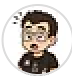
What’s the difference between Extropic, Normal Computing, and D-Wave?

Mapping the energy landscape.



ZACH
MAR 22, 2024

Note: Since this article was published, I worked at Normal Computing as their silicon engineering lead for a bit over a year. I’m currently an advisor there.

I’m leaving the article as-is — for those of you reading this in the future, I promise this isn’t just propaganda.

A couple weeks back, Extropic AI, the company founded by ex-Google X researcher Guillaume Verdon and Trevor McCourt, released their Litepaper detailing their superconducting, thermodynamic computer. It’s lacking in many details, but it does present a decent overview of the basic idea of their architecture. Still, judging by th overall response online, it seems like they’ve confused more people than they’ve satisfied. Personally, I believe the litepaper was missing some important context ab the world of energy-based computing, and how Extropic fits in with the overall bod of work on that subject that’s been developed *since 1954*.

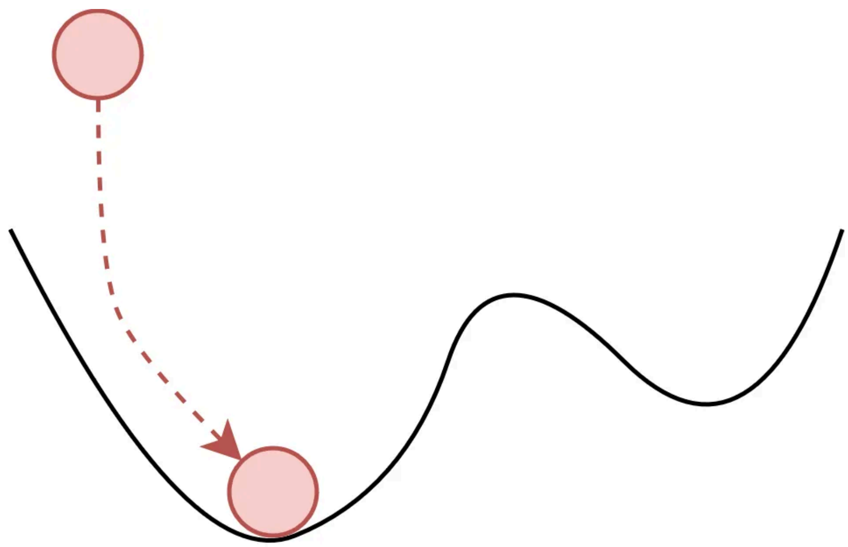
That’s right -- energy-based computing is older than the integrated circuit! Both Jo von Neumann and Eiichi Goto separately invented computers that used coupled electrical oscillators to perform calculations. Goto’s architecture, called the Parametron, was the basis of the PC-1, one of the first general-purpose computers built in Japan. Goto later went on to develop the quantum flux parametron, extendi his original architecture using superconducting Josephson junctions.

There have been a number of modern projects carrying Goto's legacy forward. Both Extropic and Normal Computing, which was also founded by former Google X engineers, are developing energy-based "thermodynamic computers", each with a different circuit technology. There's also D-Wave, which leverages a quantum form energy-based computing called quantum annealing. On the academic research side Chris Kim's group at the University of Minnesota has developed energy-based coup oscillator computers that can be implemented using normal chip technology. And Purdue has a research group devoted to investigating probabilistic spin logic, anoth form of energy-based computing that leverages specialized devices called magnetic tunnel junctions.

So, what's the difference between all of these methods? Let's start with the simples all of them: annealers.

Annealers

Annealers are specialized computers designed to use physics to minimize the energ of a system. You can think of an annealer as a ball dropped into an empty bowl. Du the force of gravity, the ball will roll down to the bottom of the bowl. Using physics the ball has "solved" the problem of finding the lowest point in the bowl. In the san way, an annealer uses physics to find the lowest-energy configuration of a physical system and solve a computational problem. Instead of a bowl and a ball, though, annealers are constructed from electronic components, and they leverage electromagnetic or quantum physics to minimize the energy of a system.

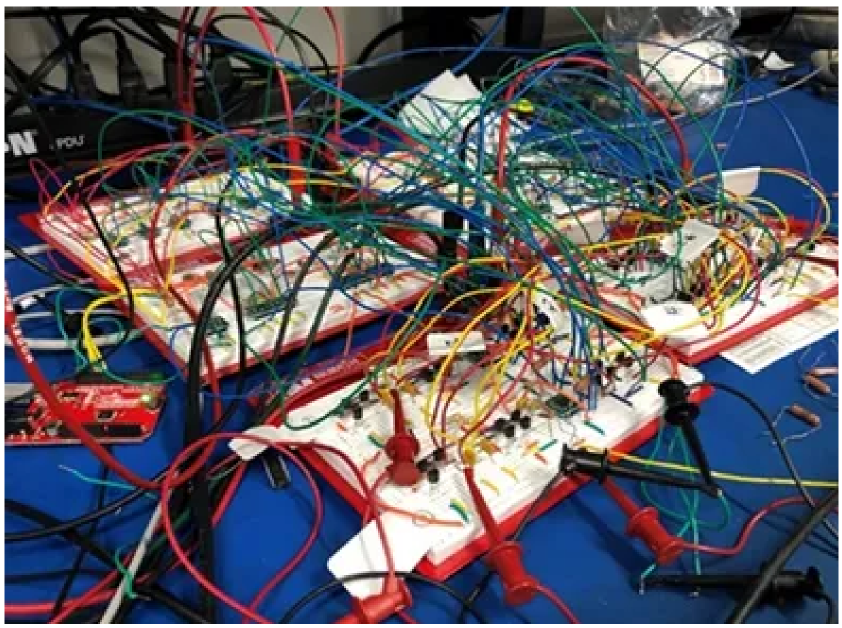


It turns out that these sorts of energy-minimization problems are both important a hard. Many of these problems are NP-complete, which means that computers take exponentially more time to solve them as they grow larger and more complex. At th same time, they have applications in many fields, like drug discovery, finance, logistics, and machine learning. So it's no surprise that researchers have tried many ways to build specialized computers to solve these problems.

Analog Annealers

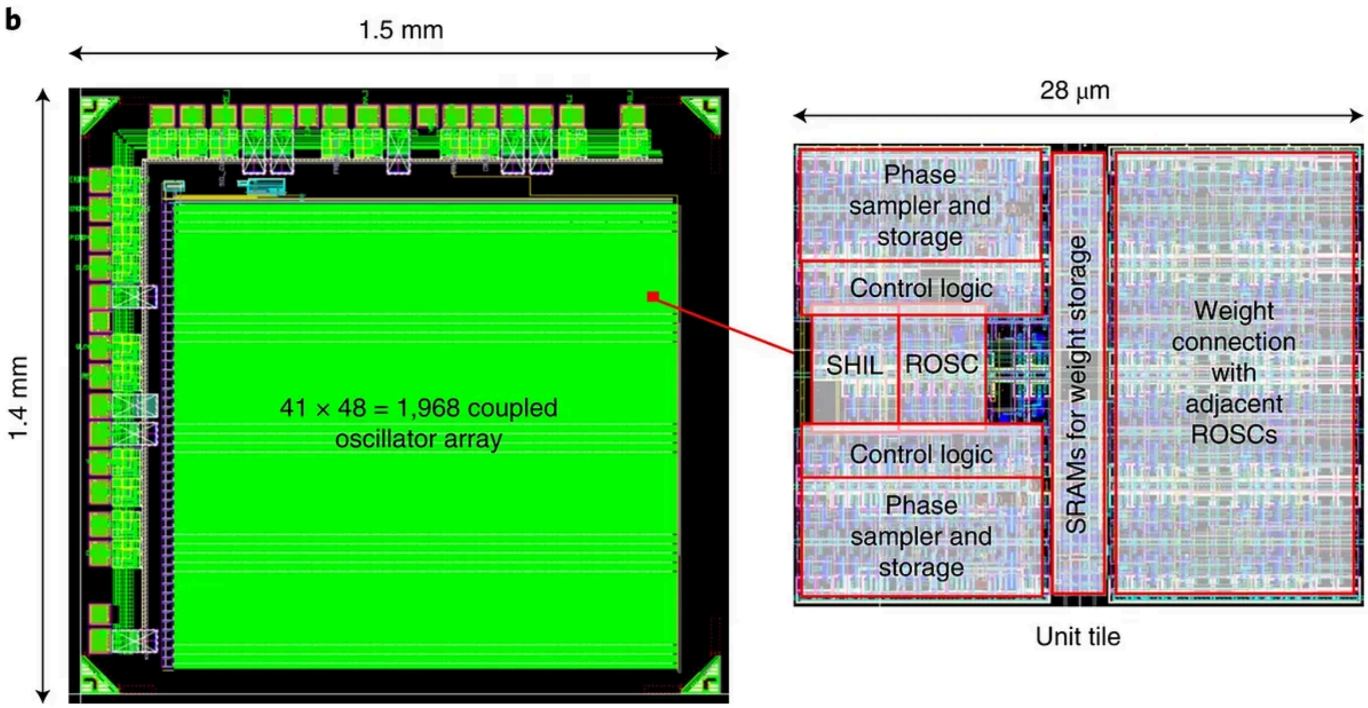
Analog solvers consist of a number of electronic oscillators -- specialized circuits designed to produce a periodic signal -- as well as analog components to allow thos oscillators to interact with one another. The strength of the interactions between tl oscillators are programmable, and define the input values to the computation being performed. By changing how the oscillators interact, the shape of the energy landscape changes, and the annealer generates a different result.

A group from MIT demonstrated a functioning analog annealer in 2019, using inductor-capacitor oscillators to solve small energy minimization problems. It was exciting proof-of-concept, but also was fairly large and a bit of a mess. There's a somewhat fundamental problem with using inductor-capacitor oscillators: capacitc are fairly hard to shrink, and inductors are extremely hard to shrink. So this design would be difficult to implement with many more oscillators, which is a key part of making it practical for solving real-world energy-minimization problems. The two authors on the MIT Nature paper actually founded a company, Sync Computing, to commercialize their work, but quickly pivoted to cloud orchestration software.



Chou et. al., 2019

Luckily, Chris Kim's group at the University of Minnesota have been figuring out practical and scalable ways to build analog annealers. Instead of using inductors an capacitors, they build their oscillators using logic gates. These kinds of oscillators, called ring oscillators, are much more practical to implement using conventional cl technology, which makes building large systems of coupled oscillators easy. They'v demonstrated a 1968-oscillator solver with local oscillator interactions, and a 50-no solver with all-to-all oscillator interactions.

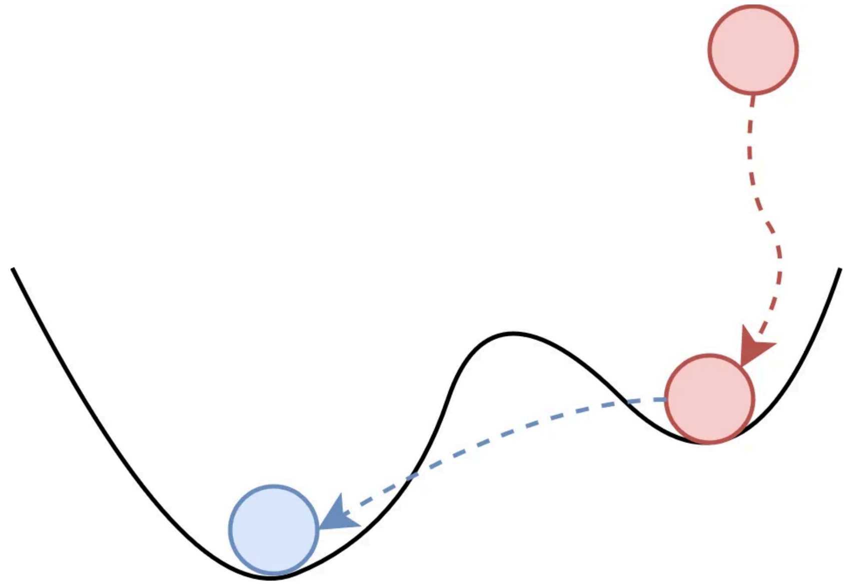


Moy et. al., 2022

Personally, the results from Chris Kim's lab make me really excited about the future analog annealers. They've definitively proved that these systems can scale to thousands of spins, the key first step in building annealers that are practical for real world problems.

Quantum Annealers

Quantum annealers solve one major problem that conventional annealers struggle with: local minima in the energy landscape. To return to the "ball-and-bowl" analogy it's possible for our ball to get stuck at a point that's not actually the lowest point in our empty bowl, as shown below in red. Here, the ball is stuck in the small dip on the right, and can't progress down to the actual lowest point on the left. Normally, annealers add noise to try to get the ball to "jump out" of the local minimum and make its way to the lowest point, but this isn't always effective.



Quantum annealers let our ball "quantum tunnel" right through the energy landsca and jump directly to the minimum-energy solution. This behavior can't be replicate by non-quantum annealers and gives quantum annealers a fundamental speedup or hard graph problems with many deep local minima.

The most well-known quantum annealer company is [D-Wave](). They've been around since 1999, and released their first commercial system, the D-Wave One, in 2011. Si then, their systems have grown from 128 qubits in the D-Wave One to over 5000 in D-Wave Advantage. D-Wave claims that these systems demonstrate "quantum supremacy": the ability of their quantum computers to quickly solve problems that non-quantum computers can't. However, there's been some controversy over wheth D-Wave systems can actually solve problems faster than non-quantum approaches i practice. Recently, D-Wave released a [preprint showing quantum supremacy]() for a r world simulation workload; it remains to be seen whether the result holds up.

D-Wave systems use superconducting Josephson junctions to solve optimization problems, which enables the quantum tunneling behavior that makes them so efficient. However, they're also far more costly. D-Wave's systems rely on superconducting Josephson junctions, which have to be cooled below 10 Kelvin.

Classical annealers, on the other hand, are fairly cheap and scalable, making them better suited to more mainstream use-cases.

D-Wave isn't the only player working on quantum annealers, though. There are alsc number of research labs working on [Coherent Ising Machines](#) (CIMs), which are systems of coupled optical parametric oscillators. CIMs are particularly exciting because they allow any oscillator to be coupled to any other oscillator; this sort of dense coupling is not supported on D-Wave systems or on many of the larger analo annealers. They can also reach immense scales of [over 100,000 oscillators](#). On the o hand, these systems are incredibly complicated and esoteric, often requiring multip kilometers of specialized optical fiber cavities that need to be carefully stabilized. I my opinion, CIMs are an incredibly promising architecture for energy-based computing, but currently have more fundamental engineering challenges than anal or superconducting annealers.

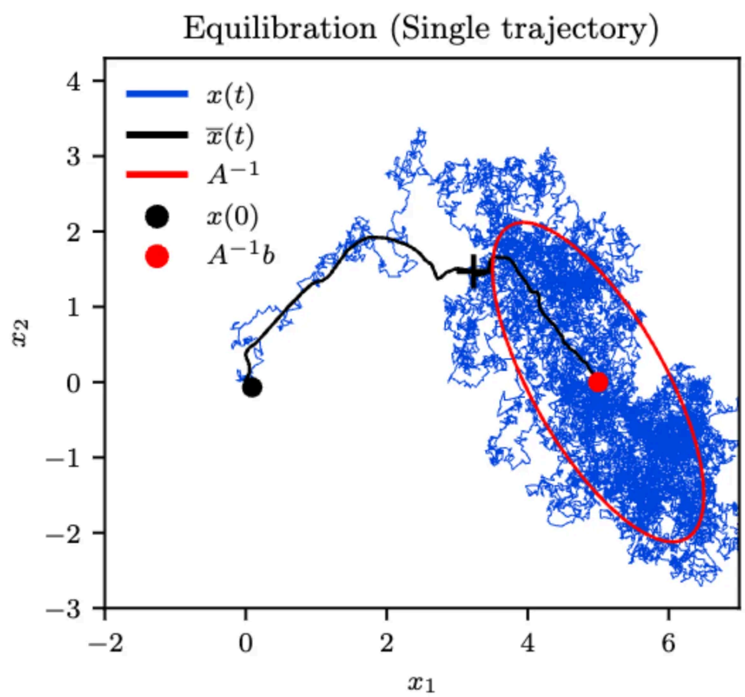
All of these energy-based annealer architectures, though, ultimately do the same th find the minimum of an energy landscape. What if you could do more with a system coupled oscillators, though? Luckily, there are some companies and researchers try to do just that.

Thermodynamic Computers

Thermodynamic computing is fairly similar to annealers, but with a twist: instead c just observing the final value that the system converges to, the computers also observes its dynamic behavior around equilibrium. With richer and more complex c from the physical system, the energy-based computer can perform additional calculations that an annealer by itself couldn't.

Normal Computing, a startup working on thermodynamic computing systems, [released a paper in 2023](#) demonstrating a working thermodynamic computer implemented on a PCB. Like an annealer, their system settles to a stable point.

However, Gaussian noise is injected into each cell, causing the system to randomly fluctuate around that stable point. By measuring the dynamics around that point, tl thermodynamic computer can calculate more complex quantities, like the inverse o matrix.



[Aifer et. al., 2023]

This is an exciting step past what annealers can do! But also, their existing hardwar implementation has the same drawbacks as that early annealer developed by MIT: relying on inductor-capacitor oscillators makes the system difficult to scale up.

As far as I can tell from Extropic's public announcement, they're building a system with similar principles but a different physical manifestation. Instead of inductor-capacitor oscillators, they use superconducting Josephson junctions. By leveraging incredibly cold, small, and low-power elements, you don't need to inject noise into coupled elements. Instead, the system fluctuates around its equilibrium point due t naturally-occuring thermal noise. Of course, the downsides here are that superconductors require incredibly cold temperatures (~1 Kelvin for the aluminum Extropic uses), and the fact that Josephson junctions aren't particularly small or scalable either. Extropic mentioned that they're also working on thermodynamic computers that use normal chip technology, rather than superconductors, but they haven't shared any more information there.

Personally, I think techniques that move away from purely analog approaches and towards mixed signal ones may show promise for thermodynamic computers too. F annealers, ring-oscillator architectures offered more compact silicon implementati than LC circuits could. But it remains to be seen whether ring-oscillator technique could be leveraged for thermodynamic computation, rather than just annealers.

Probabilistic Logic

The last kind of energy-based computing that I think is particularly exciting is [probabilistic logic], which has mostly been developed by Supriyo Datta's group at Purdue. Using magnetic tunnel junctions (MTJs), they developed a device that randomly fluctuates between two stable states. These MTJs can be connected to bu Boltzmann machines, specialized neural networks that are designed to reach low-energy states with high probability. These devices can not only solve energy-minimization problems, but also additional problems like [integer factorization].

Magnetic tunnel junctions are easier to manufacture at scale than inductor-capacit oscillators, and easier to deploy than superconductors. As a matter of fact, massive arrays of magnetic tunnel junctions are the basis of [MRAM], a new kind of dense nonvolatile memory being deployed at major semiconductor fabs like GlobalFoundries. It might be feasible to slightly modify the materials used by MRA and deliver dense arrays of probabilistic bits.

However, probabilistic logic using magnetic tunnel junctions has not been demonstrated at scale; the most mature work is still in a proof-of-concept stage, wi systems consisting of four to eight bits. Of all of the energy-based computing scher here, MTJ-based logic may have the most engineering challenges in its future, and some of those challenges may be entirely unknown at this point.

To help solve the challenge of scale, Purdue has also demonstrated a probabilistic l emulator that can be deployed on an FPGA. By replacing analog coupling with digi logic, and magnetic tunnel junctions with pseudorandom number generators, they constructed a much larger scale probabilistic system than they could with discrete MTJ cells. It's not particularly compact, with a 32-bit adder based on 434 p-bits tak over 30,000 logic cells and over 18,000 registers, but the ability to easily deploy 434 bits is unprecedented.

And they're not the first group to try to use digital logic to emulate the advantages energy-based computing!

Digital Emulation

Multiple companies, looking for short-term performance advantages, have tried to digital chips to emulate the speedups that analog annealers can deliver. The most prominent is Fujitsu's Digital Annealer, a chip that leverages parallel spin updates quickly solve an energy-minimization problem. It's powerful, and already deployed scale, but lacks the analog coupling dynamics that make analog annealers converge quickly.

Hitachi is also researching digitally emulating annealers. They developed an anneal chip that achieved much higher density than Fujitsu's -- 30,000 rather than 8192 -- leveraging ultra-compact near-memory computing and by sacrificing connectivity. While these digital emulators offer large spin counts and scalable implementations the short term, their lack of coupling dynamics mean that they will lose out to both analog and quantum annealers in the long run, as problem sizes increase and annea technology improves.

The Verdict

First of all, I think it's incredibly exciting to see so many new kinds of computing being explored. All of these technologies have the potential to offer huge speedups over conventional digital computers for certain workloads. Even better, those workloads are usually difficult and important problems, like graph optimization, matrix math, and machine learning. But all of these architectures will face challeng going from a proof-of-concept to real-world, practical adoption.

I think one of the biggest challenges these new accelerators will face is accessibility Usually, the best way to figure out how to leverage a new architecture is to get it in the hands of a community of researchers, who can develop algorithms better than a small in-house team. To do that, you need a reasonably large number of chips, whic means that the manufacturability of a chip really matters.

All of these accelerators, with the exception of digital emulators run on FPGAs, require manufacturing custom hardware. That's why I think that designs that can b manufactured using conventional chip processes have a huge leg up. Manufacturin conventional chips at scale is easy and cost-effective, while the manufacturing processes for magnetic tunnel junctions are much more complex. Superconductors only lack scaled manufacturing processes, but need extremely cold temperatures to operate, reducing accessibility even further.

This makes ring-oscillator-based annealers the most likely candidate for short term large-scale adoption. However, thermodynamic computers are more powerful than annealers in many ways -- we just haven't seen a practically scalable thermodynami

implementation from any company or research group yet. I think there could be a lo
of promise in potentially leveraging coupled ring oscillators for thermodynamic computation rather than purely annealing. It remains to be seen whether or not any
the companies or research groups working on thermodynamic computing go down that path.

---

9 Likes

Discussion about this post

Comments Restacks

Write a comment...

Grigory Sapunov May 27, 2024

Thanks for the explanation and comparison! I wrote a post about Normal Computing approach (https://gonzoml.substack.com/p/thermodynamic-ai-is-getting-hotter) and now tried to underst
what does Extropic do and how does it compare to Normal. Thanks for your post, now I understa
better!

LIKE REPLY