Q1 2024 | Network Edition

IBM **Quantum**

The Coherence Times

A Quarterly Readout

The Year of Software

Our quarterly leadership update

A note from Jay

Hi all, and welcome to the Quarter 1 2024 edition of the Coherence Times: *The year of software*.

At the 2023 IBM Quantum™ Summit, we announced that we'd entered a new era in the history of quantum computing: the era of utility. In this new era, quantum computers are no longer just research tools used to advance the field of quantum itself—they're real computational devices that developers are using to start doing real, useful work in their respective domains. But succeeding in this new era isn't just about capable hardware—it's about performant software. And that's where we must win in 2024.

Back in the Q3 2023 Coherence Times, I told you all about our vision for the first stable Qiskit® SDK release, Qiskit 1.0. And soon, we'll be bringing that vision to life. With Qiskit 1.0, we will offer the reliability and performance that users require to do utility-scale work on quantum computers and run experiments on the order of 100 qubits and 1000s of gates. We rolled out Qiskit 1.0 to users in February, with a public media release later this year.

But the Qiskit SDK is just one facet of our software strategy. Eventually, we hope that the word "Qiskit" will become synonymous with quantum computing software for the platform. We want it to be a super-performant core and associated library, built both by IBM® and our open source contributors. It will cover the entire platform (there will be vertical software above for applications), allowing technical users to explore quantum computing at the level of hand-coding the gates of a circuit while also allowing higher-level users to employ circuit libraries, patterns, and eventually quantum functions to run code in a heterogeneous architecture of classical and quantum computing.

Combined with our utility-scale hardware, Qiskit will be the tool set and ecosystem that allow our users to succeed in the era of utility. Or, as we've been saying, "Qiskit + Systems = Work."

This has been a long time coming, and it's thanks to the effort of you all that we can turn this page in our software mission—over 100 releases over the past 7 years, evolving the SDK from a first-of-its-kind research tool to a mature Python package with a Rust core for running performant, large-scale quantum experiments. Now, let's focus our sights on making Qiskit synonymous with high-performance quantum computing software here at IBM.

Logical gates and magic states

Earlier this year, we published a paper that serves as another crucial demonstration on the way toward error-corrected quantum computing. We used dynamic circuits to demonstrate a key ingredient for running logical gates: we created a magic state. And the encoded magic state performed even better than it would have had we naively coded it directly into the physical qubits.

The future of useful quantum computing relies on error correction, or encoding quantum information into more bits than we'd otherwise need to run the calculation and using those extra qubits to find and correct errors. Once we've encoded that information, we still need to manipulate it with quantum operations, which we do with logical gates. But running logical gates is really resource intensive.

Past work from our own Sergey Bravyi and Alexei Kitaev from Caltech proved that we could access the full power of quantum computing with Clifford gates—the basic set of gates we can efficiently simulate classically, like X, Y, Z, H, and CNOT—plus the ability to prepare certain error-prone states beyond what Clifford gates can access. Their method found that for certain states, they could apply a sequence of Clifford operations to distill these error-prone states into "magic states" that they can use for computation. The Clifford gates, plus the distilled magic states, allow you to run any quantum computation. In our new paper, we demonstrated a new procedure to encode magic states onto four qubits of the 27-qubit IBM Quantum Falcon processor on the *ibm_peekskill* system. Our scheme focused on starting with better states prior to the distillation process. Then, to prepare the magic states, we employed some of the important dynamic circuit techniques we introduced last year. We performed mid-circuit measurement and fed the value forward in the circuit, steering the state closer to the required magic state. By exploiting dynamic circuits, we were able to improve yield beyond what would otherwise be possible if we did not have access to these capabilities. By successfully encoding magic states into an error-correcting code, we produced better magic states than what would have otherwise been possible using raw qubits.

Ultimately, error correction is a central part of our development roadmap at IBM. Last year, we extended our roadmap to 2033, showing how we'll develop new couplers required to implement the error-correcting code we announced last year. And combined with our work on encoding magic states, as well as our efforts to improve the quality of our systems, we're painting the full picture of how we'll run error-corrected operations on those systems.

Introducing Qiskit 1.0

After 7 years of development, with over 8,000 commits from more than 500 contributors, Qiskit version 1.0 was released on February 15, 2024.

If you are a user of Qiskit, or someone who develops software projects that depend on Qiskit, there are some important things you should be aware of when upgrading to 1.0. →

01
Upgrading to 1.0 requires a fresh Python environment

Because of the complexities of the new packaging architecture for 1.0, if you are an existing user of Qiskit you will NOT be able to directly upgrade using `pip install -U qiskit`. Instead, you will need to create a new Python environment and install Qiskit from scratch. Installing Qiskit in your existing environment risks breaking your environment and causing internal issues that are difficult to debug. If you are having issues installing Qiskit 1.0, you can check out the troubleshooting section of the **migration guide**, available on the IBM Quantum Platfom and GitHub.

Note: If you are a developer who builds other software projects that depend on Qiskit, there are additional considerations you should be aware of when updating, also detailed in the ***migration guide****.*

02
The metapackage is no more!

This isn't exactly new information, as the Qiskit metapackage architecture was formally removed in Qiskit 0.44. However, it is worth remembering that a major difference between installing most pre-1.0 versions and installing 1.0 is that `pip install qiskit` will install only the core Qiskit package. You will no longer also get packages such as Qiskit Aer, IBM Q Provider, or the application modules installed by default. You will need to install them separately if you plan on using them.

Introducing Qiskit 1.0

03

A new release cycle and stability policy

The release of Qiskit 1.0 comes with a renewed commitment to stability and reliability, reflected in a new release cycle following **Semantic Versioning**. Prior to 1.0, Qiskit minor releases happened roughly once every quarter and there was no simultaneous support for different versions (so once a new minor version of Qiskit was released, support immediately ended for the previous release). With the release of 1.0, minor releases are still scheduled to happen on a quarterly basis, as well as new major releases no less than 12 months after the previous major release. In addition, each new major version has a guaranteed 18-month lifecycle, meaning support will still be available for a release up to 6 months after the release of the next version. Lastly, any deprecations introduced in a minor version will not be removed until the next major version is released.

You can find more information about Qiskit versioning in the documentation here: **https://docs.quantum.ibm.com/start/install#qiskit-versioning**.

04

Qiskit 1.0 is more performant

One of the key focuses for 1.0 and beyond is the attention to performance, both in terms of speed and memory usage. Qiskit 1.0 will see a 55% decrease in memory usage compared to Qiskit 0.39, as well as 16x faster binding and transpiling compared to Qiskit 0.33. Qiskit 1.0 is designed to scale with our hardware roadmap to be capable of constructing and transpiling circuits with increasing numbers of qubits and gate operations. Qiskit 1.0 will also feature OpenQASM3 native support, setting the standard for the implementation of the industry's lingua franca for writing and running quantum circuits. Check out the release notes for more information about the latest features released in Qiskit 1.0.

We announced our responsible quantum principles alongside a new WEF report

What it is: The World Economic Forum (WEF) released its Quantum Economy Blueprint, a report analyzing regional and national quantum development strategies. Alongside it, we publicly announced our own responsible quantum principles.

Why it matters: IBM is committed to ushering in the era of responsible quantum computing. As a leader in the field, we want to nurture a worldwide quantum computing ecosystem and proactively mitigate potentially undesired consequences of the technology. We were central participants in drafting the WEF's Quantum Computing Governance Principles—guidelines for the design and adoption of quantum technologies. The Quantum Economy Blueprint builds on that framework with strategies to incentivize the development of new quantum technologies with an emphasis on the common good.

SandboxAQ bought Good Chemistry

SandboxAQ acquired computational chemistry startup Good Chemistry

What it is: SandboxAQ acquired computational chemistry startup Good Chemistry to strengthen their position as providers of powerful simulation software. The announcement focused on the potential of Good Chemistry's QEMIST Cloud and Tangelo software to help accelerate material science breakthroughs in the pharmaceutical, energy, and finance sectors.

Why it matters: The acquisition demonstrates the growing market confidence in the immediate value of quantum chemistry simulation. Industry leaders are making aggressive moves to roll out offerings of hybrid HPC technologies that harness the power of quantum computations for useful work before the era of fault-tolerance. SandboxAQ's purchase illustrates the growing recognition that tools for quantum chemistry simulation have substantial near-term value for materials science and drug discovery.

Unitary Fund survey

A new Unitary Fund survey shows that 68.8% of developers prefer Qiskit

What it is: The Unitary Fund announced the results of their annual Quantum Open Source Survey. While Qiskit remains the most popular quantum SDK, its score fell in several categories, including “Full-stack development platforms and simulators." Given our commitment to utility scale quantum computing (which cannot be classically simulated) a reduction in score related to simulation may not be surprising.

Why it matters: The Unitary Fund survey offers insights on the community’s perception and use of Qiskit, compared to other open-source softwares. It reveals changes in users’ feelings about our tools and services in relation to our competitors and ourselves in prior years. The better we understand the shifting needs and expectations of the wider quantum community, the more effectively we can refine our user experience to remain the most popular open-source quantum SDK.

Bye to Baidu and Alibaba

Alibaba and Baidu are shuttering their quantum computing divisions

What it is: Alibaba shuttered the quantum computing division within its DAMO Academy research organization and donated the associated lab and experimental quantum equipment to Zhejiang University. Baidu followed suit, donating their lab and equipment to the Beijing Academy of Quantum Information Sciences.

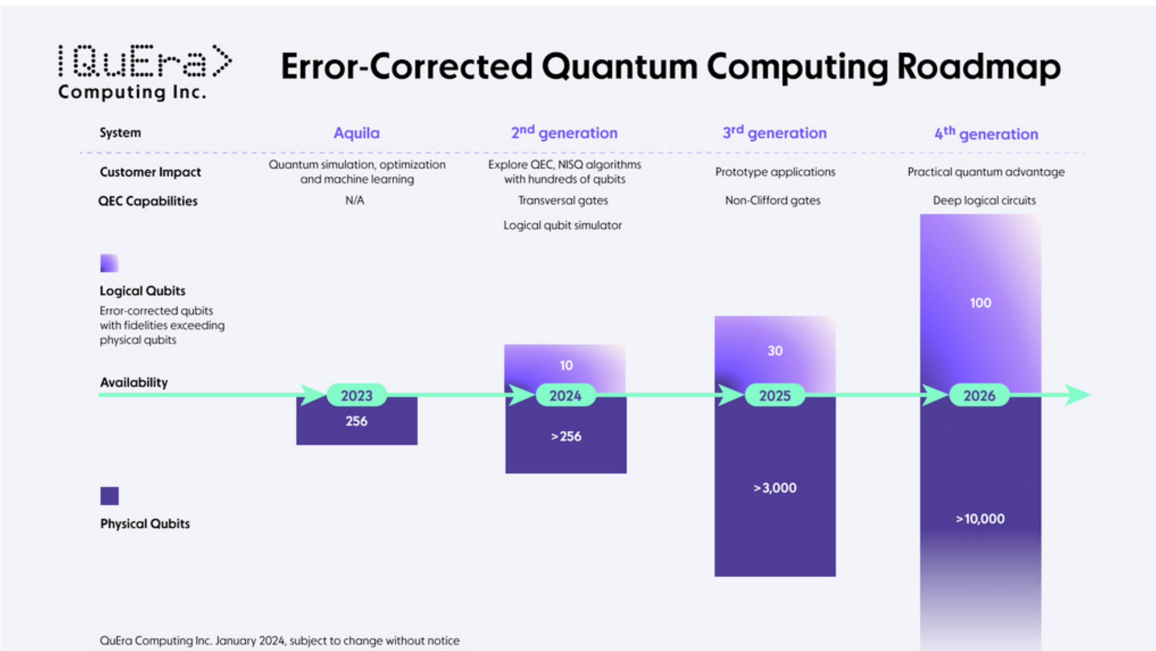
Why it matters: China’s quantum program is a complex situation. We continue monitoring to see how this plays out.

QuEra roadmap announcement

QuEra reveals their three-year quantum computing roadmap

What it is: QuEra revealed a three-year roadmap to produce a system with 100 error-corrected logical qubits based on their neutral atom qubit technology.

Why it matters: The scope of QuEra's roadmap reinforces our position that meaningful work can be done today with highly performant quantum systems of 100+ logical qubits. The announcement follows impressive new work published with collaborators at Harvard, MIT, and NIST on the successful execution of complex sampling circuits with 48 entangled logical qubits. The new roadmap and impressive results will undoubtedly lead to increased interest in quantum technology over the coming years.



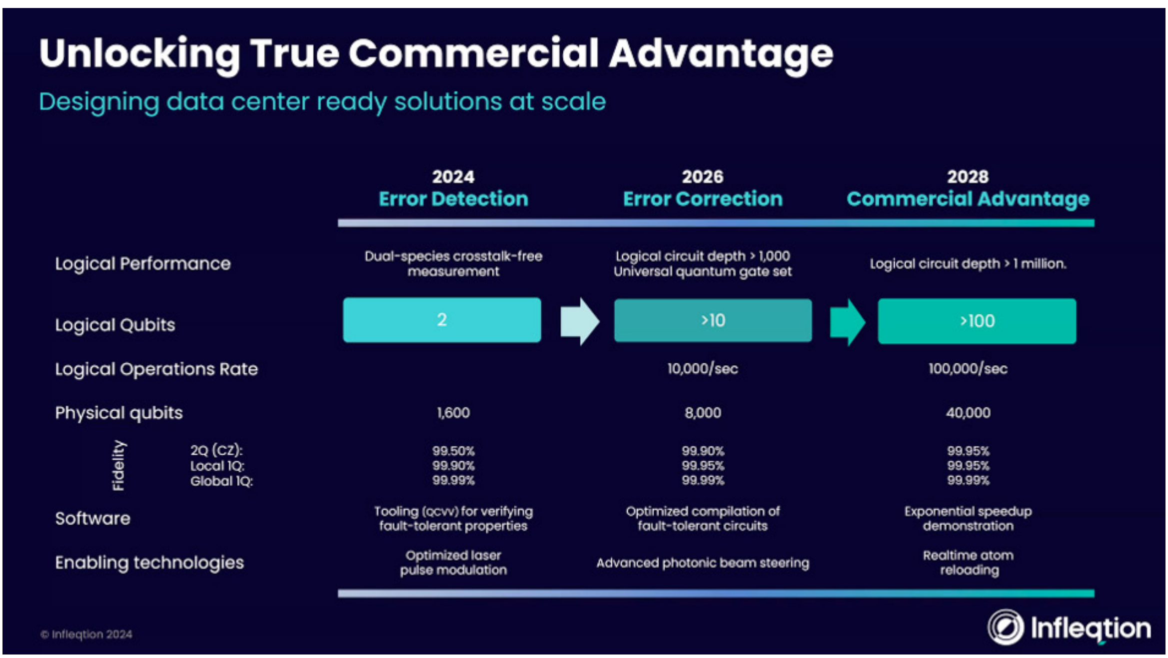
QuEra roadmap

Infleqtion roadmap announcement

Infleqtion reveals their five-year quantum computing roadmap

What it is: Infleqtion, also known as ColdQuanta, revealed a five-year roadmap with a plan to debut a system with 100 logical qubits in 2028, based on their neutral atom qubit technology.

Why it matters: Companies are following IBM Quantum's lead in demonstrating roadmaps to commercialization with a focus on delivering error-corrected quantum computation in the coming years. Infleqtion is also partnering with the National Quantum Computing Centre (NQCC) in the United Kingdom, with other partnerships in the works. As with QuEra, we expect the competing architectures to continue maturing and competing with our clients for mindshare. At IBM Quantum, we continue to work on advancing quantum processor technology based on the superconducting transmon qubit architecture and feel confident in the progress along our own roadmap.



Infleqtion roadmap

Quantum Networking

Already a world-class, prestigious institution of higher education, IIT Madras became a key partner with IBM Quantum, and India's first Quantum Innovation Center, in 2022. They have continued forming strategic partnerships like their upcoming collaboration with LTIMindtree on joint quantum research and workforce development. Last November, researchers at IIT Madras, in collaboration with IBM Quantum, applied quantum computing to a 70-year-old problem: protein folding.

Proteins are large molecules responsible for everything from binding with hormones to regulating ion exchange in cells. A protein can be thought of as a long chain of amino acids that folds into a 3D shape, the structure of which determines its function. As Professor Sanjib Senapati of the Bhupat and Jyoti Mehta School of Biosciences explained, "Many diseases and disorders are related to the misfolding of proteins, including Alzheimer's, Parkinson's, and Huntington's diseases." Understanding the process of protein folding, and the relationship between amino acid sequences and protein function, could mean treatments for hundreds of human diseases and a revolution in medicine and drug discovery. This is where quantum computing comes in.

Many quantum chemistry approaches model electronic orbitals, but this would be computationally intractable for systems of thousands of atoms. Instead, IBM Researcher Dr. Kalyan Dasgupta and the researchers at IIT Madras studied the folding of long chains of bead-like objects with some of the features of proteins, but not all. For example, different species of beads represented different amino acids present in proteins, but without a full description of the geometry of the acids. This enabled the group to focus on the folding of the larger structure. Dr. Dasgupta emphasized, "Finding a formulation of the problem that maps well onto the quantum system is of critical importance." The group used the 127-qubit *ibm_cusco* device and experimented on chains of up to 20 beads, modeled using up to 114 qubits. The energy minimization of the chain was solved using a variational quantum algorithm related to VQE (SamplingVQE). The preprint of their work can be viewed on the arXiv **here**.

This work constitutes an impressive display of utility-scale computation and demonstrates the power of applying IBM Quantum systems to real-world problems relevant to human health, drug design, and more. When asked about next steps, Prof. Senapati answered, "We will refine and extend our model. Adding the complexity of the amino acid side chains will bring us much closer to modeling real proteins."

Q&A: Jake Lishman

How would you describe your role?

I'm one of the principal Qiskit maintainers alongside Matthew Treinish for the core package—so circuits and the transpiler. I also work on the OpenQASM 3 language design and do bits and bobs all across the Qiskit ecosystem. Because Qiskit is an open-source project, I spend a lot of time talking to external, open source contributors. That means people making contributions, as well as people filing bug reports against Qiskit. I spend a lot of time figuring out whether there really is a bug, and if there is a bug, addressing it. The goal is keeping code quality up to speed.

What kind of background led you to this role?

My background is in quantum computing physics. I have a PhD in gate design for trapped-ion quantum computers from Imperial College London. While I was working on that PhD, I started contributing as an open source contributor to QuTiP, which is a language for quantum simulation. Paul Nation, who is now here at IBM, was one of its original authors. I eventually became a maintainer of QuTiP. About two and a half years ago, Paul noticed me and wrote me an email and asked me if I wanted to apply to work here at IBM, which I did.

At that time, I was still a student. So during that first year here at IBM, I was working during the day and finishing my thesis at night. I don't recommend that, but thankfully I got a lot of support through my management chain.

Is this where you imagined you would be when you started your PhD?

To be honest, I went into a physics PhD program because it was the path of least resistance. I didn't know what I wanted to do for a job, but I knew I had a path to a PhD position. During the course of that program, I realized that there are many people in the world who are better than me at physics but that what I am very good at is programming. That was when I started working on QuTiP, and that's when I realized that's what I wanted to do with my life and career afterward.

What do you do like to do outside of work?

Since the pandemic, I got into running. That was my escape during the lockdowns in the United Kingdom. I still go out and try and run longer and longer distances all the time. I also now have an 18-month-old. Being there for my kid obviously takes up a lot of my time as well.

Phase Kickback

This year marks the 20th anniversary of the first research paper published on circuit quantum electrodynamics, or CQED. And to celebrate, we're taking a look at some of the essential research that has helped build the field into what it is today.

CQED at its most basic level is an area of physics research concerning interactions between light and matter, and it has played an essential role in enabling the superconducting qubit architectures in IBM Quantum hardware. Prior to the advent of CQED, measurement techniques for superconducting quantum systems were destructive, creating lots of problems that made it difficult to manage superconducting quantum hardware. CQED gave researchers a new and improved means of building superconducting qubits and also led to the development of quantum nondemolition (QND) measurements for qubits, a tool that allows us to minimize the impact of our measurements on the qubit's stored information during our computations.

CQED was born out of an area of quantum optics research known as cavity quantum electrodynamics. In cavity QED, researchers confine light in tiny cavities filled with mirrors and study the coupling between photons in the cavity and atoms that pass through the cavity. **In 2004, researchers from Yale University published the first theory paper suggesting that an equivalent approach could serve as a simple and efficient architecture for the circuit-based artificial atoms that make up superconducting qubits.** The result would be qubits with stronger and much more controllable isolation from environmental noise.

The Yale research group would go on to write many more foundational papers in the history of CQED, with many authors joining in over the years to come—including a few names that will be familiar to the IBM Quantum team. A few months after their initial theory proposal, the group would go on to publish **another 2004 paper, this time demonstrating the first experimental implementation of the CQED architecture.**

This follow-up paper included the now-famous "vacuum Rabi splitting" experiment, which showed that the CQED architecture enabled a much stronger coupling between the qubit and its circuit-based cavity than was possible to achieve with natural atomic systems. The following year, **in 2005, the researchers published a paper demonstrating the first use of CQED for readout of a superconducting qubit.**

The Yale group published its next major CQED innovation two years later, this time with future IBM Quantum researcher Jay Gambetta onboard as second author. **Their 2007 paper was the first to propose two-qubit gates using the CQED architecture—an essential ingredient in quantum computation**—and presented definitions for not just one but five different two-qubit gates based on a variety of approaches.

Phase Kickback

Later that year, future IBM Quantum researcher Jerry Chow joined Gambetta and the other Yale coauthors for their next publication, which took the group's ideas about two-qubit gates to the next level. **That 2007 follow-up paper demonstrated how two qubits can actually be coupled to the same quantum bus—the device that stores and transfers information between qubits—and showed the first evidence of real entanglement between two qubits in the CQED architecture.**

The group achieved a key next step in their research **in 2009, not only demonstrating a real two-qubit gate on a small CQED-based quantum processor, but also successfully implementing simplified versions of the Grover's and Deutsch-Jozsa algorithms** on their two-qubit processor.

In 2010, researchers added in the key element of the Purcell filter, a device that enables fast readout of qubits while still protecting the qubits' coherence times. This allowed for much higher-fidelity readout of the qubits and would open the door to the later invention of Josephson parametric amplifiers and quantum limited amplifiers, tools that boost the readout signals.

Together, the research advances outlined in this article helped form the basis of the superconducting quantum systems we use today. Of course, these papers show only one side of the impact of CQED, which has also continued to play an influential role in the field of quantum optics, helping physicists better understand the ways in which light and matter interact.

CQED research also continued well after 2010, with research groups at Yale, IBM, and elsewhere using the CQED architecture as the foundation for numerous experiments, including those that led to the utility-scale quantum hardware we have at IBM today. Indeed, we likely would not have the ability to run quantum circuits of over 100 qubits without the two decades of CQED research that led us to where we are today.

Phase Kickback

Seminal works

2004

Cavity quantum electrodynamics for superconducting electrical circuits: an architecture for quantum computation, Blais, A., et al.

2004

Strong coupling of a single photon to a superconducting qubit using circuit quantum electrodynamics, Wallraff, A., et al.

2005

Approaching unit visibility for control of a superconducting qubit with dispersive readout, Wallraff, A., et al.

2007

Quantum-information processing with circuit quantum electrodynamics, Blais, A., et al.

2007

Coupling superconducting qubits via a cavity bus, Majer, J., et al.

2009

Demonstration of two-qubit algorithms with a superconducting quantum processor, DiCarlo, L., et al.

2010

Fast reset and suppressing spontaneous emission of a superconducting qubit, Reed, M. D., et al.

What if P = NP?

One of the biggest outstanding questions in mathematics asks: Does P = NP? Essentially, there is a class of problems that a deterministic Turing machine—an idealized model of the CPU where the computer changes between states based on the current state and inputted symbols fed into it on a tape—can efficiently solve and check classically, called P. You might hear “P” referred to as “polynomial time.” Then, there are problems that a non-deterministic Turing machine—where the machine may be able to take one of multiple actions in each situation—can efficiently solve and check, called NP, or “non-deterministic polynomial time.” We also refer to some problems as “NP-hard,” those that are at least as difficult to solve as the hardest NP problems, and “NP-complete,” or the NP-hard problems that are in NP.

Problems in P typically consist of the day-to-day problems your computer solves—like multiplying numbers together. Problems in NP are considered more difficult, often requiring an exponential amount of time to solve but only a polynomial amount of time to check. Factoring numbers into primes is in NP. It’s strongly assumed that P does not equal NP—that there really exist problems in NP that can be checked but not solved efficiently. But there’s no mathematical proof of this fact yet written.

So, we assume that we live in a world where P does not equal NP. But what if that wasn’t the case—what if P = NP? We asked two experts in the field.

Dmitri Maslov, IBM Quantum

→

A P = NP result where the complexity of the polynomial-time solution to an NP-Complete problem is something ridiculously unpractical, such as $n^{9!}$, will not have a meaningful practical application, and the world will remain unchanged for most people. If, however, P = NP comes with a practical algorithm of, say, complexity n^2, this would imply efficient solutions to all Clay Mathematics Institute problems, significant improvements to most algorithms (there may not be many problems/algorithms left with complexity exceeding n^2, except those you do not care about anyway), trivialization and phasing out of the entire field of mathematics, and disproval of no-free-lunch types of results. Basically, in layman's terms, a P = NP result like that would imply, literally, a free lunch. In this case, while contemplating the financial loss resulting from having paid for my lunches, I refuse to believe that this is the world I live in.

Scott Aaronson, UT Austin

→

Well, it would depend on whether we knew the algorithm and whether it was efficient in practice. If “yes” and “yes,” then the world would be different in many ways—for example:

- Cryptography based on hard problems (including Bitcoin and all other proof-of-work cryptocurrencies) would be dead
- Training ML models like GPT-4 would be vastly easier
- Mathematical creativity (including finding proofs of the Riemann hypothesis, etc.) could be fully automated
- There would be much less reason to try to build quantum computers

Upcoming Events

For more information on upcoming events, contact **qiskit.events@us.ibm.com**.

April 2
IBM Quantum Information Technical Exchange
Tak Hur, Yonsei University: Neural Quantum Embedding

April 23
Cleveland Innovation & Discovery Forum, hosted by Cleveland Clinic IBM Discovery Accelerator

April 24-25
HCLS Technical Working Group Meetup at Cleveland Clinic

May 7
IBM Quantum Information Technical Exchange
Michelle Grossi, CERN

May 13-15
IBM Quantum Partner Forum & EMEA Practitioners Forum in Europe

May 20-23
THINK 2024

May 21
Qiskit 102

May TBD
IBM Quantum Spring Challenge

IBM **Quantum**