

Scott Aaronson on Google's new quantum-computing paper

December 14 2015, by Larry Hardesty



Scott Aaronson Credit: Bryce Vickmark

In 2010, a Canadian company called D-Wave announced that it had begun production of what it called the world's first commercial quantum computer, which was based on theoretical work done at MIT. Quantum

computers promise to solve some problems significantly faster than classical computers—and in at least one case, exponentially faster. In 2013, a consortium including Google and NASA bought one of D-Wave's machines.

Over the years, critics have argued that it's unclear whether the D-Wave machine is actually harnessing quantum phenomena to perform its calculations, and if it is, whether it offers any advantages over classical computers. But this week, a group of Google researchers [released a paper](#) claiming that in their experiments, a quantum algorithm running on their D-Wave machine was 100 million times faster than a comparable classical algorithm.

Scott Aaronson, an associate professor of electrical engineering and computer science at MIT, has been following the D-Wave story for years. MIT News asked him to help make sense of the Google researchers' new paper.

Q: The Google researchers' paper focused on two algorithms: simulated annealing and quantum annealing. What are they?

A: Simulated annealing is one of the premier optimization methods that's used today. It was invented in the early 1980s by direct analogy with what happens when people anneal metals, which is a 7,000-year-old technology. You heat the metal up, the atoms are all jiggling around randomly, and as you slowly cool it down, the atoms are more and more likely to go somewhere that will decrease the total energy.

In the case of an algorithm, you have a whole bunch of bits that start flipping between 1 and 0 willy-nilly, regardless of what that does to the solution quality. And then as you lower the "temperature," a bit becomes

more and more unwilling to flip in a way that would make the solution worse, until at the end, when the temperature is zero, a bit will only go to the value that keeps the solution going straight downhill—toward better solutions.

The main problem with simulated annealing, or for that matter with any other local-search method, is that you can get stuck in local optima. If you're trying to reach the lowest point in some energy landscape, you can get stuck in a crevice that is locally the best, but you don't realize that there's a much lower valley somewhere else, if you would only go up and search. Simulated annealing tries to deal with that already: When the temperature is high, then you're willing to move up the hill sometimes. But if there's a really tall hill, even if it's a very, very narrow hill—just imagine it's a big spike sticking out of the ground—it could take you an exponential amount of time until you happen to flip so many bits that you happen to get over that spike.

In quantum mechanics, we know that particles can tunnel through barriers. (This is the language that the physicists use, which is a little bit misleading.) There's an important 2002 paper by Farhi, Goldstone, and Gutmann, all of whom are here at MIT, and what they showed is that if your barrier really is a tall thin spike, then [quantum annealing](#) can give you an exponential speedup over classical simulated annealing. Classical annealing is going to get stuck at the base of that spike for exponential time, and quantum annealing is going to tunnel over it and get down to the global minimum in polynomial time.

Q: So is the D-Wave machine using quantum tunneling?

A: In the current model of the D-Wave chip, there are 1,000 or so [qubits](#) [quantum bits], but they're organized into clusters of eight qubits each.

The qubits within each cluster are very tightly connected to each other, and between clusters there are only weaker connections. I think that this is the best evidence we've had so far for quantum tunneling behavior, at least at the level of the eight-bit clusters.

The main way that they got an advantage over simulated annealing in these results was by taking advantage of the fact that quantum tunneling—or anything that correlates all the qubits within the cluster—can flip all the bits within each cluster at the same time, whereas simulated annealing is going to try flipping the bits one by one, then see that that's not a good idea, then flip them all back, and not realize that by flipping all eight of them, you could get something better.

The case has now clearly been made that whatever the D-Wave device is doing, it's something that can tunnel past this eight-qubit barrier. Of course, that still doesn't mean that you're doing anything faster than you could do it classically.

Q: What does it mean, then?

A: In computer science, normally we care about asymptotic speedup: We care about, "What is your running time as a function of the size of the problem? Does it grow linearly? Does it grow quadratically?" The constant that's in front—Does it take $5N$ steps? Does it take $10N$ steps?—we don't care that much about. We just care that it's linear in N .

In the Google paper, they discuss two classical algorithms that do match the asymptotic performance—and one of them beats the real-world performance—of the D-Wave machine. So besides simulated annealing, there are two more classical algorithms that are actors in this story. One of them is quantum Monte Carlo, which is actually a classical optimization method, but it's one that's inspired by quantum mechanics.

In this new Google paper, they say that even though quantum Monte Carlo has the same asymptotic performance, the constant is way, way better for the D-Wave machine. The constant is about 100 million times better.

There are two huge issues that I would have with that. The first issue is that the problem instances where the comparison is being done are basically for the problem of simulating the D-Wave machine itself. There were \$150 million dollars that went into designing this special-purpose hardware for this D-Wave machine and making it as fast possible. So in some sense, it's no surprise that this special-purpose hardware could get a constant-factor speedup over a classical computer for the problem of simulating itself.

The qubits in their chip are organized in this particular graph topology. If you want to solve a practically important optimization problem, you need to map it somehow onto that topology. And there's always a loss when you do that mapping. It seems entirely possible that that loss would kill a constant-factor advantage.

But the other point is that there's yet another classical algorithm on the stage, which is Selby's algorithm, which I think was first announced on my blog. It's a local-search algorithm, but it's one that is able to figure out that the qubits are organized into these clusters. What the Google paper finds is that Selby's algorithm, which runs on a classical computer, totally outperforms the D-Wave machine on all the instances they tested.

If I know that these eight qubits form a cluster, and I should be thinking of them as one giant variable, then I just find the best setting of that variable, and I'm done. There are only 2^8 —2 to the 8th—cases to check. That you can do pretty quickly.

If the clusters were 800 bits, then you wouldn't be able to do this. On the

other hand, building 800 qubits that are all talking to each other is a super-hard engineering problem. And even if you did [build those qubit clusters], it's not at all clear that quantum annealing would be able to tunnel over that.

Remember, quantum annealing does best when there's a tall, thin potential barrier. When you make the potential barrier wider, as would happen if you had 800-qubit clusters, then quantum annealing would have trouble as well.

We're finally seeing clearly the logical consequences of the design decisions that D-Wave made 10, 15 years ago, which were, "Go for as many qubits as possible as quickly as possible. Don't really worry about their lifetime, about their coherence. Don't worry about error correction. Don't worry about solving something where we're confident theoretically that there's a quantum speedup."

I think of them as taking the dirty approach, and most of the others are trying to take the clean approach. Of course, it is possible that the dirty approach will get somewhere before the clean approach does. There are many precedents for that in the history of technology, where the dirty approach wins. But it hasn't won yet.

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology

Citation: Scott Aaronson on Google's new quantum-computing paper (2015, December 14) retrieved 4 June 2024 from <https://phys.org/news/2015-12-scott-aaronson-google-quantum-computing-paper.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.