TOOLBOX

ACCELERATE YOUR SCIENCE WITH GRAPHICS CARDS

Graphics processing units aren't just of interest to gamers and cryptocurrency miners — parallel processing is increasingly being used to turbocharge scientific research.

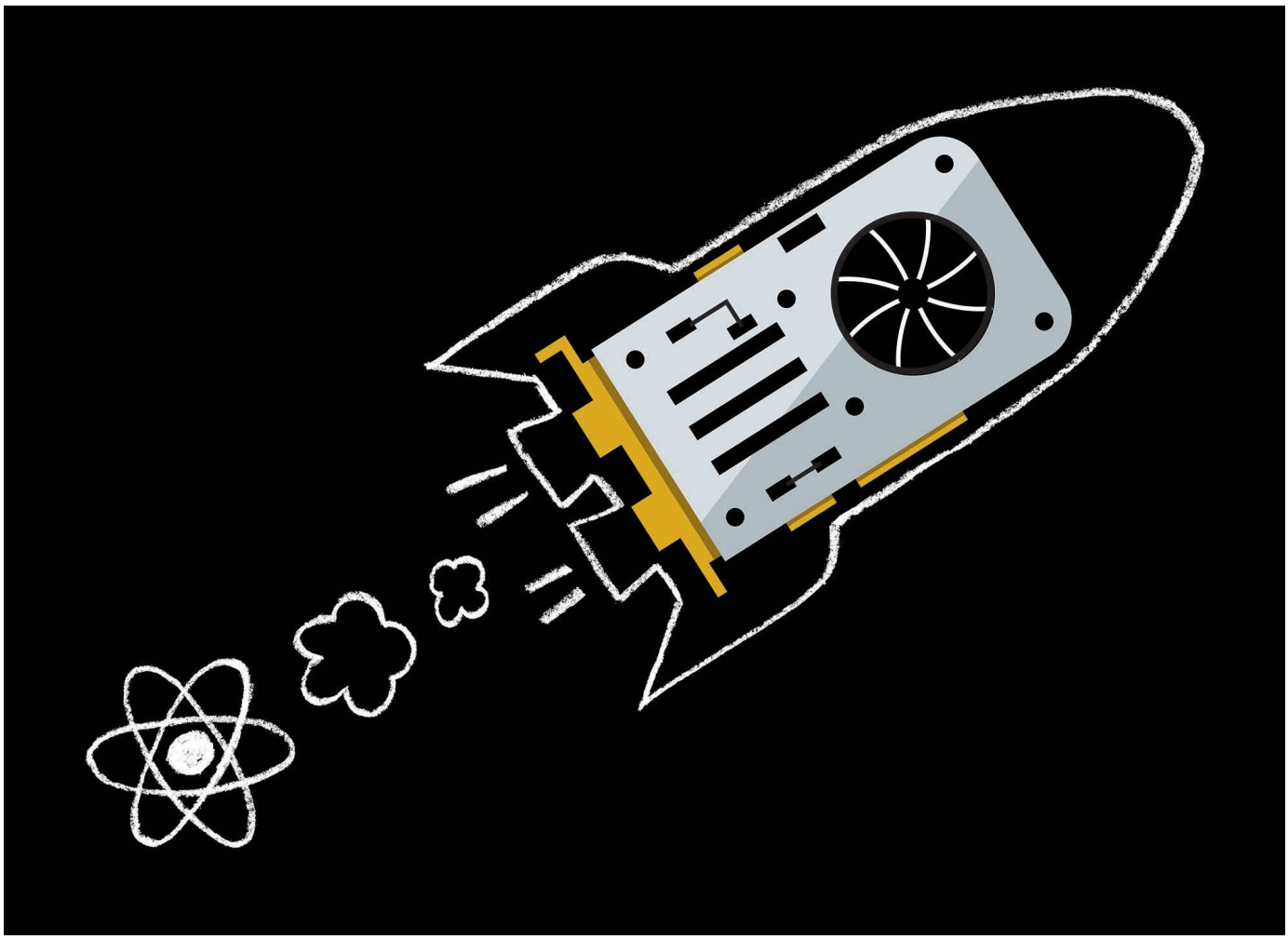

ILLUSTRATION BY THE PROJECT TWINS

BY DAVID MATTHEWS

Evan Schneider, an astrophysicist at Princeton University in New Jersey, spent her doctorate learning to harness a chip that's causing a quiet revolution in scientific data processing: the graphics processing unit (GPU).

Over the past decade or so, GPUs have been challenging the conventional workhorse of computing, the central processing unit (CPU), for dominance in computationally intensive work. As a result, chips that were designed to make video games look better are now being deployed to power everything from virtual reality to self-driving cars and cryptocurrencies.

Of the 100 most powerful supercomputing clusters in the world, 20 now incorporate GPUs from NVIDIA, one of the leading chipmakers. That includes the world's fastest computer, the Summit cluster at the US Department of Energy's Oak Ridge National Laboratory in Tennessee, which boasts more than 27,000 GPUs.

For Schneider, running astrophysical models on GPUs "has opened up a new area of my work that just wouldn't have been accessible without doing this". By rewriting her code to run on GPU-based supercomputers rather than CPU-focused ones, she has been able to simulate regions of the Galaxy in ten times as much detail, she says. As a result of the increase in resolution, she explains, the entire model now works differently — for example, giving new insights into how gas behaves on the outskirts of galaxies.

Put simply, GPUs can perform vastly ▶

more calculations simultaneously than CPUs, although these tasks have to be comparatively basic — for example, working out which colour each pixel should display during a video game. By comparison, CPUs can bring more power to bear on each task, but have to work through them one by one. GPUs can therefore drastically speed up scientific models that can be divvied up into lots of identical tasks, an approach known as parallel processing.

There are no hard rules about which types of calculation GPUs can accelerate, but proponents agree they work best when applied to problems that involve lots of things — atoms, for example — that can be modelled simultaneously. The technique has become particularly prevalent in fields such as molecular dynamics, astrophysics and machine learning.

Schneider, for example, models galaxies by splitting them up into millions of discrete areas and then dividing the work of simulating each of them between a GPU's many cores, the units that actually execute calculations. Whereas CPUs normally have at most tens of cores — the number has increased as they themselves have become more parallel — GPUs can have thousands. The difference, she explains, is that whereas each CPU core can work autonomously on a different task, GPU cores are like a workforce that has to carry out similar processes. The result can be a massive acceleration in scientific computing, making previously intractable problems solvable. But to achieve those benefits, researchers will probably need to invest in some hardware or cloud computing — not to mention re-engineering their software.

A BIT AT A TIME

Fundamentally, parallelizing work for GPUs means breaking it up into little pieces and farming those pieces out to individual cores, where they are run simultaneously. There are several ways scientists can get started. One option, Schneider says, is to attend a parallel-processing workshop, where "you'll spend literally a day or two with people who know how to program with GPUs, implementing the simplest solutions".

OpenACC, for instance, is a programming model that allows scientists to take code written for CPUs and parallelize some processes; this allows researchers to get a feel for whether their code will run significantly faster on GPUs, Schneider says. The OpenACC community runs regular workshops and group programming events called hackathons to get scientists started on porting their code to GPUs. When contemplating whether to experiment with GPUs, "the first thing to think about is: 'Is there a piece of my problem where everything could be done in parallel?'", says Schneider.

The next step is to rewrite code specifically to take advantage of GPUs. To speed up his code, Philipp Germann, a systems biologist at the European Molecular Biology Laboratory in Barcelona, Spain, learnt CUDA, an NVIDIA-created parallel processing architecture that includes a language similar to C++ that is specifically designed for GPUs. "I'm definitely not a particularly experienced programmer," he admits, but says the tool took only around two weeks to learn.

One cell-modelling program that Germann and his colleagues created proved to be two to three orders of magnitude faster on GPUs than were equivalent CPU-based programs. "We can really simulate each cell — when will it divide, how will it move, how will it signal to others," Germann says. "Before, that was simply not possible."

CUDA works exclusively on NVIDIA chips; an alternative tool, the open-source OpenCL, works on any GPU, including those from rival chipmaker AMD. Matthew Liska, an astrophysicist at the University of Amsterdam, prefers CUDA for its user-friendliness. Liska wrote GPU-accelerated code simulating black holes as part of a research project (M. Liska *et al. Mon. Not. R. Astron. Soc. Lett.* **474**, L81–L85; 2018); the GPUs accelerated that code by at least an order of magnitude, he says. Other scientists who spoke to *Nature* also said CUDA was easier to use, with plenty of code libraries and support available.

> ***"Researchers typically have seen speed-ups from a factor of two to an order of magnitude."***

You might need to block off a month or two to focus on the conversion, advises Alexander Tchekhovskoy, an astrophysicist at Northwestern University in Evanston, Illinois, who was also involved in Liska's project. Relatively simple code can work with little modification on GPUs, says Marco Nobile, a high-performance-computing specialist at the University of Milan Bicocca in Italy who has co-authored an overview of GPUs in bioinformatics, computational biology and systems biology (M. S. Nobile *et al. Brief. Bioinform.* **18**, 870–885; 2017). But, he warns, for an extreme performance boost, users might need to rewrite their algorithms, optimize data structures and remove conditional branches — places where the code can follow multiple possible paths, complicating parallelism. "Sometimes, you need months to really squeeze out performance."

And sometimes, the effort simply isn't worth it. Dagmar Iber, a computational biologist at the Swiss Federal Institute of Technology in Zurich, says that her group considered using GPUs to processes light-sheet microscopy data. In the end, the researchers managed to get acceptable results using CPUs, and decided not to explore a GPU acceleration because it would have meant making too many adaptations. And not all approaches will work, either: one of Tchekhovskoy's attempts yielded no speed improvement whatsoever over a CPU. "You can invest a lot of time, and you might not get much return on your investment," he says.

More often than not, however, that's not the case; he says that in the field of fluid dynamics, researchers typically have seen speed-ups "from a factor of two to an order of magnitude".

CLOUD PROCESSING

As for the hardware itself, all computers need a CPU, which acts as the computer's brain, but not all come with a dedicated GPU. Some instead integrate their graphics processing with the CPU or motherboard, although a separate GPU can normally be added. To add multiple GPUs, users might need a new motherboard with extra slots, says Liska, as well as a more robust power supply.

One way to test out parallel processing without having to actually buy new hardware is to rent GPU power from a cloud-computing provider such as Amazon Web Services, says Tim Lanfear, director of solution architecture and engineering for NVIDIA in Europe, the Middle East and Africa. (See, for example, this computational notebook, which exploits GPUs in the Google cloud: go.nature.com/2ngfst8.) But cloud computing can be expensive; if a researcher finds they need to use a GPU constantly, he says, "you're better off buying your own than renting one from Amazon".

Lanfear suggests experimenting with parallel processing on a cheaper GPU aimed at gamers and then deploying code on a more professional chip. A top-of-the-range gaming GPU can cost US$1,200, whereas NVIDIA's Tesla GPUs, designed for high-performance computing, have prices in the multiple thousands of dollars. Apple computers do not officially support current NVIDIA GPUs, only those from AMD.

Despite the growth of GPU computing, the technology's progress through different scientific fields has been patchy. It has reached maturity in molecular dynamics, according to Lanfear, and taken off in machine learning. That's because the algorithm implemented by a neural network "can be expressed as solving a large set of equations", which suits parallel processing. "Whenever you've got lots of something, a GPU is typically good. So, lots of equations, lots of data, lots of atoms in your molecules," he says. The technology has also been used to interpret seismic data, because GPUs can model millions of sections of Earth independently to see how they interact with their neighbours.

But as a practical matter, harnessing that power can be tricky. In astrophysics, Schneider estimates that perhaps 1 in 20 colleagues she meets have become adopters, held back by the effort it takes to rewrite code.

Germann echoes that sentiment. "More and more people are recognizing the potential," he says. But, "I think still quite a few labs are scared to some degree" because "it has the reputation of being hard to program". ■

David Matthews *is a freelance writer based in Berlin.*