Computer Science at the Frontiers of Mathematics

Robert Constable

January 15, 2020

Increasingly computer science is engaged at the frontiers of mathematics. One recent example is research in the Computer Science Department at Cornell University in the area of mathematics called *homotopy theory*. One of the world's best mathematicians, the Fields Medalist Vladimir Voevodsky, working at the *Institute for Advanced Study* near Princeton University, sought help in confirming the validity of a proposition he called the *Univalence Axiom*. One way of confirming the correctness of proposed axioms is to *formalize* the theory in which they are stated. That means formulating all of the definitions and theorems extremely precisely so that every detail of purported proofs can be *checked by a computer* using only the primitive notions of the theory and the axioms of logic. Beginning in 1985 a large team of graduate students and post doctoral assistants worked with this author to design and build the Nuprl proof assistant (pronounced as "new pearl."). It was one of the first such systems and is still actively used and further developed today 35 years later. Another well known system of this kind is the *Coq* proof assistant [5, 4]. One inspiration for these systems was the pioneering work of de Bruijn [6, 7].

The Univalence Axiom is key in precisely defining and proving Voevodsky's results in homotopy theory. He and others also created a very precisely defined version of this theory called *Homotopy Type Theory* (HoTT). By creating a formal theory it became possible to use computers to check all the details of proofs. In the case of the Univalence Axiom, this verification task was very difficult. Several experts in verification suggested that the proof assistant built, supported, and used at Cornell was the best resource for verifying the correctness of this proposed axiom. That proof assistant is called Nuprl ("new pearl"). It was designed and built from 1985 to 1986 by a large team led by this author and Joseph Bates [1]. A book about Nuprl was written in 1986 [3] by a large team. From 1986 and still being enhanced, the Nuprl proof assistant is used to build large *libraries of formal mathematics*. These libraries and the continuing evolution of the Nuprl implementation make it possible to formally verify very advanced theorems as well as proposed new axioms. The system has been used to solve open problems in mathematics [2].

Vladimir Voevodsky visited Cornell University to work with the Nuprl proof assistant in the fall of 2016. He worked with Dr. Mark Bickford, an expert in type theory and in the use of Nuprl.

For several days Dr. Bickford and Dr. Voevodsky worked to formally confirm the correctness of this critical Univalence Axiom. Their effort led to a very rich collection of results in this new theory. The National Science Foundation (NSF) supported the development and dissemination of this ground breaking work.

The formalization of Cubical Type Theory was a critical step in this work. It presented many technical challenges requiring the Cornell team to develop new tactics and prove over *one thousand lemmas*. There were two discoveries that were especially critical. The first discovery concerned the use of *nominal logic*. That logic introduces a primitive concept of names along with new rules about formulas that mention names. Fortunately Nuprl already had a type of "unguessable" atoms. The rules for atoms make Nuprl a *nominal logic*. That was precisely what was needed in this major verification task.

This ground breaking work of Bickford and Voevodsky was important in the subsequent development of Homotopy Type Theory. It also demonstrated the value of implementing an extremely rich foundational theory of mathematics. A very richly expressive theory significantly extends the reach of formal mathematics and brings to bear on hard problems the enhanced reasoning power enabled by human/machine interaction. The value of such a machine and human cooperation is known very well in the physical sciences and engineering. Now we see the value of its reach into the most advanced mathematics being created. Modern proof assistants provide a grounding for mathematics that is new to this age. Its potential is clear, and we can expect more critical discoveries and applications of this human/machine partnership as it matures with NSF and DoD support.

Having numerical data about Nuprl theories might help readers understand the data used in the formal technology of modern proof assistants. For the library of *constructive real analysis*, there are: 272 definitions, 1,731 objects. In one of the theories of real numbers there are 80 definitions and 449 lemmas. As of May 2018 there were 4,816 Definitions; 19,601 Lemmas; 107,159 Proof Steps, and 700 Million *refinements executed during replay of all theories*. The average proof step produces a tree of 7,000 primitive refinements. There are at least 1.5 Billion primitive refinement steps taken, and almost certainly over 2 Billion such steps in a *full replay* of the Nuprl Library.

References

[1] J. L. Bates and Robert L. Constable. Definition of micro-PRL. Technical Report 82–492, Cornell University, Computer Science Department, Ithaca, NY, 1981.

[2] Robert Constable and Mark Bickford. Intuitionistic Completeness of First-Order Logic. *Annals of Pure and Applied Logic*, 165(1):164–198, January 2014.

[3] Robert L. Constable, Stuart F. Allen, H. M. Bromley, W. R. Cleaveland, J. F. Cremer, R. W. Harper, Douglas J. Howe, T. B. Knoblock, N. P. Mendler, P. Panangaden, James T. Sasaki, and Scott F. Smith. *Implementing Mathematics with the Nuprl Proof Development System*. Prentice-Hall, NJ, 1986.

[4] Thierry Coquand and G. Huet. The calculus of constructions. *Information and Computation*, 76:95–120, 1988.

[5] Thierry Coquand and G. P. Huet. Constructions: A higher order proof system for mechanizing mathematics. In *EUROCAL '85*, volume 203 of *Lecture Notes in Computer Science*. Springer-Verlag, 1985.

[6] N. G. de Bruijn. The mathematical language Automath: its usage and some of its extensions. In J. P. Seldin and J. R. Hindley, editors, *Symposium on Automatic Demonstration*, volume 125 of *Lecture Notes in Mathematics*, pages 29–61. Springer-Verlag, 1970.

[7] N. G. de Bruijn. A survey of the project Automath. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays in Combinatory Logic, Lambda Calculus, and Formalism*, pages 589–606. Academic Press, 1980.