Papers in Explicit Mathematics

Thomas Studer*

2005

Abstract

This note provides an overview on my PhD work on explicit mathematics and applicative theories.

1 Object-oriented programming

The aim in this part is to give an interpretation of Java in explicit mathematics. In **A semantics for $\lambda^{\{\}}_{str}$: a calculus with overloading and late-binding** [7] and **Explicit mathematics: power types and overloading** [8], we develop type systems for overloading in late-binding in explicit mathematics.

In order to establish properties of recursive programs within theories of explicit mathematics, a least fixed point combinator is needed. We introduce a combinator if this kind in **Formalizing non-termination of recursive programs** [4].

Finally, in **Constructive foundations for Featherweight Java** [6] we present a model of (a fragment of) Java in a system of explicit mathematics.

2 Universes and induction principles

In **Extending the system T_0 of explicit mathematics: the limit and Mahlo axioms** [2], we present models for systems of explicit mathematics

*Institut für Informatik und angewandte Mathematik, Universität Bern, Switzerland, `tstuder@iam.unibe.ch`

with limit and Mahlo axioms, thus providing proof-theoretic upper bounds for them.

The basic properties for universes in explicit mathematics are studied in **Universes in explicit mathematics** [1]. There, we also introduce the principle of *name induction* to define least universes.

In **A theory of explicit mathematics equivalent to ID_1** [3], we show that name induction can be used to give a system of explicit mathematics that is proof-theoretically equivalent to the system of non-iterated positive arithmetical inductive definitions.

3 Combinatory logic

How to normalize the jay [5] is a small paper that provides a normalization proof for the J-combinator.

References

[1] Gerhard Jäger, Reinhard Kahle, and Thomas Studer. Universes in explicit mathematics. *Ann. Pure Appl. Logic*, 109(3):141–162, 2001.

[2] Gerhard Jäger and Thomas Studer. Extending the system T_0 of explicit mathematics: the limit and mahlo axioms. *Ann. Pure Appl. Logic*, 114(1-3):79–101, 2002.

[3] Reinhard Kahle and Thomas Studer. A theory of explicit mathematics equivalent to ID_1. In Peter Clote and Helmut Schwichtenberg, editors, *Proceedings of Computer Science Logic*, volume 1862 of *Lecture Notes in Computer Science*, pages 356–370. Springer, 2000.

[4] Reinhard Kahle and Thomas Studer. Formalizing non-termination of recursive programs. *J. Log. Algebr. Program.*, 49(1-2):1–14, 2001.

[5] Dieter Probst and Thomas Studer. How to normalize the jay. *Theor. Comput. Sci.*, 254(1-2):677–681, 2001.

[6] Thomas Studer. Constructive foundations for featherweight java. In Reinhard Kahle, Peter Schroeder-Heister, and Robert F. Stärk, editors, *Proceedings of the International Seminar on Proof Theory in Computer*

Science, volume 2183 of *Lecture Notes in Computer Science*, pages 202–238. Springer, 2001.

[7] Thomas Studer. A semantics for $\lambda^{\{\}}_{str}$: a calculus with overloading and late-binding. *J. Log. Comput.*, 11(4):527–544, 2001.

[8] Thomas Studer. Explicit mathematics: power types and overloading. *Ann. Pure Appl. Logic*, 134(2-3):284–302, 2005.