15 Nov 2018 | 16:00 GMT

The Case Against Quantum Computing

The proposed strategy relies on manipulating with high precision an unimaginably huge number of variables

By **Mikhail Dyakonov**

Quantum computing is all the rage. It seems like hardly a day goes by without some news outlet describing the extraordinary things this technology promises. Most commentators forget, or just gloss over, the fact that people have been working on quantum computing for decades—and without any practical results to show for it.

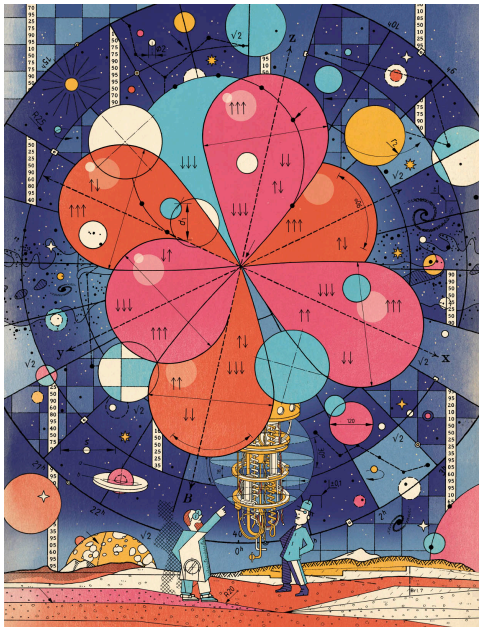

Illustration: Christian Gralingen

We've been told that quantum computers could "provide breakthroughs in many disciplines, including materials and drug discovery, the optimization of complex systems, and artificial intelligence." We've been assured that quantum computers will "forever alter our economic, industrial, academic, and societal landscape." We've even been told that "the encryption that protects the world's most sensitive data may soon be broken" by quantum computers. It has gotten to the point where many researchers in various fields of physics feel obliged to justify whatever work they are doing by claiming that it has some relevance to quantum computing.

Meanwhile, government research agencies, academic departments (many of them funded by government agencies), and corporate laboratories are spending billions of dollars a year developing quantum computers. On Wall Street, Morgan Stanley and other financial giants expect quantum computing to mature soon and are keen to figure out how this technology can help them.

It's become something of a self-perpetuating arms race, with many organizations seemingly staying in the race if only to avoid being left behind. Some of the world's top technical talent, at places like Google, IBM, and Microsoft, are working hard, and with lavish resources in state-of-the-art laboratories, to realize their vision of a quantum-computing future.

In light of all this, it's natural to wonder: When will useful quantum computers be constructed? The most optimistic experts estimate it will take 5 to 10 years. More cautious ones predict 20 to 30 years. (Similar predictions have been voiced, by the way, for the last 20 years.) I belong to a tiny minority that answers, "Not in the foreseeable future." Having spent decades conducting research in quantum and condensed-matter physics, I've developed my very pessimistic view. It's based on an understanding of the gargantuan technical challenges that would have to be overcome to ever make quantum computing work.

The idea of quantum computing first appeared nearly 40 years ago, in 1980, when the Russian-born mathematician Yuri Manin, who now works at the Max Planck Institute for Mathematics, in Bonn, first put forward the notion, albeit in a rather vague form. The concept really got on the map, though, the following year, when physicist Richard Feynman, at the California Institute of Technology, independently proposed it.

Realizing that computer simulations of quantum systems become impossible to carry out when the system under scrutiny gets too complicated, Feynman advanced the idea that the computer itself should operate in the quantum mode: "Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy," he opined. A few years later, University of Oxford physicist David Deutsch formally described a general-purpose quantum computer, a quantum analogue of the universal Turing machine.

The subject did not attract much attention, though, until 1994, when mathematician Peter Shor (then at Bell Laboratories and now at MIT) proposed an algorithm for an ideal quantum computer that would allow very large numbers to be factored much faster than could be done on a conventional computer. This outstanding theoretical result triggered an explosion of interest in quantum computing. Many thousands of research papers, mostly theoretical, have since been published on the subject, and they continue to come out at an increasing rate.

The basic idea of quantum computing is to store and process information in a way that is very different from what is done in conventional computers, which are based on classical physics. Boiling down the many details, it's fair to say that conventional computers operate by manipulating a large number of tiny transistors working essentially as on-off switches, which change state between cycles of the computer's clock.

The state of the classical computer at the start of any given clock cycle can therefore be described by a long sequence of bits corresponding physically to the states of individual transistors. With N transistors, there are 2^N possible states for the computer to be in. Computation on such a machine fundamentally consists of switching some of its transistors between their "on" and "off" states, according to a prescribed program.

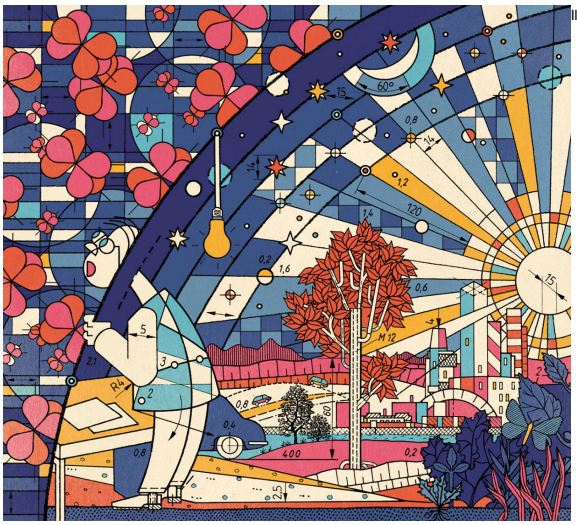

Illustration: Christian Gralingen

In quantum computing, the classical two-state circuit element (the transistor) is replaced by a quantum element called a quantum bit, or qubit. Like the conventional bit, it also has two basic states. Although a variety of physical objects could reasonably serve as quantum bits, the simplest thing to use is the electron's internal angular momentum, or spin, which has the peculiar quantum property of having only two possible projections on any coordinate axis: +1/2 or −1/2 (in units of the Planck constant). For whatever the chosen axis, you can denote the two basic quantum states of the electron's spin as ↑ and ↓.

Here's where things get weird. With the quantum bit, those two states aren't the only ones possible. That's because the spin state of an electron is described by a quantum-mechanical wave function. And that function involves two complex numbers, α and β (called quantum amplitudes), which, being complex numbers, have real parts and imaginary parts. Those complex numbers, α and β, each have a certain magnitude, and according to the rules of quantum mechanics, their squared magnitudes must add up to 1.

That's because those two squared magnitudes correspond to the *probabilities* for the spin of the electron to be in the basic states ↑ and ↓ when you measure it. And because those are the only outcomes possible, the two associated probabilities must add up to 1. For example, if the probability of finding the electron in the ↑ state is 0.6 (60 percent), then the probability of finding it in the ↓ state must be 0.4 (40 percent)—nothing else would make sense.

In contrast to a classical bit, which can only be in one of its two basic states, a qubit can be in any of a *continuum* of possible states, as defined by the values of the quantum amplitudes α and β. This property is often described by the rather mystical and intimidating statement that a qubit can exist simultaneously in both of its ↑ and ↓ states.

Yes, quantum mechanics often defies intuition. But this concept shouldn't be couched in such perplexing language. Instead, think of a vector positioned in the x-y plane and canted at 45 degrees to the x-axis. Somebody might say that this vector simultaneously points in both the x- and y-directions. That statement is true in some sense, but it's not really a useful description. Describing a qubit as being simultaneously in both ↑ and ↓ states is, in my view, similarly unhelpful. And yet, it's become almost de rigueur for journalists to describe it as such.

In a system with two qubits, there are 2^2 or 4 *basic* states, which can be written (↑↑), (↑↓), (↓↑), and (↓↓). Naturally enough, the two qubits can be described by a quantum-mechanical wave function that involves four complex numbers. In the general case of N qubits, the state of the system is described by 2^N complex numbers, which are restricted by the condition that their squared magnitudes must all add up to 1.

While a conventional computer with N bits at any given moment must be in *one* of its 2^N possible states, the state of a quantum computer with N qubits is described by the *values of the* 2^N *quantum amplitudes*, which are continuous parameters (ones that can take on any value, not just a 0 or a 1). This is the origin of the supposed power of the quantum computer, but it is also the reason for its great fragility and vulnerability.

How is information processed in such a machine? That's done by applying certain kinds of transformations—dubbed "quantum gates"—that change these parameters in a precise and controlled manner.

Experts estimate that the number of qubits needed for a useful quantum computer, one that could compete with your laptop in solving certain kinds of interesting problems, is between 1,000 and 100,000. So the number of continuous parameters describing the state of such a useful quantum computer at any given moment must be at least $2^{1,000}$, which is to say about 10^{300}. That's a very big number indeed. How big? It is much, much greater than the number of subatomic particles in the observable universe.

To repeat: A useful quantum computer *needs to process a set of continuous parameters that is larger than the number of subatomic particles in the observable universe*.

At this point in a description of a possible future technology, a hardheaded engineer loses interest. But let's continue. In any real-world computer, you have to consider the effects of errors. In a conventional computer, those arise when one or more transistors are switched off when they are supposed to be switched on, or vice versa. This unwanted occurrence can be dealt with using relatively simple error-correction methods, which make use of some level of redundancy built into the hardware.

In contrast, it's absolutely unimaginable how to keep errors under control for the 10^{300} continuous parameters that must be processed by a useful quantum computer. Yet quantum-computing theorists have succeeded in convincing the general public that this is feasible. Indeed, they claim that something called the threshold theorem *proves* it can be done. They point out that once the error per qubit per quantum gate is below a certain value, indefinitely long quantum computation becomes possible, at a cost of substantially increasing the number of qubits needed. With those extra qubits, they argue, you can handle errors by forming *logical* qubits using multiple *physical* qubits.

How many physical qubits would be required for each logical qubit? No one really knows, but estimates typically range from about 1,000 to 100,000. So the upshot is that a useful quantum computer now needs a million or more qubits. And the number of continuous parameters defining the state of this hypothetical quantum-computing machine—which was already more than astronomical with 1,000 qubits—now becomes even more ludicrous.

Even without considering these impossibly large numbers, it's sobering that no one has yet figured out how to combine many physical qubits into a smaller number of logical qubits that can compute something useful. And it's not like this hasn't long been a key goal.

In the early 2000s, at the request of the Advanced Research and Development Activity (a funding agency of the U.S. intelligence community that is now part of Intelligence Advanced Research Projects Activity), a team of distinguished experts in quantum information established a road map for quantum computing. It had a goal for 2012 that "requires on the order of 50 physical qubits" and "exercises multiple logical qubits through the full range of operations required for fault-tolerant [quantum computation] in order to perform a simple instance of a relevant quantum algorithm…." It's now the end of 2018, and that ability has still not been demonstrated.

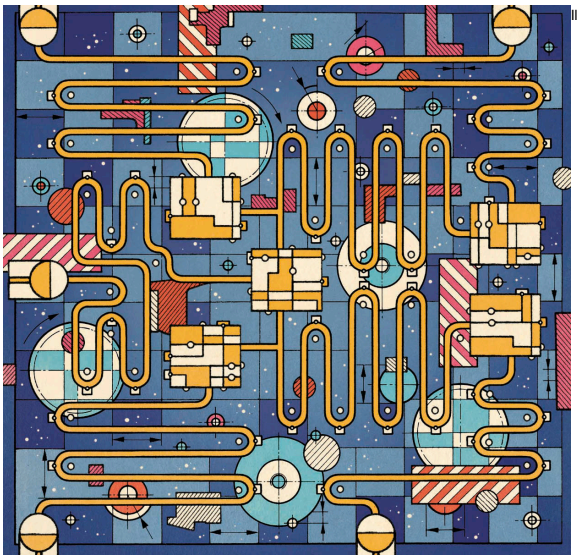

Illustration: Christian Gralingen

The huge amount of scholarly literature that's been generated about quantum-computing is notably light on experimental studies describing actual hardware. The relatively few experiments that have been reported were extremely difficult to conduct, though, and must command respect and admiration.

The goal of such proof-of-principle experiments is to show the possibility of carrying out basic quantum operations and to demonstrate some elements of the quantum algorithms that have been devised. The number of qubits used for them is below 10, usually from 3 to 5. Apparently, going from 5 qubits to 50 (the goal set by the ARDA Experts Panel for the year 2012) presents experimental difficulties that are hard to overcome. Most probably they are related to the simple fact that $2^5 = 32$, while $2^{50} = 1,125,899,906,842,624$.

By contrast, the *theory* of quantum computing does not appear to meet any substantial difficulties in dealing with millions of qubits. In studies of error rates, for example, various noise models are being considered. It has been proved (under certain assumptions) that errors generated by "local" noise can be corrected by carefully designed and very ingenious methods, involving, among other tricks, massive parallelism, with many thousands of gates applied simultaneously to different pairs of qubits and many thousands of measurements done simultaneously, too.

A decade and a half ago, ARDA's Experts Panel noted that "it has been established, under certain assumptions, that if a threshold precision per gate operation could be achieved, quantum error correction would allow a quantum computer to compute indefinitely." Here, the key words are "*under certain assumptions*." That panel of distinguished experts did not, however, address the question of whether these assumptions could ever be satisfied.

I argue that they can't. In the physical world, continuous quantities (be they voltages or the parameters defining quantum-mechanical wave functions) can be neither measured nor manipulated *exactly*. That is, no continuously variable quantity can be made to have an exact value, including zero. To a mathematician, this might sound absurd, but this is the unquestionable reality of the world we live in, as any engineer knows.

Sure, discrete quantities, like the number of students in a classroom or the number of transistors in the "on" state, can be known exactly. Not so for quantities that vary continuously. And this fact accounts for the great difference between a conventional digital computer and the hypothetical quantum computer.

Indeed, all of the assumptions that theorists make about the preparation of qubits into a given state, the operation of the quantum gates, the reliability of the measurements, and so forth, cannot be fulfilled *exactly*. They can only be approached with some limited precision. So, the real question is: What precision is required? With what exactitude must, say, the square root of 2 (an irrational number that enters into many of the relevant quantum operations) be experimentally realized? Should it be approximated as 1.41 or as 1.41421356237? Or is even more precision needed? There are no clear answers to these crucial questions.

While various strategies for building quantum computers are now being explored, an approach that many people consider the most promising, initially undertaken by the Canadian company D-Wave Systems and now being pursued by IBM, Google, Microsoft, and others, is based on using quantum systems of interconnected Josephson junctions cooled to very low temperatures (down to about 10 millikelvins).

The ultimate goal is to create a universal quantum computer, one that can beat conventional computers in factoring large numbers using Shor's algorithm, performing database searches by a similarly famous quantum-computing algorithm that Lov Grover developed at Bell Laboratories in 1996, and other specialized applications that are suitable for quantum computers.

On the hardware front, advanced research is under way, with a 49-qubit chip (Intel), a 50-qubit chip (IBM), and a 72-qubit chip (Google) having recently been fabricated and studied. The eventual outcome of this activity is not entirely clear, especially because these companies have not revealed the details of their work.

While I believe that such experimental research is beneficial and may lead to a better understanding of complicated quantum systems, I'm skeptical that these efforts will ever result in a practical quantum computer. Such a computer would have to be able to manipulate—on a microscopic level and with enormous precision—a physical system characterized by an unimaginably huge set of parameters, each of which can take on a continuous range of values. Could we ever learn to control the more than 10^{300} continuously variable parameters defining the quantum state of such a system?

My answer is simple. *No*, *never*.

I believe that, appearances to the contrary, the quantum computing fervor is nearing its end. That's because a few decades is the maximum lifetime of any big bubble in technology or science. After a certain period, too many unfulfilled promises have been made, and anyone who has been following the topic starts to get annoyed by further announcements of impending breakthroughs. What's more, by that time all the tenured faculty positions in the field are already occupied. The proponents have grown older and less zealous, while the younger generation seeks something completely new and more likely to succeed.

All these problems, as well as a few others I've not mentioned here, raise serious doubts about the future of quantum computing. There is a tremendous gap between the rudimentary but very hard experiments that have been carried out with a few qubits and the extremely developed quantum-computing theory, which relies on manipulating thousands to millions of qubits to calculate anything useful. That gap is not likely to be closed anytime soon.

To my mind, quantum-computing researchers should still heed an admonition that IBM physicist Rolf Landauer made decades ago when the field heated up for the first time. He urged proponents of quantum computing to include in their publications a disclaimer along these lines: "This scheme, like all other schemes for quantum computation, relies on speculative technology, does not in its current form take into account all possible sources of noise, unreliability and manufacturing error, and probably will not work."

Editor's note: A sentence in this article originally stated that concerns over required precision "were never even discussed." This sentence was changed on 30 November 2018 after some readers pointed out to the author instances in the literature that had considered these issues. The amended sentence now reads: "There are no clear answers to these crucial questions."

About the Author

Mikhail Dyakonov does research in theoretical physics at Charles Coulomb Laboratory at the University of Montpellier, in France. His name is attached to various physical phenomena, perhaps most famously Dyakonov surface waves.