Nathanael Arkor*

Department of Computer Science and Technology
University of Cambridge

Algebraic simple type theory: a polynomial approach

Algebraic type theory is the study of type theory qua the extension of universal algebra to richer settings (*e.g.* sorting, binding, polymorphism, dependency, and so on). In this context, this talk will focus on *simple type theories*, by which we mean ones with type structure consisting of finitary algebraic operators and with term structure consisting of binding operators, *cf.* [2]. Examples of simple type theories beyond algebraic theories are first-order logic, the untyped and simply-typed λ-calculi, and the computational λ-calculus.

First, we will be concerned with the abstract syntax of simple type theories. Their signatures will be introduced and their algebraic models defined. The approach, which is based on the theories of abstract syntax and variable binding [3] and of polynomial functors [4], is new. At its core is the general idea that type-theoretic rules are notation for polynomial functors and that the syntax generated by such rules is a free algebra, *cf.* [1]. Free algebra constructions will be discussed. Next, we will extend the theory to incorporate substitution. Algebraic models will be discussed and the preceding developments will be used to construct initial models with substitution. This relates to the introduction of substitution as a meta-operation in type theory and to the construction of a classifying category in category theory. Time permitting, we will touch upon the theory further encompassing equational presentations.

This work provides a basis and framework for our ongoing development of algebraic dependent type theory.

References:

[1] M. Fiore, Discrete generalised polynomial functors, *Proceedings of the 39th International Colloquium on Automata, Languages and Programming* (2012) 214–226.

[2] M. Fiore and C.-K. Hur, Second-order equational logic, *Proceedings of the 19th EACSL Annual Conference on Computer Science Logic* (2010) 320–335.

[3] M. Fiore, G. Plotkin and D. Turi, Abstract syntax and variable binding, *Proceedings of the 14th Symposium on Logic in Computer Science* (1999) 193–202.

[4] N. Gambino and J. Kock, Polynomial functors and polynomial monads, *Mathematical Proceedings of the Cambridge Philosophical Society* 154(1) (2013) 153–192.

*Joint work with Marcelo Fiore.