Last Minute Addition

Implementing Formulas-as-Types-as-Objects

Peter Dybjer and Qiao Haiyan
Chalmers University of Technology
Göteborg, Sweden
{peterd, qiao}@cs.chalmers.se

We give a proposal for how to implement a basic library for formal proof theory and metatheory of type theory using the proof assistant ALF for Martin-Löf type theory. The implementation concerns the basic part of the Curry-Howard-Lawvere correspondence between the natural deduction system for minimal logic, the typed lambda calculus and cartesian closed categories. The key features are the following. (i) We benefit from having dependent types in the metalanguage and implement natural deduction proof trees as a family of sets $\Gamma \vdash A$ indexed by the current assumption list Γ and the conclusion A. These proof trees may alternatively be viewed as de Bruijn like representation of terms of the typed lambda calculus. (ii) We organize numerous useful lemmas about proof trees categorically. We prove categorical properties about proof trees up to (syntactic) identity as well as up to $\beta\eta$-convertibility. The main categorical notion is that of a *simple category with families.* This notion is closely related to that of a multicategory, but has better algebraic properties and is therefore more useful for structuring purposes. It is also closely related to the $\lambda\sigma$-calculus. We also show the more traditional ccc-properties. (iii) We prove that our notion is equivalent in an appropriate sense to more traditional representations of lambda terms.

We wish to emphasize that we prove a stronger version of the "Curry-Howard-Lawvere" correspondence than is normally considered. We argue that this strong version is an appropriate way to capture essential "intensional" properties needed for proof-theoretic analysis, which are not captured by the usual correspondence between cccs and typed lambda calculi.