THE THEORY BLOGS COLUMN

BY

LUCA TREVISAN

Bocconi University
Via Sarfatti 25, 20136 Milano, Italy
L.Trevisan@UniBocconi.it
https://lucatrevisan.github.io

Boaz Barak is a professor of computer science at Harvard, known for a wide range of research interests, from the foundations of cryptography to computational complexity and combinatorial optimization, and for groundbreaking contributions to our field. Boaz has also thought a lot about computer science education: he is a co-author, with Sanjeev Arora, of a textbook on computational complexity and he has been developing a new introductory course in TCS.

In his guest column, Boaz tells us about his experience writing on the "Windows in Theory" blog, about his excellent sources of inspiration, about his thoughts on mathematics and computer science education, and much more.

WINDOWS ON THEORY

A Conversation with Boaz Barak

Q. Boaz, thanks for taking the time to talk about your blog to our readers. When did you start to blog, and what motivated you to start?

In 2012, Omer Reingold started a group blog for the amazing theoretical computer scientists of the Microsoft Research Silicon Valley lab, and called it "Windows on Theory". As a fellow MSR researcher, Omer invited me to join the blog as few months later. Joining a group blog seemed to me like an attractive proposition, since I didn't think I will have something interesting to say on a very regular basis. I liked the idea of explaining technical topics on a blog post, the way you might sketch them on a whiteboard to a colleague. Compared to a survey, where you have to cross all your t's and dot all your i's, and get all references straight, a blog post can be a good way to convey the heart of the matter without doing as much work. Indeed throughout the years, I've been inspired by several blog posts by you, Luca. Your blog is a great example of how to explain technical topics in an informal manner.

Q. Thank you so much for that! You have very broad interests in theoretical computer science, and you blog about a great variety of topics. Have there been instances where writing posts or discussing in the comment section has clarified ideas or lead to a conjecture or otherwise helped with your research?

I do think that my thinking on several questions, including structure vs. combinatorics, quantum skepticism, theory of deep learning, and more, have been shaped by both the process of writing essays and the discussion in comments or outside the blog that ensues. It is a different form of thinking than the typical scientific paper, and often when you sit down to write, it forces you to clarify your thoughts. This is similar to how often the best way to learn a topic is to teach it.

Q. I have followed on your blog your course on methods from theoretical physics and your posts on the foundations of machine learning and AI, and I know you have worked on a new approach to teach computability and complexity. What kind of TCS do you think we should teach to CS undergraduates who are interested in AI?

It's interesting because I think traditionally the critique of courses in theoretical CS was that we are teaching all this math, while students are going to be software developers and they just need to know how to write a website. Now it

turns out that we didn't teach enough math, and to participate in the AI revolution students need to know their gradients and Hessians. It's also the case that Neural networks are really just arithmetic circuits (and backpropagation has been rediscovered several times, including by Baur and Strassen in 1982, where they used it for circuit lower bounds). So I think the tools we teach as theoretical computer scientists are as relevant as ever. I did try to modernize my course, focusing on circuits, which are relevant not just for AI but also for the foundations of both cryptography and quantum computing. I also talk much more about randomness in computation. This means that some other materials, such as automata, need to be reduced or cut, but I think it's a good tradeoff.

Q. On a related note, what do you think that a future satisfactory theory of AI might look like?

As theoretical computer scientists, we are used to being way ahead of practice. For example, people are only starting now to implement the ideas of zero-knowledge and probabilistically-checkable proofs that were put forward by theorists in the 80s and 90s. Dwork and Naor suggested the "proof of work" protocol used by Bitcoin in 1992. (They were also ahead of the curve in another way: proposing to combat "junk email" before most people had access to email and the term "spam email" was even coined.)

In deep learning we are in a different setting: practice is ahead of theory, and people are implementing systems that they themselves don't understand. In that sense, these systems behave more like artifacts that are discovered (or evolved) than like ones that are designed. This forces us to use a different form of theory, and one that relies more heavily on experiments to figure out what are even the right questions to ask. So, we are not in our usual mode where there are easy-to-state but hard-to-prove conjectures, and our goal is to sit down with pen and paper and to prove them. But for me, theoretical computer science was never about the mode of operation but about the mission of understanding computation. So if understanding deep learning means that I needed to re-learn how to code, and rack up large bills for GPU computation, then so be it.

Q. Can you tell us a bit about the plans for changes in California math education and about your involvement in that debate?

Some colleagues in California have alerted me to a proposed change to the way K-12 math is taught there and that this change is part of a national movement. Part of this is the typical tension that always exists between teaching mathematical topics that are foundational (and often a bit more challenging) vs. "practical math". This is something that I mentioned also in the discussion regarding university teaching. In the context of high school, the new version of "practical math" is no longer accounting but "data science". There is also a twist in which it is

claimed that somehow data science is more "equitable", which is something I find offensive, as it tacitly assumes that people from certain groups are inherently incapable of accessing mathematical topics such as algebra and calculus. From my experience in teaching, both at university settings and in Ethiopia and Jamaica, nothing could be further from the truth

Now I am all for teaching students a course in some data literacy, including facility with spreadsheets and understanding the various ways that people can "lie with statistics". It's just not a replacement for math courses.

The truth is that, like at the university level, students need more math these days than ever before. By far the largest growth in job opportunities has been in quantitative fields. When data science is offered as an *alternative* to math, as opposed to complementing them, it basically serves as an "off ramp" that shuts students out of these fields, including, ironically, from careers in data science itself.

Q. In general, what are you thoughts about the role of public intellectuals that theoretical computer scientists could fill, and what are public debates where you would like to see more voices coming from our community?

In our field, we often have the experience of being humiliated by either discovering that our conjecture was wrong or being unable to prove it. I think this is not a bad experience to have had for public intellectuals, and so I would hope that theoretical computer scientists speak up more in the public sphere. Areas including immigration, science funding, open access to publications, and mathematical education are clearly central to our mission to advance science, but I think we can talk about more topics as well. For example, I recently signed an open letter protesting the Israeli government's efforts to weaken the judicial branch and the basic laws on human rights. Scientific progress relies on the ability to collaborate, so free speech and human rights are topics that we should talk about as well.

Q. I would like to ask you to pick one or a couple of your favorite posts, and tell us about it/them/

My first blog post[1] was an exposition of Fully Homomorphic Encryption with Zvika Brakerski. I like that post because we didn't just repeat what's in the papers but used the flexibility of the blog format to focus on optimizing simplicity and intuition as opposed to precision and computational efficiency. I think people have found it useful over the years. Another blog post[2] I am proud of is my post on "Men in Computer Science". I mostly made obvious points in that post, but heard from several women that they appreciated it.

---

[1]https://windowsontheory.org/2012/05/01/the-swiss-army-knife-of-cryptography/
[2]https://windowsontheory.org/2017/08/16/men-in-computer-science/