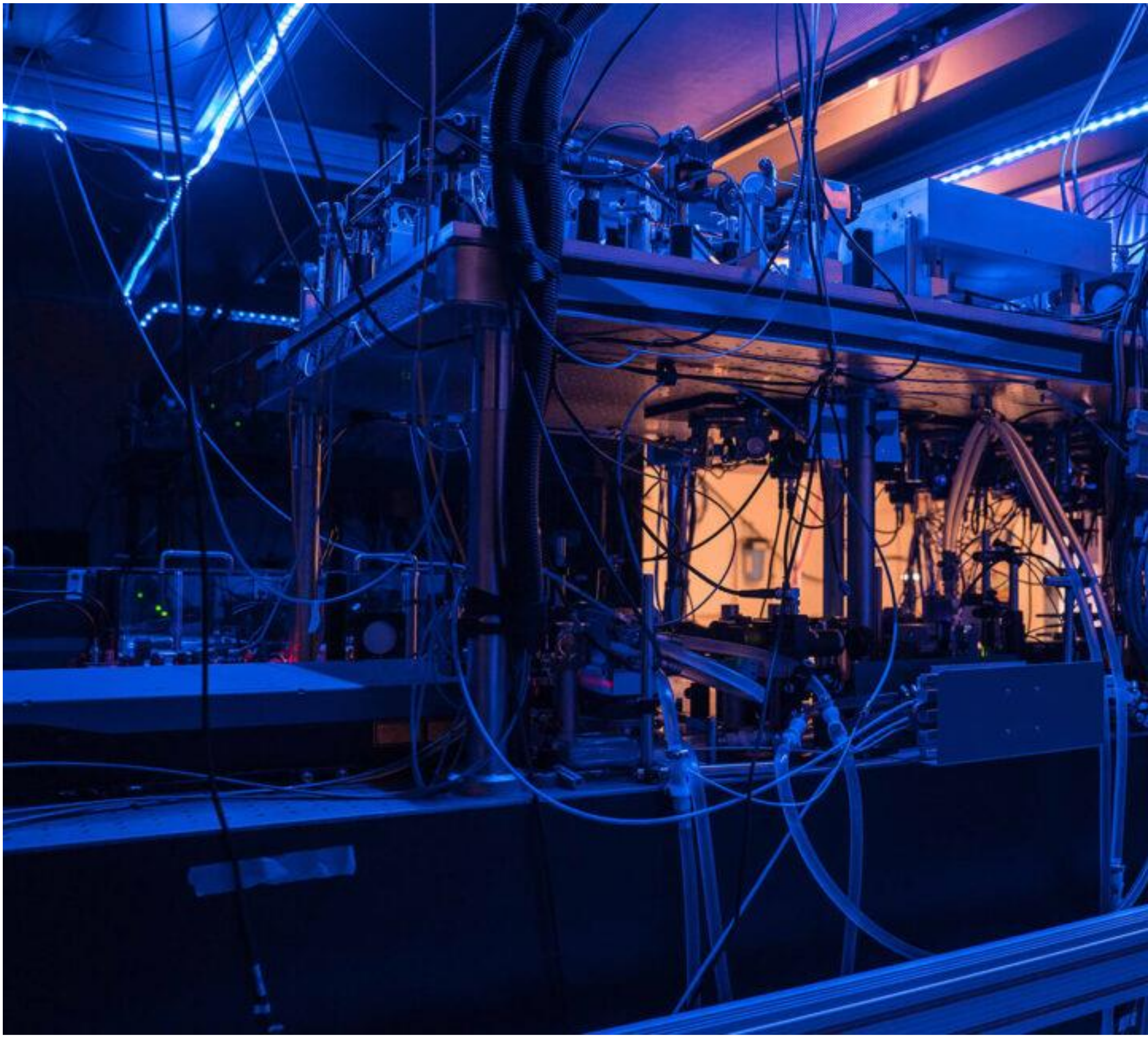
"The change is so small that every single day you don't perceive it, but when you see progress … across 10 years, that is impressive," Trombetti said on a podcast. "This is the first time ever that someone in the field of artificial intelligence did a prediction of the speed to singularity."

Although this is a novel approach to quantifying how close humanity is to approaching singularity, this definition of singularity runs into similar problems of identifying AGI more broadly. And while perfecting human speech is certainly a frontier in AI research, the impressive skill doesn't necessarily make a machine intelligent (not to mention how many researchers don't even agree on what "intelligence" is).

Whether these hyper-accurate translators are harbingers of our technological doom or not, that doesn't lessen Translated's AI accomplishment. An AI capable of translating speech as well as a human could very well change society, even if the true "technological singularity" remains ever elusive.

# Microsoft and Atom Computing combine for quantum error correction demo

New work provides a good view of where the field currently stands.



The first-generation tech demo of Atom's hardware. Things have progressed considerably since. Credit: Atom Computing

In September, Microsoft [made an unusual combination of announcements](). It demonstrated progress with quantum error correction, something that will be needed for the technology to move much beyond the interesting demo phase, using hardware from a quantum computing startup called Quantinuum. At the same time, however, the company also announced that it was forming a partnership with a different startup, Atom Computing, which uses a different technology to make qubits available for computations.

Given that, it was probably inevitable that the folks in Redmond, Washington, would want to show that similar error correction techniques would also work with Atom Computing's hardware. It didn't take long, as the two companies are releasing a draft manuscript describing their work on error correction today. The paper serves as both a good summary of where things currently stand in the world of error correction, as well as a good look at some of the distinct features of computation using neutral atoms.

## Atoms and errors

While we have various technologies that provide a way of storing and manipulating bits of quantum information, none of them can be operated error-free. At present, errors make it difficult to perform even the simplest computations that are clearly beyond the capabilities of classical computers. More sophisticated algorithms would inevitably encounter an error before they could be completed, a situation that would remain true even if we could somehow improve the hardware error rates of qubits by a factor of 1,000—something we're unlikely to ever be able to do.

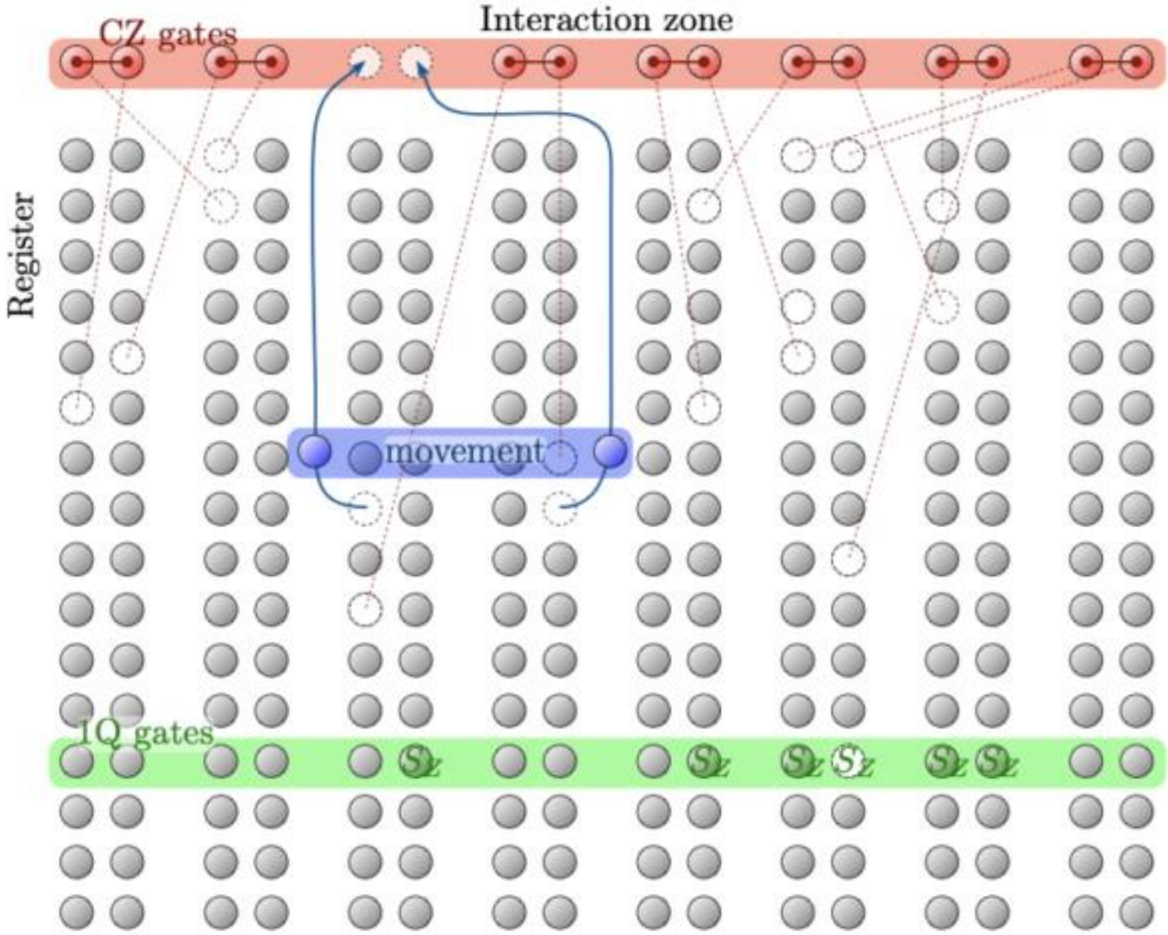
The solution to this is to use what are called logical qubits, which distribute quantum information across multiple hardware qubits and allow the detection and correction of errors when they occur. Since multiple qubits get linked together to operate as a single logical unit, the hardware error rate still matters. If it's too high, then adding more hardware qubits just means that errors will pop up faster than they can possibly be corrected.

We're now at the point where, for a number of technologies, hardware error rates have passed the break-even point, and adding more hardware qubits can lower the error rate of a logical qubit based on them. This was demonstrated using neutral atom qubits by an academic lab at Harvard University about a year ago. The new manuscript demonstrates that it also works on a commercial machine from Atom Computing.

Neutral atoms, which can be held in place using a lattice of laser light, have a number of distinct advantages when it comes to quantum computing. Every single atom will behave identically, meaning that you don't have to manage the device-to-device variability that's inevitable with fabricated electronic qubits. Atoms can also be moved around, allowing any atom to be entangled with any other. This any-to-any connectivity can enable more efficient algorithms and error-correction schemes. The quantum information is typically stored in the spin of the atom's nucleus, which is shielded from environmental influences by the cloud of electrons that surround it, making them relatively long-lived qubits.

Operations, including gates and readout, are performed using lasers. The way the physics works, the spacing of the atoms determines how the laser affects them. If two atoms are a critical distance apart, the laser can perform a single operation, called a two-qubit gate, that affects both of their states. Anywhere outside this distance, and a laser only affects each atom individually. This allows a fine control over gate operations.

That said, operations are relatively slow compared to some electronic qubits, and atoms can occasionally be lost entirely. The optical traps that hold atoms in place are also contingent upon the atom being in its ground state; if any atom ends up stuck in a different state, it will be able to drift off and be lost. This is actually somewhat useful, in that it converts an unexpected state into a clear error.




Atom Computing's system. Rows of atoms are held far enough apart so that a single laser sent across them (green bar) only operates on individual atoms. If the atoms are moved to the interaction zone (red bar), a laser can perform gates on pairs of atoms. Spaces where atoms can be held can be left empty to avoid performing unneeded operations. Credit: Reichardt, et al.

The machine used in the new demonstration hosts 256 of these neutral atoms. Atom Computing has them arranged in sets of parallel rows, with space in between to let the atoms be shuffled around. For single-qubit gates, it's possible to shine a laser across the rows, causing every atom it touches to undergo that operation. For two-qubit gates, pairs of atoms get moved to the end of the row and moved a specific distance apart, at which point a laser will cause the gate to be performed on every pair present.

Atom's hardware also allows a constant supply of new atoms to be brought in to replace any that are lost. It's also possible to image the atom array in between operations to determine whether any atoms have been lost and if any are in the wrong state.

## It's only logical

As a general rule, the more hardware qubits you dedicate to each logical qubit, the more simultaneous errors you can identify. This identification can enable two ways of handling the error. In the first, you simply discard any calculation with an error and start over. In the second, you can use information about the error to try to fix it, although the repair involves additional operations that can potentially trigger a separate error.

For this work, the Microsoft/Atom team used relatively small logical qubits (meaning they used very few hardware qubits), which meant they could fit more of them within 256 total hardware qubits the machine made available. They also checked the error rate of both error detection with discard and error detection with correction.

The research team did two main demonstrations. One was placing 24 of these logical qubits into what's called a cat state, named after Schrödinger's hypothetical feline. This is when a quantum object simultaneously has non-zero probability of being in two mutually exclusive states. In this case, the researchers placed 24 logical qubits in an entangled cat state, the largest ensemble of this sort yet created. Separately, they implemented what's called the Bernstein-Vazirani algorithm. The classical version of this algorithm requires individual queries to identify each bit in a string of them; the quantum version obtains the entire string with a single query, so is a notable case of something where a quantum speedup is possible.

Both of these showed a similar pattern. When done directly on the hardware, with each qubit being a single atom, there was an appreciable error rate. By detecting errors and discarding those calculations where they occurred, it was possible to significantly improve the error rate of the remaining calculations. Note that this doesn't eliminate errors, as it's possible for multiple errors to occur simultaneously, altering the value of the qubit without leaving an indication that can be spotted with these small logical qubits.

Discarding has its limits; as calculations become increasingly complex, involving more qubits or operations, it will inevitably mean every calculation will have an error, so you'd end up wanting to discard everything. Which is why we'll ultimately need to correct the errors.

In these experiments, however, the process of correcting the error—taking an entirely new atom and setting it into the appropriate state—was also error-prone. So, while it could be done, it ended up having an overall error rate that was intermediate between the approach of catching and discarding errors and the rate when operations were done directly on the hardware.

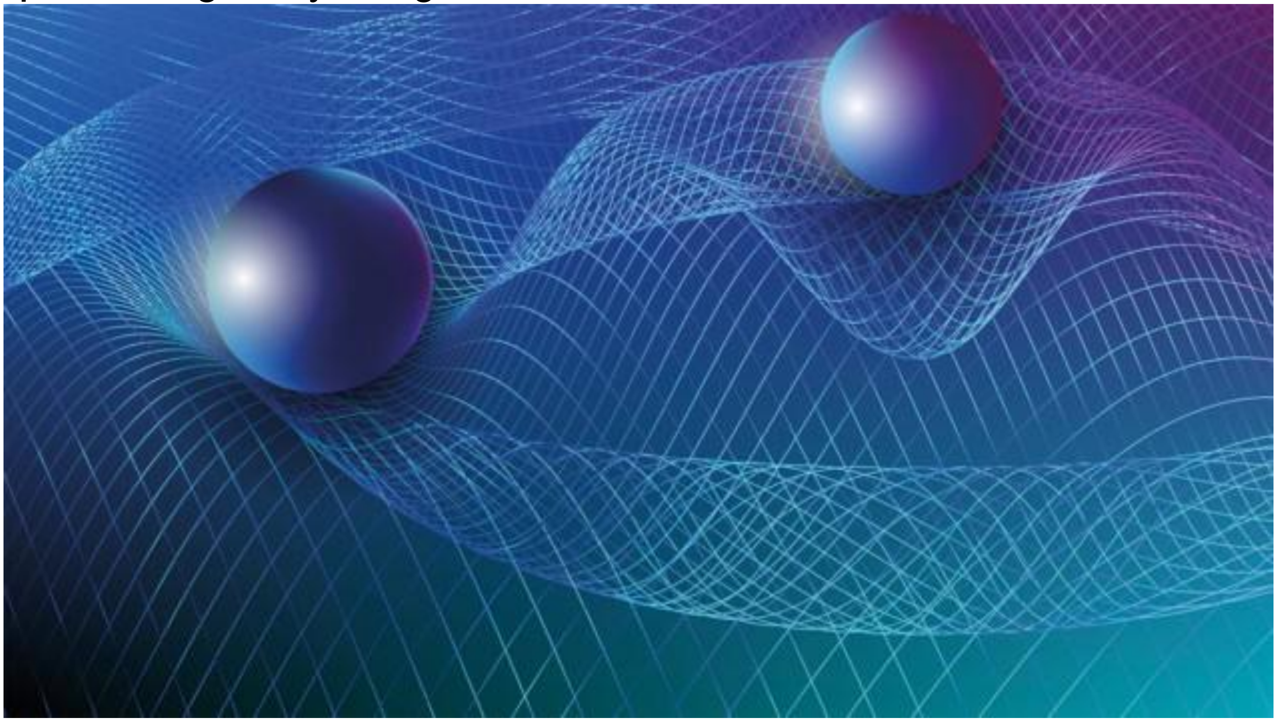
In the end, the current hardware has an error rate that's good enough that error correction actually improves the probability that a set of operations can be performed without producing an error. But not good enough that we can perform the sort of complex operations that would lead quantum computers to have an advantage in useful calculations. And that's not just true for

Atom's hardware; similar things can be said for other error-correction demonstrations done on different machines.

There are two ways to go beyond these current limits. One is simply to improve the error rates of the hardware qubits further, as fewer total errors make it more likely that we can catch and correct them. The second is to increase the qubit counts so that we can host larger, more robust logical qubits. We're obviously going to need to do both, and Atom's partnership with Microsoft was formed in the hope that it will help both companies get there faster.

# Quantum computing

## Quantum error correction research yields unexpected quantum gravity insights



Quantum link: New research has revealed an unexpected connection between the physics of approximate error-correcting codes and quantum gravity. (Courtesy: Shutterstock/Evgenia Fux)

In computing, quantum mechanics is a double-edged sword. While computers that use quantum bits, or qubits, can perform certain operations much faster than their classical counterparts, these qubits only maintain their quantum nature – their superpositions and entanglement – for a limited time. Beyond this so-called coherence time, interactions with the environment, or noise, lead to loss of information and errors. Worse, because quantum states cannot be copied – a consequence of quantum mechanics known as the no-cloning theorem – or directly observed without collapsing the state, correcting these errors requires more sophisticated strategies than the simple duplications used in classical computing.

One such strategy is known as an approximate quantum error correction (AQEC) code. Unlike exact QEC codes, which aim for perfect error correction, AQEC codes help quantum computers return to almost, though not exactly, their intended state. "When we can allow mild degrees of approximation, the code can be much more efficient," explains Zi-Wen Liu, a theoretical physicist who studies quantum information and computation at China's Tsinghua University. "This is a very worthwhile trade-off."

The problem is that the performance and characteristics of AQEC codes are poorly understood. For instance, AQEC conventionally entails the expectation that errors will become negligible as system size increases. This can in fact be achieved simply by appending a series of redundant qubits to the logical state for random local noise; the likelihood of the logical information being affected would, in that case, be vanishingly small. However, this approach is ultimately unhelpful. This raises the questions: What separates good (that is, non-trivial) codes from bad ones? Is this dividing line universal?

**Establishing a new boundary**

So far, scientists have not found a general way of differentiating trivial and non-trivial AQEC codes. However, this blurry boundary motivated Liu, Daniel Gottesman of the University of Maryland, US; Jinmin Yi of Canada's Perimeter Institute for Theoretical Physics; and Weicheng Ye at the University of British Columbia, Canada, to develop a framework for doing so.

To this end, the team established a crucial parameter called subsystem variance. This parameter describes the fluctuation of subsystems of states within the code space, and, as the team discovered, links the effectiveness of AQEC codes to a property known as quantum circuit complexity.

Circuit complexity, an important concept in both computer science and physics, represents the optimal cost of a computational process. This cost can be assessed in many ways, with the most intuitive metrics being the minimum time or the “size” of computation required to prepare a quantum state using local gate operations. For instance, how long does it take to link up the individual qubits to create the desired quantum states or transformations needed to complete a computational task?

The researchers found that if the subsystem variance falls below a certain threshold, any code within this regime is considered a nontrivial AQEC code and subject to a lower bound of circuit complexity. This finding is highly general and does not depend on the specific structures of the system. Hence, by establishing this boundary, the researchers gained a more unified framework for evaluating and using AQEC codes, allowing them to explore broader error correction schemes essential for building reliable quantum computers.

**A quantum leap**

But that wasn’t all. The researchers also discovered that their new AQEC theory carries implications beyond quantum computing. Notably, they found that the dividing line between trivial and non-trivial AQEC codes also arises as a universal “threshold” in other physical scenarios – suggesting that this boundary is not arbitrary but rooted in elementary laws of nature.

One such scenario is the study of topological order in condensed matter physics. Topologically ordered systems are described by entanglement conditions and their associated code properties. These conditions include long-range entanglement, which is a circuit complexity condition, and topological entanglement entropy, which quantifies the extent of long-range entanglement. The new framework clarifies the connection between these entanglement conditions and topological quantum order, allowing researchers to better understand these exotic phases of matter.

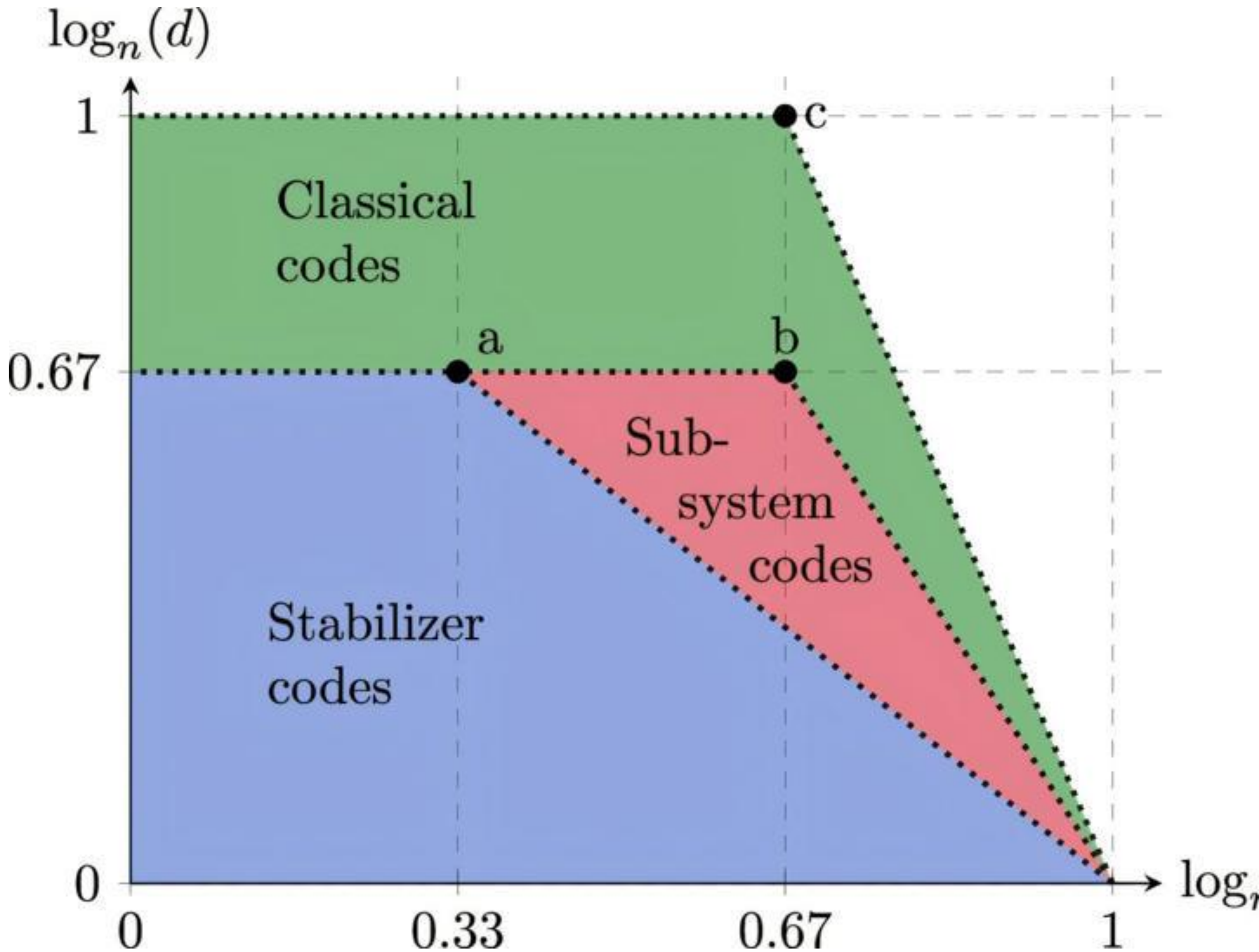
A more surprising connection, though, concerns one of the deepest questions in modern physics: how do we reconcile quantum mechanics with Einstein’s general theory of relativity? While quantum mechanics governs the behavior of particles at the smallest scales, general relativity accounts for gravity and space-time on a cosmic scale. These two pillars of modern physics have some incompatible intersections, creating challenges when applying quantum mechanics to strongly gravitational systems.

In the 1990s, a mathematical framework called the anti-de Sitter/conformal field theory correspondence (AdS/CFT) emerged as a way of using CFT to study quantum gravity even though it does not incorporate gravity. As it turns out, the way quantum information is encoded in CFT has conceptual ties to QEC. Indeed, these ties have driven recent advances in our understanding of quantum gravity.

By studying CFT systems at low energies and identifying connections between code properties and intrinsic CFT features, the researchers discovered that the CFT codes that pass their AQEC threshold might be useful for probing certain symmetries in quantum gravity. New insights from AQEC codes could even lead to new approaches to spacetime and gravity, helping to bridge the divide between quantum mechanics and general relativity.

Some big questions remain unanswered, though. One of these concerns the line between trivial and non-trivial codes. For instance, what happens to codes that live close to the boundary? The researchers plan to investigate scenarios where AQEC codes could outperform exact codes, and to explore ways to make the implications for quantum gravity more rigorous. They hope their study will inspire further explorations of AQEC's applications to other interesting physical systems.

# Compact error correction: Toward a more efficient 'quantum hard drive'



An illustration of existing bounds on local codes in 3D, and constructions known to saturate them. Credit: Nature Communications (2024). DOI: 10.1038/s41467-024-53881-3

Two quantum information theorists at the University of Sydney Nano Institute have solved a decades-old problem that will require fewer qubits to suppress more errors in quantum hardware.

University of Sydney quantum researchers Dominic Williamson and Nouédyn Baspin have revealed a transformative new architecture for managing errors that emerge in the operation of quantum computers.

Their innovative theoretical approach promises to not only enhance the reliability of quantum information storage but also significantly reduce the physical computing resources needed to create "logical qubits" (or "quantum switches" that can perform useful calculations). This should lead to the development of a more compact quantum hard drive.

Lead author Dr. Williamson from the University of Sydney Nano Institute and School of Physics said, "There remain significant barriers to overcome in the development of a universal quantum computer. One of the biggest is the fact we need to use most of the qubits—quantum switches at the heart of the machines—to suppress the errors that emerge as a matter of course within the technology.

"Our proposed quantum architecture will require fewer qubits to suppress more errors, liberating more for useful quantum processing," said Dr. Williamson, who is currently working for 12 months as a quantum researcher at IBM.

At the heart of their theoretical architecture is a three-dimensional structure that allows for quantum error correction across two-dimensions. Current error correction architecture, also constructed within a 3D system of qubits, works to reduce errors in just one dimension along a single line of connected qubits.

Error correction is performed by writing code that operates through the qubit structure, a latticework of how the quantum switches are organized. The objective is to win an "arms race" where physical qubits are used to suppress errors as they emerge, by using as few qubits as possible to reduce errors.

Dr. Williamson said, "Current 3D codes in a block of dimensions L x L x L can only manage L errors. Our codes can handle errors that scale like $L^2$ (LxL)—a significant improvement."

It has been known for more than a decade that a three-dimensional quantum error correction architecture (LxLxL) had an upper limit of LxL, but no such codes had been discovered.

Ph.D. student and co-author Baspin said, "This means that we have discovered new states of quantum matter in three dimensions that have properties never seen before."

Quantum computers promise to solve complex problems that are currently beyond the reach of classical computers. However, one of the major challenges in realizing practical quantum computing is the need for robust error correction mechanisms.

Traditional quantum error correction methods, such as the widely studied surface code, have limitations in terms of scalability and resource efficiency.

Williamson and Baspin's research introduces a three-dimensional architecture that effectively manages quantum errors within two-dimensional layers. By leveraging this three-dimensional topological code, the researchers have demonstrated that it is possible to achieve optimal scaling while significantly reducing the number of physical qubits needed. This advance is crucial for the development of scalable quantum computers, as it allows for a more compact construction of quantum memory systems.

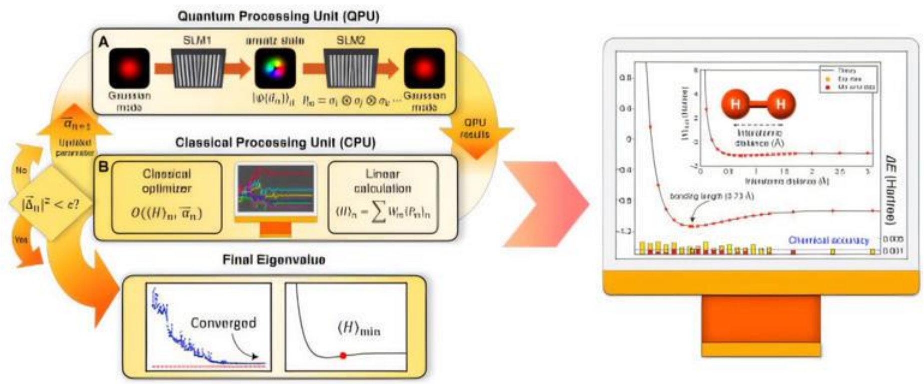
By reducing the physical qubit overhead, the findings pave the way for the creation of a more compact quantum hard drive—an efficient quantum memory system capable of storing vast amounts of quantum information reliably.

Quantum theorist and Director of the University of Sydney Nano Institute, Professor Stephen Bartlett, said, "This advancement could help transform the way quantum computers are built and operated, making them more accessible and practical for a wide range of applications, from cryptography to complex simulations of quantum many-body systems."

**More information:** Dominic J. Williamson et al, Layer codes, *Nature Communications* (2024). DOI: 10.1038/s41467-024-53881-3

Provided by University of Sydney

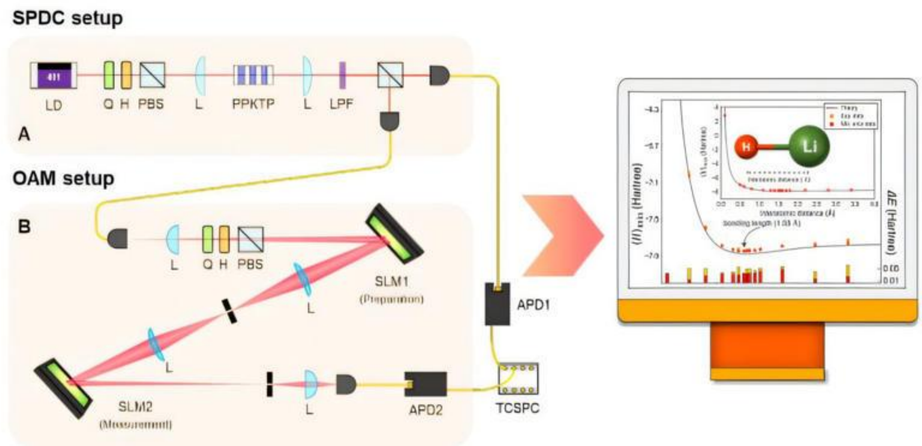
# Photon qubits challenge AI, enabling more accurate quantum computing without error-correction techniques




Orbital Angular Momentum Quantum-based VQE - Hydrogen (H2) Molecule / A quantum processing device based on orbital angular momentum qubit states is implemented by using spatial light modulators. The ground state energy of a H2 molecular model based is estimated on VQE. Credit: Korea Institute of Science and Technology

In an era where AI and data are driving the scientific revolution, quantum computing technology is emerging as another game-changer in the development of new drugs and new materials.

Dr. Hyang-Tag Lim's research team at the Center for Quantum Technology at the Korea Institute of Science and Technology (KIST) has implemented a quantum computing algorithm that can estimate interatomic bond distances and ground state energies with chemical accuracy using fewer resources than conventional methods, and has succeeded in performing accurate calculations without the need for additional quantum error mitigation techniques.



Orbital Angular Momentum Quantum Based VQE - LiH Molecules / Scheme for orbital angular momentum qudit based VQE experiment. Estimation of the ground state energy of the LiH molecular model which corresponds to 16 dimensions with the same experimental setup for the four-dimensional hydrogen molecule. Credit: Korea Institute of Science and Technology

The work is published in the journal *Science Advances*.

Quantum computers have the disadvantage of rapidly increasing errors as the computational space grows at the current level. To overcome this, the Variational Quantum Eigensolver (VQE) method, which combines the advantages of classical and quantum computers, has emerged.

VQE is a hybrid algorithm designed to use a Quantum Processing Unit (QPU) and a Classical Processing Unit (CPU) together to perform faster computations.

Global research teams, including IBM and Google, are investigating it in a variety of quantum systems, including superconducting and trapped-ion systems. However, qubit-based VQE is currently only implemented up to 2 qubits in photonic systems and 12 qubits in superconducting systems, and is challenged by error issues that make it difficult to scale when more qubits and complex computations are required.

Instead of qubits, the team utilized a higher-dimensional form of quantum information called a qudit. A qudit is a quantum unit that can have multiple states, including 0, 1, and 2, in addition to the 0 and 1 that a traditional qubit can represent, which is advantageous for complex quantum computations.

In this study, a qudit was implemented by the orbital angular momentum state of a single-photon, and dimensional expansion was possible by adjusting the phase of a photon through holographic images. This allowed for high-dimensional calculations without complex quantum gates, reducing errors.

The team used the method to perform quantum chemistry calculations with VQE to estimate the bond length between hydrogen molecules in four dimensions and lithium hydride (LiH) molecules in 16 dimensions, the first time 16-dimensional calculations have been realized in photonic systems.

While conventional VQEs from IBM, Google, and others are required error mitigation techniques for chemical accuracy, the KIST team's VQE achieved chemical accuracy without any error mitigation techniques. This demonstrates how high accuracy can be achieved with fewer resources, showing the potential for widespread application in industries where molecular properties are important. It is also expected to be useful in solving complex problems such as climate modeling.

"By securing qudit-based quantum computing technology that can achieve chemical accuracy with fewer resources, we expect it to be used in various practical fields, such as developing new drugs and improving battery performance," said Dr. Hyang-Tag Lim of KIST.

**More information:** Byungjoo Kim et al, Qudit-based variational quantum eigensolver using photonic orbital angular momentum states, *Science Advances* (2024). DOI: 10.1126/sciadv.ado3472

Provided by National Research Council of Science and Technology

# AlphaQubit tackles one of quantum computing's biggest challenges

Quantum computers have the potential to revolutionize drug discovery, material design and fundamental physics — that is, if we can get them to work reliably.

Certain problems, which would take a conventional computer billions of years to solve, would take a quantum computer just hours. However, these new processors are more prone to noise than conventional ones. If we want to make quantum computers more reliable, especially at scale, we need to accurately identify and correct these errors.

In a paper published today in Nature, we introduce AlphaQubit, an AI-based decoder that identifies quantum computing errors with state-of-the-art accuracy. This collaborative work brought together Google DeepMind's machine learning knowledge and Google Quantum AI's error correction expertise to accelerate progress on building a reliable quantum computer.

Accurately identifying errors is a critical step towards making quantum computers capable of performing long computations at scale, opening the doors to scientific breakthroughs and many new areas of discovery.

## Correcting quantum computing errors

Quantum computers harness the unique properties of matter at the smallest scales, such as superposition and entanglement, to solve certain types of complex problems in far fewer steps than classical computers. The technology relies on qubits, or quantum bits, which can sift through vast sets of possibilities using quantum interference to find an answer.

The natural quantum state of a qubit is fragile and can be disrupted by various factors: microscopic defects in hardware, heat, vibration, electromagnetic interference and even cosmic rays (which are everywhere).

Quantum error correction offers a way forward by using redundancy: grouping multiple qubits into a single logical qubit, and regularly performing consistency checks on it. The decoder preserves quantum information by using these consistency checks to identify errors in the logical qubit, so they can be corrected.

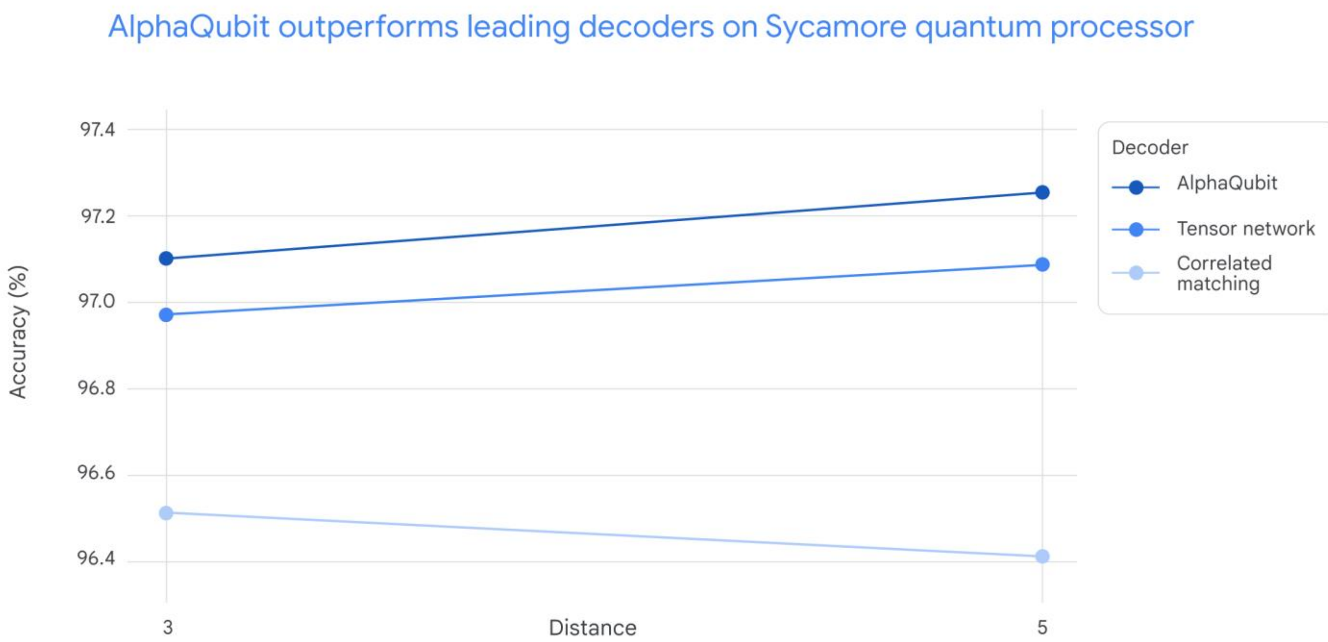
Here, we illustrate how nine physical qubits (small gray circles) in a qubit grid of side length 3 (code distance) form a logical qubit. At each step, 8 more qubits perform consistency checks (square and semicircle areas, blue and magenta when failing and gray otherwise) at each time step which inform the neural network decoder (AlphaQubit). At the end of the experiment, AlphaQubit determines what errors occurred.

# Creating a neural-network contender for decoding

AlphaQubit is a neural-network based decoder drawing on Transformers, a deep learning architecture developed at Google that underpins many of today's large language models. Using the consistency checks as an input, its task is to correctly predict whether the logical qubit — when measured at the end of the experiment — has flipped from how it was prepared.

We began by training our model to decode the data from a set of 49 qubits inside a Sycamore quantum processor, the central computational unit of the quantum computer. To teach AlphaQubit the general decoding problem, we used a quantum simulator to generate hundreds of millions of examples across a variety of settings and error levels. Then we finetuned AlphaQubit for a specific decoding task by giving it thousands of experimental samples from a particular Sycamore processor.

When tested on new Sycamore data, AlphaQubit set a new standard for accuracy when compared with the previous leading decoders. In the largest Sycamore experiments, AlphaQubit makes 6% fewer errors than tensor network methods, which are highly accurate but impractically slow. AlphaQubit also makes 30% fewer errors than correlated matching, an accurate decoder that is fast enough to scale.

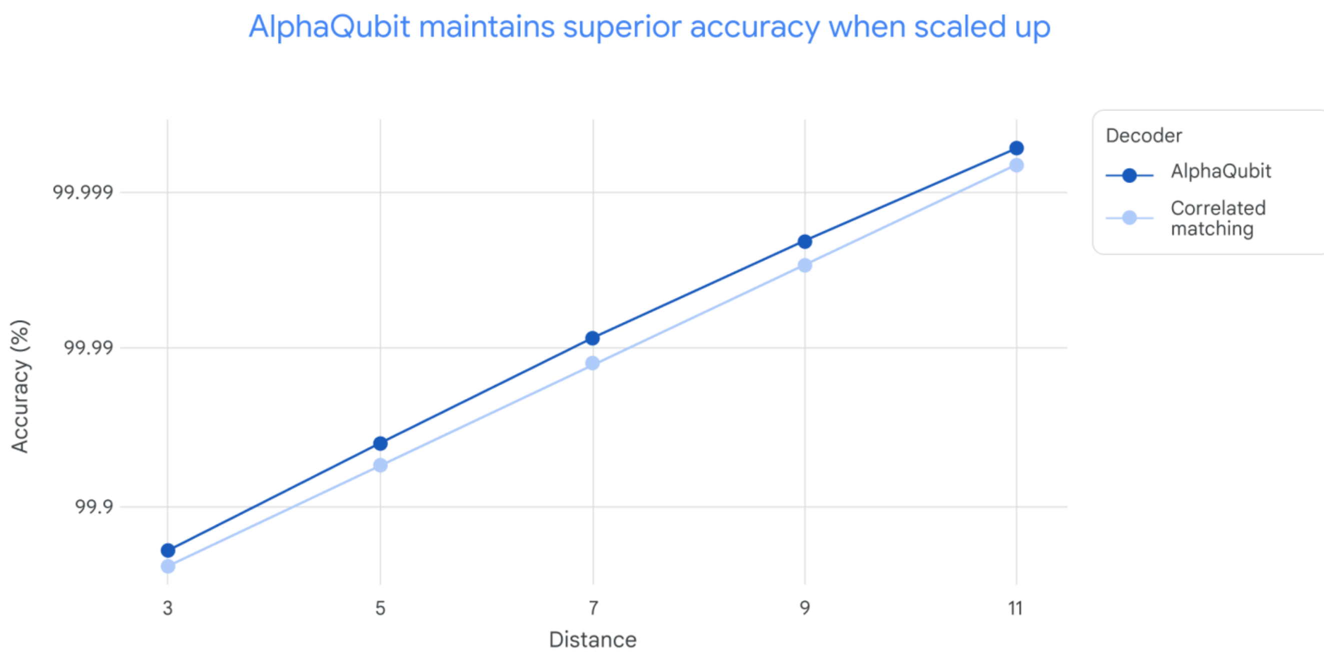



Decoding accuracies for small and large Sycamore experiments (distance 3 = 17 physical qubits, and distance 5 = 49 physical qubits). AlphaQubit is more accurate than the tensor network (TN, a method that is not expected to scale at large experiments) and correlated matching (an accurate decoder with the speed to scale).

# Scaling AlphaQubit for future systems

We expect quantum computers to advance beyond what's available today. To see how AlphaQubit would adapt to larger devices with lower error levels, we trained it using data from simulated quantum systems of up to 241 qubits, as this exceeded what was available on the Sycamore platform.

Again, AlphaQubit outperformed leading algorithmic decoders, suggesting it will also work on mid-sized quantum devices in the future.




Decoding accuracies for different scaling/simulated experiments, from distance 3 (17 qubits) to distance 11 (241 qubits). The Tensor Network decoder does not appear in this graph, as it is too slow to run at large distances. The accuracy of the other two decoders increases when increasing distance (that is, when using more physical qubits). At each distance, AlphaQubit is more accurate than correlated matching.

Our system also demonstrated advanced features like the ability to accept and report confidence levels on inputs and outputs. These information-rich interfaces can help further improve the performance of the quantum processor.

And when we trained AlphaQubit on samples that included up to 25 rounds of error correction, it maintained good performance on simulated experiments of up to 100,000 rounds, showing its ability to generalize to scenarios beyond its training data.

# Moving towards practical quantum computing

AlphaQubit represents a major milestone in using machine learning for quantum error correction. But we still face significant challenges involving speed and scalability.

For example, each consistency check in a fast superconducting quantum processor is measured a million times every second. While AlphaQubit is great at accurately identifying errors, it's still too slow to correct errors in a superconducting processor in real time. As quantum computing grows toward the potentially millions of qubits needed for commercially relevant applications, we'll also need to find more data-efficient ways of training AI-based decoders.

Our teams are combining pioneering advances in machine learning and quantum error correction to overcome these challenges — and pave the way for reliable quantum computers that can tackle some of the world's most complex problems.

# Ask a Techspert: What is quantum computing?

*Editor's Note: Do you ever feel like a fish out of water? Try being a tech novice and talking to an engineer at a place like Google. Ask a Techspert is a new series on the Keyword asking Googler experts to explain complicated technology for the rest of us. This isn't meant to be comprehensive, but just enough to make you sound smart at a dinner party.*

Quantum computing sounds like something out of a sci-fi movie. But it's real, and scientists and engineers are working to make it a practical reality. Google engineers are creating chips the size of a quarter that could revolutionize the computers of tomorrow. But what *is* quantum computing, exactly?

The Keyword's very first Techspert is Marissa Giustina, a research scientist and quantum electronics engineer in our Santa Barbara office. We asked her to explain how this emerging technology actually works.

**What do we need to know about conventional computers when we think about quantum computers?**

At a first glance, "information" seems like an abstract concept. Sure, information can be stored by writing and drawing—humans figured that out a long time ago. Still, there doesn't seem to be anything physically tangible about the process of thinking.

Enter the personal computer. It's a machine—a purely physical object—that manipulates information. So how does it do that, if it's a physical machine and information is abstract? Well, information is actually physical. Computers store and process rich, detailed information by breaking it down. At a low level, a computer represents information as a series of "bits." Each bit can take a value of either [0] or [1], and physically, these bits are tiny electrical switches that can be either open [0] or closed [1]. Emails, photos and videos on YouTube are all represented by long sequences of bits—long rows of tiny electrical switches inside a computer.

The computer "computes" by manipulating those bits, like changing between [0] and [1] (opening or closing a switch), or checking whether two bits have equal or opposite values and setting another bit accordingly. These bit-level manipulations are the basis of even the fanciest computer programs.

**Ones and zeros, like "The Matrix." Got it. So then what is a quantum computer?**

A quantum computer is a machine that stores and manipulates information as quantum bits, or "qubits," instead of the "classical" bits we were talking about before. Quantum bits are good at storing and manipulating a different kind of information than classical bits, since they are governed by rules of quantum mechanics—the same rules that govern the behavior of atoms and molecules.

**What's the difference between a bit and a qubit?**

This is where it gets more complicated. Remember that a classical bit is just a switch: it has only two possible configurations: [open] or [closed]. A qubit's configuration has a lot more possibilities. Physicists often think of a qubit like a little globe, with [0] at the north pole and [1] at the south pole, and the qubit's configuration is represented by a point on the globe. In manipulating the qubit, we can send any point on the globe to any other point on the globe.

At first, it sounds like a qubit can hold way more information than a regular bit. But there's a catch: the "rules" of quantum mechanics restrict what kinds of information we can get out of a qubit. If we want to know the configuration of a classical bit, we just look at it, and we see that the switch is either open [0] or closed [1]. If we want to know the configuration of a qubit, we measure it, but the only possible measurement outcomes are [0] (north pole) or [1] (south pole). A qubit that was situated on the equator will measure as [0] 50 percent of the time and [1] the other 50 percent of the time. That means we have to repeat measurements many times in order to learn about a qubit's actual configuration.

Researcher Marissa Giustina (right) in the Google AI Quantum hardware lab shares quantum computing hardware with Google executives. On the left, you can see the coldest part of a cryostat and some quantum hardware mounted to the bottom.

**So if qubits are so tricky to measure, how can you build a quantum computer?**

Well, you're right—it's complicated! My main focus at Google, together with my teammates, is to figure out how to build a quantum computer and how we can use it. Years of research have given us a pretty good idea of how to build and control a few quantum bits, but the process of scaling up to a full quantum processor is not just "copy-paste." We're also continuing to investigate possible uses of quantum computers, where there's a lot that's unknown. It's wrong to think of a quantum computer as a more powerful version of your regular computer. Instead, each is a machine that's good at certain—and different—kinds of tasks. If you're going to your local grocery store, you'd take a car or walk, but you wouldn't take a plane or a spaceship.

**What does a quantum computer look like?**

In our hardware at Google, the qubits are resonant electrical circuits made of patterned aluminum on a silicon chip. In our qubits, electricity sloshes around the circuit at a lower or higher energy to encode the quantum version of [0] and [1]. We use aluminum because at very low temperatures aluminum becomes superconducting, which means it experiences no electrical loss. By "very low temperatures" I mean that we operate our quantum processors in a special refrigerator called a cryostat, which cools the chips to below 50 millikelvin—significantly colder than outer space!

When you see pictures of "a quantum computer," usually you notice the cryostat—which is bigger than a person. But that's just the shell, providing the proper environment for the processor to function. The quantum processor itself is a silicon chip installed in the cryostat, and is closer to the size of a coin. The qubits are small, roughly 0.1 mm across, but not that small—you can see them with the naked eye (though it's easier with a magnifying glass or microscope).

**Do you know what we would use a quantum computer for?**

As I mentioned, a quantum computer is a novel kind of computing machine—not a speedier or beefier version of your laptop. However, quantum computers, with their fundamentally different way of encoding and manipulating information, promise to be good at some problems that would choke regular computers. One example is the simulation of chemical reactions.

Suppose a chemist wants to develop a material—for example a better fertilizer, an anti-corrosion coating, or an efficient solar cell. Even if the chemist knows the structure of a new molecule they're developing, they won't know how that molecule behaves in the real world until they make it and test it. This makes materials research laborious and expensive. It would be much more efficient if researchers could simulate the behavior of a new molecule before synthesizing it in the lab. However, every atom in a molecule is affected by every other atom, which means that each time you add an atom to a molecule, there are twice as many parameters to include in the simulation. As a result, chemistry simulation becomes impossible for a classical computer, even for relatively small molecules. The quantum computer, in contrast, is based in the same physics that governs the molecule's behavior. I'm optimistic that quantum computers could change the way we do research on materials.

# Ask a Techspert: What's the difference between a CPU, GPU and TPU?

Our latest TPU, Trillium, is now available in preview. Learn more about our latest TPU, Trillium, from a Google expert — as well as what a TPU, CPU and GPU are and what makes them all different.

Back at I/O in May, we announced Trillium, the sixth generation of our very own custom-designed chip known as the Tensor Processing Unit, or TPU — and today, we announced that it's now available to Google Cloud Customers in preview. TPUs are what power the AI that makes your Google devices and apps as helpful as possible, and Trillium is the most powerful and sustainable TPU yet.

But what exactly is a TPU? And what makes Trillium "custom"? To really understand what makes Trillium so special, it's important to learn not only about TPUs, but also other types of compute processors — CPUs and GPUs — as well as what makes them different. As a product manager who works on AI infrastructure at Google Cloud, Chelsie Czop knows exactly how to break it all down. "I work across multiple teams to make sure our platforms are as efficient as possible for our customers who are building AI products," she says. And what makes a lot of Google's AI products possible, Chelsie says, are Google's TPUs.

**Let's start with the basics! What are CPUs, GPUs and TPUs?**

These are all chips that work as processors for compute tasks. Think of your brain as a computer that can do things like reading a book or doing a math problem. Each of those activities is similar to a compute task. So if you use your phone to take a picture, send a text or open an application, your phone's brain, or processor, is doing those compute tasks.

**What do the different acronyms stand for?**

Even though CPUs, GPUs and TPUs are all processors, they're progressively more specialized. CPU stands for Central Processing Unit. These are general-purpose chips that can handle a diverse range of tasks. Similar to your brain, some tasks may take longer if the CPU isn't specialized in that area.

Then there's the GPU, or Graphics Processing Unit. GPUs have become the workhorse of accelerated compute tasks, from graphic rendering to AI workloads. They're what's known as a type of ASIC, or application-specific integrated circuit. Integrated circuits are generally made using silicon, so you might hear people refer to chips as "silicon" — they're the same thing (and yes, that's where the term "Silicon Valley" comes from!). In short, ASICs are designed for a single, specific purpose.

The TPU, or Tensor Processing Unit, is Google's own ASIC. We designed TPUs from the ground up to run AI-based compute tasks, making them even more specialized than CPUs and GPUs. TPUs have been at the heart of some of Google's most popular AI services, including Search, YouTube and DeepMind's large language models.

**Got it, so all of these chips are what make our devices work. Where would I find CPUs, GPUs and TPUs?**

CPUs and GPUs are inside very familiar items you probably use every day: You'll find CPUs in just about every smartphone, and they're in personal computing devices like laptops, too. A GPU you'll find in high-end gaming systems or some desktop devices. TPUs you'll only find in Google data centers: warehouse-style buildings full of racks and racks of TPUs, humming along 24/7 to keep Google's, and our Cloud customers', AI services running worldwide.

**What made Google start thinking about creating TPUs?**

CPUs were invented in the late 1950s, and GPUs came around in the late '90s. And then here at Google, we started thinking about TPUs about 10 years ago. Our speech recognition services were getting much better in quality, and we realized that if every user started "talking" to Google for just three minutes a day, we would need to *double* the number of computers in our data centers. We knew we needed something that was a lot more efficient than off-the-shelf hardware that was available at the time — and we knew we were going to need a lot more processing power out of each chip. So, we built our own!

**And that "T" stands for Tensor, right? Why?**

Yep — a "tensor" is the generic name for the data structures used for machine learning. Basically, there's a bunch of math happening under the hood to make AI tasks possible. With our latest TPU, Trillium, we've increased the amount of calculations that can happen: Trillium has 4.7x peak compute performance per chip compared to the prior generation, TPU v5e.

**What does that mean, exactly?**

It basically means that Trillium is able to work on all the calculations required to run that complex math 4.7 times faster than the last version. Not only does Trillium work faster, it can also handle larger, more complicated workloads.

**Is there anything else that makes it an improvement over our last-gen TPU?**

Another thing that's better about Trillium is that it's our most sustainable TPU yet — in fact, it's 67% more energy-efficient than our last TPU. As the demand for AI continues to soar, the industry needs to scale infrastructure sustainably. Trillium essentially uses less power to do the same work.

**Now that customers are starting to use it, what kind of impact do you think Trillium will have?**

We're already seeing some pretty incredible developments powered by Trillium! We have customers using it in technologies that analyze RNA for various diseases, turn written text into videos at incredible speeds and more. And that's just from our very initial round of users — now that Trillium's in preview, we can't wait to see what people can do with it.

# Ask a Techspert: What is on-device processing?

Learn about how on-device processing actually works, plus how it powers features across Google products, like Pixel, Nest and more.

Every time a new Pixel phone comes out, you might hear that "on-device processing" makes its cool new features possible. Just take a look at the new Pixel 9 phones — things like Pixel Studio and Call Notes run "on device." And it's not just phones: Nest cameras, Pixel smartwatches and Fitbit devices also use this whole "on-device processing" thing. Given the devices that use it and the features it's powering, it sounds pretty important.

It's safe to assume that the, er, processing, is happening on the, uh…well, the device. But to get a better understanding of what that means, we talked to Trystan Upstill, who has been at Google for nearly 20 years working on engineering teams across Android, Google News and Search.

**You were on a team that helped develop some of the exciting features that shipped with our new Pixel devices — can you tell me a little about what you worked on?**

Most recently, I worked within Android where I led a team that focuses on melding Google's various technology stack into an amazing experience that's meaningful to the user. Then figuring out how to build it and ship it.

**Since we're improving technologies and introducing new ones quite often, it seems like that would be a never-ending job.**

Exactly! Within recent years, there's been this explosion in generative AI capabilities. At first when we started thinking about running large language models on devices, we thought it was kind of a joke — like, "Sure we can do that, but maybe by 2026." But then we began scoping it out, and the technology performance evolved so quickly that we were able to launch features using Gemini Nano, our on-device model, on Pixel 8 Pro in December 2023.

**That's what I want to know more about: "on-device processing." Let's break it down and start with what exactly "processing" means.**

The main processor, or system-on-a-chip (SoC), in your devices, has a number of what are called Processing Units designed specifically to handle the tasks you want to do with that device. That's why you'll see the chip (like the Tensor chip found in Pixels) referred to as a "system-on-a-chip: There's not just one processor, but several processing units, memory, interfaces and much more, all together on one piece of silicon.

Let's use Pixel smartphones as an example: The processing units include a Central Processing Unit, or CPU, as the main "engine" of sorts; a Graphics Processing Unit, or GPU, which renders visuals; and now today we have a Tensor Processing Unit, or TPU, specially designed by Google to run AI/ML workloads on a device. These all work together to help your phone get things done — aka, processing.

For example, when you take photos, you're often using all elements of your phone's processing power to good effect. The CPU will be busy running core tasks that control what the phone is doing, the GPU will be helping render what the lens is seeing and, on a premium Android device like a Pixel, there's also a lot of work happening on the TPU to process what the optical lens sees to make your photos look awesome.

**Got it. "On-device" processing implies there's off-device. Where is "off-device processing" happening, exactly?**

Off-device processing happens in the cloud. Your device connects to the internet and sends your request to servers elsewhere, which perform the task, and then send the output back to your phone. So if we wanted to take that process and make it happen on device, we'd take the large machine learning model that powered that task in the cloud and make it smaller and more efficient so it can run on your device's operating system and hardware.

**What hardware makes that possible?**

New, more powerful chipsets. For example, with the Pixel 9 Pro, that's happening thanks to our SoC called Tensor G4. Tensor G4 enables these phones to run models like Gemini Nano — it's able to handle these high-performance computations.

**So basically, Tensor is designed specifically to run Google AI, which is *also* what powers a lot of Pixel's new gen AI capabilities.**

Right! And the generative AI features are definitely part of it, but there are lots of other things on-device processing makes possible, too. Rendering video, playing games, HDR photo editing, language translation — most everything you do with your phone. These are all happening on your phone, not being sent up to a server for processing.

TalkBack with Gemini, which analyzes images and reads descriptions out loud to blind or low-vision users, is an example of on-device processing that makes use of Tensor, Pixel's system on a chip.

The computation your phone can do today is pretty incredible. Today's smartphones are thousands of times faster than early high-performance computers, even those that were the size of rooms. Back in the day, those high-performance computers were the state of the art in terms of data analysis, image processing, anomaly detection and early AI research. Now we can do this all on device, and it opens up all sorts of neat opportunities to build helpful features that use this processing capability.

**Is on-device processing better than off-device?**

Not necessarily. If you were to use Search entirely on-device, that would be really slow or really limited or both, because when you're searching the web, you're sort of looking for a needle in a haystack. To fit the entire web index on your phone would be too much! Instead, when you use Search, you're tapping into the cloud and our data centers to access trillions of web pages to find what you're looking for.

But if you want to perform a more specific task, then on-device processing is really useful. For starters, there's latency — if something's being processed directly on the device, you may get the result faster. Then there's also the fact that features that are fully on device work without an internet connection, meaning better availability and reliability.

Finally, given the AI chip is in your pocket rather than being served through a cloud backend, it's free for apps to leverage the LLM capabilities.

All this said, there are distinct advantages to both: Cloud has more powerful models and can house lots of important data. Lots of your data, like photos, videos and more, sits in the cloud today. It also helps support actions like searching massive databases, like Drive, Gmail and Google Photos.

**I'm already pretty impressed with what my Pixel can do today, but from what you're saying, I'd imagine it's only going to get better.**

Yes, the models we're using to do these complex tasks on Android devices are getting more capable. And of course it's not just about better models and better technology: We also put a lot of work and research into thinking about what's actually going to benefit people. We don't want to just introduce products because the on-device processing can handle it; we want to make sure it's something that people want to use on their phones in their everyday lives.

# How we built AlphaFold 3 to predict the structure and interaction of all of life's molecules

Since its launch in 2020, more than 2 million researchers have used Google DeepMind's AlphaFold 2 model for protein predictions in their work on vaccine development, cancer treatments and more — helping solve a problem that researchers had been working on for over 50 years. After helping scientists predict hundreds of millions of structures, it would've been easy for the team to rest on their laurels.

Instead, they got started on AlphaFold 3. This newer model, which the teams at Google DeepMind and Isomorphic Labs launched in May, builds on our previous models by predicting not just protein folding structures, but predicting the structure and interactions of all of life's molecules, including DNA, RNA and ligands (small molecules that bind to proteins).

"With AlphaFold 2, we made enormous progress on this decades-old open problem of protein folding, but if you look at recent high-impact research, researchers are moving beyond that," says Jonas Adler, research scientist at Google DeepMind. "Their conclusions were often about something more detailed, like binding of small molecules, or RNA, which AlphaFold 2 couldn't do. Things had moved on experimentally and so to get to the frontier of where things are today in biology and chemistry, we really needed to be able to cover all biomolecules."

"Everything" includes ligands, which make up about half of all drugs. "At Isomorphic Labs, we see the tremendous potential of AlphaFold 3 for rational drug design, and we're already using it in our day-to-day work," says Adrian Stecula, research leader at Isomorphic Labs. "Investigating the binding of novel small molecules to novel drug targets, answering questions like, 'How do proteins engage with DNA and RNA?,' looking into the effects of chemical modifications on protein structure — the new model unlocks all of those capabilities."

Adding in these additional molecular types introduced an order of magnitude more possible combinations. "Proteins are very ordered. For example, there are only 20 standard amino acids," Jonas says. "Whereas for small molecules, there's an infinitely large space — they can do basically anything. They're extremely diverse."

2:45

AlphaFold Server allows researchers to access AlphaFold 3.

That meant making a database with all the capabilities would have been impossible. Instead, we've released AlphaFold Server, a free tool that lets scientists plug in their own sequences that AlphaFold can then generate molecular complexes for. Since launching in May, researchers have already used it to generate over 1 million structures.

"It's like Google Maps for molecular complexes," says Lindsay Willmore, research engineer at Google DeepMind. "Any user who doesn't know how to code at all can just copy and paste the sequences of their proteins, DNA, RNA or the name of their small molecule, press a button and wait a few minutes. Their structure and the confidence metrics will come out so that they're able to look at and evaluate their prediction."

In order to get AlphaFold 3 to work with this much wider range of biomolecules, the team vastly expanded the data that the newer model was trained on to include DNA, RNA, small molecules and more. "We were able to say, 'Let's just train on everything that exists in this dataset that helped us so much with proteins and let's see how far we can get,'" Lindsay says. "And it turns out we can get pretty far."

Another major change in AlphaFold 3 is a shift in architecture for the final part of the model that generates the structure. Where AlphaFold 2 used a complex custom geometry-based module, AlphaFold 3 uses a generative model that's based on diffusion — similar to our other cutting-edge image generation models, like Imagen — which greatly simplified how the model handles all the new molecule types.

That shift led to a new issue, though: Since so-called "disordered regions" of proteins weren't included in the training data, the diffusion model would try to create an inaccurate "ordered" structure with a defined spiral shape, instead of predicting disordered regions.

So the team turned to AlphaFold 2, which is already extremely good at predicting which interactions would be disordered — which look like a pile of chaotic spaghetti — and which ones were not. "We were able to use those predicted structures from AlphaFold 2 as distillation training for AlphaFold 3, so that AlphaFold 3 could learn to predict disorder," Lindsay says.

"We have a saying: 'Trust the fusilli, reject the spaghetti,'" adds Jonas.

An example of a prediction from AlphaFold 3 with ordered “fusilli” regions in blue and disordered “spaghetti” regions in orange. The colors represent the model’s confidence of predicted accuracy.

The team is looking forward to seeing how researchers will use AlphaFold 3 to advance fields like genomics research, drug design and more.

“It's incredible to see how much progress we made,” Jonas says. “What used to be very hard has become very easy. What used to be impossible has become possible — and while there are still very hard problems here to solve, we’re excited about the potential for AlphaFold 3 to help solve them.”

# 7 pieces of AI news we announced in October

Here’s a recap of seven AI updates from October, including Google Maps’ biggest AI update ever, tips for getting started with NotebookLM, and more ways people can ask questions, search for information and get an AI Overview.

For more than 20 years, we’ve invested in machine learning and AI research, tools and infrastructure to build products that make everyday life better for more people. Teams across Google are working on ways to unlock AI’s benefits in fields as wide-ranging as healthcare, crisis response and education. To keep you posted on our progress, we're doing a regular roundup of Google's most recent AI news across products, research, and more. Here’s a look back at just some of our AI announcements from October.

**We gave Google Maps its biggest AI update ever**. For nearly 20 years, Google Maps has transformed the way we navigate and explore. Today, thanks to its biggest AI update ever, Maps is getting even more helpful. Now, you’ll be able to ask Maps more complex queries, like “things to do with friends” to get answers curated with Gemini, quick answers to questions about a place — in addition to helpful review summaries when you don’t have time to read through each one. Plus, to help take the stress out of driving, we added new features to help you with every step of your drive — like the ability to explore along your route before you head out. The latest AI update in Maps means that whether you’re traveling across town or around the world, you can get the most up-to-date information possible, when you need it. Make sure to also check out the new updates we're making to Waze, Google Earth and our developer products.

**An expert Googler shared tips for using NotebookLM**. Steven Johnson is a popular author who often writes on the intersection of science, technology and the human experience. In his role at Google, he's also helped launch NotebookLM — the product gaining a lot of attention for helping people synthesize large volumes of information and quickly understand complex ideas. You can upload PDFs, Google Docs, websites, YouTube videos and more to NotebookLM to glean new insights and get deeper dives on new topics, which Steven details (along with other helpful tips) in his post.

**Google Shopping rolled out new AI to help you pick the right product for you**. The new Google Shopping — which is available in the U.S. to start — uses AI to help take the guesswork out of finding the right products. For example, when researching a product, an AI-generated brief will give you more details about the most important things to know before buying. And, as you browse the results, you will see AI-generated briefs about what to consider before making a purchase, plus the products that may (or may not) be a good fit for your needs. The Google Shopping home page will also now feature a feed of product recommendations, videos and a personalized deal page — so you can find products you want at the lowest price.

**Search got a major AI update, expanding the types of questions people can ask**. AI has already helped us reimagine Google Search, so people can ask questions in new ways — whether typing a query, searching with a camera or even humming a tune. In October, we added even more updates, including helping people to identify songs in Circle to Search, shop for what they see and search with video. For example, when people visit the aquarium they can open Lens in the Google app and hold down the shutter button to record the fish while asking aloud, "why are they swimming together?" — AI will make sense of the video and produce an AI Overview, along with helpful resources from across the web. The world is in constant motion, and Search can now help people understand it, as it is.

**We announced that all Chromebooks, including two new models, will come with built-in AI**. All Chromebooks will now come with the Gemini app, and Chromebook Plus laptops include new Google AI productivity tools like Live Translate, Help me write, the Recorder app and Welcome Recap, which helps people pick right back up where they left off when logging into a Chromebook. The new Samsung Galaxy Chromebook Plus is the thinnest and lightest Chromebook Plus, and features a new Quick Insert key to access AI features; and Lenovo's latest Chromebook Duet 11" is compact, durable and converts between tablet and desktop mode in seconds.

**We announced new grants for AI training at our annual Google Public Sector Summit**. AI is poised to help the U.S. public sector bolster cybersecurity, enhance data analysis and find new ways to improve citizen services. To train the U.S. government workforce in responsible AI, Google.org announced $15 million in new training grants for two leading public sector organizations, the Partnership for Public Service and InnovateUS. Those grants will help cultivate AI leadership and talent within the federal government, and increase access to AI training for more than 100,000 public sector workers across more than 30 states.

**The Nobel Prize was awarded to three Google AI pioneers** for advancing the science of AI in a way that benefits all people. Google alum Geoff Hinton received the Nobel Prize in Physics for his pioneering work in neural networks. And Google DeepMind's Demis Hassabis and John Jumper received the Nobel Prize in Chemistry for their groundbreaking work with AlphaFold 2, which predicted the structures for nearly all proteins known to science. It's been used by more than 2 million researchers around the world, accelerating scientific discovery in important areas like malaria vaccines, cancer treatments, and more.

# 9 ways AI is advancing science

We're sharing a recap of some of the biggest scientific breakthroughs in recent years brought about by AI.

We're living in a time when applied science, human ingenuity and new technologies are offering deep insights into some of humanity's biggest (and oldest) questions. While we often think of scientific progress as fast and unrelenting, for many decades, progress has actually slowed. While the scientific community continues to debate the cause of this slowdown, much of today's technology — from jets to manufacturing processes — is not significantly different than half a century ago.

But in just the past few years, breakthroughs in formerly nascent fields like artificial intelligence and quantum computing have dramatically accelerated the pace of scientific discovery. And from healthcare advances to finding plastic-eating enzymes, we're already benefiting from it.

These breakthroughs are built on decades of collaboration between researchers, technologists, policymakers, civil organizations and many people from across society. And they offer a blueprint for how applying AI to science can dramatically improve human life.

It's with this in mind that today The Royal Society in partnership with Google DeepMind is cohosting the first AI for Science Forum. This event in London brings together the scientific community, policymakers, and industry leaders to look at the transformative potential of AI to accelerate science and the role of public-private partnerships in innovation.

To explore how we got here and where we can go next, here's a look at nine recent moments that have set the stage for so much of the scientific progress on the horizon:

# 1. Cracking the 50-year "grand challenge" of protein structure prediction

Experts have described demystifying protein folding as a "grand challenge" for decades. In 2022, Google DeepMind shared the predicted structures of 200 million proteins from their AlphaFold 2 model. Previously, determining the 3D structure of a single protein typically took a year or more — AlphaFold can predict these shapes with remarkable accuracy in minutes. By releasing the protein structure predictions in a free database, this has enabled scientists around the world to accelerate progress in areas like developing new medicines, fighting antibiotic resistance and tackling plastic pollution. As a next step, the AlphaFold 3 model builds on AlphaFold 2 to predict the structure and interaction of all of life's molecules.

# 2. Showing the human brain in unprecedented detail, to support health research

Few things have held more mystery throughout time than the human brain. Developed over 10 years of connectomics research, Google partnered with others, including the the Lichtman Lab at Harvard, to map a tiny piece of the human brain to a level of detail never previously achieved. This project, released in 2024, revealed never-before-seen structures within the human brain. And the full dataset, including AI-generated annotations for each cell, has been made publicly available to help accelerate research.

# 3. Saving lives with accurate flood forecasting

When Google's flood forecasting project began in 2018, many believed it was impossible to accurately deliver flood forecasting at scale, given the scarcity of data. But researchers were able to develop an AI model that achieves reliability in predicting extreme riverine events in ungauged watersheds at up to a five-day lead time with reliability matching or exceeding that of nowcasts (zero-day lead time). In 2024, Google Research expanded this coverage to 100 countries and 700 million people worldwide — and improved the AI model so it offers the same accuracy at a seven-day lead time as the previous model had at five.

# 4. Spotting wildfires earlier to help firefighters stop them faster

Wildfires are increasingly upending communities around the world due to hotter and drier climates. In 2024, Google Research partnered with the U.S. Forest Service to develop FireSat, an AI model and new global satellite constellation designed specifically to detect and track wildfires the size of a classroom by providing higher-resolution imagery within

20 minutes. This will allow fire authorities to respond more quickly, potentially saving lives, property and natural resources.

# 5. Predicting weather faster and with more accuracy

In 2023, Google DeepMind launched and open sourced the model code for GraphCast, a machine learning research model that predicts weather conditions up to 10 days in advance more accurately and much faster than the industry gold-standard weather simulation system (HRES). GraphCast can also predict the tracks of cyclones (and associated risks like flooding) with greater accuracy, and accurately predicted Hurricane Lee would hit Nova Scotia three days before traditional models.

# 6. Advancing the frontier of mathematical reasoning

AI has always struggled with complex math due to a lack of data and reasoning skills. Then, in 2024, Google DeepMind announced AlphaGeometry, an AI system that solved complex geometry problems at a level approaching a human Olympiad gold-medalist — a breakthrough in AI performance and the pursuit of more advanced general AI systems. The subsequent Gemini-trained model, AlphaGeometry 2, was then combined with a new model AlphaProof, and together they solved 83% of all historical International Mathematical Olympiad (IMO) geometry problems from the past 25 years. In demonstrating AI's growing ability to reason, and potentially solve problems beyond current human abilities, this moved us closer to systems that can discover and verify new knowledge.

# 7. Using quantum computing to accurately predict chemical reactivity and kinetics

Google researchers worked with UC Berkeley and Columbia University to perform the largest chemistry simulations to date on a quantum computer. The results, published in 2022, were not only competitive with classical methods, but they also did not require the burdensome error mitigation typically associated with quantum computing. The ability to conduct these simulations will offer even more accurate predictions of chemical reactivity and kinetics, which is a precursor for applying chemistry in new ways to help solve real-world challenges.

# 8. Accelerating materials science and the potential for more sustainable solar cells, batteries and superconductors

In 2023, Google DeepMind announced Graph Networks for Materials Exploration (GNoME), a new AI tool that has already discovered 380,000 materials that are stable at low temperatures, according to simulations. At a time when our world is looking for new approaches to energy, processing power and materials science, this work could pave the way to better solar cells, batteries and potential superconductors. Plus, to help this technology benefit everyone, Google DeepMind made GNoME's most stable predictions available via the Materials Project on their open database.

## 9. Taking a meaningful step toward nuclear fusion — and abundant clean energy

As the old joke goes, "Fusion is the energy of the future — and it always will be." Controlling and using the energy that fuels stars — including our own sun — has been beyond the realm of science. Then in 2022, Google DeepMind announced that it developed AI that can control the plasma inside a nuclear fusion reactor autonomously. By collaborating with the Swiss Plasma Center at EPFL, Google DeepMind built the first system capable of autonomously stabilizing and shaping the plasma within an operational fusion reactor, taking a critical step toward stable fusion and abundant clean energy for everyone.

# A new era of discovery

At today's AI for Science Forum, James Manyika laid out how AI is accelerating scientific breakthroughs and can help to address the world's most pressing challenges.

*Editor's note: Today in London, Google DeepMind and the Royal Society co-hosted the inaugural AI for Science Forum, which brought together Nobel laureates, the scientific community, policymakers, and industry leaders to explore the transformative potential of AI to drive scientific breakthroughs, address the world's most pressing challenges, and lead to a new era of discovery.*

*Google's Senior Vice President for Research, Technology and Society, James Manyika, delivered the opening address; what follows is a transcript of his remarks, as prepared for delivery.*

AI's impact in science has been in the headlines lately, but the potential of AI to advance science has long been a motivating force for many in the field, dating back to early AI researchers, such as Alan Turing and Christopher Longuet-Higgins, and to many in recent decades including my colleagues at Google DeepMind and Google Research.

The excitement around AI and science is not because of a belief that AI is a replacement for scientists, but because many confounding problems in science benefit from the use of computational techniques — thus making AI a powerful tool to assist scientists.

We saw early signs of that assistive potential with Hodgkin and Huxley's use of computational approaches to describe how nerve impulses travel along neurons, work that would win them the Nobel Prize in 1963.

Fast forward to my colleagues Demis Hassabis, John Jumper and the AlphaFold team whose work using AI recently won the Nobel Prize in Chemistry, solving the "protein-folding problem" posed by Nobel laureate Christian Anfinsen in the 1970s.

# So how is AI helping advance science?

I'll start with speed. In some areas of science, increasingly capable AI is making it possible for us to condense hundreds or even thousands of years of research into a few years, months, or even days.

AI is also helping expand the scope of research – enabling scientists to look at many things at once — and in new ways — rather than one by one.

AI advances — along with access to insights from using it — are enabling many more people to participate in research, so that we can further accelerate scientific discovery.

# AI is enabling landmark progress in multiple scientific disciplines

Let me share briefly a few examples of how AI is enabling landmark advances, starting with AlphaFold:

With AlphaFold, over the course of a year my colleagues were able to predict the structure of nearly every protein known to science — over 200 million of them. And with Alphafold 3, they have extended beyond proteins to all of life's bio-molectures including DNA, RNA and ligands.

To date, AlphaFold has been used by more than 2M researchers in more than 190 countries, working on problems ranging from neglected diseases to drug-resistant bacteria.

AlphaMissense, which builds on AlphaFold, enabled my colleagues to categorize almost 90% of 71M possible missense variants — single letter substitutions in DNA — as likely pathogenic or likely benign. By contrast, only 0.1% have been confirmed by human experts, albeit in more detail.

When the human genome was initially sequenced — an incredible achievement — it was based on a single genomic assembly.

Last year, my colleagues in Google Research, using AI tools and working with a consortium of academic collaborators, released the first draft reference human pangenome.

This was based on 47 genomic assemblies, thus better representing human genetic diversity.

In neuroscience, a 10-year collaboration between my colleagues in Google Research, the Max Planck Institute, and the Lichtman Lab at Harvard, recently produced a nano-scale mapping of a piece of the human brain — that is a level of detail never previously achieved.

This project revealed never-before-seen structures in the human brain that may change our understanding of how the human brain works. This will perhaps lead us to new approaches to understanding and tackling neurological diseases like Alzheimer's and others. The full mapping has been made publicly available for researchers to build on

Beyond the life sciences, we're seeing progress in other domains.

In a landmark achievement for climate modeling, we combined machine learning with a traditional, physics-based approach to build NeuralGCM.

This allows us to simulate the atmosphere more accurately and efficiently — NeuralGCM can simulate over 70,000 days of the atmosphere in the time it would take a state-of-the-art, physics-based model to simulate only 19 days.

There are other similar breakthroughs such as the work by my colleagues at Google DeepMind on GraphCast, a state-of-the-art AI model that predicts weather conditions up to 10 days in advance more accurately and much faster than the industry gold-standard weather simulation system.

Our Quantum AI team is making progress on questions that previously were the realm of science fiction, like studying the characteristics of traversable wormholes.

This opens up new possibilities for testing quantum gravity theories originally posed with the Einstein-Rosen bridge almost ninety years ago.

In fact, Quantum is an area where we're beginning to see promising bidirectional reinforcement between AI and science.

In one direction, AI is advancing our progress in quantum computing — in the other, quantum is helping advance research in AI.

There are many other such examples that we are working on in material science, fusion, mathematics and more – all of these, in collaboration with many academic scientists.

# Scientific advances enabled by AI are having real world impact

Beyond such breakthroughs, AI is also advancing science in ways that are already providing tangible benefits for real people in areas like climate and healthcare.

Let me start with an example from climate adaptation. Flood forecasting is a more frequent and urgent problem due to climate change. Now, advances in AI have enabled us to fill in large gaps in data to predict riverine flooding up to 7 days in advance with the same accuracy as nowcasts. After an initial pilot in Bangladesh, our early-warning platform — Flood Hub — now covers over 100 countries and 700 million people.

And for an example in climate mitigation, consider the following: the formation of contrails has long been a known driver of emissions in aviation — accounting for as much as 35% of aviation's global warming impact.

My colleagues in Google Research developed an AI model that predicts where contrails are likely to form, and in partnership with American Airlines, tested it on 70 flights. We measured the impact and found a 54% reduction in emissions.

Similarly, AI offers much promise for disease detection. For example, eight years ago, Google researchers found that AI could help accurately interpret retinal scans to detect diabetic retinopathy, a preventable cause of blindness that affects roughly 100 million people.

We developed a screening tool that has been used in more than 600,000 screenings worldwide. And new partnerships in Thailand and India will enable 6 million screenings over the next decade.

# The Road Ahead

We have been implementing other examples including in tuberculosis, colorectal cancer, breast cancer and maternal health.

Despite the progress, this is just the beginning. There’s so much still to do.

I see three key areas to focus on to fully realize AI’s potential to help advance science and bring tangible societal benefits:

First, we need to continue to make progress on AI’s current limitations and shortcomings — and to increase AI’s capabilities to be able to assist in developing novel scientific concepts, theories, experiments and more.

Second, we need a sustained commitment to the scientific method and to responsible approaches to using AI to advance science.

We need scientists, ethicists and safety experts — like many in this room — working together to address the risks most particular to science, like viruses and bioweapons, as well as challenges like bias in data sets, privacy preservation, and environmental impacts.

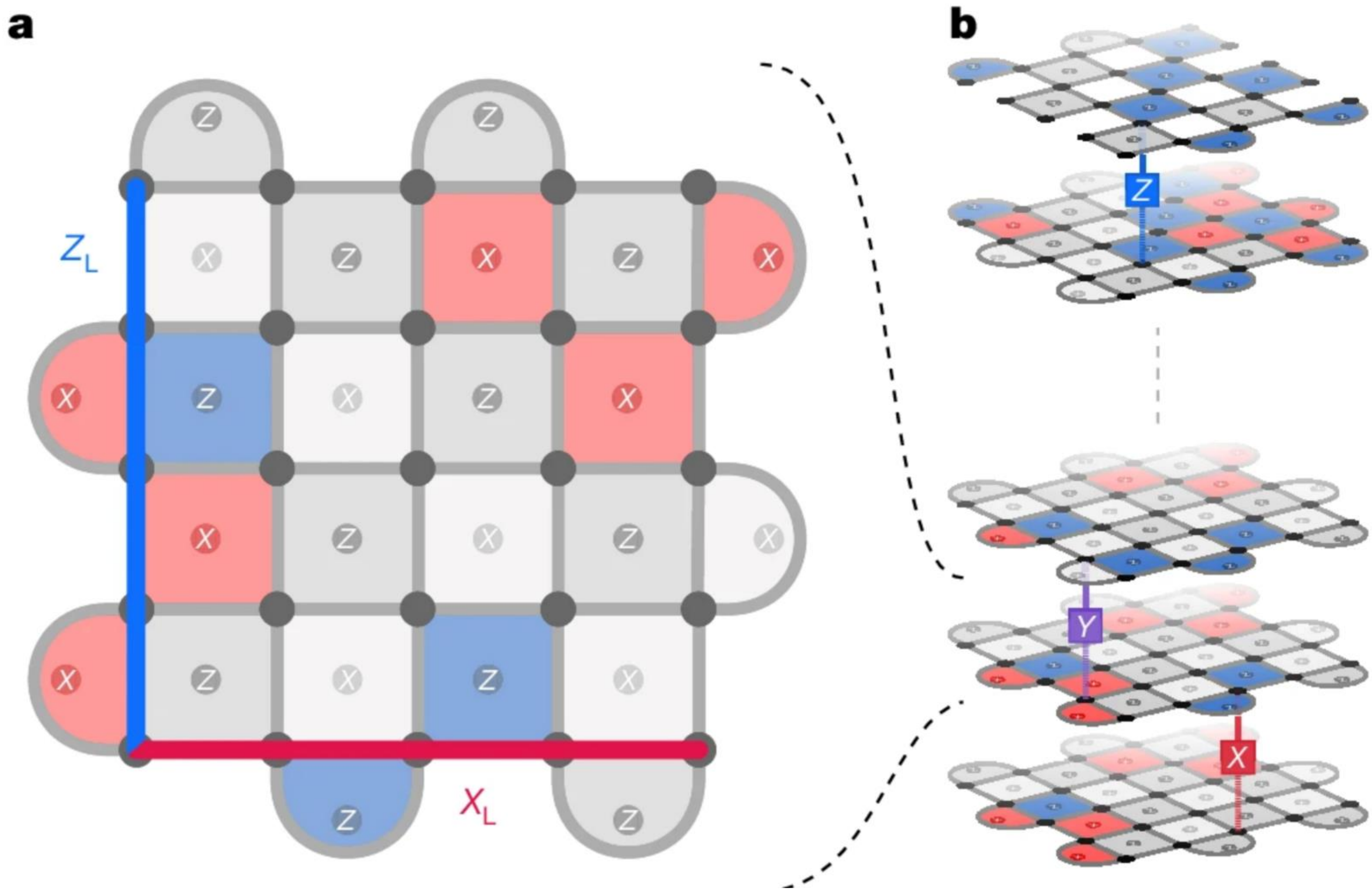
Third, we need to prioritize making AI-enabled research, tools and resources more accessible to more scientists in more places — and to make sure the progress we make benefits people everywhere.

I am excited about what lies ahead in this new era of discovery.

There is so much we can do together to build tools that help advance science to benefit everyone.

And there is so much we can do to enable the amazing scientists here and elsewhere in their work — we'll hear from some of them today.

# AI Power For Quantum Errors: Google Develops AlphaQubit to Identify, Correct Quantum Errors




**Insider Brief**

- Google researchers introduced AlphaQubit, an AI-powered decoder that improves quantum error correction, reducing errors by 6% compared to tensor networks and 30% compared to correlated matching.

- AlphaQubit's two-stage training — pretraining on synthetic data and finetuning with experimental data — enables it to adapt to complex real-world noise, including cross-talk and leakage, showcasing machine learning's potential in quantum computing.
- While AlphaQubit excels in accuracy, challenges remain in achieving real-time speed and scalability, highlighting the need for further optimization to support fault-tolerant quantum systems.

Researchers from Google Quantum AI and DeepMind have developed AlphaQubit, a machine-learning decoder that surpasses existing methods in identifying and correcting quantum computing errors. This advance, outlined in Nature and detailed in a company blog post, could help make quantum computers reliable enough to solve complex problems currently beyond the reach of conventional systems.

AlphaQubit, a neural network, processes error information from quantum processors to improve the accuracy of quantum error correction. Testing on Google's Sycamore quantum processor demonstrated that AlphaQubit reduces errors by 6% compared to tensor network methods and by 30% compared to correlated matching, a widely used decoder.

"This collaborative work brought together Google DeepMind's machine learning knowledge and Google Quantum AI's error correction expertise to accelerate progress on building a reliable quantum computer," researchers stated in a Google blog post. "Accurately identifying errors is a critical step towards making quantum computers capable of performing long computations at scale, opening the doors to scientific breakthroughs and many new areas of discovery."

## NEW BENCHMARK FOR QUANTUM ERROR CORRECTION?

Quantum computers, which leverage principles like superposition and entanglement, are poised to solve specific problems exponentially faster than classical machines, according to the post. However, qubits—the building blocks of quantum computers—are highly susceptible to noise, leading to frequent errors. Overcoming this vulnerability is critical to scaling quantum devices for practical applications.

The team writes in the post: "The natural quantum state of a qubit is fragile and can be disrupted by various factors: microscopic defects in hardware, heat, vibration, electromagnetic interference and even cosmic rays (which are everywhere)."

To counteract this, quantum error correction uses redundancy: multiple physical qubits are grouped into a single logical qubit, and consistency checks are performed to detect and correct errors.

The challenge lies in decoding these checks efficiently and accurately, especially as quantum processors scale up. Current hardware typically exhibits error rates of 1% to 10% per operation, far too high for reliable computations. Future systems will require error rates below 0.000000001% for practical applications like drug discovery, materials design, and cryptographic tasks.

## HOW ALPHAQUBIT WORKS

AlphaQubit is built on the Transformer architecture — Transformer refers to a type of neural network architecture designed to process sequential data efficiently by, for example, focusing on the most important parts of the data it analyzes. This helps AlphaQubit to decode quantum errors accurately.

As the name suggests, neural networks are meant to mimic the human brain's neurons — generally speaking. Just like people have to learn before they master a new skill and continually hone that skill, neural networks have to learn and practice, too. AlphaQubit employs a two-stage training process: Pretraining and Finetuning.

In the pretraining phase, the model is first exposed to synthetic examples generated by a quantum simulator. This enables it to learn general error patterns under various noise conditions. Then, the system goes through the fine tuning. Here, the model is further trained on real-world error data from Google's Sycamore processor, tailoring it to the specific noise characteristics of the hardware.

The decoder adapts to complex error types, including "cross-talk" (unwanted qubit interactions) and "leakage" (qubits drifting into non-computational states). It also utilizes soft readouts—probabilistic measurements that provide richer information about qubit states.

In experiments with Sycamore's surface codes — which are a leading method for quantum error correction — AlphaQubit maintained its advantage across multiple configurations, from 17 qubits (distance 3) to 49 qubits (distance 5). The distance refers to the three errors (distance 3) or five errors (distance 5) that are required to break the logical qubit's encoded information.

Simulations extended this performance to systems with up to 241 qubits, suggesting the decoder's potential for larger quantum devices.

## IMPLICATIONS AND CHALLENGES

The team suggests that their success with AlphaQubit's performance represents a significant step forward in the integration of machine learning and quantum computing. By automating the decoding process, the model reduces the reliance on hand-crafted algorithms, which often struggle with the complexity of real-world noise.

“Although we anticipate that other decoding techniques will continue to improve, this work supports our belief that machine-learning decoders may achieve the necessary error suppression and speed to enable practical quantum computing,” the researchers write in the study.

However, the system is not without limitations. Current implementations of AlphaQubit may initially be on the slow side for real-time error correction on high-speed superconducting quantum processors, which perform a million consistency checks per second. Additionally, training the model for larger systems requires substantial computational resources, highlighting the need for more data-efficient approaches.

They write: “AlphaQubit represents a major milestone in using machine learning for quantum error correction. But we still face significant challenges involving speed and scalability.”

## BROADER IMPACT AND FUTURE DIRECTIONS

As noted above, quantum error correction is essential for achieving fault-tolerant quantum computing, so mastering errors becomes a prerequisite for tackling some of the most pressing challenges in science and industry. As AlphaQubit matures, it could reduce the number of physical qubits needed to form logical qubits, making quantum computers more compact and cost-effective.

The model’s architecture is also versatile, with potential applications beyond surface codes. Researchers plan to explore its adaptation to other quantum error-correction frameworks, such as color codes and low-density parity-check codes.

Further improvements will likely involve integrating AlphaQubit with hardware advancements, including custom processors designed for machine-learning tasks. Techniques like weight pruning and lower-precision inference could also enhance the model’s efficiency.

While challenges remain and there is more work to do, the researchers suggest that AlphaQubit serves as a way to give machine learning a role in the quest for reliable quantum computation. The vision for the future, then, would be one where quantum hardware and AI models evolve in tandem — and the dream of fault-tolerant quantum computers capable of solving real-world problems inches closer to reality.

“AlphaQubit represents a major milestone in using machine learning for quantum error correction. But we still face significant challenges involving speed and scalability,” the team writes in their post. “Our teams are combining pioneering advances in machine learning and quantum error correction to overcome these challenges—and pave the way for reliable quantum computers that can tackle some of the world’s most complex problems.”

# Understanding Google's Quantum Error Correction Breakthrough

Imagine trying to balance thousands of spinning tops at the same time—each top representing a qubit, the fundamental building block of a quantum computer. Now imagine these tops are so sensitive that even a slight breeze, a tiny vibration, or a quick peek to see if they're still spinning could make them wobble or fall. That's the challenge of quantum computing: Qubits are incredibly fragile, and even the process of controlling or measuring them introduces errors.

This is where Quantum Error Correction (QEC) comes in. By combining multiple fragile physical qubits into a more robust logical qubit, QEC allows us to correct errors faster than they accumulate. The goal is to operate below a critical threshold—the point where adding more qubits reduces, rather than increases, errors. That's precisely what Google Quantum AI has achieved with their recent breakthrough [1].

## Google's Breakthrough Achievement

To grasp the significance of Google's result, let's first understand what success in error correction looks like. In classical computers, error-resistant memory is achieved by duplicating bits to detect and correct errors. A method called majority voting is often used, where multiple copies of a bit are compared, and the majority value is taken as the correct bit. In quantum systems, physical qubits are combined to create logical qubits, where errors are corrected by monitoring correlations among qubits instead of directly observing the qubits themselves. It involves redundancy like majority voting, but does not rely on observation but rather entanglement. This indirect approach is crucial because directly measuring a qubit's state would disrupt its quantum properties. Effective quantum error correction maintains the integrity of logical qubits, even when some physical qubits experience errors, making it essential for scalable quantum computing.

However, this only works if the physical error rate is below a critical threshold. In fact, intuition says that increasing the number of physical qubits that make a logical qubit should allow for better error correction. In truth if each physical qubit is very error-prone, adding qubits makes errors accumulate faster than we can detect and correct them. In other words, quantum error correction works only if each qubit can operate below an error threshold even before any error correction. Having more physical qubits allows to increase the QEC code distance, which is a measure of a quantum code's ability to detect and correct errors.

By showing logical error decreased by a factor of 2.14 when increasing code distance from five to seven, Google has now demonstrated below-threshold operation using surface codes—a specific type of quantum error correction code.  This reduction in errors (which is exponential with increasing code distance) is the smoking gun proving that their QEC strategy works. With this, Google could show that their logical qubit lasted more than twice as long as their best physical qubit, as shown in Figure 1, demonstrating that logical qubits didn't just survive—they outperformed physical ones.

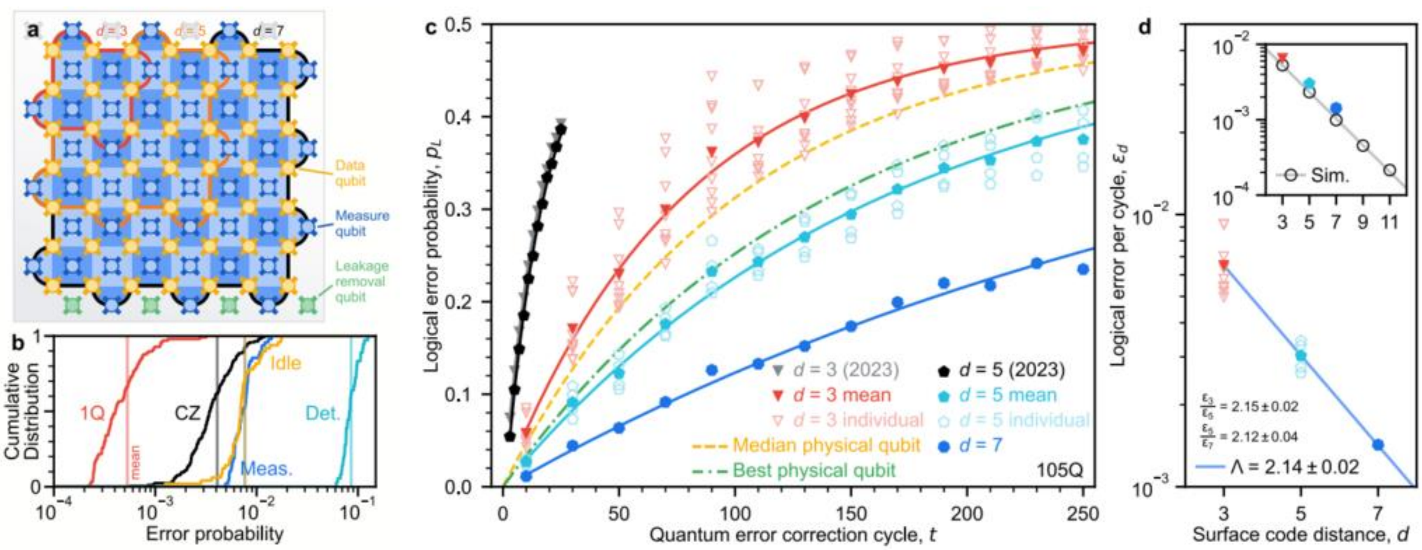



*Fig. 1 – An adapted plot showing logical qubit error rates versus code distance, highlighting exponential suppression of logical errors as the code distance increases. The figure illustrates the transition to below-threshold performance and the "beyond break-even" behavior achieved with distance-7 codes. (Adapted from [1] by Google Quantum AI, CC BY 4.0)*

A distance-7 surface code on 101 qubits effectively doubled the logical qubit's lifetime (blue line in Figure 1c) compared to uncorrected physical qubits (green line in Figure 1c). This accomplishment demonstrates that error-corrected qubits can preserve coherence for longer periods, which is crucial for running extended quantum algorithms and computations.

## A Control Engineering Perspective: How Google Made It Work.

The experiment wasn't just a test of surface codes—it was a carefully orchestrated feat of engineering and control. The control system had to deliver flawless precision on multiple fronts—synchronization, frequency control, measurement fidelity, real-time decoding, and stability—over many hours of operation. Let's stop for a second to talk about some of these interesting challenges.

At the heart of the system was **real-time synchronization**. Every correction cycle had to complete within 1.1 μs—a narrow window in which the qubits were measured. The precision of this synchronization was critical to preventing errors from accumulating and destabilizing the computation. Achieving this required precise coordination of control signals across the qubit array, ensuring that every gate operation, measurement, was perfectly aligned.

One of the most important components was **real-time decoding**. Decoding refers to the process of analyzing measurement data to determine where and how errors have occurred. To use logical qubits to perform universal quantum computation, certain gates called non-Clifford gates must be applied. Applying these gates, required correcting errors in real-time based on the real-time decoding. In Google's system, the real-time decoder maintained a constant latency of about 63 μs while operating over one million correction cycles. Namely, the real-time error correction pipeline could process the measurements fast enough to avoid congestion. This rapid decoding process was essential, as any delay could allow errors to propagate and accumulate, potentially destabilizing the logical qubits.

The experiment also demanded **high-fidelity gate operations**. Errors in qubit gates could easily propagate through the system, jeopardizing the stability of the logical qubit. Google achieved single-qubit gate errors below 0.1% and two-qubit CZ gate errors around 0.3%—thresholds essential to keeping logical qubits stable over time. For this goal, high performance of the control electronics is paramount, as fidelity can directly be impaired by errors of control pulses. These fidelities are especially critical when scaling surface codes, where even minor gate errors could degrade the effectiveness of error correction.

As quantum computers scale to more qubits and longer computations, these and more control requirements will only grow more demanding, making the development of advanced control hardware essential for the future of fault-tolerant quantum computing.

Out of the requirements above, real-time decoding, in particular, is fundamental for any scalable quantum computing system, as it provides the rapid response required to keep quantum information stable.

## A deeper dive into real-time decoding

Google's work highlights that the feasibility of the decoding depends on the decoder latency and throughput, as one of the most important pieces for running QEC below threshold.

Decoding is a classical compute task, and it can be done effectively on various classical architectures, such as FPGAs or GPUs. However, there is usually a trade-off between computational resources. FPGAs for example, are limited in computing power, but operate deterministically and in strict timing, making them suitable to manage the qubit control and measurement tasks as well as perform dedicated classical computations with low latency. On the other hand, CPUs or GPUs might have increased latency but enable far more advanced and larger computation. At Quantum Machines, we partnered with NVIDIA to deliver a unique platform, called DGX Quantum, that provides a unique combination of ultra-low controller-decoder latency, high-performance computational power, and flexible SW programmability. Our platform, which includes a less than 4 μs communication between our controller, OPX1000 and the CPU/GPU, allows to easily program and execute QEC workflows, including real-time decoding such as Google's decoding. The SW programmability allows iterating over the decoding algorithm and scheme very quickly. A feature we believe is key for faster progress towards scalable and effective QEC. The truth is that a lot more experimentation and benchmarking is needed to learn what decoders to use, which classical resources optimize performance and meet requirements and how to design systems that can eventually run QEC on a much larger scale. What we know so far is that the latency of decoders should be less than 10 μs for QEC schemes to converge. Watch our CEO Itamar Sivan explaining this further with the example of Shor's algorithm for factorizing the number 21.

DGX-quantum is already live, showcasing less than 4 μs round-trip latency between controller and GPU. To learn more, watch the IEEE QCE 2024 tutorial below, on DGX-quantum, co-authored by QM and NVIDIA.

## So, what's next?

Google's demonstration of below-threshold quantum error correction marks a milestone towards fault-tolerant quantum computing. By demonstrating that logical qubits can outperform physical qubits and showing that errors can be corrected faster than they accumulate, they've paved the way for scalable quantum processors.

However, this is just the beginning. In the future, to perform universal quantum computation with error corrected logical qubits, the full feedback loop must be closed, meaning that the control system needs to make decisions in real-time based on the decoder computation. Future developments will require faster decoders, better error mitigation strategies, automated calibrations embedded within quantum programs to stabilize parameters, and control hardware that tightly integrates and manages classical and quantum workflows.

Google's achievement signifies a substantial step toward fault-tolerant quantum computing. By demonstrating that logical error rates can be exponentially suppressed through the use of surface codes, the work provides a scalable and practical pathway to reliable quantum computing. As code distance increases, errors decrease at a rapid rate, setting the stage for quantum processors capable of handling complex operations with higher fidelity. Furthermore, this implementation of fast decoding represents a fundamental advancement in QEC. This technique allows for correction of errors faster than their propagation, minimizing the chance for errors to propagate through the quantum system.

# Quantum Error Correction and the Vision for Fault Tolerance

Real-time, low-latency feedback loops are going to be an essential element of future fault tolerant quantum devices, to ensure that errors are corrected faster than they accumulate. This principle resonates across the broader quantum computing community, where rapid and robust control mechanisms are viewed as the key to achieving large-scale, reliable quantum operations.

By focusing on low-latency, high-fidelity feedback and decoding, the broader quantum technology field is advancing toward the shared goal of fault-tolerant quantum computing, just as Google's milestone achievement shows. The evolution of quantum control systems that support agile error correction and real-time adaptability will continue to play a central role in the pursuit of stable, scalable quantum computing systems that can be deployed in practical applications. And with DGX-quantum, we are just starting this exciting journey, so stay tuned for what's to come!


*The DGX Quantum solution, co-developed by NVIDIA and Quantum Machines, enables quantum error correction, calibration, and fast retuning for large-scale quantum computers. It allows the use of robust classical resources (GPUs and CPUs) for quantum computer operation, with ultra-fast data round-trip delays of under 4 μs.*

## Reference

[1] Acharya, Rajeev, et al. "Quantum error correction below the surface code threshold." arXiv preprint arXiv:2408.13687 (2024).

# Quantum error correction

**Quantum error correction** (**QEC**) is a set of techniques used in quantum computing to protect quantum information from errors due to decoherence and other quantum noise. Quantum error correction is theorised as essential to achieve fault tolerant quantum computing that can reduce the effects of noise on stored quantum information, faulty quantum gates, faulty quantum state preparation, and faulty measurements. Effective quantum error correction would allow quantum computers with low qubit fidelity to execute algorithms of higher complexity or greater circuit depth.[1]

Classical error correction often employs redundancy. The simplest albeit inefficient approach is the repetition code. A repetition code stores the desired (logical) information as multiple copies, and—if these copies are later found to disagree due to errors introduced to the system-determines the most likely value for the original data by majority vote. E.g. suppose we copy a bit in the one (on) state three times. Suppose further that noise in the system introduces an error which corrupts the three-bit state so that one of the copied bits becomes zero (off) but the other two remain equal to one. Assuming that errors are independent and occur with some sufficiently low probability $p$, it is most likely that the error is a single-bit error and the intended message is three bits in the one state. It is possible that a double-bit error occurs and the transmitted message is equal to three zeros, but this outcome is less likely than the above outcome. In this example, the logical information is a single bit in the one state and the physical information are the three duplicate bits. Creating a physical state that represents the logical state is called *encoding* and determining which logical state is encoded in the physical state is called *decoding*. Similar to classical error correction, QEC codes do not always correctly decode logical qubits, but instead reduce the effect of noise on the logical state.

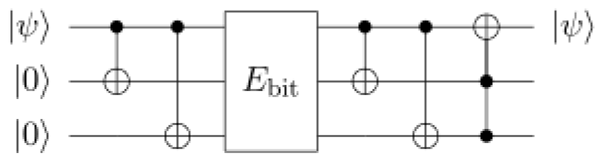
Copying quantum information is not possible due to the no-cloning theorem. This theorem seems to present an obstacle to formulating a theory of quantum error correction. But it is possible to *spread* the (logical) information of one logical qubit onto a highly entangled state of several (physical) qubits. Peter Shor first discovered this method of formulating a *quantum error correcting code* by storing the information of one qubit onto a highly entangled state of nine qubits.[2]

In classical error correction, *syndrome decoding* is used to diagnose which error was the likely source of corruption on an encoded state. An error can then be reversed by applying a corrective operation based on the syndrome. Quantum error correction also employs syndrome measurements. It performs a multi-qubit measurement that does not disturb the quantum information in the encoded state but retrieves information about the error. Depending on the QEC code used, syndrome measurement can determine the occurrence, location and type of errors. In most QEC codes, the type of error is either a bit flip, or a sign (of the phase) flip, or both (corresponding to the Pauli matrices X, Z, and Y). The measurement of the syndrome has the projective effect of a quantum measurement, so even if the error due to the noise was arbitrary, it can be expressed as a combination of basis operations called the error basis (which is given by the Pauli matrices and the identity). To correct the error, the Pauli operator corresponding to the type of error is used on the corrupted qubit to revert the effect of the error.

The syndrome measurement provides information about the error that has happened, but not about the information that is stored in the logical qubit—as otherwise the measurement would destroy any quantum superposition of this logical qubit with other qubits in the quantum computer, which would prevent it from being used to convey quantum information.

# Bit flip code

The repetition code works in a classical channel, because classical bits are easy to measure and to repeat. This approach does not work for a quantum channel in which, due to the no-cloning theorem, it is not possible to repeat a single qubit three times. To overcome this, a different method has to be used, such as the *three-qubit bit flip code* first proposed by Asher Peres in 1985.[3] This technique uses entanglement and syndrome measurements and is comparable in performance with the repetition code.

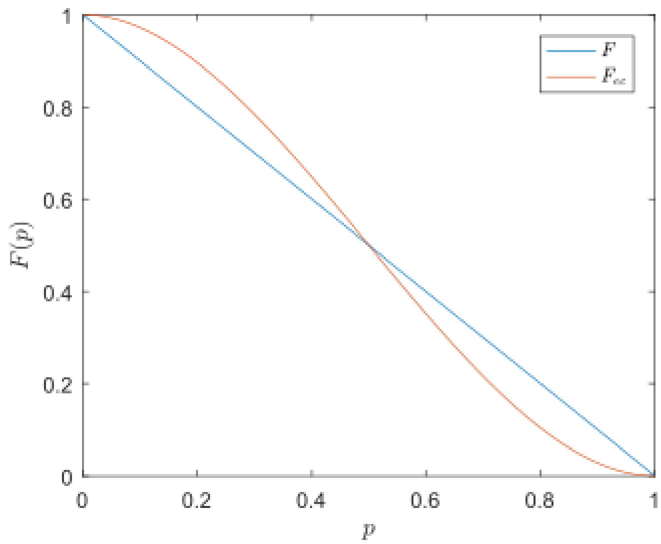


Quantum circuit of the bit flip code

Consider the situation in which we want to transmit the state of a single qubit  through a noisy channel. Let us moreover assume that this channel either flips the state of the qubit, with probability , or leaves it unchanged. The action of  on a general input  can therefore be written as .

Let  be the quantum state to be transmitted. With no error correcting protocol in place, the transmitted state will be correctly transmitted with probability  . We can however improve on this number by *encoding* the state into a greater number of qubits, in such a way that errors in the corresponding logical qubits can be detected and corrected. In the case of the simple three-qubit repetition code, the encoding consists in the mappings  and  . The input state  is encoded into the state  . This mapping can be realized for example using two CNOT gates, entangling the system with two ancillary qubits initialized in the state  .[4] The encoded state  is what is now passed through the noisy channel.

The channel acts on  by flipping some subset (possibly empty) of its qubits. No qubit is flipped with probability , a single qubit is flipped with probability , two qubits are flipped with probability, and all three qubits are flipped with probability . Note that a further assumption about the channel is made here: we assume that  acts equally and independently on each of the three qubits in which the state is now encoded. The problem is now how to detect and correct such errors, while not corrupting the transmitted state.




Comparison of output *minimum* fidelities, with (red) and without (blue) error correcting via the three qubit bit flip code. Notice how, for , the error correction scheme improves the fidelity.

Let us assume for simplicity that is small enough that the probability of more than a single qubit being flipped is negligible. One can then detect whether a qubit was flipped, without also querying for the values being transmitted, by asking whether one of the qubits differs from the others. This amounts to performing a measurement with four different outcomes, corresponding to the following four projective measurements:
This reveals which qubits are different from the others, without at the same time giving information about the state of the qubits themselves. If the outcome corresponding to is obtained, no correction is applied, while if the outcome corresponding to is observed, then the Pauli *X* gate is applied to the -th qubit. Formally, this correcting procedure corresponds to the application of the following map to the output of the channel:
Note that, while this procedure perfectly corrects the output when zero or one flips are introduced by the channel, if more than one qubit is flipped then the output is not properly corrected. For example, if the first and second qubits are flipped, then the syndrome measurement gives the outcome , and the third qubit is flipped, instead of the first two. To assess the performance of this error-correcting scheme for a general input we can study the fidelity between the input and the output . Being

the output state correct when no more than one qubit is flipped, which happens

with probability , we can write it as , where the dots denote components

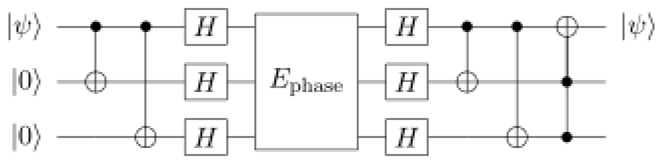
of resulting from errors not properly corrected by the protocol. It follows that
This fidelity is to be compared with the corresponding fidelity obtained when no error-

correcting protocol is used, which was shown before to equal . A little algebra then

shows that the fidelity *after* error correction is greater than the one without for .
Note that this is consistent with the working assumption that was made while deriving

the protocol (of being small enough).

# Sign flip code



Quantum circuit of the phase flip code

Flipped bits are the only kind of error in classical computer, but there is another possibility of an error with quantum computers, the sign flip. Through the transmission in

a channel the relative sign between and can become inverted. For instance, a

qubit in the state may have its sign flip to

The original state of the qubit will be changed into the state
In the Hadamard basis, bit flips become sign flips and sign flips become bit flips.

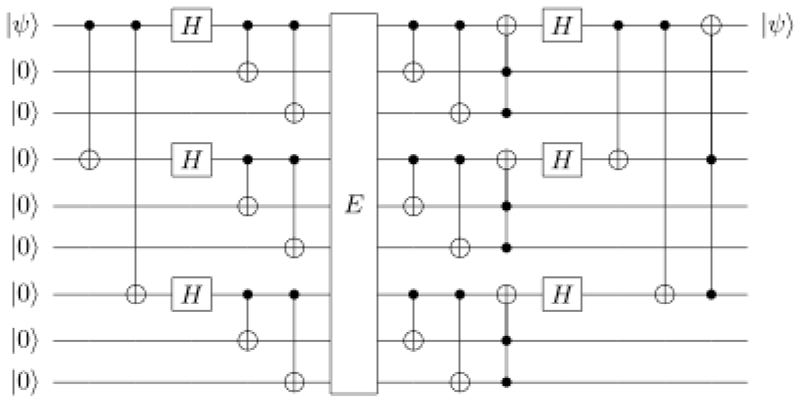
Let be a quantum channel that can cause at most one phase flip. Then the bit flip

code from above can recover by transforming into the Hadamard basis before and

after transmission through .

# Shor code

The error channel may induce either a bit flip, a sign flip (i.e., a phase flip), or both. It is possible to correct for both types of errors on a logical qubit using a well-designed QEC code. One example of a code that does this is the Shor code, published in 1995.[2][5]: 10 Since these two types of errors are the only types of errors that can result after a projective measurement, a Shor code corrects arbitrary single-qubit errors.

 Quantum circuit to encode a single logical qubit with the Shor code and then perform bit flip error correction on each of the three blocks.

Let be a quantum channel that can arbitrarily corrupt a single qubit. The 1st, 4th and 7th qubits are for the sign flip code, while the three groups of qubits (1,2,3), (4,5,6), and (7,8,9) are designed for the bit flip code. With the Shor code, a qubit state will be transformed into the product of 9 qubits , where

If a bit flip error happens to a qubit, the syndrome analysis will be performed on each block of qubits (1,2,3), (4,5,6), and (7,8,9) to detect and correct at most one bit flip error in each block.

If the three bit flip group (1,2,3), (4,5,6), and (7,8,9) are considered as three inputs, then the Shor code circuit can be reduced as a sign flip code. This means that the Shor code can also repair a sign flip error for a single qubit.

The Shor code also can correct for any arbitrary errors (both bit flip and sign flip) to a single qubit. If an error is modeled by a unitary transform U, which will act on a qubit , then can be described in the form where , , , and are complex constants, I is the identity, and the Pauli matrices are given by

If *U* is equal to *I*, then no error occurs. If , a bit flip error occurs. If , a sign flip error occurs. If then both a bit flip error and a sign flip error occur. In other words, the Shor code can correct any combination of bit or phase errors on a single qubit.

# Bosonic codes

Several proposals have been made for storing error-correctable quantum information in bosonic modes.[clarification needed] Unlike a two-level system, a quantum harmonic oscillator has infinitely many energy levels in a single physical system. Codes for these systems include cat,[6][7][8] Gottesman-Kitaev-Preskill (GKP),[9] and binomial codes.[10][11] One insight offered by these codes is to take advantage of the redundancy within a single system, rather than to duplicate many two-level qubits.

### Binomial code[10]

Written in the Fock basis, the simplest binomial encoding is where the subscript L indicates a "logically encoded" state. Then if the dominant error mechanism of the system is the stochastic application of the bosonic lowering operator the corresponding error states are and respectively. Since the codewords involve only even photon number, and the error states involve only odd photon number, errors can be detected by measuring the photon number parity of the system.[10][12] Measuring the odd parity will allow correction by application of an appropriate unitary operation without knowledge of the specific logical state of the qubit. However, the particular binomial code above is not robust to two-photon loss.

### Cat code[6][7][8]

Schrödinger cat states, superpositions of coherent states, can also be used as logical states for error correction codes. Cat code, realized by Ofek et al.[13] in 2016, defined two sets of logical states: and , where each of the states is a superposition of coherent state as follows

Those two sets of states differ from the photon number parity, as states denoted with only occupy even photon number states and states with indicate they have odd parity. Similar to the binomial code, if the dominant error mechanism of the system is the stochastic application of the bosonic lowering operator , the error takes the logical states from the even parity subspace to the odd one, and vice versa. Single-photon-loss errors can therefore be detected by measuring the photon number parity operator using a dispersively coupled ancillary qubit.[12]

Still, cat qubits are not protected against two-photon loss , dephasing noise , photon-gain error , etc.

## General codes

In general, a *quantum code* for a quantum channel is a subspace ,

where is the state Hilbert space, such that there exists another quantum

channel with where is the orthogonal projection onto . Here is known as the *correction operation*.

A *non-degenerate code* is one for which different elements of the set of correctable errors produce linearly independent results when applied to elements of the code. If distinct of the set of correctable errors produce orthogonal results, the code is considered *pure*.[14]

# Models

Over time, researchers have come up with several codes:

- Peter Shor's 9-qubit-code, a.k.a. the Shor code, encodes 1 logical qubit in 9 physical qubits and can correct for arbitrary errors in a single qubit.
- Andrew Steane found a code that does the same with 7 instead of 9 qubits, see Steane code.
- Raymond Laflamme and collaborators found a class of 5-qubit codes that do the same, which also have the property of being fault-tolerant. A 5-qubit code is the smallest possible code that protects a single logical qubit against single-qubit errors.
- A generalisation of the technique used by Steane, to develop the 7-qubit code from the classical [7, 4] Hamming code, led to the construction of an important class of codes called the CSS codes, named for their inventors: Robert Calderbank, Peter Shor and Andrew Steane. According to the quantum Hamming bound, encoding a single logical qubit and providing for arbitrary error correction in a single qubit requires a minimum of 5 physical qubits.
- A more general class of codes (encompassing the former) are the stabilizer codes discovered by Daniel Gottesman, and by Robert Calderbank, Eric Rains, Peter Shor, and N. J. A. Sloane; these are also called additive codes.
- Two dimensional Bacon–Shor codes are a family of codes parameterized by integers *m* and *n*. There are *nm* qubits arranged in a square lattice.[15]
- Alexei Kitaev's topological quantum codes, introduced in 1997 as the toric code, and the more general idea of a topological quantum computer are the basis for various code types.[16]
- Todd Brun, Igor Devetak, and Min-Hsiu Hsieh also constructed the entanglement-assisted stabilizer formalism as an extension of the standard stabilizer formalism that incorporates quantum entanglement shared between a sender and a receiver.
- The ideas of stabilizer codes, CSS codes, and topological codes can be expanded into the 2D planar surface code, of which various types exist.[17] As of June 2024, the 2D planar surface code is generally considered the most

well-studied type of quantum error correction, and one of the leading contenders for practical use in quantum computing.[18][19]

That these codes allow indeed for quantum computations of arbitrary length is the content of the quantum threshold theorem, found by Michael Ben-Or and Dorit Aharonov, which asserts that you can correct for all errors if you concatenate quantum codes such as the CSS codes—i.e. re-encode each logical qubit by the same code again, and so on, on logarithmically many levels—*provided* that the error rate of individual quantum gates is below a certain threshold; as otherwise, the attempts to measure the syndrome and correct the errors would introduce more new errors than they correct for.

As of late 2004, estimates for this threshold indicate that it could be as high as 1–3%,[20] provided that there are sufficiently many qubits available.

# Experimental realization

There have been several experimental realizations of CSS-based codes. The first demonstration was with nuclear magnetic resonance qubits.[21] Subsequently, demonstrations have been made with linear optics,[22] trapped ions,[23][24] and superconducting (transmon) qubits.[25]

In 2016 for the first time the lifetime of a quantum bit was prolonged by employing a QEC code.[13] The error-correction demonstration was performed on Schrödinger-cat states encoded in a superconducting resonator, and employed a quantum controller capable of performing real-time feedback operations including read-out of the quantum information, its analysis, and the correction of its detected errors. The work demonstrated how the quantum-error-corrected system reaches the break-even point at which the lifetime of a logical qubit exceeds the lifetime of the underlying constituents of the system (the physical qubits).

Other error correcting codes have also been implemented, such as one aimed at correcting for photon loss, the dominant error source in photonic qubit schemes.[26][27]

In 2021, an entangling gate between two logical qubits encoded in topological quantum error-correction codes has first been realized using 10 ions in a trapped-ion quantum computer.[28][29] 2021 also saw the first experimental demonstration of fault-tolerant Bacon-Shor code in a single logical qubit of a trapped-ion system, i.e. a demonstration for which the addition of error correction is able to suppress more errors than is introduced by the overhead required to implement the error correction as well as fault tolerant Steane code.[30][31][32]

In 2022, researchers at the University of Innsbruck have demonstrated a fault-tolerant universal set of gates on two logical qubits in a trapped-ion quantum computer. They have performed a logical two-qubit controlled-NOT gate between two instances of the seven-qubit colour code, and fault-tolerantly prepared a logical magic state.[33]

In February 2023 researchers at Google claimed to have decreased quantum errors by increasing the qubit number in experiments, they used a fault tolerant surface code measuring an error rate of 3.028% and 2.914% for a distance-3 qubit array and a distance-5 qubit array respectively.[34][35][36]

In April 2024, researchers at Microsoft claimed to have successfully tested a quantum error correction code that allowed them to achieve an error rate with logical qubits that is 800 times better than the underlying physical error rate.[37]

This qubit virtualization system was used to create 4 logical qubits with 30 of the 32 qubits on Quantinuum's trapped-ion hardware. The system uses an active syndrome extraction technique to diagnose errors and correct them while calculations are underway without destroying the logical qubits.[38]

# Quantum error-correction without encoding and parity-checks

In 2022, research at University of Engineering and Technology Lahore demonstrated error-cancellation by inserting single-qubit Z-axis rotation gates into strategically chosen locations of the superconductor quantum circuits.[39] The scheme has been shown to effectively correct errors that would otherwise rapidly add up under constructive interference of coherent noise. This is a circuit-level calibration scheme that traces deviations (e.g. sharp dips or notches) in the decoherence curve to detect and localize the coherent error, but does not require encoding or parity measurements.[40] However, further investigation is needed to establish the effectiveness of this method for the incoherent noise.[39]

# References


1. ^ *Cai, Weizhou; Ma, Yuwei (2021). "Bosonic quantum error correction codes in superconducting quantum circuits". Fundamental Research.* ***1*** *(1): 50–67. arXiv:2010.08699. Bibcode:2021FunRe...1...50C. doi:10.1016/j.fmre.2020.12.006. A practical quantum computer that is capable of large circuit depth, therefore, ultimately calls for operations on logical qubits protected by quantum error correction*
2. ^ Jump up to:a b *Shor, Peter W. (1995). "Scheme for reducing decoherence in quantum computer memory". Physical Review A.* ***52*** *(4): R2493–R2496. Bibcode:1995PhRvA..52.2493S. doi:10.1103/PhysRevA.52.R2493. PMID 9912632.*
3. ^ *Peres, Asher (1985). "Reversible Logic and Quantum Computers". Physical Review A.* ***32*** *(6): 3266–3276. Bibcode:1985PhRvA..32.3266P. doi:10.1103/PhysRevA.32.3266. PMID 9896493.*
4. ^ *Nielsen, Michael A.; Chuang, Isaac L. (2000). Quantum Computation and Quantum Information. Cambridge University Press.*
5. ^ *Devitt, Simon J; Munro, William J; Nemoto, Kae (2013-06-20). "Quantum error correction for beginners". Reports on Progress in Physics.* ***76*** *(7):*

076001. arXiv:0905.2794. Bibcode:2013RPPh...76g6001D. doi:10.1088/0034-4885/76/7/076001. ISSN 0034-4885. PMID 23787909. S2CID 206021660.

6. ^ Jump up to:a b Cochrane, P. T.; Milburn, G. J.; Munro, W. J. (1999-04-01). "Macroscopically distinct quantum-superposition states as a bosonic code for amplitude damping". Physical Review A. **59** (4): 2631–2634. arXiv:quant-ph/9809037. Bibcode:1999PhRvA..59.2631C. doi:10.1103/PhysRevA.59.2631. S2CID 119532538.
7. ^ Jump up to:a b Leghtas, Zaki; Kirchmair, Gerhard; Vlastakis, Brian; Schoelkopf, Robert J.; Devoret, Michel H.; Mirrahimi, Mazyar (2013-09-20). "Hardware-Efficient Autonomous Quantum Memory Protection". Physical Review Letters. **111** (12): 120501. arXiv:1207.0679. Bibcode:2013PhRvL.111l0501L. doi:10.1103/physrevlett.111.120501. ISSN 0031-9007. PMID 24093235. S2CID 19929020.
8. ^ Jump up to:a b Mirrahimi, Mazyar; Leghtas, Zaki; Albert, Victor V; Touzard, Steven; Schoelkopf, Robert J; Jiang, Liang; Devoret, Michel H (2014-04-22). "Dynamically protected cat-qubits: a new paradigm for universal quantum computation". New Journal of Physics. **16** (4): 045014. arXiv:1312.2017. Bibcode:2014NJPh...16d5014M. doi:10.1088/1367-2630/16/4/045014. ISSN 1367-2630. S2CID 7179816.
9. ^ Daniel Gottesman; Alexei Kitaev; John Preskill (2001). "Encoding a qubit in an oscillator". Physical Review A. **64** (1): 012310. arXiv:quant-ph/0008040. Bibcode:2001PhRvA..64a2310G. doi:10.1103/PhysRevA.64.012310. S2CID 18995200.
10. ^ Jump up to:a b c Michael, Marios H.; Silveri, Matti; Brierley, R. T.; Albert, Victor V.; Salmilehto, Juha; Jiang, Liang; Girvin, S. M. (2016-07-14). "New Class of Quantum Error-Correcting Codes for a Bosonic Mode". Physical Review X. **6** (3): 031006. arXiv:1602.00008. Bibcode:2016PhRvX...6c1006M. doi:10.1103/PhysRevX.6.031006. S2CID 29518512.
11. ^ Albert, Victor V.; Noh, Kyungjoo; Duivenvoorden, Kasper; Young, Dylan J.; Brierley, R. T.; Reinhold, Philip; Vuillot, Christophe; Li, Linshu; Shen, Chao; Girvin, S. M.; Terhal, Barbara M.; Jiang, Liang (2018). "Performance and structure of single-mode bosonic codes". Physical Review A. **97** (3): 032346. arXiv:1708.05010. Bibcode:2018PhRvA..97c2346A. doi:10.1103/PhysRevA.97.032346. S2CID 51691343.
12. ^ Jump up to:a b Sun, L.; Petrenko, A.; Leghtas, Z.; Vlastakis, B.; Kirchmair, G.; Sliwa, K. M.; Narla, A.; Hatridge, M.; Shankar, S.; Blumoff, J.; Frunzio, L.; Mirrahimi, M.; Devoret, M. H.; Schoelkopf, R. J. (July 2014). "Tracking photon jumps with repeated quantum non-demolition parity measurements". Nature. **511** (7510): 444–448. arXiv:1311.2534. Bibcode:2014Natur.511..444S. doi:10.1038/nature13436. ISSN 1476-4687. PMID 25043007. S2CID 987945.
13. ^ Jump up to:a b Ofek, Nissim; Petrenko, Andrei; Heeres, Reinier; Reinhold, Philip; Leghtas, Zaki; Vlastakis, Brian; Liu, Yehan; Frunzio, Luigi; Girvin, S. M.; Jiang, L.; Mirrahimi, Mazyar (August 2016). "Extending the lifetime of a quantum bit with error correction in superconducting circuits". Nature. **536** (7617): 441–445. Bibcode:2016Natur.536..441O. doi:10.1038/nature18949. ISSN 0028-0836. PMID 27437573. S2CID 594116.
14. ^ Calderbank, A. R.; Rains, E. M.; Shor, P. W.; Sloane, N. J. A. (1998). "Quantum Error Correction via Codes over GF(4)". IEEE Transactions on

*Information Theory.* ***44*** *(4): 1369–1387. arXiv:quant-ph/9608006. doi:10.1109/18.681315. S2CID 1215697.*

15. ^ *Bacon, Dave (2006-01-30). "Operator quantum error-correcting subsystems for self-correcting quantum memories". Physical Review A.* ***73*** *(1): 012340. arXiv:quant-ph/0506023. Bibcode:2006PhRvA..73a2340B. doi:10.1103/PhysRevA.73.012340. S2CID 118968017.*
16. ^ *Kitaev, Alexei (1997-07-31). "Quantum Error Correction with Imperfect Gates". Quantum Communication, Computing, and Measurement. Springer. pp. 181–188. doi:10.1007/978-1-4615-5923-8.*
17. ^ *Fowler, Austin G.; Mariantoni, Matteo; Martinis, John M.; Cleland, Andrew N. (2012-09-18). "Surface codes: Towards practical large-scale quantum computation". Physical Review A.* ***86*** *(3): 032324. arXiv:1208.0928. Bibcode:2012PhRvA..86c2324F. doi:10.1103/PhysRevA.86.032324. ISSN 1050-2947.*
18. ^ *Gidney, Craig; Newman, Michael; Brooks, Peter; Jones, Cody (2023). "Yoked surface codes". arXiv:2312.04522 [quant-ph].*
19. ^ *Horsman, Dominic; Fowler, Austin G; Devitt, Simon; Meter, Rodney Van (2012-12-01). "Surface code quantum computing by lattice surgery". New Journal of Physics.* ***14*** *(12): 123011. arXiv:1111.4022. Bibcode:2012NJPh...14l3011H. doi:10.1088/1367-2630/14/12/123011. ISSN 1367-2630.*
20. ^ *Knill, Emanuel (2004-11-02). "Quantum Computing with Very Noisy Devices". Nature.* ***434*** *(7029): 39–44. arXiv:quant-ph/0410199. Bibcode:2005Natur.434...39K. doi:10.1038/nature03350. PMID 15744292. S2CID 4420858.*
21. ^ *Cory, D. G.; Price, M. D.; Maas, W.; Knill, E.; Laflamme, R.; Zurek, W. H.; Havel, T. F.; Somaroo, S. S. (1998). "Experimental Quantum Error Correction". Phys. Rev. Lett.* ***81*** *(10): 2152–2155. arXiv:quant-ph/9802018. Bibcode:1998PhRvL..81.2152C. doi:10.1103/PhysRevLett.81.2152. S2CID 11662810.*
22. ^ *Pittman, T. B.; Jacobs, B. C.; Franson, J. D. (2005). "Demonstration of quantum error correction using linear optics". Phys. Rev. A.* ***71*** *(5): 052332. arXiv:quant-ph/0502042. Bibcode:2005PhRvA..71e2332P. doi:10.1103/PhysRevA.71.052332. S2CID 11679660.*
23. ^ *Chiaverini, J.; Leibfried, D.; Schaetz, T.; Barrett, M. D.; Blakestad, R. B.; Britton, J.; Itano, W. M.; Jost, J. D.; Knill, E.; Langer, C.; Ozeri, R.; Wineland, D. J. (2004). "Realization of quantum error correction". Nature.* ***432*** *(7017): 602–605. Bibcode:2004Natur.432..602C. doi:10.1038/nature03074. PMID 15577904. S2CID 167898.*
24. ^ *Schindler, P.; Barreiro, J. T.; Monz, T.; Nebendahl, V.; Nigg, D.; Chwalla, M.; Hennrich, M.; Blatt, R. (2011). "Experimental Repetitive Quantum Error Correction". Science.* ***332*** *(6033): 1059–1061. Bibcode:2011Sci...332.1059S. doi:10.1126/science.1203329. PMID 21617070. S2CID 32268350.*
25. ^ *Reed, M. D.; DiCarlo, L.; Nigg, S. E.; Sun, L.; Frunzio, L.; Girvin, S. M.; Schoelkopf, R. J. (2012). "Realization of Three-Qubit Quantum Error Correction with Superconducting Circuits". Nature.* ***482*** *(7385): 382–*

385. arXiv:1109.4948. Bibcode:2012Natur.482..382R. doi:10.1038/nature10786. PMID 22297844. S2CID 2610639.

26. ^ Lassen, M.; Sabuncu, M.; Huck, A.; Niset, J.; Leuchs, G.; Cerf, N. J.; Andersen, U. L. (2010). "Quantum optical coherence can survive photon losses using a continuous-variable quantum erasure-correcting code". Nature Photonics. **4** (10): 700. arXiv:1006.3941. Bibcode:2010NaPho...4..700L. doi:10.1038/nphoton.2010.168. S2CID 55090423.
27. ^ Guo, Qihao; Zhao, Yuan-Yuan; Grassl, Markus; Nie, Xinfang; Xiang, Guo-Yong; Xin, Tao; Yin, Zhang-Qi; Zeng, Bei (2021). "Testing a quantum error-correcting code on various platforms". Science Bulletin. **66** (1): 29–35. arXiv:2001.07998. Bibcode:2021SciBu..66...29G. doi:10.1016/j.scib.2020.07.033. PMID 36654309. S2CID 210861230.
28. ^ "Error-protected quantum bits entangled for the first time". phys.org. 2021-01-13. Retrieved 2021-08-30.
29. ^ Erhard, Alexander; Poulsen Nautrup, Hendrik; Meth, Michael; Postler, Lukas; Stricker, Roman; Stadler, Martin; Negnevitsky, Vlad; Ringbauer, Martin; Schindler, Philipp; Briegel, Hans J.; Blatt, Rainer; Friis, Nicolai; Monz, Thomas (2021-01-13). "Entangling logical qubits with lattice surgery". Nature. **589** (7841): 220–224. arXiv:2006.03071. Bibcode:2021Natur.589..220E. doi:10.1038/s41586-020-03079-6. ISSN 1476-4687. PMID 33442044. S2CID 219401398.
30. ^ Bedford, Bailey (2021-10-04). "Foundational step shows quantum computers can be better than the sum of their parts". phys.org. Retrieved 2021-10-05.
31. ^ Egan, Laird; Debroy, Dripto M.; Noel, Crystal; Risinger, Andrew; Zhu, Daiwei; Biswas, Debopriyo; Newman, Michael; Li, Muyuan; Brown, Kenneth R.; Cetina, Marko; Monroe, Christopher (2021-10-04). "Fault-tolerant control of an error-corrected qubit". Nature. **598** (7880): 281–286. Bibcode:2021Natur.598..281E. doi:10.1038/s41586-021-03928-y. ISSN 0028-0836. PMID 34608286. S2CID 238357892.
32. ^ Ball, Philip (2021-12-23). "Real-Time Error Correction for Quantum Computing". Physics. **14**. 184. Bibcode:2021PhyOJ..14..184B. doi:10.1103/Physics.14.184. S2CID 245442996.
33. ^ Postler, Lukas; Heußen, Sascha; Pogorelov, Ivan; Rispler, Manuel; Feldker, Thomas; Meth, Michael; Marciniak, Christian D.; Stricker, Roman; Ringbauer, Martin; Blatt, Rainer; Schindler, Philipp; Müller, Markus; Monz, Thomas (2022-05-25). "Demonstration of fault-tolerant universal quantum gate operations". Nature. **605** (7911): 675–680. arXiv:2111.12654. Bibcode:2022Natur.605..675P. doi:10.1038/s41586-022-04721-1. PMID 35614250. S2CID 244527180.
34. ^ Google Quantum AI (2023-02-22). "Suppressing quantum errors by scaling a surface code logical qubit". Nature. **614** (7949): 676–681. Bibcode:2023Natur.614..676G. doi:10.1038/s41586-022-05434-1. ISSN 1476-4687. PMC 9946823. PMID 36813892.
35. ^ Boerkamp, Martijn (2023-03-20). "Breakthrough in quantum error correction could lead to large-scale quantum computers". Physics World. Retrieved 2023-04-01.
36. ^ Conover, Emily (2023-02-22). "Google's quantum computer reached an error-correcting milestone". ScienceNews. Retrieved 2023-04-01.

37.^ Smith-Goodson, Paul (2024-04-18). "Microsoft And Quantinuum Improve Quantum Error Rates By 800x". Forbes. Retrieved 2024-07-01.
38.^ Yirka, Bob (2024-04-05). "Quantinuum quantum computer using Microsoft's 'logical quantum bits' runs 14,000 experiments with no errors". Phys.org. Retrieved 2024-07-01.
39.^ Jump up to:[a] [b] Ahsan, Muhammad; Naqvi, Syed Abbas Zilqurnain; Anwer, Haider (2022-02-18). "Quantum circuit engineering for correcting coherent noise". Physical Review A. **105** (2): 022428. arXiv:2109.03533. Bibcode:2022PhRvA.105b2428A. doi:10.1103/physreva.105.022428. ISSN 2469-9926. S2CID 237442177.
40.^ Steffen, Matthias (2022-10-20). "What's the difference between error suppression, error mitigation, and error correction?". IBM Research Blog. Retrieved 2022-11-26.

# Further reading


- Daniel Lidar and Todd Brun, ed. (2013). Quantum Error Correction. Cambridge University Press.
- La Guardia, Giuliano Gadioli, ed. (2020). Quantum Error Correction: Symmetric, Asymmetric, Synchronizable, and Convolutional Codes. Springer Nature.
- Frank Gaitan (2008). Quantum Error Correction and Fault Tolerant Quantum Computing. Taylor & Francis.
- Freedman, Michael H.; Meyer, David A.; Luo, Feng (2002). "$Z_2$-Systolic freedom and quantum codes". Mathematics of quantum computation. Comput. Math. Ser. Boca Raton, FL: Chapman & Hall/CRC. pp. 287–320.
- Freedman, Michael H.; Meyer, David A. (1998). "Projective plane and planar quantum codes". Found. Comput. Math. **2001** (3): 325–332. arXiv:quant-ph/9810055. Bibcode:1998quant.ph.10055F.


# Suppressing quantum errors by scaling a surface code logical qubit

# Abstract

Practical quantum computing will require error rates well below those achievable with physical qubits. Quantum error correction[1,2] offers a path to algorithmically relevant error rates by encoding logical qubits within many physical qubits, for which increasing the number of physical qubits enhances protection against physical errors. However, introducing more qubits also increases the number of error sources, so the density of errors must be sufficiently low for logical performance to improve with increasing code size. Here we report the measurement of logical qubit performance scaling across several code sizes, and demonstrate that our system of superconducting qubits has sufficient performance to overcome the additional errors from increasing qubit number. We find that our distance-5 surface code logical qubit modestly outperforms an ensemble of distance-3 logical qubits on average, in terms of both logical error probability over 25 cycles and logical error per cycle ($(2.914 \pm 0.016)\%$ compared to $(3.028 \pm 0.023)\%$). To investigate damaging, low-probability error sources, we run a distance-25 repetition code and observe a $1.7 \times 10^{-6}$ logical error per cycle floor set by a single high-energy event ($1.6 \times 10^{-7}$ excluding this event). We accurately model our experiment, extracting error budgets that highlight the biggest challenges for future systems. These results mark an experimental demonstration in which quantum error correction begins to improve performance with increasing qubit number, illuminating the path to reaching the logical error rates required for computation.

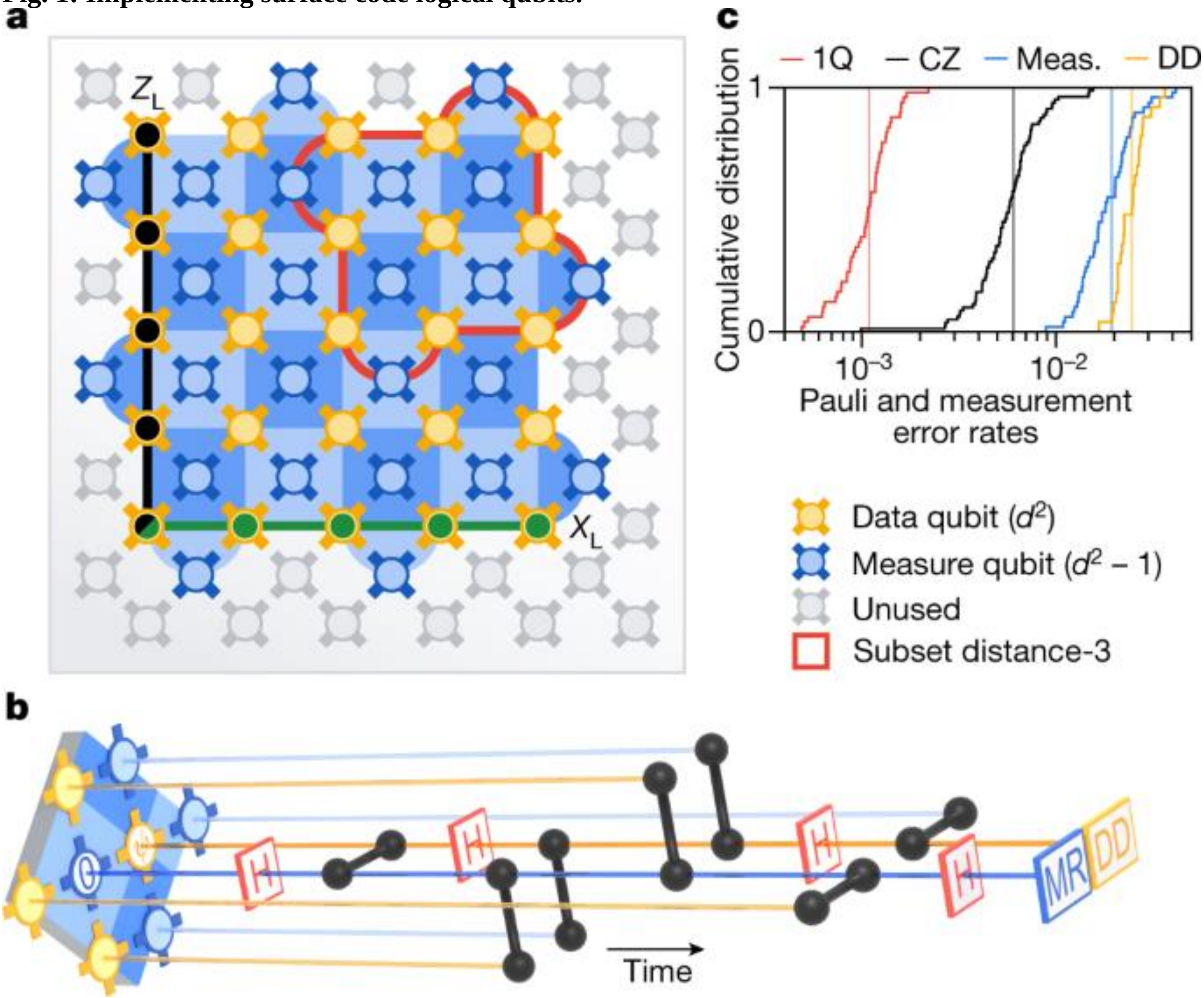
Since Feynman's proposal to compute using quantum mechanics[3], many potential applications have emerged, including factoring[4], optimization[5], machine learning[6], quantum simulation[7] and quantum chemistry[8]. These applications often require billions of quantum operations[9,10,11] and state-of-the-art quantum processors typically have error rates around $10^{-3}$ per gate[12,13,14,15,16,17], far too high to execute such large circuits. Fortunately, quantum error correction can exponentially suppress the operational error rates in a quantum processor, at the expense of temporal and qubit overhead[18,19].

Several works have reported quantum error correction on codes able to correct a single error, including the distance-3 Bacon–Shor[20], colour[21], five-qubit[22], heavy-hexagon[23] and surface[24,25] codes, as well as continuous variable codes[26,27,28,29]. However, a crucial question remains of whether scaling up the error-correcting code size will reduce logical error rates in a real device. In theory, logical errors should be reduced if physical errors are sufficiently sparse in the quantum processor. In practice, demonstrating reduced logical error requires scaling up a device to support a code that can correct at least two errors, without sacrificing state-of-the-art performance. In this work we report a 72-qubit superconducting device supporting a 49-qubit distance-5 ($d = 5$) surface code that narrowly outperforms its average subset 17-qubit distance-3 surface code, demonstrating a critical step towards scalable quantum error correction.

# Surface codes with superconducting qubits

Surface codes[30,31,32,33,34] are a family of quantum error-correcting codes that encode a logical qubit into the joint entangled state of a $d \times d$ square of physical qubits, referred to as data qubits. The logical qubit states are defined by a pair of anti-commuting logical observables $X_L$ and $Z_L$. For the example shown in Fig. 1a, a $Z_L$ observable is encoded in the joint $Z$-basis parity of a line of qubits that traverses the lattice from top to bottom, and likewise an $X_L$ observable is encoded in the joint $X$-basis parity traversing left to right. This non-local encoding of information protects the logical qubit from local physical errors, provided we can detect and correct them.

**Fig. 1: Implementing surface code logical qubits.**




**a**, Schematic of a 72-qubit Sycamore device with a distance-5 surface code embedded, consisting of 25 data qubits (gold) and 24 measure qubits (blue). Each measure qubit is associated with a stabilizer (blue coloured tile, dark: $X$, light: $Z$). Representative logical operators $Z_L$ (black) and $X_L$ (green) traverse the array, intersecting at the lower-left data qubit. The upper right quadrant (red outline) is one of four subset distance-3 codes (the four quadrants) that we compare to distance-5. **b**, Illustration of a stabilizer measurement, focusing on one data qubit (labelled $\psi$) and one measure qubit (labelled 0), in perspective view with time progressing to the right. Each qubit participates in four CZ gates (black) with its four nearest neighbours, interspersed with Hadamard gates (H), and finally, the measure qubit is measured and reset to $|0\rangle$ (MR). Data qubits perform dynamical decoupling (DD) while waiting for the measurement and reset. All stabilizers are measured in this manner concurrently. Cycle duration is 921 ns, including 25-ns single-qubit gates, 34-ns two-qubit gates, 500-ns measurement and 160-ns reset (see Supplementary Information for compilation details). The readout and reset take up most of the cycle time, so the concurrent data qubit idling is a dominant source of error. **c**, Cumulative distributions of errors for single-qubit gates (1Q), CZ gates, measurement (Meas.) and data qubit dynamical decoupling (idle during measurement and reset), which we refer to as component errors. The circuits were benchmarked in simultaneous operation using random circuit techniques, on the 49 qubits used in distance-5 and the 4 CZ layers from the stabilizer circuit[38,59] (see Supplementary Information). Vertical lines are means.

To detect errors, we periodically measure $X$ and $Z$ parities of adjacent clusters of data qubits with the aid of $d^2 - 1$ measure qubits interspersed throughout the lattice. As shown in Fig. 1b, each measure qubit interacts with its neighbouring data qubits to map the joint data qubit parity onto the measure qubit state, which is then measured. Each parity measurement, or stabilizer, commutes with the logical observables of the encoded qubit as well as every other stabilizer. Consequently, we can detect errors when parity measurements change unexpectedly, without disturbing the logical qubit state.

A decoder uses the history of stabilizer measurement outcomes to infer likely configurations of physical errors on the device. We can then determine the overall effect of these inferred errors on the logical qubit, thus preserving the logical state. Most surface code logical gates can be implemented by maintaining logical memory and executing different sequences of measurements on the code boundary[35,36,37]. Thus, we focus on preserving logical memory, the core technical challenge in operating the surface code.

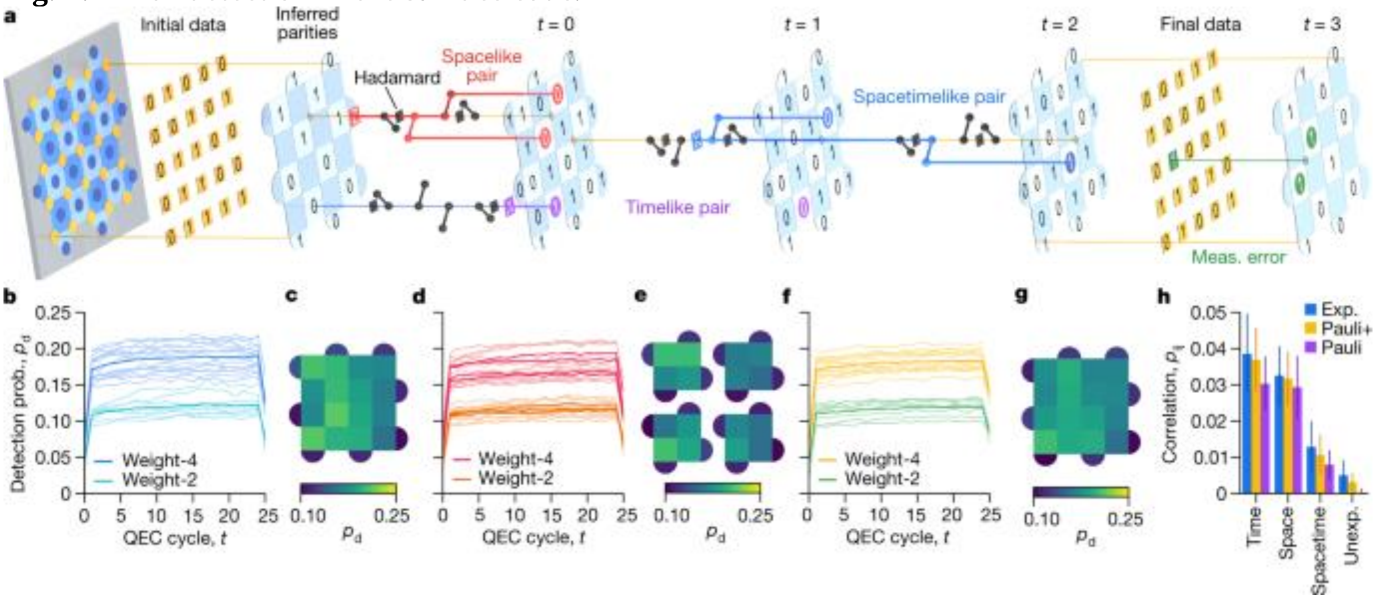
We implement the surface code on an expanded Sycamore device[38] with 72 transmon qubits[39] and 121 tunable couplers[40,41]. Each qubit is coupled to four nearest neighbours except on the boundaries, with mean qubit coherence times $T_1 = 20$ μs and $T_{2,\mathrm{CPMG}} = 30$ μs, in which CPMG represents Carr–Purcell–Meiboom–Gill. As in ref. 42, we implement single-qubit rotations, controlled-$Z$ (CZ) gates, reset and measurement, demonstrating similar or improved simultaneous performance as shown in Fig. 1c.

The distance-5 surface code logical qubit is encoded on a 49-qubit subset of the device, with 25 data qubits and 24 measure qubits. Each measure qubit corresponds to one stabilizer, classified by its basis ($X$ or $Z$) and the number of data qubits involved (weight, 2 or 4). Ideally, to assess how logical performance scales with code size, we would compare distance-5 and distance-3 logical qubits under identical noise. Although device inhomogeneity makes this comparison difficult, we can compare the distance-5 logical qubit to the average of four distance-3 logical qubit subgrids, each containing nine data qubits and eight measure qubits. These distance-3 logical qubits cover the four quadrants of the distance-5 code with minimal qubit overlap, capturing the average performance of the full distance-5 grid.

In a single instance of the experiment, we initialize the logical qubit state, run several cycles of error correction, and then measure the final logical state. We show an example in Fig. 2a. To prepare a $Z_L$ eigenstate, we first prepare each data qubit in $|0\rangle$ or $|1\rangle$, an eigenstate of the $Z$ stabilizers. The first cycle of stabilizer measurements then projects the data qubits into an entangled state that is also an eigenstate of the $X$ stabilizers. Each cycle contains CZ and

Hadamard gates sequenced to extract $X$ and $Z$ stabilizers simultaneously, and ends with the measurement and reset of the measure qubits. In the final cycle, we also measure the data qubits in the $Z$ basis, yielding both parity information and a measurement of the logical state. Preparing and measuring $X_L$ eigenstates proceeds analogously. The instance succeeds if the corrected logical measurement agrees with the known initial state; otherwise, a logical error has occurred.

**Fig. 2: Error detection in the surface code.**




**a**, Illustration of a surface code experiment, in perspective view with time progressing to the right. We begin with an initial data qubit state that has known parities in one stabilizer basis (here, $Z$). We show example errors that manifest in detection pairs: a $Z$ error (red) on a data qubit (spacelike pair), a measurement error (purple) on a measure qubit (timelike pair), an $X$ error (blue) during the CZ gates (spacetimelike pair) and a measurement error (green) on a data qubit (detected in the final inferred $Z$ parities). **b**, Detection probability for each stabilizer over a 25-cycle distance-5 experiment (50,000 repetitions). Darker lines: average over all stabilizers with the same weight. There are fewer detections at timestep $t = 0$ because there is no preceding syndrome extraction, and at $t = 25$ because the final parities are calculated from data qubit measurements directly. QEC, quantum error correction. **c**, Detection probability heatmap, averaging over $t = 1$ to 24. **d**,**e**, Similar to **b**,**c** for four separate distance-3 experiments covering the four quadrants of the distance-5 code. **f**,**g**, Similar to **b**,**c** using a simulation with Pauli errors plus leakage, crosstalk and stray interactions (Pauli+). **h**, Bar chart summarizing the detection correlation matrix $p_{ij}$, comparing the distance-5 experiment from **b** to the simulation in **f** (Pauli+) and a simpler simulation with only Pauli errors. We aggregate four groups of correlations: timelike pairs; spacelike pairs; spacetimelike pairs expected for Pauli noise; and spacetimelike pairs unexpected for Pauli noise (Unexp.), including correlations over two timesteps. Each bar shows a mean and standard deviation of correlations from a 25-cycle, 50,000-repetition dataset.

Our stabilizer circuits contain a few modifications to the standard gate sequence described above (see Supplementary Information), including phase corrections to correct for unintended qubit frequency shifts and dynamical decoupling gates during qubit idles[43]. We also remove certain Hadamard gates to implement the $ZXXZ$ variant of the surface code[44,45], which helps symmetrize the $X$- and $Z$-basis logical error rates. Finally, during initialization, the data qubits are prepared into randomly selected bitstrings. This ensures that we do not preferentially measure even parities in the first few cycles of the code, which could artificially lower logical error rates owing to bias in measurement error (see Supplementary Information).

# Error detectors

After initialization, parity measurements should produce the same value in each cycle, up to known flips applied by the circuit. If we compare a parity measurement to the corresponding measurement in the preceding cycle and their values are inconsistent, a detection event has occurred, indicating an error. We refer to these comparisons as detectors.

The detection event probabilities for each detector indicate the distribution of physical errors in space and time while running the surface code. In Fig. 2, we show the detection event probabilities in the distance-5 code (Fig. 2b,c) and the distance-3 codes (Fig. 2d,e) running for 25 cycles, as measured over 50,000 experimental instances. For the weight-4 stabilizers, the average detection probability is $0.185 \pm 0.018$ ($1\sigma$) in the distance-5 code and $0.175 \pm 0.017$ averaged over the distance-3 codes. The weight-2 stabilizers interact with fewer qubits and hence detect fewer errors. Correspondingly, they yield a lower average detection probability of $0.119 \pm 0.012$ in the distance-5 code and $0.115 \pm 0.008$ averaged over the distance-3 codes. The relative consistency between code distances suggests that growing the lattice does not substantially increase the component error rates during error correction.

The average detection probabilities exhibit a relative rise of 12% for distance-5 and 8% for distance-3 over 25 cycles, with a typical characteristic risetime of roughly 5 cycles (see Supplementary Information). We attribute this rise to data qubits leaking into non-computational excited states and anticipate that the inclusion of leakage-removal techniques on data qubits would help to mitigate this rise[42,46,47,48]. We reason that the greater increase in detection probability in the distance-5 code is due to increased stray interactions or leakage from simultaneously operating more gates and measurements.

We test our understanding of the physical noise in our system by comparing the experimental data to a simulation. We begin with a depolarizing noise simulation based on the component error information in Fig. 1c, and then extend to a Pauli simulation with qubit-specific $T_1$ and $T_{2,\mathrm{CPMG}}$, transitions to leaked states, and stray interactions between qubits during CZ gates (see Supplementary Information). We refer to this simulation as Pauli+. Figure 2f shows that this second simulator accurately predicts the average detection probabilities, finding $0.180 \pm 0.013$ for the weight-4 stabilizers and $0.116 \pm 0.011$ for the weight-2 stabilizers, with average detection probabilities increasing 7% over 25 cycles (distance-5).

# Understanding errors through correlations

We next examine pairwise correlations between detection events, which give us fine-grained information about which types of error are occurring during error correction. Figure 2a illustrates a few examples of pairwise detections that are generated by $X$ or $Z$ errors in the surface code. Measurement and reset errors are detected by the same stabilizer in two consecutive cycles, which we classify as a timelike pair. Data qubits may experience an $X$ ($Z$) error while idling during measurement that is detected by its neighbouring *Z* (*X*) stabilizers in the same cycle, forming a spacelike pair. Errors during CZ gates may cause a variety of pairwise detections to occur, including spacetimelike pairs that are separated in both space and time. More complex clusters of detection events arise when a *Y* error occurs, which generates detection events for both *X* and *Z* errors.

To estimate the probability for each detection event pair from our data, we compute an appropriately normalized correlation $p_{ij}$ between detection events occurring on any two detectors *i* and *j* (refs. [42,49]; see Supplementary Information). In Fig. 2h, we show the estimated probabilities for experimental and simulated distance-5 data, aggregated and averaged according to the different classes of pairs. In addition to the expected pairs, we also quantify how often detection pairs occur that are unexpected in a local depolarizing circuit model. Overall, the Pauli simulation systematically underpredicts these probabilities compared to experimental data, whereas the Pauli+ simulation is closer and predicts the presence of unexpected pairs, which we surmise are related to leakage and stray interactions. These errors can be especially harmful to the surface code because they can generate multiple detection events distantly separated in space or time, which a decoder might wrongly interpret as multiple independent component errors. We expect that mitigating leakage and stray interactions will become increasingly important as error rates decrease.

# Decoding and logical error probabilities

We next examine the logical performance of our surface code qubits. To infer the error-corrected logical measurement, the decoder requires a probability model for physical error events. This information may be expressed as an error hypergraph: detectors are vertices, physical error mechanisms are hyperedges connecting the detectors they trigger, and each hyperedge is assigned its corresponding error mechanism probability. We use a generalization of $p_{ij}$ to determine these probabilities[42,50].

Given the error hypergraph, we implement two different decoders: belief-matching, an efficient combination of belief propagation and minimum-weight perfect matching[51]; and tensor network decoding, a slow but accurate approximate maximum-likelihood decoder. The belief-matching decoder first runs belief propagation on the error hypergraph to update hyperedge error probabilities based on nearby detection events[51,52]. The updated error hypergraph is then decomposed into a pair of disjoint error graphs, one each for *X* and *Z* errors[31]. These graphs are decoded efficiently using minimum-weight perfect matching[53] to select a single probable set of errors.

By contrast, a maximum-likelihood decoder considers all possible sets of errors consistent with the detection events, splits them into two groups on the basis of whether they flip the logical measurement, and chooses the group with the greater total likelihood. The two likelihoods are each expressed as a tensor network contraction[51,54,55] that exhaustively sums the probabilities of all sets of errors within each group. We can contract the network approximately, and verify that the approximation converges. This yields a decoder that is nearly optimal given the hypergraph error priors, but is considerably slower. Further improvements could come from a more accurate prior, or by incorporating more fine-grained measurement information[47,56].

Figure 3 shows a comparison of the logical error performance of the distance-3 and distance-5 codes using the approximate maximum-likelihood decoder. As the *ZXXZ* variant of the surface code symmetrizes the $X$ and $Z$ bases, differences between the two bases' logical error per cycle are small and attributable to spatial variations in physical error rates. Thus, for visual clarity, we report logical error probabilities averaged between the $X$ and $Z$ basis; the full dataset may be found in the Supplementary Information. Note that we do not post-select on leakage or high-energy events to capture the effects of realistic non-idealities on logical performance. Over all 25 cycles of error correction, the distance-5 code realizes lower logical error probabilities $p_L$ than the average of the subset distance-3 codes.

**a**, Logical error probability $p_L$ versus cycle comparing distance-5 (blue) to distance-3 (pink: four separate quadrants, red: average), all averaged over $Z_L$ and $X_L$. Each individual data point represents 100,000 repetitions. Solid line: fit to experimental average, $t = 3$ to 25 (see main text). Dotted line: comparison to Pauli+ simulation. **b**, Logical fidelity $F = 1 - 2p_L$ versus cycle, semilog plot. The datapoints and fits are the experimental averages and fits from **a**. **c**, Summary of experimental progression comparing logical error per cycle $\varepsilon_d$ (specifically plotting $1 - \varepsilon_d$) between distance-3 and distance-5, for which system improvements lead to faster improvement for distance-5 (see main text). Each open circle is a comparison to a specific distance-3 code, and filled circles average over several distance-3 codes measured in the same session. Markers are coloured chronologically from light to dark. Typical $1\sigma$ statistical and fit uncertainty is 0.02%, smaller than the points.

We fit the logical fidelity $F = 1 - 2p_L$ to an exponential decay. We start the fit at $t = 3$ to avoid two phenomena that advantage the larger code: the lower detection probability during the first cycle relative to subsequent cycles (Fig. 2b,d), and the higher effective threshold caused by the confinement of errors to thin time slices in few-cycle experiments[31]. We obtain a logical error per cycle $\varepsilon_5 = (2.914 \pm 0.016)\%$ ($1\sigma$ statistical and fit uncertainty) for the distance-5 code, compared to an average of $\varepsilon_3 = (3.028 \pm 0.023)\%$ for the subset distance-3 codes, a relative error reduction of about 4%. When decoding with the faster belief-matching decoder, we fit a logical error per cycle of $(3.056 \pm 0.015)\%$ for the distance-5 code, compared to an average of $(3.118 \pm 0.025)\%$ for the distance-3 codes, a relative error reduction of about 2%. We note that the distance-5 logical error per cycle is slightly higher than those of two of the distance-3 codes individually, and that leakage accumulation may cause distance-5 performance to degrade faster than that of distance-3 as logical error probability approaches 50%.

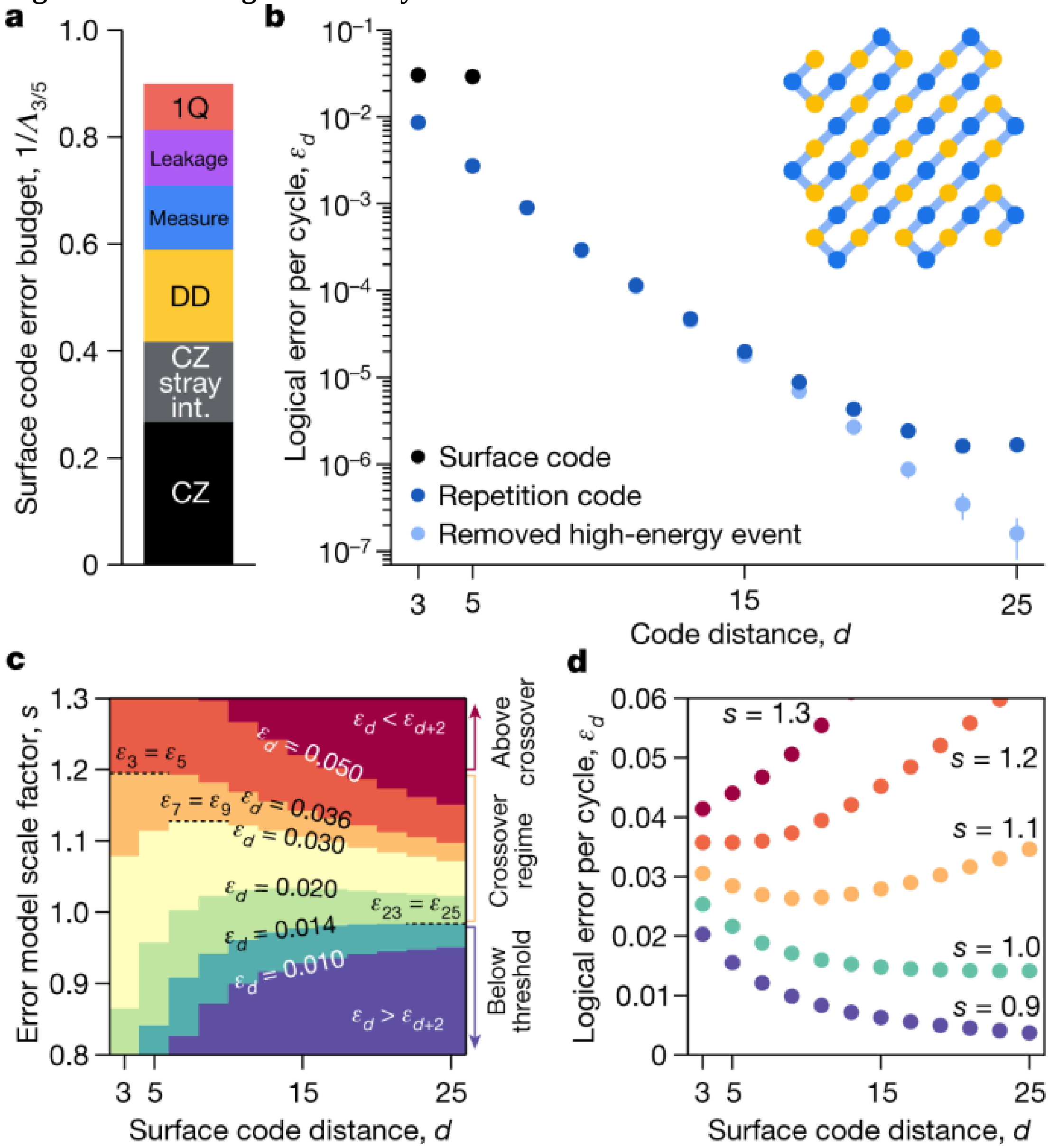
In principle, the logical performance of a distance-5 code should improve faster than that of a distance-3 code as physical error rates decrease[33]. Over time, we improved our physical error rates, for example by optimizing single- and two-qubit gates, measurement and data qubit idling (see Supplementary Information). In Fig. 3c, we show the corresponding performance progression of distance-5 and distance-3 codes. The larger code improved about twice as fast until finally overtaking the smaller code, validating the benefit of increased-distance protection in practice.

To understand the contributions of individual components to our logical error performance, we follow ref. 42 and simulate the distance-5 and distance-3 codes while varying the physical error rates of the various circuit components. As the logical-error-suppression factor

$$\Lambda_{d/(d+2)}=\varepsilon_d/\varepsilon_{d+2} \tag{1}$$

is approximately inversely proportional to the physical error rate, we can budget how much each physical error mechanism contributes to $1/\Lambda_{3/5}$ (as shown in Fig. 4a) to assess scaling. This error budget shows that CZ error and data qubit decoherence during measurement and reset are dominant contributors.

**Fig. 4: Towards algorithmically relevant error rates.**



**a**, Estimated error budget for the surface code, based on component errors (see Fig. 1c) and Pauli+ simulations. $\Lambda_{3/5} = \varepsilon_3/\varepsilon_5$. CZ, contributions from CZ error (excluding leakage and stray interactions). CZ stray int., CZ error from unwanted interactions. DD, dynamical decoupling (data qubit idle error during measurement and reset). Measure, measurement and reset error. Leakage, leakage during CZs and due to heating. 1Q, single-qubit gate error. **b**, Logical error for repetition codes. Inset: schematic of the distance-25 repetition code, using the same data and measure qubits as the distance-5 surface code. Smaller codes are subsampled from the same distance-25 data[42]. A high-energy event resulted in an apparent error floor around $10^{-6}$. After removing the instances nearby (light blue), error decreases more rapidly with code distance. The dataset has 50 cycles, $5 \times 10^5$ repetitions. We also plot the surface code error per cycle from Fig. 3b in black. **c**, Contour plot of simulated surface code logical error per cycle $\varepsilon_d$ as a function of code distance $d$ and a scale factor $s$ on the error model in Fig. 1c (Pauli simulation, $s = 1.0$ corresponds to the current device error model). **d**, Horizontal slices from **c**, each for a value of error-model scale factor $s$. $s = 1.3$ is above threshold (larger codes are worse), and $s = 1.2$ to 1.0 represent the crossover regime, for which progressively larger codes get better until a turnaround. $s = 0.9$ is below threshold (larger codes are better).

## Algorithmically relevant error rates

Even as known error sources are suppressed in future devices, new dominant error mechanisms may arise as lower logical error rates are realized. To test the behaviour of codes with substantially lower error rates, we use the bit-flip repetition code, a one-dimensional version of the surface code. The bit-flip repetition code does not correct for phase-flip errors and is thus unsuitable for quantum algorithms. However, correcting only bit-flip errors allows it to achieve much lower logical error probabilities.

Without post-selection, we achieve a logical error per cycle of $(1.7 \pm 0.3) \times 10^{-6}$ using a distance-25 repetition code decoded with minimum-weight perfect matching (Fig. 4b). We attribute many of these logical errors in the higher-distance codes to a high-energy impact, which can temporarily impart widespread correlated errors to the system[57]. These events may be identified by spikes in detection event counts[42], and such error mechanisms must be mitigated for scalable quantum error correction to succeed. In this case, there was one such event; after removing it (0.15% of trials), we observe a logical error per cycle of $(1.6 \pm 0.8) \times 10^{-7}$ (see Supplementary Information). The repetition code results demonstrate that low logical error rates are possible in a superconducting system, but finding and mitigating highly correlated errors such as cosmic ray impacts will be an important area of research moving forwards.

## Towards large-scale quantum error correction

To understand how our surface code results project forwards to future devices, we simulate the logical error performance of surface codes ranging from distance-3 to 25, while also scaling the physical error rates shown in Fig. 1c. For efficiency, the simulation considers only Pauli errors. Figure 4c,d illustrates the contours of this parameter space, which has three distinct regions. When the physical error rate is high (for example, the initial runs of our surface code in Fig. 3c), logical error probability increases with increasing system size ($\varepsilon_{d+2} > \varepsilon_d$). On the other hand, low physical error rates show the desired exponential suppression of logical error ($\varepsilon_{d+2} < \varepsilon_d$). This threshold behaviour can be subtle[58], and there exists a crossover regime in which, owing to finite-size effects, increasing system size initially suppresses the logical error per cycle before later increasing it. We believe our experiment lies in this regime.

Although our device is close to threshold, reaching algorithmically relevant logical error rates with manageable resources will require an error-suppression factor $\Lambda_{d/(d+2)} \gg 1$. On the basis of the error budget and simulations in Fig. 4, we estimate that component performance must improve by at least 20% to move below threshold, and substantially improve beyond that to achieve practical scaling. However, these projections rely on simplified models and must be validated experimentally, testing larger code sizes with longer durations to eventually realize the desired logical performance. This work demonstrates the first step in that process, suppressing logical errors by scaling a quantum error-correcting code—the foundation of a fault-tolerant quantum computer.

# Data availability

The data that support the findings of this study are available at https://doi.org/10.5281/zenodo.6804040.

# References


1. Shor, P. W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **52**, R2493 (1995).

   CAS ADS PubMed Google Scholar

2. Gottesman, D. *Stabilizer Codes and Quantum Error Correction*. PhD thesis, California Institute of Technology (1997).
3. Feynman, R. P. Simulating physics with computers. *Int. J. Theor. Phys.* **21**, 467–488 (1982).
4. Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **41**, 303–332 (1999).

   MathSciNet MATH ADS Google Scholar

5. Farhi, E. et al. A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. *Science* **292**, 472–475 (2001).

MathSciNet CAS MATH ADS PubMed Google Scholar

6. Biamonte, J. et al. Quantum machine learning. *Nature* **549**, 195–202 (2017).

CAS ADS PubMed Google Scholar

7. Lloyd, S. Universal quantum simulators. *Science* **273**, 1073–1078 (1996).

MathSciNet CAS MATH ADS PubMed Google Scholar

8. Aspuru-Guzik, A., Dutoi, A. D., Love, P. J. & Head-Gordon, M. Simulated quantum computation of molecular energies. *Science* **309**, 1704–1707 (2005).

CAS ADS PubMed Google Scholar

9. Reiher, M., Wiebe, N., Svore, K. M., Wecker, D. & Troyer, M. Elucidating reaction mechanisms on quantum computers. *Proc. Natl Acad. Sci. USA* **114**, 7555–7560 (2017).

CAS ADS PubMed PubMed Central Google Scholar

10. Gidney, C. & Ekera, M. How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum* **5**, 433 (2021).

Google Scholar

11. Kivlichan, I. D. et al. Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via trotterization. *Quantum* **4**, 296 (2020).

Google Scholar

12. Ballance, C., Harty, T., Linke, N., Sepiol, M. & Lucas, D. High-fidelity quantum logic gates using trapped-ion hyperfine qubits. *Phys. Rev. Lett.* **117**, 060504 (2016).

CAS ADS PubMed Google Scholar

13. Huang, W. et al. Fidelity benchmarks for two-qubit gates in silicon. *Nature* **569**, 532–536 (2019).

CAS ADS PubMed Google Scholar

14. Rol, M. et al. Fast, high-fidelity conditional-phase gate exploiting leakage interference in weakly anharmonic superconducting qubits. *Phys. Rev. Lett.* **123**, 120502 (2019).

CAS ADS PubMed Google Scholar

15. Jurcevic, P. et al. Demonstration of quantum volume 64 on a superconducting quantum computing system. *Quantum Sci. Technol.* **6**, 025020 (2021).

16. Foxen, B. et al. Demonstrating a continuous set of two-qubit gates for near-term quantum algorithms. *Phys. Rev. Lett.* **125**, 120504 (2020).

CAS ADS PubMed Google Scholar

17. Wu, Y. et al. Strong quantum computational advantage using a superconducting quantum processor. *Phys. Rev. Lett.* **127**, 180501 (2021).

CAS ADS PubMed Google Scholar

18. Knill, E., Laflamme, R. & Zurek, W. H. Resilient quantum computation. *Science* **279**, 342–345 (1998).

[CAS](#) [MATH](#) [ADS](#) [Google Scholar](#)

19. Aharonov, D. & Ben-Or, M. Fault-tolerant quantum computation with constant error rate. *SIAM J. Comput.* **38**, 1207–1282 (2008).
20. Egan, L. et al. Fault-tolerant control of an error-corrected qubit. *Nature* **598**, 281–286 (2021).

[CAS](#) [ADS](#) [PubMed](#) [Google Scholar](#)

21. Ryan-Anderson, C. et al. Realization of real-time fault-tolerant quantum error correction. *Phys. Rev. X* **11**, 041058 (2021).

[CAS](#) [Google Scholar](#)

22. Abobeih, M. et al. Fault-tolerant operation of a logical qubit in a diamond quantum processor. *Nature* **606**, 884–889 (2022).

[CAS](#) [ADS](#) [PubMed](#) [PubMed Central](#) [Google Scholar](#)

23. Sundaresan, N. et al. Matching and maximum likelihood decoding of a multi-round subsystem quantum error correction experiment. Preprint at https://arXiv.org/abs/2203.07205 (2022).
24. Krinner, S. et al. Realizing repeated quantum error correction in a distance-three surface code. *Nature* **605**, 669–674 (2022).

[CAS](#) [ADS](#) [PubMed](#) [Google Scholar](#)

25. Zhao, Y. et al. Realization of an error-correcting surface code with superconducting qubits. *Phys. Rev. Lett.* **129**, 030501 (2022).

[CAS](#) [ADS](#) [PubMed](#) [Google Scholar](#)

26. Ofek, N. et al. Extending the lifetime of a quantum bit with error correction in superconducting circuits. *Nature* **536**, 441–445 (2016).

CAS ADS PubMed Google Scholar

27. Flühmann, C. et al. Encoding a qubit in a trapped-ion mechanical oscillator. *Nature* **566**, 513–517 (2019).

ADS PubMed Google Scholar

28. Campagne-Ibarcq, P. et al. Quantum error correction of a qubit encoded in grid states of an oscillator. *Nature* **584**, 368–372 (2020).

CAS PubMed Google Scholar

29. Grimm, A. et al. Stabilization and operation of a Kerr-cat qubit. *Nature* **584**, 205–209 (2020).

CAS ADS PubMed Google Scholar

30. Kitaev, A. Y. Fault-tolerant quantum computation by anyons. *Ann. Phys.* **303**, 2–30 (2003).

MathSciNet CAS MATH ADS Google Scholar

31. Dennis, E., Kitaev, A., Landahl, A. & Preskill, J. Topological quantum memory. *J. Math. Phys.* **43**, 4452–4505 (2002).

MathSciNet MATH ADS Google Scholar

32. Raussendorf, R. & Harrington, J. Fault-tolerant quantum computation with high threshold in two dimensions. *Phys. Rev. Lett.* **98**, 190504 (2007).

ADS PubMed Google Scholar

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

ADS Google Scholar

34. Satzinger, K. et al. Realizing topologically ordered states on a quantum processor. *Science* **374**, 1237–1241 (2021).

CAS ADS PubMed Google Scholar

35. Horsman, C., Fowler, A. G., Devitt, S. & Meter, R. V. Surface code quantum computing by lattice surgery. *New J. Phys.* **14**, 123011 (2012).

MathSciNet MATH ADS Google Scholar

36. Fowler, A. G. & Gidney, C. Low overhead quantum computation using lattice surgery. Preprint at https://arXiv.org/abs/1808.06709 (2018).

37. Litinski, D. A game of surface codes: large-scale quantum computing with lattice surgery. *Quantum* **3**, 128 (2019).

Google Scholar

38. Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).

CAS ADS PubMed Google Scholar

39. Koch, J. et al. Charge-insensitive qubit design derived from the Cooper pair box. *Phys. Rev. A* **76**, 042319 (2007).

ADS Google Scholar

40. Neill, C. *A Path towards Quantum Supremacy with Superconducting Qubits*. PhD thesis, Univ. California Santa Barbara (2017).

41. Yan, F. et al. Tunable coupling scheme for implementing high-fidelity two-qubit gates. *Phys. Rev. Appl.* **10**, 054062 (2018).

CAS ADS Google Scholar

42. Chen, Z. et al. Exponential suppression of bit or phase errors with cyclic error correction. *Nature* **595**, 383–387 (2021).

Google Scholar

43. Kelly, J. et al. Scalable in situ qubit calibration during repetitive error detection. *Phys. Rev. A* **94**, 032321 (2016).

ADS Google Scholar

44. Wen, X.-G. Quantum orders in an exact soluble model. *Phys. Rev. Lett.* **90**, 016803 (2003).

ADS PubMed Google Scholar

45. Bonilla Ataides, J. P., Tuckett, D. K., Bartlett, S. D., Flammia, S. T. & Brown, B. J. The XZZX surface code. *Nat. Commun.* **12**, 2172 (2021).

CAS ADS PubMed PubMed Central Google Scholar

46. Aliferis, P. & Terhal, B. M. Fault-tolerant quantum computation for local leakage faults. *Quantum Inf. Comput.* **7**, 139–156 (2007).

MathSciNet MATH Google Scholar

47. Suchara, M., Cross, A. W. & Gambetta, J. M. Leakage suppression in the toric code. *Proc. 2015 IEEE International Symposium on Information Theory (ISIT)* 1119–1123 (2015).

48. McEwen, M. et al. Removing leakage-induced correlated errors in superconducting quantum error correction. *Nat. Commun.* **12**, 1761 (2021).

CAS ADS PubMed PubMed Central Google Scholar

49. Spitz, S. T., Tarasinski, B., Beenakker, C. W. & O'Brien, T. E. Adaptive weight estimator for quantum error correction in a time-dependent environment. *Adv. Quantum Technol.* **1**, 1800012 (2018).

Google Scholar

50. Chen, E. H. et al. Calibrated decoders for experimental quantum error correction. *Phys. Rev. Lett.* **128**, 110504 (2022).

CAS ADS PubMed Google Scholar

51. Higgott, O., Bohdanowicz, T. C., Kubica, A., Flammia, S. T. & Campbell, E. T. Fragile boundaries of tailored surface codes and improved decoding of circuit-level noise. Preprint at https://arXiv.org/abs/2203.04948 (2022).

52. Criger, B. & Ashraf, I. Multi-path summation for decoding 2D topological codes. *Quantum* **2**, 102 (2018).

Google Scholar

53. Fowler, A. G., Whiteside, A. C. & Hollenberg, L. C. Towards practical classical processing for the surface code. *Phys. Rev. Lett.* **108**, 180501 (2012).

ADS PubMed Google Scholar

54. Bravyi, S., Suchara, M. & Vargo, A. Efficient algorithms for maximum likelihood decoding in the surface code. *Phys. Rev. A* **90**, 032326 (2014).

ADS Google Scholar

55. Chubb, C. T. & Flammia, S. T. Statistical mechanical models for quantum codes with correlated noise. *Ann. Inst. Henri Poincaré D* **8**, 269–321 (2021).

MathSciNet MATH Google Scholar

56. Pattison, C. A., Beverland, M. E., da Silva, M. P. & Delfosse, N. Improved quantum error correction using soft information. Preprint at https://arXiv.org/abs/2107.13589 (2021).

57. McEwen, M. et al. Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits. *Nat. Phys.* **18**, 107–111 (2022).

CAS Google Scholar

58. Stephens, A. M. Fault-tolerant thresholds for quantum error correction with the surface code. *Phys. Rev. A* **89**, 022321 (2014).

ADS Google Scholar

59. Emerson, J., Alicki, R. & Życzkowski, K. Scalable noise estimation with random unitary operators. *J. Opt. B* **7**, S347 (2005).

MathSciNet ADS Google Scholar

# Brighter, cutting-edge OLED technology wins our Innovation Award 2024


Brighter, cutting-edge OLED technology wins our Innovation Award 2024© Apple

OK so Tandem OLED car displays have been a thing since 2019; but 2024 is the year the tech has come to devices you can watch movies on. Of those devices, the Porsche Design Honor Magic6 RSR smartphone and Apple iPad Pro M4 launched globally within days of one another. But the latter has a much larger screen and the former hasn't even been submitted for review, so it's the Apple implementation that gets our Award.

In short, a Tandem OLED panel has two OLED layers sandwiched together, rather than a single layer as is the norm for OLED displays. These two OLED layers can work together (in tandem, if you will) to massively boost brightness; but Tandem OLED panels can also be thinner, more power-efficient and more durable. Apple's implementation of Tandem OLED (or 'Ultra Retina XDR') in the iPad Pro M4 is stunning.

Mini LED technology, which is what the previous iPad Pro used, is generally able to go much brighter than OLED, but the new Tandem OLED iPad Pro matches its predecessor for peak and full-screen HDR brightness, with figures of 1600 nits and 1000 nits respectively. It's this full-screen figure of 1000 nits that is most impressive. We have become used to next-gen MLA OLED and QD-OLED TVs with peak brightness figure claims in the thousands, but those are achievable only with tiny highlights. Full-screen brightness on even the latest, brightest OLED TVs measures in only the low hundreds.

But why is the iPad's Tandem OLED display matching the previous model's brightness figure so exciting? Primarily because that brightness is now combined with perfect OLED blacks and pixel-by-pixel contrast control. Movies pop from the screen in a way that the previous model can't match. The switch to OLED has made the display more responsive, too, which makes motion smoother, and there's a cinematic richness to the delivery.

The Tandem OLED display appears to have enabled Apple to reduce the thickness of the new iPad Pro which, at 5.1mm, is the thinnest product Apple has ever produced. This might be down to the thinness of the Tandem OLED panel, but it's also possible that the panel's power efficiency has allowed Apple to fit slimmer batteries.

So, are Tandem OLED TVs on the way? Not immediately, but it's easy to see why a Tandem OLED TV would appeal: who wouldn't want an OLED TV that could go brighter, draw less power from the grid and be even thinner than current models? Whether Tandem OLED makes it to TVs or not, there's no denying how innovative it is – or how impressive it is in the new iPad Pro.

# What is tandem OLED screen tech? How does it work?

## Getting inside the new iPad Pro's display




(Image credit: Apple)

As rumoured, Apple announced its first iPad with an OLED screen in early May. But the latest iPad Pro doesn't just have any old OLED screen – it uses tandem OLED technology in what Apple calls an ultra retina XDR display. Typically Apple, it claims this is "the world's most advanced display".

We're here to unpack exactly what that means. What does tandem OLED technology entail? And it might surprise you to know that Apple wasn't the first to use it...

## What is tandem OLED technology?

(Image credit: Apple)
As the name suggests, tandem OLED uses two OLED panels 'in tandem' to combine the light from both. By layering two lots of OLED pixels on top of each other, you get extra brightness, which addresses the criticism that's most often levelled at OLED screens: that they're not bright enough, especially compared to LED and LCD screens. That's why we've seen such a glut of brightness-boosting technologies come to OLED TVs lately, like heatsinks, Micro Lens Array (MLA) and Quantum Dots.
Tandem OLED technology was first developed by LG Display in 2019 for use in automobiles to make heads-up displays more durable, so they could last the lifespan of the car. The iPad Pro is the first consumer tablet to feature the tech.

## How does it work?

(Image credit: Apple)
Because OLED pixels generate their own light, having twice as many generates much more light. And not just in peak highlights, but across the whole screen.

This part is key. It means the picture is brighter overall, rather than just one or two key details really popping. Apple says the new iPad Pro can reach 1000 nits of full-screen brightness for SDR and HDR.
It goes even brighter for smaller highlights – Apple claims 16000 nits for peak HDR brightness. Admittedly that's not as bright as the latest OLED TVs (which reach 3000-4000 nits), but it's still pretty impressive.

It's worth noting that these figures are the same that Apple claimed for its previous Mini LED 12.9-inch iPad Pro. It's undoubtedly an achievement to hit these stats with an OLED panel and all the advantages that brings, but it won't be so bright you'll have to wear sunglasses.

## What are the advantages of tandem OLED?

(Image credit: Apple)

Increased brightness is the most obvious one, but there are other benefits too.

For starters, higher brightness doesn't just mean that the picture has more 'pop'. It reveals more dark detail, so scenes in murky environments or with lots of shadows become clearer. And it will make the picture easier to see in bright sunlight. It also enables sub-millisecond control over both the colour and luminance of each pixel, resulting in greater responsiveness during motion and less blooming.

One downside to OLED screens is that they're susceptible to burn-in. This is where one image stays on-screen so long that it becomes indelibly 'burned' into the screen – think the BBC logo if you watch too much of the BBC News channel. With tandem OLED, each layer actually operates at a less bright level than a single array of pixels would have to, making them more efficient in terms of energy and heat use. Which should make it less likely you have the BBC News logo permanently etched into the corner of your screen, no matter how much Jeremy Bowen you watch.

This higher efficiency should also help the screen last longer. Chinese display maker BOE claims its tandem OLED technology increases an OLED screen's lifespan by six times. And with a 40 per cent increase in efficiency, we should see a longer battery life per charge as well.

Tandem OLED has all the benefits of standard OLED technology too, like absolute black levels, great colour reproduction, slimmer dimensions and ultra-wide viewing angles.

## What's the downside of tandem OLED?

(Image credit: Apple)
One word: price. As you can imagine, increasing the number of pixels comes at a cost, and in the case of the latest iPad Pro, it's a hefty one. Prices start at £999 / $999 / AU$1699 for the 11-inch and £1299 / $1299 / AU$2199 for the 13-inch – that's £50 / $200 / AU$300 more expensive than the previous 12.9-inch model, and £200 / $200 / AU$300 more than the previous 11-inch iPad Pro. But then the new models do have other extras, like the brand-new M4 processor and slimline design.

Hopefully we'll see prices come down as the tech becomes more widespread.

## Where else is tandem OLED tech used?

(Image credit: Honor)
Apple isn't the only company using tandem OLED screen tech. The Honor Magic 6 Ultimate launched in March 2024 as the first device to use tandem OLED screen tech made by Chinese display maker BOE. But it hasn't launched in the US or UK, and there's no word on whether it ever will.

Tandem OLED tech has been used previously in cars, and according to specialist site *OLED Info*, it will also be used in laptops, tablets and monitors where the UI is more susceptible to burn in.
One of OLED's most prominent uses is in TVs, but so far there's been no mention of TVs using tandem OLED tech. That's presumably due to the cost and complexity of developing the technology for these much larger displays. But as the costs come down, you never know...

With Apple ordering millions of tandem OLED displays, expect other manufacturers to follow suit, which should make the technology much more mainstream and possibly used in a wider range of devices. Looks like we'll be hearing a lot more about tandem OLED in future.

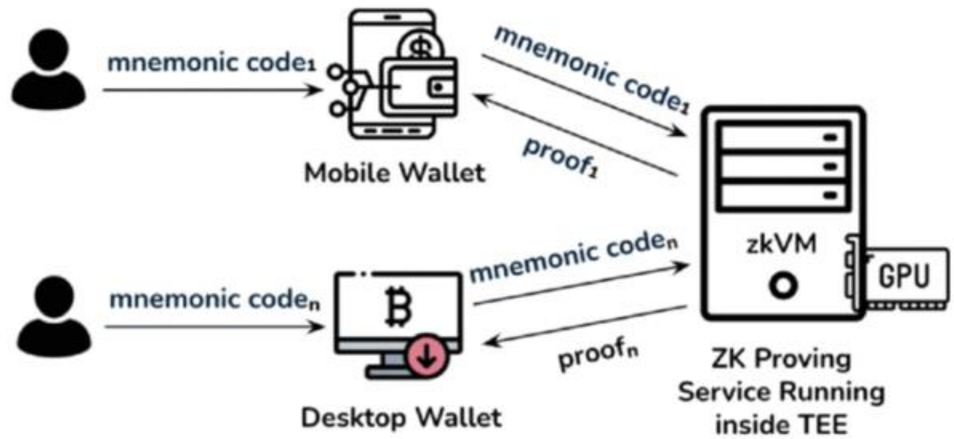
# Hash-based zero-knowledge tech can quantum-proof Ethereum — XinXin Fan

Dr. XinXin Fan, the head of cryptography at IoTeX, recently co-authored a research paper titled *Enabling a Smooth Migration Towards Post-Quantum Security for Ethereum.* The research paper received a Best Paper award from the 2024 International Conference for Blockchain and argued that hash-based zero-knowledge technology is the most user-friendly way to quantum-proof the Ethereum network and other similar cryptographic systems.
In an interview with Cointelegraph, Dr. Fan explained that the elliptical curve digital signature algorithms (ECDSA) employed in current blockchain systems to sign transactions are quantum-vulnerable. However, this vulnerability can be addressed by attaching a hash-based zero-knowledge proof — such as a zero-knowledge scalable transparent argument of knowledge (ZK-Stark) — to each transaction.

The researcher said this method also ensures the smoothest transition for users — avoiding the complexity of other proposed quantum-resistance methods. "The way we are implementing this allows the user to use their current wallet, but we attach each transaction with a zero-knowledge proof that is quantum-safe," Dr. Fan said.

"We need to consider both the security aspect and also the usability aspect," Dr. Fan continued. The researcher stressed that balancing user experience with security needs was key to ensuring a timely migration to post-quantum standards.

Fig. 6.




A ZK Proving Service Running inside a Trusted Execution Environment (TEE)

Full size image ›

Cryptography, Cybersecurity, Quantum Computing, zk-STARK© Cointelegraph

# The quantum scare of 2024

A smooth transition to post-quantum security for end users is paramount, as the National Institute of Standards and Technology (NIST) recently published the first hard deadline for legacy systems to migrate to post-quantum signature standards — advising institutions to adopt quantum-resistant measures before 2035.

In Oct. 2024, a report from the South Morning China Post claimed that researchers at Shanghai University successfully breached cryptographic algorithms using a quantum computer.

However, an analysis by YouTuber "Mental Outlaw" later revealed that the quantum computer used in the experiment only broke a 22-bit key. For context, modern encryption standards use keys between 2048 and 4096 bits — meaning that quantum computers have not yet cracked encryption standards.

Other researchers also agreed the threat posed by quantum computers is exaggerated at this point due to the stark divergence between the current ability of quantum computers to factor numbers and the length of modern encryption keys.

# How Schrödinger's cat could make quantum computers work better

A quantum bit inspired by Schrödinger’s cat has managed to resist making errors for an unusually long time in a quantum computing experiment. This may make it a promising building block for more reliable quantum computers in the future.

Researchers have long believed that quantum computers can solve problems that are impossible for conventional computers, but there have been very few demonstrations of such capability so far. This is because quantum computers tend to make errors as they compute, but building a quantum computer powerful enough to correct its own errors is technically difficult.

Zaki Leghtas at the École Normale Supérieure in France and his colleagues, in collaboration with the quantum computing start-up Alice & Bob, have now created a quantum bit, or qubit, that avoids making a particularly common type of error for the unprecedentedly long time of 10 seconds.

Read more

Quantum computers are revealing an unexpected new theory of reality

They made their qubit by trapping light in a small hole on a chip filled with tiny circuits made from perfectly conducting – or “superconducting” – wires. The light could oscillate back and forth in two different ways inside the hole. But instead of forcing it to oscillate one way only, the team made it do both – creating a quantum superposition similar to the one involving the cat in Erwin Schrödinger’s famous thought experiment. This type of qubit is, accordingly, called a "cat qubit".

Leghtas says that for more than 10 years, physicists have theorised that cat qubits should be particularly unlikely to make so-called bit-flip errors, which are equivalent to the digital 0s in a conventional computer spontaneously becoming 1s, or vice versa. But demonstrating that cat qubits in the lab are so resistant to bit-flips is not straightforward.

For several years, he says he and his colleagues were detecting bit-flip errors in their cat qubit every few milliseconds. Recently, however, they realised that many of these errors were actually induced by the way they were measuring the cat qubit's states. Redesigning that process led them to a major technical leap: their cat qubit can now function for 10 seconds without bit-flipping, which is 10,000 times longer than in any past experiment.

The researchers have only built one cat qubit with this property so far, but building more of them could be a step towards reliably useful quantum computers. This is because a computer built with the cat qubits could devote more of them to computation, rather than reserving just a few for computation and using the others to correct bit-flip errors in the computational qubits. Leghtas says that using these cat qubits could cut the number of qubits needed for error-correction by about 10 times compared with other qubit designs involving superconducting circuits.

Christian Andersen at the Delft University of Technology in the Netherlands says that while 10 seconds in between bit-flips is a very long time for a qubit, it is not the only qubit property that matters. There is a trade-off between making the cat qubit more resilient to bit-flip errors and having it inadvertently become more prone to other kinds of errors. Future studies will have to find the most practical way to deal with that, he says.

"This is really cool, it's nice progress, but there are also many challenges," says Andersen.

Journal reference:

*Nature* DOI: 10.1038/s41586-024-07294-3

# Will we ever have quantum laptops?

Roughly 80 years ago, the world was at war. Under a shroud of secrecy, scientists in the U.K., Germany and the U.S. were creating the first electronic computers. These computers filled rooms, demanded vast quantities of electricity and enabled previously impossible calculations. Few of the people involved could have imagined that decades later, computers orders of magnitude more powerful would fit in a backpack — yet that's exactly what happened.

So, as we sit on the threshold of genuinely useful quantum computing, could we ever see quantum laptops? "I think it's possible," Mario Gely, a quantum computing researcher at the University of Oxford, told Live Science. "It's highly speculative, but I can't think of a fundamental reason why a quantum laptop would not be possible."

## Scaling up qubit number

Before scientists can make a quantum laptop, they need to make a useful quantum computer, period. Questions remain over how many qubits — quantum equivalents to digital bits — are needed to create a genuinely useful quantum computer, or one that can solve a range of useful, real-world problems that elude the best superclassical computers. But it's definitely higher than is currently possible.

Stephen Bartlett, a theoretical quantum physicist and director of the University of Sydney's Nano Institute, thinks we could see genuinely useful quantum computers by the end of this decade. "There's a bunch of open scientific challenges, which makes that pathway a bit murky, but we're getting close," Bartlett told Live Science.

**Related: What is the largest known prime number?**

For instance, newly developed quantum charge-coupled device (QCCD) architecture could be used to make two-dimensional arrays of qubits rather than one-dimensional ones — which would increase the density, and potentially the number, of qubits.

## Reducing the errors in quantum computers

But scaling brings another challenge in building a miniature quantum computer: correcting errors, or "noise." "Our existing quantum components are noisy, so we need error correction, and that necessitates a large amount of redundancy," Bartlett said. Scientists need to either reduce errors or build error correction into quantum computers, and that requires even more qubits. Many scientists are trying to solve this problem.

For example, a December 2023 study tried to reduce errors by building a quantum computer with "logical qubits." In another paper, published in April 2024, scientists designed a new type of qubit that behaved like an error-correcting logical qubit. Some scientists have even proposed using photons (light particles) as qubits, including another study that used a laser pulse. According to Peter van Loock, a professor of theoretical quantum optics at Johannes Gutenberg University of Mainz in Germany and co-author of the study, this approach has an "inherent capacity to correct errors".

So if, within a decade or two, powerful and useful quantum computers exist, the next step would be miniaturization.

## Choosing different types of qubits

But to get really small, quantum computers may need to focus on a different type of qubit than is currently popular. Some of the most advanced quantum computers today — such as those made by IBM and Google — rely on quantum processing units filled with superconducting qubits. But the first quantum laptop probably won't use this technology.

That's because, by their nature, superconducting qubits must be cooled to a fraction above absolute zero — around 20 millikelvin — and that requires filling a room with dilution refrigerators. And companies like IBM aren't trying to get around this size constraint. For example, IBM's current quantum computing roadmap sets out goals that include a 2,000-qubit quantum computer by 2033 — which would fill many rooms rather than one.

Quantum laptops may instead rely on trapped ion qubits, charged particles that exist in multiple states at once and that are suspended using electromagnetic fields, Bartlett and Gely explained. Although trapped ion systems work at room temperature and don't rely on room-sized refrigerators, the lasers they use are gigantic.

"At the moment, our laser system occupies approximately a cubic meter [35 cubic feet]," Gely said. "If we assume that ion traps are the future, then we need the lasers to become smaller."

And lasers must not only shrink but also become more advanced. Current systems are geared to constrain 100 ions. "How many qubits you can control with this volume of laser equipment is unclear," Gely said. "You can control more qubits than we have today, but certainly not the millions of qubits of a fully fledged quantum computer."

However, two recent advances could help with miniaturization. First, future QCCDs could aid miniaturization by increasing qubit density. Second, in July, Stanford researchers created titanium-sapphire lasers that are 10,000 times smaller than the ones they replace.

**Related: How does a secure phone line work?**

# Miniaturization efforts will ramp up

Right now, scientists are focused on making quantum computers more powerful, not on shrinking them. "The drive for miniaturization is not as strong at the moment as the drive for performance, and that mimics the early days of conventional computers when we had mainframes," Bartlett said. "People thought of the most powerful computers as taking up a building. And you know, why would anyone seriously consider carrying one around in your backpack?"

The history of computers suggests quantum computers will roll out first for industrial, military and government applications before shifting to consumers. The apocryphal 1943 quote from Thomas Watson Sr. that there would be a "world market for maybe five computers" springs to mind.

Of course, the world market for PCs and laptops is immense, so could there ever be a similar explosion of demand for quantum PCs and laptops? "The question I always get in my quantum computing classes is, you know, 'When can I play Doom on a quantum computer?'" Bartlett said. "But why would you want to when you can play Doom perfectly well on your computer today?"

Instead, Bartlett suggested there might be "quantum personal apps like finance or something niche around information security" — but the truth is, nobody knows. Gely made the alternative suggestion of a quantum processor sitting alongside a classical processor. "It could be like you have a graphics card, but it would only be useful for certain tasks," Gely said.

It's not yet clear that quantum laptops would be useful for consumers. What experts can say with a high level of confidence is that all of the hardware obstacles — scaling the number of qubits, correcting errors cand miniaturizing components — can be overcome. And yet, a future quantum laptop probably won't play Doom.

# The Holy Grail of Quantum Machines May Finally Be Near

- Quantum computers require reliable qubits, but those can be hard to come by and are limited in their error correction abilities.
- A new study from the University of Sydney has announced the development of a 2D error-correction architecture that could spot quantum errors using fewer qubits, thereby making them more efficient.
- This efficiency could lead to more compact quantum hard drives that can encode quantum information more reliably.

The promise of quantum computers has *not* been overstated. By using the incredible calculating properties of qubits—the quantum equivalent of bits—these machines could speed up medical breakthroughs, discover exotic materials, create unbreakable cybersecurity, and even help solve climate change. For all the amazing advancements made possible by the classical number-crunching of ones and zeroes, quantum computers would elevate human innovation into a whole new era.

There's just one problem: qubits have a nasty habit of producing errors. This side effect is understandable when you understand what it takes to sustain these little capsules of computation in the first place (near absolute temperatures, superconducting magnets, etc...). Eventually, qubits will experience decoherence if they experience what's known as "noise"—created by any temperature fluctuation or electromagnetic interference—causing them to producing errors. And if a qubit can't be held in the quantum state of superposition where it can be both one and zero *at the same time*, then it's basically just a classical bit.

These errors are one of the primary limiting factors of quantum computers. By one estimate, one error occurs every 1,000 operations, and error rates need to be something like one in a trillion for quantum computing to *really* take off. Thankfully, two quantum information theorists from the University of Sydney Nano Institute have developed new architecture capable of suppressing errors in a quantum system using fewer qubits. This breakthrough not only makes quantum information storage more reliable, but it brings the world one step closer to the long sought-after quantum hard drive. The results of the study were published in the journal *Nature Communications*.

"There remain significant barriers to overcome in the development of a universal quantum computer," University of Sydney's Dominic Williamson, a co-author of the study, said in a press statement. "One of the biggest is the fact we need to use most of the qubits—quantum switches at the heart of these machines—to suppress the errors that emerge as a matter of course within the technology. Our proposed quantum architecture will require fewer qubits to suppress more errors, liberating more for useful quantum processing."

This new, efficient architecture takes error correction to a whole new dimension—literally. Usually, 3D error-correction can only mitigate errors along a single line of qubits. But this new technique, according to the paper, uses a 3D lattice with "topological codes with optimal scaling code parameters" that essentially allow for error correction across two dimensions within the 3D structure. The single-dimension nature of previous error correction methods limited the system, but this new architecture could push forward a new era of quantum machines, including quantum hard drives.

“This advancement could help transform the way quantum computers are built and operated, making them more accessible and practical for a wide range of applications, from cryptography to complex simulations of quantum many-body systems,” Stephen Bartlett, director of the University of Sydney Nano Institute, said in a press statement.

Of course, creating this kind of lab-based quantum error correction is one thing—scaling such a breakthrough is another challenge entirely. But this new step toward 2D error correction brings humanity undeniably closer to realizing its quantum potential.