M.E. Sokhatsky*

Issue I: Internalizing Martin-Löf Type Theory

Igor Sikorsky Kyiv Polytechnical Institute, Kyiv, Ukraine
*Corresponding author: maxim@synrc.com

Annotation

This article demonstrates formal Martin-Löf Type Theory (MLTT) embedding into the host type system with constructive proofs of complete set of inference rules. This was recently made possible by introducing cubical type theory and cubical type checker **cubicaltt**[1]in 2017. The long road from pure type systems of AUTOMATH by de Brujin to type checkers with homotopical core was made. This artcile touches only formal MLTT core type system with Π, Σ types (that correspond to $\forall$ and $\exists$ quantifiers for mathematical reasoning) and identity type.

Each language implementation needs to be checked. The one of possible test cases for type checkers is the direct embedding of type theory model into the language of type checker. As types are formulated using 5 types of rules (formation, introduction, elimination, computation, uniqueness), we construct aliases for the host language primitives and use type checker to prove that it has realization of MLTT. This could be seen as ultimate test sample for type checker as intro-elimination fusion resides in beta-eta rules, so by proving them we prove properties of the host type checker.

More formally cubical MLTT internalization proofs that was impossible to express contructively before cubical type system: the J eliminator and its beta rule. Also this issue opens a series of articles dedicated to examples of formalization in cubical type theory the foundations of mathematics, MLTT modeling and cubical proof verification. As many may not be familiar with Π and Σ types, this issue presents different interpretations of core types from other areas of mathematics.

We should note that this is only entrance to internalization technique, and after formal MLTT embedding we could go through inductive types up to embedding of CW-complexes as pushout gluenings.

[1]http://github.com/mortberg/cubicaltt