Quantum computation and physical law

Leonard J. Schulman
Caltech

The relationship between science and engineering is unequal. Science has custody of the noble truths; engineering is in charge of getting things done. One engineering proposal, however, defies this asymmetry. A successful quantum computer would verify as-yet untested predictions of quantum mechanics. Such verification is not a foregone conclusion.

Forty years ago, when computers began to enter the academic and commercial world, researchers asked what types of problems could forseeably be solved on them. As computers were improved from year to year, it was clear that this question must be couched in a suitably abstract framework. Soon, the computer scientist Jack Edmonds and others had focused on what we still consider the most essential distinction: between those problems which can solved in time *polynomial* in the input size, and those which cannot. The class of tractable problems, "Poly-time," turned out not to depend in any way that people could identify, on the technology used to build the computers. The class was perceived, then, as a property of physical reality—a limit on the computational power of physical devices. Although the concept "Poly-time" was originally formulated to answer an *engineering* question, it was quickly absorbed into a *scientific* assertion about what is physically possible in our universe.

One problem that seemed to be intractable, or outside of "Poly-time" is this: Given a whole number, find its factorization. This task has intrigued mathematicians at least since the early nineteenth century. Confidence in its intractability was so strong that in the 1970s, in work for which they were recently given the Turing award, Ron Rivest, Adi Shamir and Leonard Adleman invented a cryptosystem whose security depended on this assertion [4]. Today that cryptosystem is widely used for commercial and other transactions.

Early in the 1980s the physicist Richard Feynman observed that computers were having difficulty with another kind of computation: simulating quantum mechanical dynamics [2]. This may at first seem unremarkable: all manner of physical processes, such as the weather, can be complex, unstable, and hard to compute. But Feynman's difficulty was of an altogether different scale. In the mathematical theory of quantum mechanics, the number of parameters needed to accurately describe a many-particle system grows *exponentially* in the number of particles. This is because each particle of the system is, to varying degrees, in each of its possible states at once—what is called a "superposition"—and because to write down the state of the whole system we need to keep track of each way of *combining* the states of all the particles. As far as we know, the system cannot be simulated without doing so. This leaves simulation of quantum dynamics far outside of "Poly-time."

Now, something about this picture is suspicious. Many real-world problems are hard to compute and it would be no surprise for physicists to have encountered an important one. But the problem of simulating quantum dynamics should not be on this list. After all, the universe performs these computations all the time—and it keeps up with itself, too. What gives? Feynman suggested two possibilities.

(1) There is some clever, mysterious way of computing quantum mechanical simulations that doesn't require writing down all those parameters.

Feynman couldn't think of one, nor has anyone else, yet it would have to be something basic. So we'll dismiss the possibility for the rest of this essay.

(2) Devices operating on the principles of quantum mechanics have inherently greater computational power than those operating on the principles of classical mechanics.

Feynman did not have the mathematical framework (complexity theory) to take possibility (2) further, but a decade later, the computer scientists Ethan Bernstein and Umesh Vazirani did so [1]. They were able to show (under certain abstract assumptions) that the class of tractable (poly-time) problems is indeed greater in a quantum-mechanical world, than it would be in a classical world: a deep *scientific* statement about what is or is not physically possible in our universe. Within a year, the computer scientist Peter Shor had derived from it a great *engineering* accomplishment: a poly-time algorithm for factoring numbers [5]!

To date, the prototype quantum computers that have been built are very limited, however. Shor's algorithm can only realize its potential on a quantum computer capable of factoring numbers large enough to be out of reach for ordinary computers. Such a device must be able to produce, and maintain over an extended time, particular kinds of quantum superpositions simultaneously involving many (at least a few hundred) particles. Superpositions like this have never been observed. Indeed, the prediction that they exist has troubled physicists since the inception of quantum theory. Erwin Schrödinger, a founder of the theory, memorably told of a (hypothetical) cat in a simultaneous superposition of two states: alive and dead. The whole point of this image is that it is ridiculous—nothing as complex as a cat has ever straddled reality so delicately. Yet subatomic particles are always in superpositions, and quantum theory knows no size limit: what it prescribes for particles, it predicts for cats...and for computers.

What quantum computation has done is to take Schrodinger's improbable feline spectre out of the shadows of quantum theory, where it was always spoken of, but never seen, and into the arena of testable experimental predictions. Since the computational implications of these predictions are remarkable, it behooves us to consider an alternative remarkable possibility—that a quantum computer of a useful size is a physical impossibility, that large numbers cannot be quickly factored, that Schrödinger's cat was never in danger—in short, maybe:

(3) Quantum theory breaks down for large, complex systems.

Large quantum systems are so hard to control in the laboratory that their theory is only an extrapolation of what we know for small systems. Like earlier extrapolations—Newtonian mechanics, which Albert Einstein revised at high velocities, or the flatness of the earth, which the ancient Greeks revised at large distances—it might be wrong. Quantum computers, as computers, will probably not be useful until they contain hundreds of "quantum bits" (basically, particles involved in the computation). As experimental tests of quantum mechanics, however, they are already charting new terrain: recent experiments have reached a dozen quantum bits [3].

Where do we stand? Quantum algorithms—nothing but engineering designs—are so good that they pose a test to the laws of physics. In a manner of speaking, these algorithms have given teeth to that troublesome cat that has always lurked precariously in quantum theory. We have a choice

between possibility (2)—the scenario in which our old understanding of computation is revealed to have been fundamentally flawed—and (3)—in which our understanding of physics turns out to have been no better. There are serious researchers in both camps. The only test is in the laboratory.

References

[1] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997. (STOC 1993).

[2] R. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7):467–488, 1982.

[3] C. Negrevergne, T. S. Mahesh, C. A. Ryan, M. Ditty, F. Cyr-Racine, W. Power, N. Boulant, T. Havel, D. G. Cory, , and R. Laflamme. Benchmarking quantum control methods on a 12-qubit system. *Phys. Rev. Lett.*, 96, 2006. 170501.

[4] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *C. ACM*, 21:120–126, 1978.

[5] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Computing*, 26:1484–1509, 1997. (FOCS 1994).