Abstracts of invited talks in the Special Session on Logical Foundations of Programming Semantics

► ANDREJ BAUER AND CHRISTOPHER STONE, *From theories to signatures*.
Department of Mathematics and Physics, University of Ljubljana, Jadranska 19 Slovenia.
E-mail: `Andrej.Bauer@andrej.com`.
Computer Science Department, Harvey Mudd College, 1250 North Dartmouth Avenue, Claremont, CA 91711, USA.
E-mail: `stone@cs.hmc.edu`.

Much work has been done on extraction of programs from proofs. A less ambitious, but possibly equally rewarding, endeavor is extraction of abstract data types from axiomatizations of first-order theories. We use a realizability interpretation of an (intuitionistic) first-order theory to translate an axiomatization of a mathematical structure to an ML signature for a structure which implements it. In addition, the interpretation gives us a set of negative formulas which express criteria that a concrete implementation of the signature must satisfy, if it is to be a correct implementation of the original first-order theory. The fact that the correctness criteria are expressed as negative formulas is desirable, because their classical reading is the same as their intuitionistic reading. One practical benefit of this is that also programmers who are not familiar with intuitionistic logic will correctly understand correctness.

► MARTÍN ESCARDÓ, *Topology via higher-order logic*.
School of Computer Science, The University of Birmingham, Birmingham B15 2TT, England.
E-mail: `m.escardo@cs.bham.ac.uk`.

For suitable topologies, computable functions are continuous, and semidecidable properties of their inputs/outputs are open, but the converses of these two statements fail. Some authors attempted to resolve this mismatch by considering effective or constructive versions of topology.

In recent work [1], we instead propose a synthetic approach, in which both classical topology and various computational flavours arise as special cases. The idea is to (i) take continuity to mean definability in a given base language, and (ii) reduce other topological notions (such as open set, closed set, compact set, discrete space, Hausdorff space) to that of continuity with the aid of the Sierpinski space, (iii) use the lambda calculus to prove theorems about them. We shall give an overview, with some interesting computational applications.

However, the main aim of this talk is to reformulate and redevelop synthetic topology in the internal language of a topos. Exploiting the fact that excluded middle fails, one can define a non-trivial topology on the one-point set, where not all unions of open sets are necessarily postulated to be open. Technically, one obtains a subset of the subobject classifier, known as a dominance, which is to be thought of as a "Sierpinski set". This induces topologies on all sets, making all functions continuous.

However, because virtually all theorems of classical topology require excluded middle (and even choice), it would be useless to reduce all other topological notions to the notion of open set in the usual way. So, for example, our synthetic definition of compactness for a set X says that, for any Sierpinski-valued predicate p on X, the truth value of the statement "for all x in X, $p(x)$" lives in the Sierpinski set. The proposed synthetic notions interact in the expected way. Moreover, we show that there are embeddings of full subcategories of topological spaces into toposes that translate the classical notions into the synthetic ones, and, similarly, we look at computational manifestations of the synthetic notions by embedding programming languages into toposes.

[1] M. H. ESCARDO, ***Topology of data types and computability concepts***, 93pp. School of

Computer Science, University of Birmingham, UK.
http://www.cs.bham.ac.uk/~mhe/papers/barbados.pdf

▶ NICOLA GAMBINO, *Wellfounded trees, fixpoints, and free monads*.
DPMMS, University of Cambridge, Centre for Mathematical Sciences, Wilberforce Road, Cambridge CB3 0WB, UK.
E-mail: N.Gambino@dpmms.cam.ac.uk.

We will study of the consequences of the assumption of types of wellfounded trees in dependent type theories. In order to do so, we investigate the categorical notion of wellfounded tree recently introduced by Moerdijk and Palmgren. Our main results shows that wellfounded trees allow us to define initial algebras for a wide class of endofunctors on locally cartesian closed categories. We give some applications of this result to the theory of fixpoints and free monads.

▶ ALEX SIMPSON, *Axioms for synthetic domain theory*.
LFCS, School of Informatics, University of Edinburgh, JCMB, King's Buildings, Edinburgh EH9 3JZ, UK.
E-mail: Alex.Simpson@ed.ac.uk.

Synthetic domain theory aims to provide a simple theory of "domains", for use in programming language semantics, in which domains are merely (special) sets, and morphisms of domains are arbitrary set-theoretic functions. For several reasons, no such theory is possible within ordinary classical set theory. However, Dana Scott, long ago indicated that such a theory should be compatible with intuitionistic set theory. In this talk I will outline the synthetic domain theory program, and summarize recent progress towards finding simple and usable set-theoretic axioms which finally make synthetic domain theory a viable tool for applications to the semantics of programming languages.