Abstracts of invited tutorials

▶ JEREMY AVIGAD, *Proof mining*.
Department of Philosophy, Carnegie Mellon University, Pittsburgh, PA 15213, USA.
E-mail: avigad@cmu.edu, http://www.andrew.cmu.edu/~avigad.

Hilbert's program can be viewed as an attempt to study infinitary, nonconstructive mathematics in more "concrete" terms. Gödel's incompleteness theorems show the impracticability of Hilbert's original goal of proving the consistency of full-blown set-theoretic reasoning using only finitary methods. But proof theorists have since been pursuing modified programs that aim to justify, interpret, or understand restricted portions of mathematical practice, with respect to constructive, computational, or otherwise explicit forms of reasoning.

The general strategy has been to show that workable developments of, say, infinitary algebra and analysis can be obtained in formal theories representing restricted fragments of set-theoretic reasoning; and that, for interesting classes of formulas, such theories are conservative over finitary or constructive counterparts. This tells us that, at least in *principal*, proofs in the general frameworks have more explicit translations.

"Proof mining" involves putting this strategy into *practice*, that is, using metamathematical techniques to obtain useful information from proofs where it is not readily apparent. Although this general idea can be found in suggestions by Georg Kreisel as early as the 1950's, it has received remarkably little attention since then. In recent years, however, there have been some interesting advances. Most notably, Ulrich Kohlenbach has had some striking successes with applications to functional analysis.

Such a program requires a general understanding of the appropriate formal theories and their metamathematical properties, as well as issues that arise when one tries to formalize mathematics in restricted frameworks. It also requires specialized techniques that are adapted to the specific domains of application. In this series of lectures, I will try to convey a sense of some of the tools and methods available, and the kinds of information that can be obtained.

▶ TONIANN PITASSI, *Recent advances in proof complexity*.
Department of Computer Science, University of Toronto, 10 Kings College Rd., Toronto ON M5S 3G4, Canada, and Institute for Advanced Study, Princeton, New Jersey, USA.
E-mail: toni@cs.toronto.edu.

One of the most basic questions of logic is the following: Given a universally true statement (tautology) what is the length of the shortest proof of the statement in some standard axiomatic proof system? The propositional logic version of this question is particularly important in computer science for both theorem proving and complexity theory. Important related algorithmic questions are: Is there an efficient algorithm that will produce a proof of any tautology? Is there an efficient algorithm to produce the shortest proof of any tautology? Such questions of theorem proving and complexity inspired Cook's seminal paper on NP-completeness and were contemplated even earlier by Gödel in his now well-known letter to von Neumann.

The above questions have fundamental implications for complexity theory. As formalized by Cook and Reckhow, there exists a propositional proof system giving rise to short (polynomial-size) proofs of all tautologies if and only if NP equals coNP. Cook and Reckhow were the first to propose a program of research aimed at attacking the NP versus coNP problem by systematically studying and proving strong lower bounds for standard proof systems of increasing complexity. This program has led to many beautiful results as well as to new connections with circuit complexity within the last twenty years.

In this tutorial, we will try to highlight some of the main discoveries, with emphasis on the interplay and connections with bounded arithmetic, complexity theory, and machine learning.