Analog Computing: a different way to think about building a (quantum) computer

November 24, 2016

1 What is an analog computer?

Most of the computers we have around us today, such as desktops, laptops, and smartphones are digital. Everything they do can be represented as a series of logic operations on a string of ones and zeros. But there is another type of computer which used to be quite common, *analog computers*, and these types of computers may be a promising route for building a quantum computer. Before talking about quantum computing though, it is worth discussing what an analog computer is. Unlike a digital computer, the behaviour of an analog computer cannot be described as a series of logical operations on ones and zeros.

A simple example of a small analog computer is a slide rule, which enables computation to be done by reading numbers from different parts which move relative to each other, one design of slide rule looks like this:

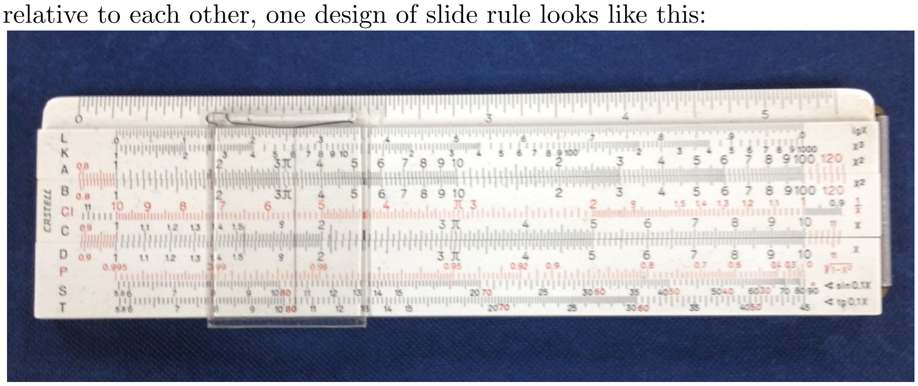


Image credit: I have this on my desk, I took a picture of it, feel free to reuse without attribution.

Before computers as we know them, slide rules were used for most important calculations. A slide rule can be used to compute things, but it clearly does not encode data in a digital (ones and zeros) way. In the case of the slide rule, it is the position of the slide, which can be changed continuously, which encodes the information, and the carefully placed numbers which allow the result to be read out. This kind of device has existed since ancient times, the oldest known analog computer is the Antikythera mechanism, a geared device used by the ancient Greeks to predict astronomical events, below you can see what this device looks like now (left) and a reconstruction of what if may have originally looked like (right).

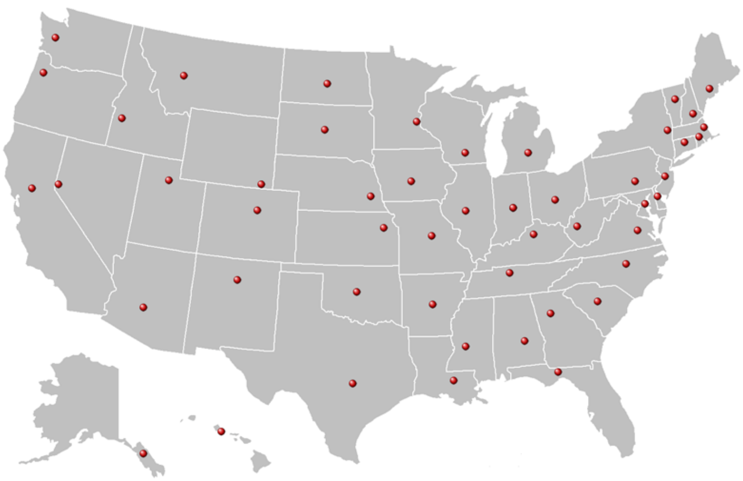
Image credit: (Left) Can be found on wikimedia commons, produced by user Marsyas (Right) Computer-generated graphic for front of Antikythera mechanism by Tony Freeth.

Analog computers do not have to be mechanical, like the two examples I have given here, in fact electrical analog computers were also used fairly frequently in the mid 20th century.

2 Why someone might want to use an analog quantum computer?

For digital computing, one reaches a solution through a series of logical operations, a *digital algorithm.* It is relatively clear how to do this for many tasks, such as multiplying numbers together. For some problems however, there isn't an obvious digital algorithm which works well. Consider for instance the travelling salesperson problem, a salesperson has to go to N different cities to sell things and come back home, but can go in any order, which order requires the least total distance to travel? The most obvious digital algorithm to check this is to list the distance for every possible order and pick whatever one is the least, you could do this by hand for 3 cities (6 possibilities), or maybe 4 if you were really patient (24 possibility), but what about 20 cites? Lets say a fast computer can check one order every nanosecond, it would take over 70 years to check every possible order, for 26 cities, it would take the same machine longer than the amount of time the universe has existed.

Obviously doing these calculations this way is not practical, even for a relatively small number of cities, there are other digital algorithms to calculate this and they can do better, but none of them do very well. Most computer scientists suspect (although haven't proven) that there is **no** algorithm which can do particularly well at solving this kind of problem. Fortunately, the salesperson doesn't necessarily have to find the absolutely most efficient order to go to the cities, but still doesn't want to waste gas and time, a 'pretty good' solution, is still useful, even if it isn't the absolute best. In fact for many problems like this, which come up in the real world, coming up with a *slightly* better solution can make a big difference. Imagine for instance if a company employs 100 travelling salespeople, if they could make each of their routes even 1% more efficient, this would add up to a huge savings!



Travelling salesperson problem: lets say I want to visit all 50 state capitols but don't care what order I do it in, what is the best order? the answer is not easy. Image credit: wikimedia commons, uploaded by user Roke.

This kind of problem, where you want to find the best possible solution you can are known as *optimization* problems, and hard optimization problems, like the travelling salesperson problem are a place where doing something new and clever (such as using a quantum analog computer) can make a big difference.

3 Making nature solve our problems for us

So now we know *why* someone might want to try something new to tackle hard problems, but we need to think about how. Fortunately, the laws of physics can help us here, for instance, we know that when things are cooled down they tend to seek out low energy states. Therefore, if we can build a physical system which 'maps' to our problem so that lower energy means a better solution, than we can just cool the system down and nature will solve the problem for us.

This works pretty well, but much of the time it actually easier to simulate a system on a computer rather than actually build it. *Simulated annealing*, where problems are solved by simulating the cooling of a physical system has proven very successful at solving many optimization problems. Very successful, but not perfect, the laws of physics tell us that a system which is cooled seeks out low energy states, but it doesn't always find the lowest energy. Even nature isn't a perfect problem solver. Finding out the best way to arrange atoms in glass is a very hard problem, and nature doesn't solve it correctly (humans don't know how to solve it correctly either), glass therefore can be viewed as a real world physical example of nature trying and failing at a hard optimization problem.

Sorry mate, you failed, better luck next time. Image credit: wikimeida commons, uploaded by user: Fir0002/Flagstaffotos

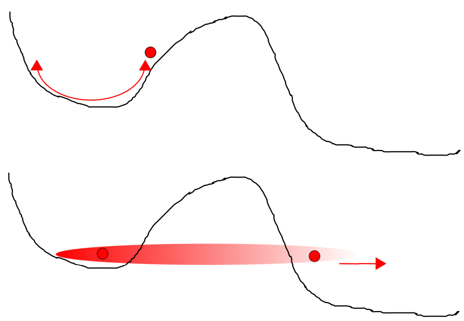
If we want to do even better, we have look at what other tools nature has put at our disposal, and one obvious one is quantum mechanics, the physics which governs the behaviour of atoms and molecules. Simulations have limits however and quantum mechanics is very hard to simulate in many cases, therefore if we want a system which uses quantum mechanics to help solve these kinds of problems, we are going to have to actually build something.

4 Quantum Mechanics

Before talking about how analog quantum machines could be built, lets step back a bit and discuss what quantum mechanics actually is. Quantum mechanics comes about because matter is made up of waves, which must behave smoothly, as a result of this, in quantum mechanical systems, certain quantities, such as energy and momentum are only allowed to come in discrete chunks or 'quanta'. When we measure something in quantum mechanics we force or 'project' the system into a state which has one definite value for whatever we are measuring. In fact you generally **cannot** measure a quantum system without changing it, even in principle, and even if your measuring apparatus is perfect. It is worth pointing out that even though quantum mechanics usually describes small systems like atoms and electrons, some quantum systems can be pretty big, for instance superconducting circuits can behave quantum mechanically and can be millimetres to centimetres across.

If you measure one property of a system, you might change a different property, and therefore both properties cannot be known exactly at the same time, this is known as the *Heisenberg uncertainty principle* (no relation to Walter White). Because the state of a quantum system is fundamentally uncertain, these systems have some strange behaviours, one of which is *quantum tunnelling*.

If I take a ball, (a classical particle) and trap it behind a hill but I don't give it enough momentum to go over the hill, it will be trapped, and no matter how long I wait, it will never get over the hill. On the other hand, if I do the same thing with a quantum particle, replacing the hill with a region that it costs energy to cross, it will eventually escape (although this may take a long time) through a phenomena known as *quantum tunnelling*. Because quantum systems can do things which regular classical systems cannot, they might be able to help us to do calculations in ways which don't make sense classically, for instance in the case of the travelling salesperson problems to tunnel from a bad solution to a better solution.



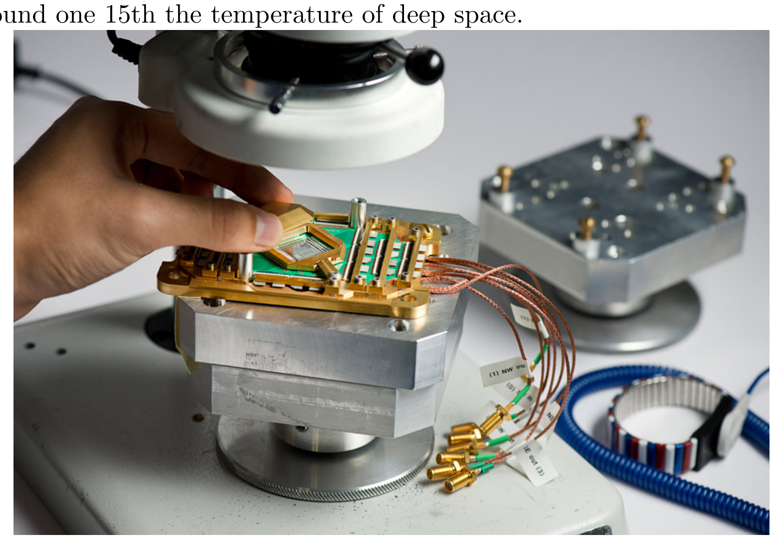
Top: Classical ball trapped behind a hill. Bottom: quantum particle is able to tunnel and escape. Image credit: I drew this, feel free to reuse without attribution.

As I mentioned before, many quantum mechanical things are very hard to simulate, exactly because it has such unfamiliar behaviours, therefore another use for an analog quantum machine is to simulate quantum things which exist in nature. This is a particularly important problem because the behaviour of electrons, which determines all of chemistry is both quantum and very hard to simulate. This idea, known as *quantum simulation* isn't what I will talk about for the rest of this document, but it is worth mentioning as a very cool application.

5 Putting it all Together: a Quantum Annealer

Now that we have all of the ingredients, we can think of how think of an entirely different way of solving problems from the way the computers we are used to do. If we build a quantum system where the energy is mapped to the quality of solutions to interesting problems, make it quantum, and make it very cold, then it can use quantum tunnelling to try to find the best answer, or at least a better answer than anyone else can find. This idea is called *quantum annealing*. There are also other ways to build an analog quantum computer, but these are a bit more technical, so I will not talk about them here, and of course there are also ideas for *digital* quantum computers, but that is a subject for another time.

Building a device to get nature to use quantum mechanics to solve your problems for you is a hard thing to do, but people have tried to do this, for instance D-Wave Systems Inc. has been building progressively larger *quantum annealers* (and have been able to sell a few for around $10 million each). These devices are based on circuits, like the ones in your desktop or laptop computer, but which are made out of special materials, magnetically shielded, and cooled to around one 15th the temperature of deep space.



A D-Wave quantum annealer being built. Image credit: Courtesy of D-Wave Systems Inc..