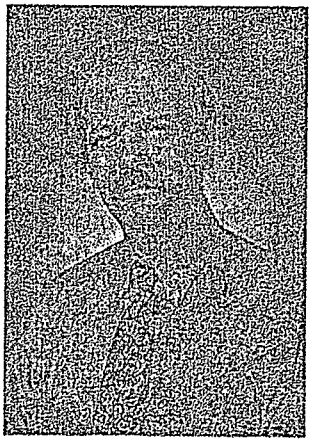
ickly

same
s for
effort
s will
help

uality
uring
mil-
look-

Chapter 25

## Moshe Y. Vardi: From Theory and Practice in Computing to Research Ethics and the Surveillance State



*Professor Moshe Y. Vardi, www. cs. rice. edu/~vardi/ is the George Distinguished Service Professor in Computational Engineering and Director of the Ken Kennedy Institute for Information Technology at Rice University. He is the author and co-author of over 450 papers, as well as two books: Reasoning about Knowledge and Finite Model Theory and Its Applications. He is the co-recipient of three IBM Outstanding Innovation Awards, the ACM SIGACT Gödel Prize, the ACM Kanellakis Award, the ACM SIGMOD Codd Award, the Blaise Pascal Medal, the IEEE Computer Society Goode Award, the EATCS Distinguished Achievements Award and the Southeastern Universities Research Association's Distinguished Scientist Award. He is a Fellow of the Association for Computing Machinery, the*

*American Association for Artificial Intelligence, the American Association for the Advancement of Science and the Institute for Electrical and Electronic Engineers. He is a member of the US National Academy of Engineering, the American Academy of Arts and Science, the European Academy of Science and the Academia Europaea. He holds honorary doctorates from the Saarland University in Germany and Orleans University in France. He is currently the Editor-in-Chief of the* Communications of the ACM.

**CC**: You have referred to logic as "the calculus of computer science". Are mathematical proofs relevant for computer science? Can you cite a practical application of your results which can be understood by a non-mathematician?

**MV**: I should point out first that the most fundamental ideas in computer science, such as computability, universal machines that can execute arbitrary programs, the distinction between hardware (machines) and software (programs), and programming languages (formalisms to describe computations), all came out from an investigation in the 1920s and 1930s into the nature of mathematical proofs. Once people started programming in the 1950s and 1960s, they realised that reasoning about correctness of computer programs highly resemble reasoning about the correctness of mathematical theorems. This led to intensive research in "program verification". This research borrowed heavily from various branches of mathematical logic, such as lattice theory, model theory and automata theory. My own contribution has been in developing the connection with automata theory. Today there are industrial verification tools that are used daily, which are based on an "esoteric" concept such as "automata on infinite words".

**CC**: You have worked in many different subjects, including database theory, finite-model theory, knowledge in multi-agent systems, computer-aided verification and reasoning. Which of your results do you like most?

**MV**: I happen to like the most results which have been the most influential. As a PhD student I discovered that when analysing the computational complexity of database query evaluation one has to distinguish between the contribution of the data to that complexity and the contribution of the query. These are known as "data complexity" and "query complexity", and have become the standard way of looking at the complexity of query evaluation. I also discovered that query complexity is usually exponentially higher than data complexity. Fortunately for us, queries are typically short while databases are large. As a postdoc I discovered the automata-theoretic perspective to program verification. At first this did not lead to new algorithms as much as offered a very simple way to understand existing algorithms. (In fact, it was so simple that the paper was at first rejected when we submitted it to a conference). Over time, the automata-theoretic perspective has been highly useful, leading to many developments in automata-theory-based algorithms. This is is still a very active research area.

**CC**: You are a co-winner of the 2000 Gödel Prize. Do you see any relevance of incompleteness for mathematics or computer science?

**MV**: Gödel's Incompleteness theorems established limits on the power of the mathematical approach. It showed that mathematics is a human activity (or, as sociologists would say "a social construct"), with all the limitations that are implied by that. In a similar way, Church–Turing's Undecidability theorem established the limits of the computational approach. It showed that computing, which we tend to think of as a mechanical activity, is at its heart a human activity, since computers and programs are designed by humans.

**CC**: Is computer-aided verification "practical"? What about automated proof-checking?

**MV**: For many years there was a fierce debate whether computer-aided verification could ever be made practical. See, for example, the article by DeMillo, Lipton and Perlis on "Social Processes and Proofs of Theorems and Programs", *Commun. ACM* 22(5): 271–280

(1979). Today, model checkers such as FormalCheck, Rulebase, SPIN, and others are in standard industrial usage, so the philosophical debate has become somewhat moot. Proof checking is by nature more labour-intensive than algorithmic methods such as model checking, so its industrial applicability is more limited. Nevertheless, it is used to verify certain critical pieces of hardware designs, such as floating-point arithmetics. The development of highly scalable decision procedures for fairly expressive logics further reduces the distinction between model checking and proof checking.

**CC**: Please tell us more about the Ken Kennedy Institute for Information Technology (K2I) at Rice.

**MV**: The basic observation is that the information revolution is transforming society-creating new careers, new industries, new academic disciplines, and the need for new educational and research programmes. We wish to see Rice as a leading institution in the information age. For Rice to be an academic leader in the information revolution, it is imperative to make information technology a *key institutional priority* and take bold, coordinated, pervasive steps to enhance Rice's position in this area. The information technology initiative at Rice needs to encompass information technology research, the digital library effort, information technology education and information technology *in* education.

The role of the Ken Kennedy Institute (K2I) (`www.k2i.rice.edu`) is to complement the traditional academic structures of the university in order to provide more flexibility to respond to new educational and research needs in the fast-paced information age. K2I counteracts the compartmentalisation of the university by becoming a focal point for academic activities in information technology. It focuses resources to seed and nurture the development of both existing areas and emerging activities. It serves the broad and strong student interest in information technology, and incubates new concentrations, programmes, majors and departments as needs emerge. K2I faculty members tie their home departments to K2I and to one another, providing new channels for cross-disciplinary activity related to information technology. K2I is a home that encourages

and facilitates exchanges between information technology and the broadest possible range of disciplines at Rice University.

**CC**: It seems that hardware evolves faster than software. Do you see in this trend any chance for theoretical computer science?

**MV**: Theoretical research can play different roles in computer science. On the one hand, its role is to clarify the fundamental principles of our discipline. These principles are quite independent from technical development. Much of complexity theory, for example, falls in this category. On the other hand, theoretical research can follow technical developments, trying to explain experimental observations, as well as solving technical challenges that practitioners face. For example, much of the current research on computer-aided verification is driven by the growing complexity of hardware designs. Thus, Boolean decision diagrams (BDDs), which are used to represent Boolean functions compactly, came out of theoretical research on automata theory and branching programs, motivated by the need to effectively manipulate very large Boolean functions.

**CC**: Improper scientific conduct undermines public trust in the results and methods of science, and threatens public funding and support for scientific research. You have organised a "Research Ethics Seminar" at Rice. Please tell us more about it.

**MV**: Researchers face ethical issues on a daily basis, managing research funds obtained from public sources, supervising graduate students whose educational interests may diverge from those of their supervisors, dealing with intellectual property issues and more. Even a decision on who should be an author on a paper has ethical dimensions. At the same time, graduate students, who are essentially apprentice researchers, typically receive little training and education in thinking about ethical aspects of research. The Research Ethics Seminar was an attempt to address that. It was a success and a failure at the same time. On the one hand, the students loved it. On the other hand, we did not find enough faculty members with an interest in the topic to make this seminar a regular course offering.

**CC**: Some public opinion leaders think that theoretical computer science has little relevance for core informatics. Many talented young people don't regard this subject as exciting. Do you agree? Please tell us about the challenges of teaching theoretical computer science.

**MV**: Twenty and thirty years ago, most computer science students arrived from mathematics. These students enjoyed learning theoretical computer science as a mathematical theory, so we taught theoretical computer science as a mathematical theory. Today's computer science students are interested in computing, yet we still often teach theoretical computer science as a mathematical theory. The challenge is on us to make the theory relevant to the practice of computing. For example, I teach "Logic in Computer Science" at Rice. Part of the final project of this course is for the students to implement a Boolean satisfiability solver and use it to solve "Einstein's Puzzle". This demonstrates to the students that Boolean satisfiability is not some abstract mathematical concept, but rather a very powerful generic problem-solving framework.

**CC**: Non-funded research seems to be declining. Do you agree? Is this a good trend?

**MV**: I am not sure I agree with that. If you go back to pre-WWII time, you see that the research enterprise was rather small. It grew enormously after the war, supported by research funding, when it became clear that research can contribute to national security as well as to economic prosperity. There is no question that today's funded research dwarves non-funded research, but it is not clear to me that non-funded research has actually declined. What clearly has declined is scientifically driven, industrially funded research. No industrial lab today is anything like Bell Labs in its heyday. That is a clear loss to science. Microsoft Research today is the only industrial research lab that supports curiosity driven research on a large scale. It will be interesting to see how it evolves in the coming years, as Microsoft loses its dominant position in the IT industry.

**CC**: Which topics cultivated today in theoretical computer science will survive the end of the century?

**MV**: If we just knew, we'd all be working on these topics! The most fundamental work will clearly survive. It is hard to imagine that the basic concepts and results of the complexity theory will not be used in 100 years (though if it turns out that P=NP, then much of current complexity theory evaporates!), including such fundamental algorithmic techniques such as breadth-first search and depth-first search. Some current theoretical topics, such as quantum computation, will either be extremely fundamental or completely irrelevant, depending on whether quantum computing will turn out to be a reality.

**CC**: In a recent editorial in *CACM* you pointed out that one role of theoretical computer science is to provide guidance to engineering. Referring to the Boolean satisfiability problem you noted that the current theories of complexity seem to offer little guidance for problems that are theoretically — worst case or average case — difficult to solve but tractable in practice. Could you give a more detailed picture of this phenomenon? Where a solution may be sought?

**MV**: I believe that complexity theory is facing a major challenge. This theory is based typically on worst-case complexity analysis, which focuses on instances that are the most difficult to solve. Worst-case complexity analysis has proven to be quite tractable mathematically, much more, than say average-case complexity analysis. It also seems intuitive from a practical point of view; for example, a worst-case upper bound for an algorithm offers an absolute upper bound on its running time in practice. Thus, worst-case analysis is the standard approach in complexity theory. What has become clear, however, is that worst-case analysis actually sheds very little light on the behaviour of algorithms in real-life instances. For example, theorists have demonstrated that current SAT-solving algorithms must take exponential time to solve certain families of SAT instances. Practitioners simply shrug at such bounds, while they continue to apply their solvers to very large but practically solvable SAT instances. Indeed, one role of theory is to provide guidance to engineering, but worst-case (and average-case) complexity seems to offer little

guidance for problems that are difficult in theory but feasible in practice. What is needed is a new computational complexity model, which will better capture the concept of "complexity in practice". We need a new kind of complexity theory!

**CC**: In March 2013 you tweeted: "The Internet is a surveillance state". This was followed by an editorial in November 2013 in *CACM* titled "The End of The American Network" which ends with the statement:

> The real question, I believe, is whether we can have an Internet that is free, or at least freer, from government meddling than today's Internet. In view of the Internet's centrality in our information-saturated lives, this is a question of the utmost importance.

Meantime the US Government has signalled its intention to end its historical agreement with the Internet Corporation for Assigned Names and Numbers in late 2015, and ICANN will develop a new global governance model.

Do you see any significant improvement?

**MV**: This may be a positive development, but the devil is in the detail. While we are justified in complaining about the Orwellian activities of the US Government, other governments may not only eavesdrop by also censor. As we have also learned, it is not clear that we can trust corporations with stewardship of the Internet. Developing a new Internet governance model that is transparent and free of meddling by governments and corporations is not an easy task.

**CC**: How do you manage to do so many different things well?

**MV**: I do not. I have a huge pile of unfinished projects.