Science & Technology

WHY ALGORITHMS SUCK AND ANALOG COMPUTERS ARE THE FUTURE

f

𝕏

G+

in

• • •

Bernd Ulmann | 06.07.2017

Analog computing, which was the predominant form of high-performance computing well into the 1970s, has largely been forgotten since today's stored program digital computers took over. But the time is ripe to change this.

I know this article's headline may sound slightly provocative, but it's true: Algorithmic computing just doesn't scale too well when it comes to problems requiring immense amounts of processing power.

Just have a look at the latest Top500-list of high-performance computers. Currently, the most powerful supercomputer is the Sunway TaihuLight at the National

Supercomputing Center in Wuxi, China. This behemoth delivers a whopping 93 petaflops (one petaflop equals a quadrillion floating point operations per second), which is really unfathomable.

SUPERCOMPUTERS AND ENERGY CONSUMPTION

However, such enormous processing power comes at a cost. In this case it requires 10.649.600 processing units, so-called cores, that consume 15.371 megawatts – an amount of electricity that could power a small city of about 16.000 inhabitants based on an average energy consumption equal to that of San Francisco.

If we turn our attention away from the most powerful supercomputers and focus on the most energy-efficient ones instead, we'll find the system TSUBAME3.0 at the Tokyo Institute of Technology on top of the Green500-list. It delivers 14.11 gigaflops/watt with an overall power consumption of "only" 142 kilowatts required by its 36288 cores. In terms of energy efficiency, this is about 2 times better than TaihuLight. But 142 kilowatts is still an awful lot of electrical power.

But aren't there any supercomputers that don't require this much energy to run? Indeed there are: The human brain is a great example – its processing power is estimated at about 38 petaflops, about two-fifths of that of TaihuLight. But all it needs to operate is about 20 watts of energy. Watts, not megawatts! And yet it performs tasks that – at least up until now – no machine has ever been able to execute.

What causes this difference? First of all, the brain is a specialized computer, so to speak, while systems like TaihuLight and TSUBAME3.0 are much more general-purpose machines, capable of tackling a wide variety of problems.

Machines like these are programmed by means of an algorithm, a so-called program. Basically this is a sequence of instructions that each processor executes by reading them from a memory subsystem, decoding instructions, fetching operands, performing the requested operation, storing back the results etc.

> **"Aren't there any supercomputers that don't require this much energy to run? Indeed there are: The human brain is a great example."**

AN ECONOMICAL SUPERCOMPUTER: THE HUMAN BRAIN

All these memory requests require a substantial amount of energy and slow things down considerably. Having hundreds of thousands of individual cores also requires an intricate interconnect fabric to exchange data between cores as required which also adds to the time and power necessary to perform a computation.

A brain, on the other hand, works with a completely different approach. There is no program involved in its operation. It is simply "programmed" by the interconnections between its active components, mostly so-called neurons.

The brain doesn't need to fetch instructions or data from any memory, decode instructions etc. Neurons get input data from other neurons, operate upon this data and generate output data which is fed to receiving neurons. OK – this is a bit oversimplified but it is enough to bring us to the following questions:

1. **Isn't there an electronic equivalent of such a computer architecture?**

2. **Do we really need general-purpose supercomputers?**

3. **Could a specialized system be better for tackling some problems?**

f

🐦

G+

in

• • •

The answer to the first question is: Yes, there is an electronic equivalent of such an architecture.

It's called an analog computer.

ENTER ANALOG COMPUTERS

"Analog" derives from the Greek word "analogon" which means "model". And that's exactly what an analog computer is: A model for a certain problem that can then be used to solve that very problem by means of simulating it.

So-called direct analogs – relying on the same underlying principle as the problem being investigated – form the simplest class of such analog methods. Soap bubbles used to generate minimal surfaces are a classic example for this.

"An analog computer is a model for a certain problem that can then be used to solve that very problem by means of simulating it."

A more generally usable class contains indirect analogies which require a mapping between the problem and the computing domain. Typically such analogs are based on analog electronic circuits such as summers, integrators and multipliers. But they can also be implemented using digital components in which case they are called digital differential analyzers.

Both approaches share one characteristic: There is no stored program that controls the operation of such a computer. Instead, you program it by changing the interconnection between its many computing elements – kind of like a brain.

FROM ANALOG TO HYBRID COMPUTERS

Consider my own analog computer prototype, which is an example of a classic analog-electronic computer. The crisscross of wires can be described as the program of the machine. In this case it is set up to simulate the airflow around a so-called Joukowsky airfoil – not a trivial task.
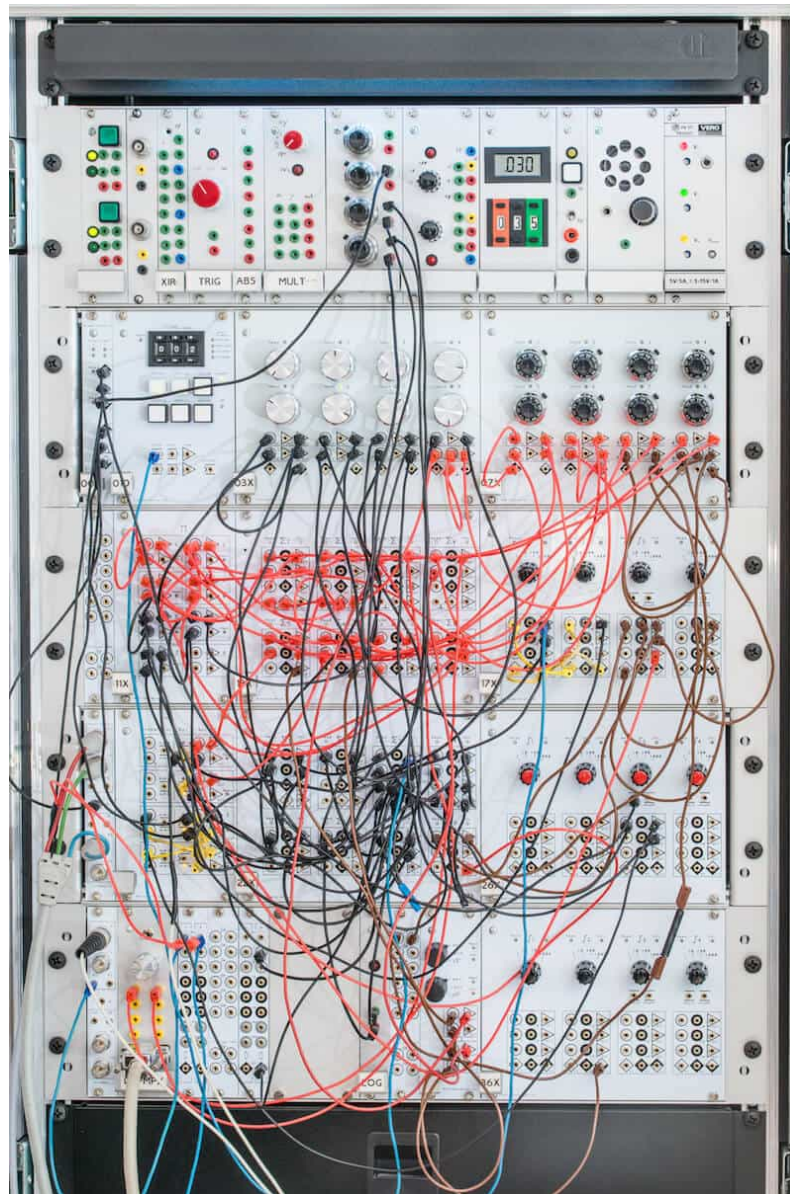
This picture has been taken by Tibor Florestan Pluto and shows the author's own analog computer prototype.

All of the machine's computing elements work in complete parallelism with no central or distributed memory to access and to wait for. You program the computer by plugging the patch cords in and out. This can be quite useful for a single researcher and for teaching purposes, but for a true general-purpose analog computer one would have to get rid of this manual patching procedure.

Luckily, with today's electronic technology it is possible to build integrated circuits containing not only the basic computing elements but also a crossbar that can be programmed from an attached digital computer, thus eliminating the rat's nest of wires altogether.

Our latest articles, sent to your inbox once a month.

Enter your e-mail address

SIGN UP NOW   ▶

Such analog computers reach extremely high computational power for certain problem classes. Among others, they are unsurpassed for tackling problems based in differential equations and systems thereof – which applies to many if not most of today's most relevant problems in science and technology.

For instance, in a 2005 paper , Glenn E. R. Cowan described a Very-Large-Scale-Integrated Analog Computer (VLSI), i.e. an analog computer on a chip, so to speak. This chip delivered whopping 21 gigaflops per watt for a certain class of differential equations, which is better than today's most power-efficient system in the Green500-list.

THE ANALOG FUTURE OF COMPUTING

Analog computing, which was the predominant form of high-performance computing well into the 1970s, has largely been forgotten since today's stored program digital computers took over. But the time is ripe to change this.

Tomorrow's applications demand stronger computing powers at much lower energy consumption levels. But digital computers simply can't provide this out of the box. So it's about time to start developing modern analog co-processors that take off the load of solving complex differential equations from traditional computers. The result would be so-called hybrid computers.

f

🐦

G+

in

...

> **"Tomorrow's applications demand stronger computing powers at much lower energy consumption levels. Digital computers can't provide this."**

These machines could offer higher computational power than today's supercomputers at comparable energy consumption or could be used in areas where only tiny amounts of energy are available, like implanted controllers in medicine and other embedded systems.

FUTURE CHALLENGES

Clearly analog computing holds great promise for the future. One of the main problems to tackle will be that programming analog computers differs completely from everything students learn in university today.

In analog computers there are no algorithms, no loops, nothing as they know it. Instead there are a couple of basic, yet powerful computing elements that have to be

interconnected cleverly in order to set up an electronic analog of some mathematically described problem.

Technological challenges such as powerful interconnects or highly integrated yet precise computing elements seem minor in comparison to this educational challenge. In the end, changing the way people think about programming will be the biggest hurdle for the future of analog computing.

[The author would like to thank Malcolm Blunden for proofreading this post. Images by Tibor Florestan Pluto.]

f
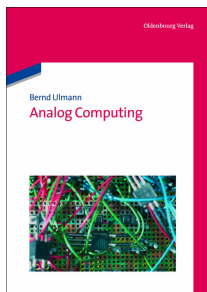
Bernd Ulmann

Analog Computing

Bernd Ulmann

Bernd Ulmann is a professor for business informatics at the FOM (Frankfurt/Main). He studied mathematics at the Johannes Gutenberg University Mainz with philosophy as his minor.

He wrote his diploma thesis about cryptographically secure pseudorandom number generators and his Ph.D. thesis at the University

Hamburg about analog computing, the field he is obsessed with (to quote his wife and friends). He devotes his leisure time to collect and restore historic analog computers and devise modern analog computers , and loves programming both of them.

SUBSCRIBE TO OUR NEWSLETTER

Our latest articles, sent to your inbox once a month.

f

Enter your e-mail address

y

SIGN UP NOW ▶

G+

in

• • •

VISIT OUR SHOP

De Gruyter publishes over 1,300 new book titles each year and more than 750 journals in the humanities, social sciences, medicine, mathematics, engineering, computer sciences, natural sciences, and law.

VISIT SHOP