



Contributions of Quantum Factoring on Quantum Research

Clayton Ferner, University of North Carolina Wilmington

This installment of Computer's series highlights work published in IEEE Symposium on Foundations of Computer Science. Although we typically highlight articles from work published in IEEE Computer Society journals, this month's exception is due to the prevalence and significance of quantum computing research progress in recent years.



Quantum computing is a new, hot area of computer science that promises great potential for solving problems that are currently intractable for traditional or classical computers. The

concept of using quantum mechanics as a basis for computing was first introduced around 1980 by Paul Benioff.¹ Since that time, researchers have been working on developing actual quantum computers as well as algorithms using quantum computing. Algorithms have been developed that demonstrate quantum supremacy, that is, quantum computers can solve problems with a super polynomial speedup over classical computers.

Until recently, quantum computing has remained on the fringes of computer science research. This is primary because there haven't been quantum computers built beyond a few qubits (the quantum equivalent

of a bit), and the algorithms were on problems that were quite theoretical and of not much interest to people outside of computer science. That is until recently. Quantum computing is now gaining speed and moving out of the fringe area and toward a more mainstream computer science research area. There are two main reasons for the surge in interest in quantum computing. First, IBM has

been able to build a larger quantum computer, just recently announcing a 127-qubit machine with plans for a 433-qubit processor later this year.² Figure 1 shows a close-up of a quantum computer. Second, Peter Shor presented a quantum algorithm capable of solving a real-world problem that caught everyone's attention.

also demonstrated quantum supremacy on a very real-world problem.

The reader should not be overly concerned that public-key encryption, which is essential to the success of e-commerce, can be broken. First, it will still take some time before quantum computers with enough qubits to perform factorization on 2,048-bit keys

alternative to the Diffie-Hellman Key Exchange, which would be broken by Shor's algorithm.⁵

Shor makes two important contributions. First, he provides an algorithm that, once quantum computers are built large enough and sophisticated enough, will break public-key encryption in its current form. Second, his algorithm sparked interest in quantum computing, which is a research area that is now gaining popularity.

Shor's algorithm uses a combination of classical computing with a quantum component. That quantum component is the quantum Fourier transformation, which does a discrete Fourier transform of a sinusoidal signal in the time domain to a frequency domain. The purpose of the Fourier transform is because factorization techniques have shown that factoring is equivalent to period finding. The idea behind factoring a large number N , starting with an initial guess g that is relatively prime with N , is to find an integer p such that g^p is one more than a multiple of N . If one can find this p , the factors of N are easily computed. The trouble is, finding p is no easier than a brute-force search for the factors of N .

It turns out that the integers that are congruent modulo N repeat with a period k . In other words, the powers of g that are one more than a multiple of N repeat with a period. Finding this period would allow one to find p . That is where the Fourier transform comes in. If one finds the frequency of the remainders from the modulus, then inverting that frequency would give the period. Still, this is no easier than a brute-force search for the factors of N .

Shor based his algorithm on Simon's algorithm. Simon developed an algorithm that solved the problem of finding a secret n -bit string s that satisfies a certain function. Although Simon's algorithm provided a quantum solution to a problem that didn't have a significant real-world application, it did provide a mechanism to solve period-finding algorithms. It was this contribution that served as a catalyst to Shor's algorithm.

Not only were many people surprised and interested in the notion of breaking public-key encryption, but it also demonstrated quantum supremacy on a very real-world problem.

At the annual IEEE Symposium on Foundations of Computer Science held in Santa Fe, New Mexico, in November 1994, Peter Shor presented an algorithm to factor integers using quantum computing in polynomial time with respect to the input size.³ The algorithm is a combination of classical computing with a quantum computing component. The implication of this is that the types of encryptions that are based upon the intractability of factoring large numbers, such as the Rivest-Shamir-Adleman (RSA) public-key encryption, would then be rendered useless. Not only were many people surprised and interested in the notion of breaking public-key encryption, but it

can be built. Second, it will take a significant amount of time before quantum algorithms with so many qubits can deal with the propagation of errors. Third, quantum encryption is an active research area that may provide an alternative to RSA public-key encryption before RSA is broken.⁴ Fourth, quantum computers are unable to break some forms of encryption, such as those demonstrated to be unbreakable. The one-time pad (OTP) is a provably unbreakable encryption if the key is not reused. Fifth, quantum algorithms have already been designed to provide a secure way to exchange keys needed for the OTP, already providing for an

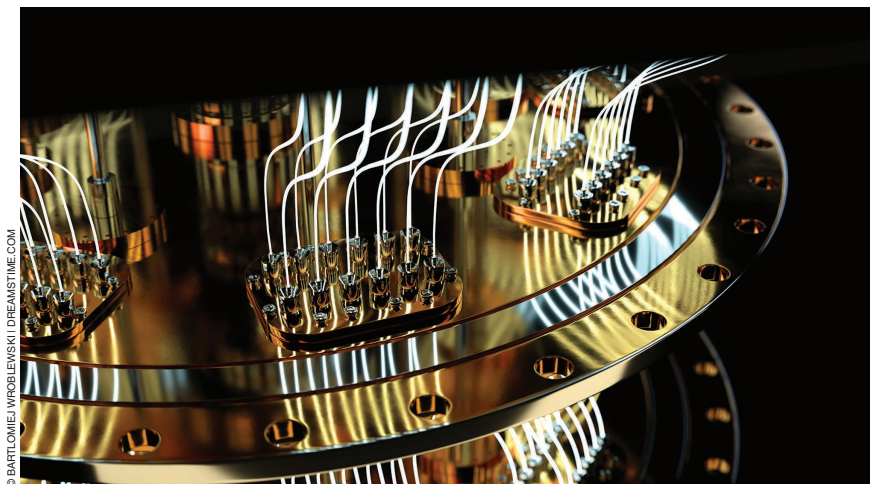



FIGURE 1. Close-up of a quantum computer with glowing wires with depth of field (https://www.dreamstime.com/pepasystem_info).

Shor's algorithm can find the frequency, and therefore the period, in polynomial time by using quantum computing as a way that allows the undesirable frequencies to cancel out, leaving the one frequency that leads to the correct period.

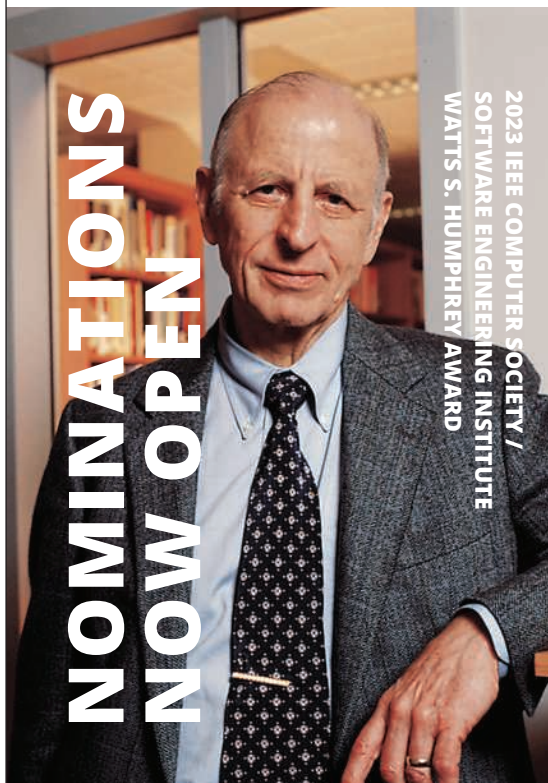
So, when can we expect quantum computers to become mainstream, able to decrypt all our communications? This is a difficult question. A lot of it depends on continued advances in quantum computers with a significant number of qubits, the capability to make the results as reliable as classical computers, and the ability to program quantum computers efficiently. Based on recent commercial interests and associated hardware advances, this time will come sooner than most people think. 

REFERENCES

1. P. Benioff, "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines," *J. Statist. Phys.*, vol. 22, no. 5, pp. 563–591, 1980, doi: 10.1007/BF01011339.
2. "IBM unveils new roadmap to practical quantum computing era; plans to deliver 4,000+ qubit system." IBM Newsroom. <https://newsroom.ibm.com/2022-05-10-IBM-Unveils-New-Roadmap-to-Practical-Quantum-Computing-Era-Plans-to-Deliver-4,000-Qubit-System> (Accessed: May 20, 2022).
3. P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Found. Comput. Sci.*, 1994, pp. 124–134, doi: 10.1109/SFCS.1994.365700.
4. "Post-quantum cryptography initiative," NIST Computer Security Resource Center, Gaithersburg, MD, USA, 2022. Accessed: May 24, 2022. [Online]. Available: <https://csrc.nist.gov/Projects/post-quantum-cryptography>
5. "Qrypt provides secure alternative to quantum key distribution." Business Wire. <https://www.businesswire.com/news/home/20220215005201/en/Qrypt-Provides-Secure-Alternative-to-Quantum-Key-Distribution> (Accessed: Jun. 9, 2022).

CLAYTON FERNER is a full professor in the Department of Computer Science at the University of North Carolina Wilmington, Wilmington, North Carolina, 28403, USA. Contact him at cferner@uncw.edu.

Carnegie Mellon University Software Engineering Institute



Since 1994, the SEI and the Institute of Electrical and Electronics Engineers (IEEE) Computer Society have cosponsored the award, which recognizes outstanding achievements in improving an organization's ability to create and evolve high-quality software-dependent systems.

The Humphrey Award nominee's productivity improvement must, to an exceptional degree, be **significant, measured, sustained, and shared.**

TO NOMINATE YOURSELF OR A COLLEAGUE, GO TO computer.org/volunteering/awards/humphrey-software-process-achievement

Nominations due by September 1, 2022.

FOR MORE INFORMATION

resources.sei.cmu.edu/news-events/events/watts