The Interview Column

by

Chen Avin and Stefan Schmid

Ben Gurion University, Israel and TU Berlin, Germany
{chenavin,schmiste}@gmail.com

Know the Person behind the Papers

Today: Bruce Maggs

Bio: *Bruce Maggs is the Pelham Wilder Professor of Computer Science at Duke University. He received the S.B. (1985), S.M. (1986), and Ph.D. (1989) degrees in computer science from the Massachusetts Institute of Technology. His advisor was Charles Leiserson. After spending one year as a Postdoctoral Associate at MIT, he joined NEC Research Institute in Princeton. In 1994, he moved to Carnegie Mellon, where he stayed until joining Duke in 2009. While on a two-year leave-of-absence from Carnegie Mellon, Maggs helped to launch Akamai Technologies, leading all engineering efforts as its Vice President for Research and Development. In 2018 he was one of the inaugural winners of the SIGCOMM Networking Systems Award for the Akamai content delivery network, and was named a Fellow of the ACM. From 2019 to 2024 he served as Director of Engineering for Emerald Innovations.*

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

Bruce: I go windsurfing whenever I can! One weekend last winter I flew to Tampa because the wind forecast looked promising. Alas, the weather didn't cooperate. The wind was light and the air was cold (8°C). As you can see from the photo, I went out anyway.

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

Bruce: I had an unusual childhood for an American boy. My hometown is Urbana, Illinois. My father is a law professor and when I was a boy his speciality was Soviet law. When I was young we lived for months at a time in Bulgaria, Yugoslavia, Romania, and the Soviet Union. (We also lived in Hawaii for eight months.)

My father was also an early computer hobbyist. I remember watching him lying on his stomach in the living room filling out IBM coding sheets. Later he would go to the university and punch the corresponding cards and then run his programs. In 1976, when I was thirteen, he had a PLATO computer terminal installed in our home, connected by a 1200-baud dedicated line. PLATO was an innovative mainframe-based computer system developed jointly by Control Data and the University of Illinois. The creators of PLATO also invented the plasma panel. PLATO was intended to enable computer-aided instruction, but it really excelled at multi-player games. At first I was only interested in playing these games. My friend Andrew Shapira and I worshipped the game authors and eventually we decided that we wanted to be the objects of this adulation. So in junior high school we started writing a dungeons and dragons game called "Avatar." We were soon joined by a third author, C. David Sides and received some help from Mike Berger. Avatar eventually became the most popular program on the PLATO system.

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

Bruce: As a graduate student I was blown away by Bob Tarjan's analysis of the union-find algorithm [5, 6], which, I think, was the first analysis to incorporate the inverse Ackermann function in the running time of an algorithm. It was hard for me to imagine where his "multiple partition" method had come from. The paper demonstrates that seemingly intractible problems can solved with enough ingenuity. But I can't limit myself to one paper! Others that had a big impact on my work were the papers that describe the AKS sorting network [1, 3], and the papers that introduce Valiant's idea of first routing to a random destination [7, 8].

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

Bruce: I guess my most "famous" paper was co-authored with Tom Leighton and Satish Rao. Simplifying just a little, it shows that in any network, any packet routing problem can be solved in $O(c + d)$ steps, where c is the congestion of the paths of the packets in the network, and d is the dilation [2]. Congestion is the maximum number of packets that cross any edge of the network, and dilation is the length of the longest path. The key to the paper is a beautiful (if I may say so) application of the Lovász Local Lemma [4, pp. 57–58]. As a graduate student I presented this paper at FOCS. It was my first such presentation. Later I shared an elevator with Bob Tarjan (see above) who I hadn't met yet. He said, "Nice talk."

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

Bruce: I'm often so busy during the day that I can't think straight. Over and over again I've seen the solution to a problem just after lying down to sleep at night, when I can finally relax. Sometimes this makes me feel like I didn't try very hard during the day!

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

Bruce: I feel like I'm stuck 99% of the time. When I'm frustrated I remind myself that it's possible to have a highly successful research career with only one or two breakthroughs per year. Then I go back to banging my head against the wall. But, as I was taught by Tom Leighton, one of my mentors in graduate school, after a breakthrough occurs you must explore the consequences with all your energy!

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

Bruce: I transferred to MIT as a junior in college. The very first class I attended was 6.045, "Automata, Computability and Complexity Theory," taught by Charles Leiserson. I loved this subject. At the end of the semester I approached Charles and asked if he would be willing to supervise an undergraduate research project on complexity theory. But he told me that complexity theory wasn't really his research area. He said, however, that he would be happy to introduce me to some outstanding faculty members at MIT who worked in the area. I thought for just a moment, and then said no, I really liked the way he presented the material in class, and I wanted to work with him. So I asked what his research was about. Eventually he became my Ph.D. advisor. Choosing to work with Charles was the best career decision I ever made.

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

Bruce: Invest your time trying to solve problems that are important to you. Your motivation will be better, and your work will be more likely to have an impact. You might also consider taking some time off from academia to work at start-up companies. I really enjoy the teamwork aspect of building a new company. By necessity you'll learn many new things! At my first start-up, Akamai Technologies, I learned a lot about computer networks and distributed systems and decided to change my research focus from the theory of parallel algorithms and architectures to computer networking and distributed systems. My former Ph.D. student Anja Feldmann served as my mentor as I transitioned into the computer networking community. Most recently I've been working at a start-up called Emerald Innovations. Emerald is an MIT spin-off founded by Dina Katabi and three of her former Ph.D. students, Rumen Hristov, Hariharan Rahul, and Zac Kabelac. Emerald has developed a "human radar" device that can track the breathing, sleep cycles, and mobility of a person at home. The Emerald device collects data in the homes of volunteer participants in clinical drug trials to help assess the effectiveness of new medications.

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

Bruce: By far the most important trait is persistence. The most successful graduate students just don't give up. When one approach fails, they try another. Each time they come to your office they surprise you with something they found under a new rock they turned over.

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

Bruce: The biggest challenge remains translating a real-world problem into a theoretical framework in a way that's amenable to analysis, but still provides guidance on solving the original problem. I don't know that I'm particular adept at that.

Please complete the following sentences?

- *Being a researcher is a privilege! Very few people get to decide what is most important to solve and go after it.*
- *My first research discovery was a "communication efficient" algorithm for computing minimum-cost spanning trees on a fat-tree parallel computer.*
- *Enjoying research is the key to being a happy academic. It makes up for all the overhead that academic life entails.*
- *Theoretical computer scientists 100 years from now will look back and marvel that primitive humans struggled to prove $P \neq NP$.*

References

[1] M. Ajtai, J. Komlós, and E. Szemerédi. Sorting in $c \log n$ parallel steps. *Combinatorica*, 3:1–19, 1983.

[2] F. T. Leighton, B. M. Maggs, and S. B. Rao. Packet routing and job-shop scheduling in O(congestion + dilation) steps. *Combinatorica*, 14(2):167–180, 1994.

[3] M. S. Paterson. Improved sorting networks with $O(\log N)$ depth. *Algorithmica*, 5:75–92, 1990.

[4] J. Spencer. *Ten Lectures on the Probabilistic Method*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1987.

[5] R. E. Tarjan. Efficiency of a good but not linear set union algorithm. *Journal of the ACM*, 22(2):215–225, Apr. 1975.

[6] R. E. Tarjan. *Data Structures and Network Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1983.

[7] L. G. Valiant. A scheme for fast parallel communication. *SIAM Journal on Computing*, 11(2):350–361, May 1982.

[8] L. G. Valiant and G. J. Brebner. Universal schemes for parallel communication. In *Conference Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computing*, pages 263–277, May 1981.