Know the Person behind the Papers

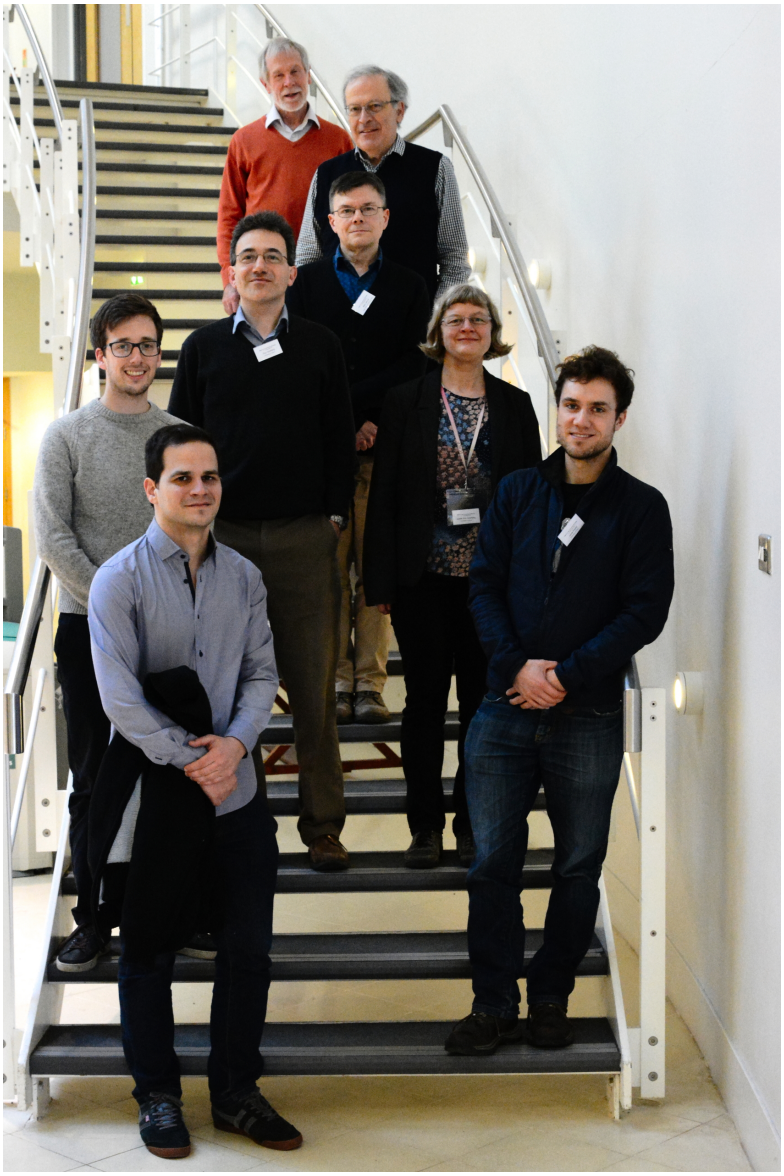
Leslie Ann Goldberg

Bio: *Leslie Ann Goldberg is the Head of the Department of Computer Science at the University of Oxford. Her work is in the area of Randomised Algorithms and in the related area of Approximate Counting. Leslie received her BA in 1987 from Rice University and her PhD in 1992 from the University of Edinburgh. She was awarded an ERC Advanced Grant in 2014, and was elected to Academia Europaea in the same year. In 2016 she received a Suffrage Science Award. She and her co-authors have won four best-paper prizes at ICALP.*

We ask all interviewees to share a photo with us. Can you please tell us a little bit more about the photo you shared?

Leslie: Like some others, I've included two photos here. The first one was taken in 2016 in the "green room" at the Hay Festival of Literature and Arts. I was there because I was giving a public lecture entitled "Algorithms and their Limitations" about P vs NP. I really like doing this kind of outreach, and I chose

this photo partly for that reason, and partly because my main non-academic hobby is reading fiction. The annual festival has grown from its original literary purpose to now include science, politics, and music. I'm pretty sure that I look both nervous and excited in that photo. Partly nervous about my talk, and partly about being in the presence of literary giants. Salman Rushdie had just walked by!



The second photo is a "context" photo. This is a tiny piece of Mike Paterson's computer science "family tree". Standing below Mike is his once-student Les Valiant, and below Les is his once-student, Mark Jerrum. Below Mark are my husband, Paul Goldberg, and myself, both once-students of Mark. On the next

row are Paul's then-student Edwin Lock and my then-student Jacob Focke. On the bottom row is Andreas Galanis, a really important collaborator for me. Andreas fits into the picture because he is a former student of Eric Vigoda, who is a former student of Alistair Sinclair, who is also a former student of Mark. I chose the photo because of the great TCS context — so many great colleagues to whom I owe so much!

Can you please tell us something about you that probably most of the readers of your papers don't know?

Leslie: I was a latecomer to Maths and Computer Science. My dream, as a young person, was to be a civil rights lawyer. At Rice University, I took a double major in Political Science and Computer Science. Political Science, to prepare for postgraduate study in law, and Computer Science because I mistook it for vocational training which would give me a way of paying for my study in law! Theoretical Computer Science was my first introduction to open problems in mathematics. I was overwhelmed by how fascinating it was.

Is there a paper which influenced you particularly, and which you recommend other community members to read?

Leslie: I think that papers in Theoretical Computer Science do age a bit with time, given the speed of advances in the field. Two papers that were hugely influential on me when I was a PhD student were Valiant's "The Complexity of Computing the Permanent" and "The Complexity of Enumeration and Reliability Problems", which essentially introduced the field of Computational Counting (one of the many fields that Les has initiated!). I was definitely also influenced by Jerrum and Sinclair's "Approximating the Permanent" which introduced some great techniques and ideas. They also write very well.

Is there a paper of your own you like to recommend the readers to study? What is the story behind this paper?

Leslie: I can't really imagine recommending that anybody should "study" one of my own papers! That feels very arrogant and strange. Let me say instead that I am usually most enthusiastic about some of my recent work. At the moment I am very excited about contention resolution, which is something that I really liked working on a long time ago with Mike Paterson and others, and which I've very recently come back to with John Lapinskas. I'm very excited about our new paper "Instability of backoff protocols with arbitrary arrival rates".

When (or where) is your most productive working time (or place)?

Leslie: I'd love to be one of those people who can work perfectly well in noisy places in odd snatches of 10 minutes, but in fact I work best when I have

long quiet periods alone, especially in the morning. When I was a PhD student I once solved a problem that I'd been stuck on for many months during a walk. I was with others who had much better gear for Scottish peaks (something I fixed later!) and I was too cold to continue to the top with them, so I spent the day with the sheep much lower down. It was a blow to the pride to drop out of the excursion to the top, but it was good compensaton to be rewarded with an idea for my problem!

What do you do when you get stuck with a research problem? How do you deal with failures?

Leslie: I think the right approach to getting stuck is to divide time between (a) persisting and (b) working on something else. You want to do the first because you can't solve a problem if you don't even try. You want to do the second because the original problem might not be solvable! Failures are actually kind of nice because they give you problems that you can "carry around" to come back to later in life. The main open problem from my PhD (the complexity of approximating the cycle index polynomial) is still open. The main problem that I was working on with Mike Paterson at Warwick around 20 years ago is the source for what John Lapinskas and I have picked up recently.

Is there a nice anecdote from your career you like to share with our readers?

Leslie: When I was a young researcher Martin Dyer asked me whether I'd been invited to a certain Oberwolfach meeting and I admitted that I hadn't been invited. He rushed to explain "Oh, don't worry. It isn't what you know. It is *who* you know." I hadn't actually been very upset about not being invited — after all, only a small number of people can attend — but I was very touched by Martin's kind reassurance. He was a good mentor, to me and to others.

Do you have any advice for young researchers? In what should they invest time, what should they avoid?

Leslie: The following advice is something that I learned from Robin Milner. At the time I was struggling to find a PhD topic and my supervisor, Mark Jerrum, was also pretty young, so we went to Robin for advice. He advised me to stop thinking so much about the "big picture" and instead to focus on a small problem that I enjoyed working on. His comment was that the small problem would always lead to something else, so there was no need to do so much planning. I think it is good advice.

What are the most important features you look for when searching for graduate students?

Leslie: Enthusiasm for the topic. Problem-solving ability. Being curious about problems. Being hard-working.

Do you see a main challenge or opportunity for theoretical computer scientists for the near future?

Leslie: I see many opportunities for theoretical computer science! Computing is becoming more and more important and I don't expect that to change. But as computing becomes more and more ubiquitous, foundational questions like "how long does it take" and "what is actually possible" become increasingly important. Right now "deep learning" is proving to be useful for many practical problems. But it will have limits and I expect TCS to be at the forefront of figuring out what those limits are, and what can be done instead.

What kind of opportunities should EATCS offer to researchers, and especially to young researchers?

Leslie: In addition to offering outstanding conferences, EATCS offers a sense of community. I think this is particularly important for young researchers.

What can be the role of EATCS in solving the challenges of our society?

Leslie: I'm strongly of the belief that the best way to obtain research that has a societal impact is to support "blue-skies" curiosity-driven research. There are lots of examples of curiosity-driven research that later turned out to have a big "impact" (consider, for example, radar, x-rays, or even calculus). This is true in science generally, and it is also true in the Theory of Computing. Oded Goldreich and Avi Wigderson have written a nice essay about this entitled "The Theory of Computing: A scientific Perspective". My main point is that the EATCS does and should stay focussed on "discovery of truth" and "foundational understanding". This is the best way to contribute to societal challenges.

Please complete the following sentences?

- *My favorite movie is...* perhaps "Before Sunrise" — I'm not really the sort of person that has a "favourite movie". I'm not even sure whether there are any movies that I've seen more than twice, apart from maybe "Casablanca".
- *Being a researcher...* is a very rewarding career!
- *My first research discovery...* was probably written on a piece of paper that I don't have anymore!
- *Theoretical computer science in 100 years from now...* will be making lots of fascinating discoveries that we can't predict now.
- *EATCS in 50 years from now...* will hopefully continue to be at the centre of this fascinating area.
- intellectual curiosity ... *is key to being a happy academic.*