Make some noise: the mathematical theories behind data privacy

From shopping online to posting on social media, we generate data about ourselves all the time. To keep us safe and avoid cybercrime, it is essential our data is private. However, building efficient algorithms that maintain data privacy is challenging, especially as cybercriminals become increasingly sophisticated in their operations. **Dr Clément Canonne** is a theoretical computer scientist at **The University of Sydney** in Australia. He is researching how advanced computational techniques can best future-proof our data, while maintaining efficiency.

Dr Clément Canonne

ARC DECRA Fellow, Senior Lecturer, School of Computer Science, The University of Sydney, Australia

PROFILE

Fields of research
Theoretical computer science, computational learning, data privacy

Research project
Investigating fundamental trade-offs between data, computation and privacy

Funder
This work was supported by the Australian Research Council, through an ARC Discovery Early Career Researcher Award (DE230101329)

doi: 10.33424/FUTURUM644

Talk like a ...

theoretical computer scientist

Algorithm — in computation, the procedure – or 'step-by-step recipe' – used by a computer to solve a certain problem

Anonymous — when a person's identity is unknown

Data — information and statistics collected for analysis or reference

Data privacy — an individual's right to control how their personal information is collected and used by others

Differential privacy (DP) — a rigorous mathematical framework that helps guarantee privacy

Random noise — in data, an unpredictable variation within a dataset that does not represent meaningful information

Every time we use a computer or phone, we generate data about ourselves. As more and more applications monitor our activity and use algorithms to learn about our individual interests, lifestyle and networks, concern is growing about how this data may be used – and abused. In particular, if malicious actors get hold of this data, they could use it for any number of nefarious purposes: fraud, impersonation, blackmail, stalking – the list goes on.

This introduces the need for data privacy, especially when sensitive information is involved. If the outputs (the conclusions drawn from such information) can be used to identify individuals, then there is a big privacy issue. But guaranteeing full privacy is very difficult, especially when we consider that the hackers of the future will be using computational tricks that have not yet been invented. This is where theoretical computer science comes in – to understand the fundamental mathematics behind computation and how it can help us build systems that guarantee privacy.

Testing the theory behind data privacy

Dr Clément Canonne is a theoretical computer scientist at The University of Sydney. He studies the mathematical foundations, capabilities and limitations of computation and algorithms. "Most of my work is on pen and paper: coming up with an algorithm, or building one from several existing algorithms," he explains. "Then, I mathematically prove its properties."

Once the mathematical foundations are in place, Clément tests his algorithm's usefulness by coding it and running it using example data. "This is a good

way of testing the accuracy of the algorithm and whether it solves the required task," he says. "What is a lot more difficult, however, is testing how private it is." This is because a malicious party with a lot of time, resources and extra information could exploit the algorithm and the data it contains in ways that are difficult to predict and test.

Anonymity is not enough

At first, data privacy might seem straightforward – it simply requires us not to store any of our personal, identifying information. However, keeping data anonymous does not always prevent malicious parties linking it to specific people. "The example I like to give is a logic puzzle called Einstein's riddle," says Clément. "You are given little bits of information: there are five people, each has a pet, the Norwegian drinks coffee, the tea-drinker has a cat, and so on. Then you're asked: who owns the zebra?" Through deduction and reasoning, it is possible to work out who the zebra belongs to, even though that information is not given explicitly. Malicious parties can use similar techniques with the aid of their own algorithms that can extrapolate such conclusions from seemingly anonymous datasets.

"The key message here is that removing a piece of information doesn't prevent that information from being known," says Clément. "A lot of clues still remain, and a malicious party only needs a few pieces of information to work out your unique identity." This is important for businesses or governments that want to collect information from individuals without compromising their privacy – not to mention important for the individuals involved.

> ***"... a malicious party only needs a few pieces of information to work out your unique identity."***

Differential privacy

Fortunately, there is a mathematically rigorous way of solving this issue, known as differential privacy (DP). DP was invented by Professor Cynthia Dwork, Dr Franck McSherry, Professor Kobbi Nissim and Professor Adam D. Smith in 2006. The significance of their invention is highlighted by their receipt of the prestigious Gödel Prize for outstanding papers in theoretical computer science in 2017.

"DP involves randomising the data-processing algorithms," explains Clément. "This is achieved through the injection of some carefully calibrated 'random noise' into the calculations." This 'noise' means that, whether the data of any one individual is included or omitted, the output will stay the same or extremely similar. This is what makes the algorithm 'differentially private'.

This is important, because it means the output cannot be 'post-processed' to reveal information about an individual. "This future-proofs the output against any possible efforts using any possible methods," says Clément. "No matter how cleverly they try, even with the most powerful computer on the planet, there is no way to learn more from the differentially private output." Doing so is not just computationally impossible, but mathematically impossible.

Is DP worth it?

A common question about DP is whether the introduction of randomised data contaminates the outputs. "Sometimes people don't like the idea of DP because it implies the output isn't 100% accurate," says Clément. "But any data already has noise in the form of random errors and inaccuracies. At least DP noise is controlled, so can be accounted for by statisticians." A more credible concern is that while DP makes the output future-proof, the input data – and the algorithm that processed it – are not covered. "DP does not protect you if the server used to store the dataset is compromised, for instance," says Clément. This shows the necessity of combining DP with other technologies and concepts to fully protect the privacy of the individuals involved.

Clément is also reckoning with another tricky aspect of DP: convincing people that it is useful and trustworthy. "For many people, DP seems unnecessary or overly opaque," he says. "Yet, data breach after data breach has shown its worth." Clément is interested in further researching trust in computing and how to build it. "There's a responsibility that comes with the things we design," he says. "We need to be able to guarantee that people can trust what we do with their data, and that they maintain agency over how we use it."

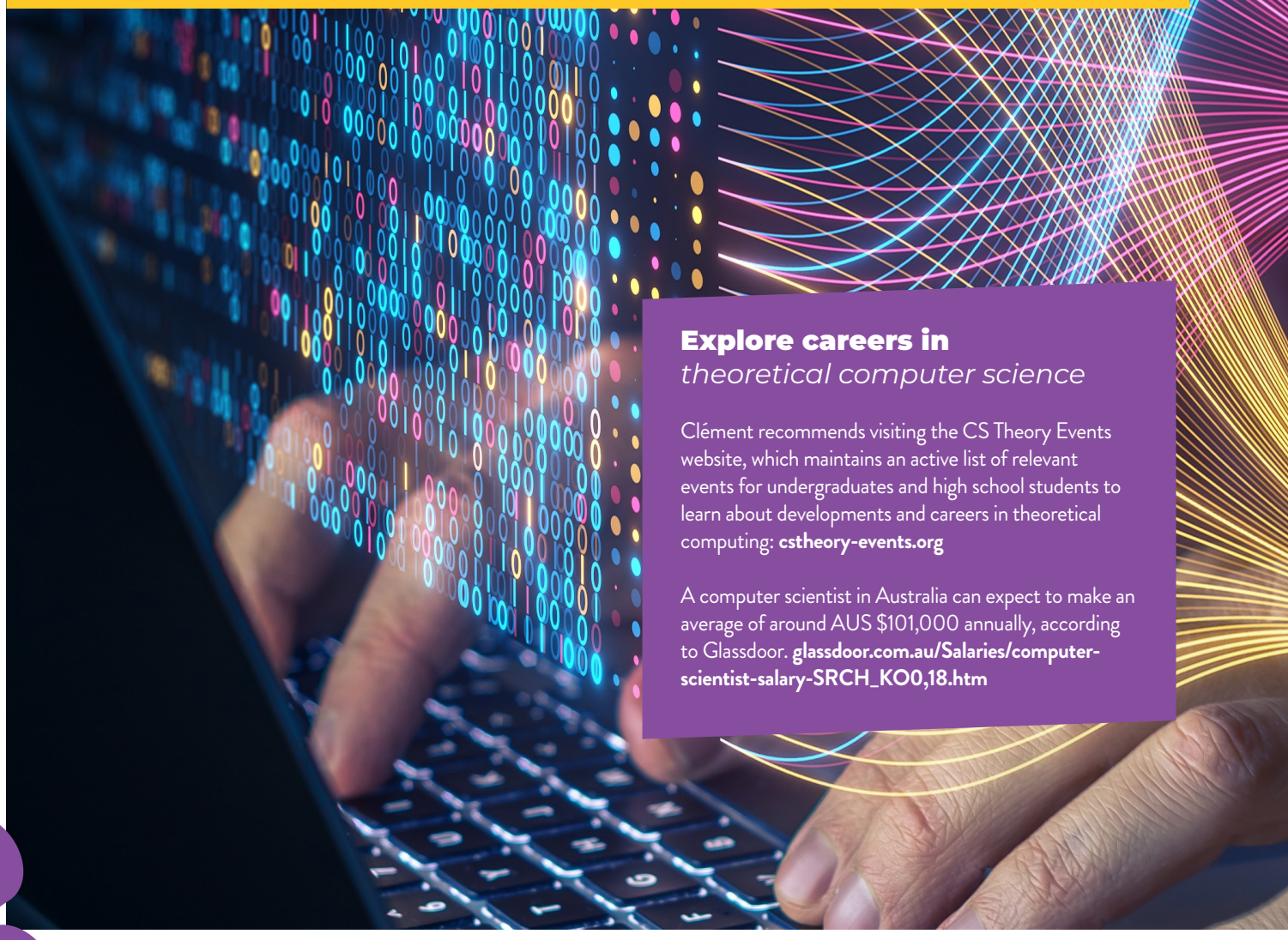
About *theoretical computer science*

Unlike applied computer science, theoretical computer science is less about coding and more about exploring the fundamental laws and limits of computation and information processing. These fundamentals are found within mathematics, which can be explored using abstract models of computation.

Much of Clément's work focuses on examining the 'cost' of meaningful privacy guarantees – for example, whether better privacy entails slower algorithms, larger datasets or less accurate models. "A typical day of research involves a mix of meetings with collaborators and PhD students to discuss ideas and suggest possible approaches, along with hours in front of a piece of paper trying to prove a concept," he says. "Staring into the void does yield some ideas, but more often, the ideas arrive while discussing things with other people. Research is truly a collaborative endeavour."

Another aspect that Clément enjoys is the freedom to choose where to focus his attention. "There is an endless supply of interesting questions, which means there is the luxury of choice of which to tackle," he explains. "We also have a lot more flexibility than, say, theoretical physicists, as we get to design the rules of the systems we study." Having a strong understanding of these rules is essential to be able to theorise about their applications. "It's important to understand how computers work and how programming languages work," says Clément. "But the crucial set of core skills is a strong scientific and mathematical background, alongside heaps of curiosity."

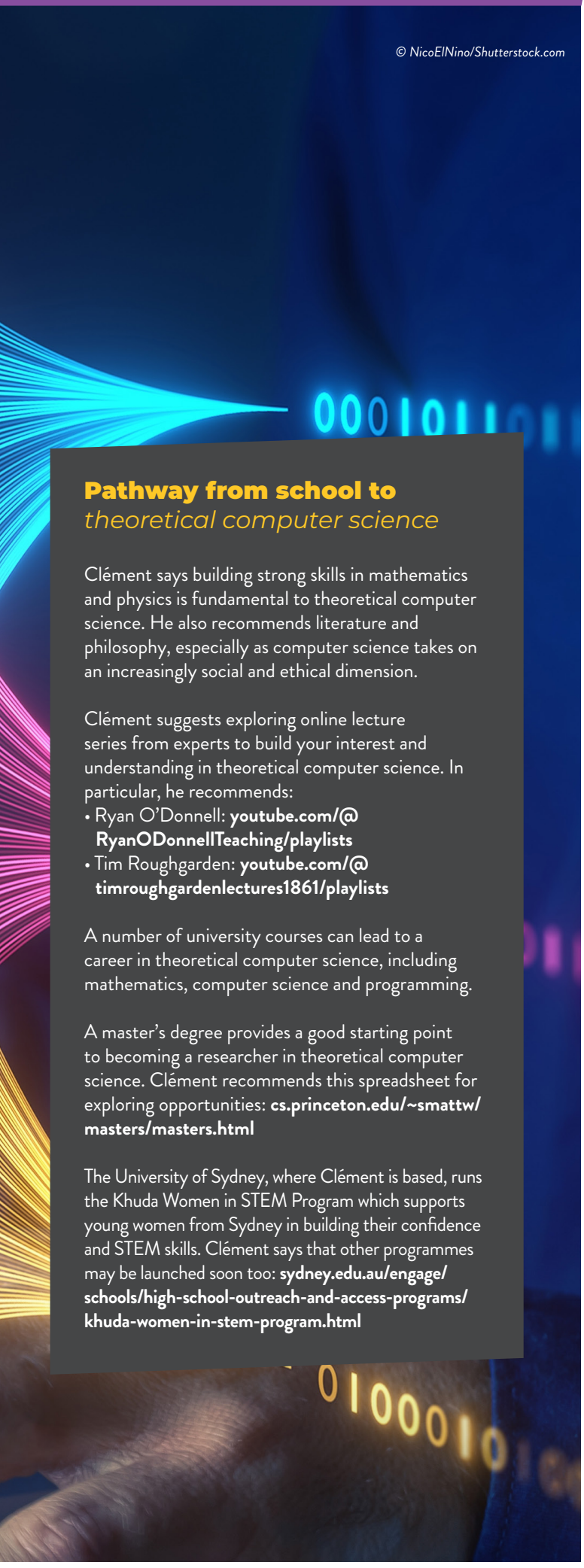
With regards to the future direction of theoretical computer science, Clément says that it is anyone's guess. "It's really hard to predict what will come next," he says. "All I am sure of is that there will be plenty of opportunities!"



Explore careers in *theoretical computer science*

Clément recommends visiting the CS Theory Events website, which maintains an active list of relevant events for undergraduates and high school students to learn about developments and careers in theoretical computing: **cstheory-events.org**

A computer scientist in Australia can expect to make an average of around AUS $101,000 annually, according to Glassdoor. **glassdoor.com.au/Salaries/computer-scientist-salary-SRCH_KO0,18.htm**



Pathway from school to *theoretical computer science*

Clément says building strong skills in mathematics and physics is fundamental to theoretical computer science. He also recommends literature and philosophy, especially as computer science takes on an increasingly social and ethical dimension.

Clément suggests exploring online lecture series from experts to build your interest and understanding in theoretical computer science. In particular, he recommends:

- Ryan O'Donnell: **youtube.com/@RyanODonnellTeaching/playlists**
- Tim Roughgarden: **youtube.com/@timroughgardenlectures1861/playlists**

A number of university courses can lead to a career in theoretical computer science, including mathematics, computer science and programming.

A master's degree provides a good starting point to becoming a researcher in theoretical computer science. Clément recommends this spreadsheet for exploring opportunities: **cs.princeton.edu/~smattw/masters/masters.html**

The University of Sydney, where Clément is based, runs the Khuda Women in STEM Program which supports young women from Sydney in building their confidence and STEM skills. Clément says that other programmes may be launched soon too: **sydney.edu.au/engage/schools/high-school-outreach-and-access-programs/khuda-women-in-stem-program.html**

Meet *Clément*

When I was a kid, I wanted to become an archaeologist. I guess that didn't pan out! But other interests also called me. I remember reading a fascinating book called *The Number Devil*, by Hans Magnus Enzensberger, which really captured my interest in mathematics.

My career has been shaped by two very lucky breaks. The first was during my undergraduate studies, when I got the chance to spend an exchange semester at Princeton University in the US. I decided to take a class in algorithms, taught by an amazing researcher, Professor Moses Charikar. This defined what I chose for my PhD.

My second stroke of luck was during my PhD. While trying to find a good research question to focus on, Professor Dana Ron, from Tel Aviv University, was visiting for a sabbatical. I joined her and my PhD supervisor in meetings and that's what led to my first paper – and my field of study for the rest of my career!

I'm really proud of my undergraduate students. Some of their theses have even led to peer-reviewed publications. My first PhD students will soon be graduating, and I'm already looking forward to seeing where their careers take them.

I aim to keep focusing on research that I find interesting and meaningful. More practically, I am looking forward to seeing Sydney's theoretical computer science and privacy research communities expand. This is a great place to be, with excellent people, so I see a lot of potential for growth.

Clément's *top tips*

Be curious, ask questions, and don't be ashamed to contact people. While they are always busy, most researchers also love what they do and love discussing it – so reach out to them!